TURKISH NAVAL ACADEMY
NAVAL SCIENCE AND ENGINEERING INSTITUTE
DEPARTMENT OF COMPUTER ENGINEERING

A DATA MINING BASED TARGET CLASSIFICATION FOR TACTICAL UNDERWATER SENSOR NETWORKS

MASTER THESIS

YAŞAR DOĞAN

Advisor: Assist.Prof. Vedat Coşkun
Co-Advisor : Assoc.Prof. Erdal Çayırcı

İstanbul, 2004

A DATA MINING BASED CLASSIFICATION ALGORITHM FOR TACTICAL UNDERWATER SENSOR NETWORKS

Submitted in partial fulfillment of the requirements for degree of

MASTER OF SCIENCE IN COMPUTER ENGINEERING

from the

TURKISH NAVAL ACADEMY

Author :

Yaşar Doğan

Defense Date : 26/07/2004
Approved by: 13 /08/2004

Assist.Prof. Vedat Coşkun (Advisor)

Assoc.Prof. Erdal Çayırcı (Co-Advisor)

Prof. M. Yahya Karslıgil (Defense Comittee Member)

Prof. Şebnem Baydere (Defense Comittee Member)

Assist.Prof. Gökhan Yavuz (Defense Comittee Member)

ABSTRACT (TURKISH)

SU ALTI TAKTİK DUYARGA AĞLARINDA VERİ MADENCİLİĞİ TABANLI HEDEF SINIFLANDIRMASI

Anahtar Kelimeler : Su altı duyarga ağları, Sınıflandırma veri madenciliği, Karar ağacı, Hedef sınıflandırma, Yakınlık duyargaları, Radyasyon(Işınım) duyargaları, Manyetik duyargalar, Akustik duyargalar.

Mikroelektro-mekanik sistemler, kablosuz iletişim ve dijital elektronikteki gelişmeler çok sayıda küçük boyutta, ucuz maliyetli algılama, işlem ve iletişim kapasitelerine sahip duyargaların üretilmesine olanak sağlamıştır. Aynı zamanda düşük maliyet yüzlerce hatta binlerce duyargadan oluşan ağlara ve bu ağlarda verilerin güvenilirliğini, doğruluğunu ve kapsanan alanı artırmaya olanak sağlar. Herbir duyarganın çevreyi bağımsız olarak algılayabildiği, ancak birlikte işlem yaptıklarında işgal tespiti, hedef takibi, çevresel gözlem, uzaktan algılama gibi karmaşık bilgileri birleştirebilen ve yayabilen büyük miktardaki duyargalar bölgeye süratle yerleştirilebilir.

Duyarga ağ uygulamalarının en önemlilerinden bir tanesi savaş alanında hedef sınıflandırmasıdır.Ucuz, çoklu duyargalı akıllı aygıtlar hedef sistemlerinin izlenmesi ve kontrolü için olanak sağlarlar. Duyargalar değişik veri formlarını işleme ve elde etme kapasitesine sahiptir. Çoklu sinyal işleme ve birleştirme algoritmaları güvenilir ve etkili biçimde karar vermek için ağdaki nodlar arasındaki dağıtık veriyi birleştirir.

Bu tezde su altı duyarga ağlarında veri madenciliği tabanlı sınıflandırma algortimasını önerdik. Açık, sığ ve çok sığ sularda denizaltı, küçük sualtı taşıma araçları, su altı mayınları ve dalgıçları tespit etme ve sınıflandırmada yalnızca

birkaç dolar maliyeti olan mikroduyargalar kullanılmıştır. Algoritma, yüzeydeki şamandıralara bağlı ve ayarlanabilir derinliklere indirilebilen duyargalardan oluşan taktik su altı duyarga ağları için tasarlanmıştır. Dinamik olarak denizaltı, küçük sualtı taşıma aracı, mayın ve dalgıç tespitinde, sınıflandırma veri madenciliği tekniği olarak, karar ağacı algoritmaları kullanılmıştır. Sınıflandırma için yakın bir cismin varlık bilgisi, manyetik ve akustik veriler birlikte kullanılmıştır.

ABSTRACT (ENGLISH)

A DATA MINING BASED TARGET CLASSIFICATION FOR TACTICAL UNDERWATER SENSOR NETWORKS

Keywords : Underwater sensor networks, Classification mining, Decision tree, Target detection, Proximity microsensor, Radiation microsensor, Magnetic microsensor, Acoustic microsensor.

Advances in MEMS, wireless communications, and digital electronics have made it possible to produce large amount of small-size, low-cost sensors which integrate sensing, processing, and communication capabilities together. Also, the low cost makes it possible to have a network of hundreds or thousands of these sensors, thereby enhancing the reliability and accuracy of data and the area coverage. Large amount of these sensors can be quickly deployed in the field, where each sensor independently senses the environment but collaboratively achieves complex information gathering and dissemination tasks like intrusion detection, target tracking, environmental monitoring, remote sensing, global surveillance, etc.

Inexpensive, smart devices with multiple sensors provide opportunities for instrumenting, monitoring and controlling targeting systems. Sensor nodes have capability for acquiring and embedded processing of variety of data forms. Collaborative signal processing and fusion algorithms are needed to aggregate the distributed data from among the nodes in the network, including possibly multiple modalities of data within a sensor node, to make decisions in a reliable and efficient manner. One of the important sensor network applications is target classification in battlefields.

In this thesis we proposed a classification mining based detection algorithm for underwater sensor networks is introduced. Microsensors, which cost only a couple of dollars, are used to tackle the challenge of detecting and classifying submarines, SDV (small delivery vehicles), underwater mines and divers in open, shallow and very shallow water. The algorithm is designed for wireless tactical underwater sensor network architectures where sensors attached to a surface buoy can be lowered to an adjustable depth. Decision tree algorithms are used as a classification mining technique for dynamic diver, mine, submarine, small delivery vehicle detection. The sensed proximity, magnetic and acoustic data are used for collaborative classification.

DISCLAIMER STATEMENT

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Turkish Navy, Naval Academy and Naval Science and Engineering Institute.

DEDICATION

To my wife Seda and my son Göktürk Buğra.

ACKNOWLEDGEMENT

I would like to thank my advisors Assoc.Prof.Erdal Çayırcı and Assist.Prof.Vedat Coşkun for their helpful discussions, comments, and Prof. Suleyman Özkaynak for obtaining the required materials and support.

Finally, I would like to express my deepest gratitude to my wife Seda Doğan and my little son Göktürk Buğra Doğan for their love, patience and everlasting support.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS, ACRONYMS, AND SYMBOLS

WSN	Wireless Sensor Network
MEMS	Micro-electro-mechanical system
CMDC	Classification Mining Based Detection and Classification
SDV	Small Delivery Vehicle
NBC	Nuclear, Biological and Chemical
3D	Three-Dimensional
LCD	Liquid Crystal Display
IC	Integrated Circuit
ER	Energy Ray
UV	Ultraviolet
IR	Infrared
VIS	Visible
NIR	Near Infrared
ASIC	Application Specific Integrated Circuit
KDD	Knowledge Discovery in Databases
AI	Artificial Intelligence
OLAP	On-Line Analytical Processing
DT	Decision Tree
ID3	Induction of Decision Tree
CART	Classification Mining and Regression Trees
GP	Cgenetic Programming
CdS	Cadmium Sulfate
SAW	Surface Acoustic Wave
WEKA	Waikato Environment for Knowledge Analysis
SLIQ	Supervised Learning In Quest
BPNN	Back Propagation Neural Network
JSTARS	Joint Surveillance Target Attach Radar System

I. INTRODUCTION

A. MOTIVATION

1. What is a Wireless Sensor Network?

Wireless Sensor Networks are composed of wireless sensors, each capable of process, transmit and receive information. Sensors may be randomly deployed over an area of observation, operating under energy constraints in an unattended mode. These networks are intended for a broad range of environmental sensing applications from weather data collection to vehicle tracking and habitat monitoring.

A sensor node is constituted four main components which are shown in Figure 1. These main components are: a sensing unit, a processing unit, a transceiver unit and a power unit. A sensor node may also has additional components such as location finding system, a power generator and a mobilizer according to the application [1].

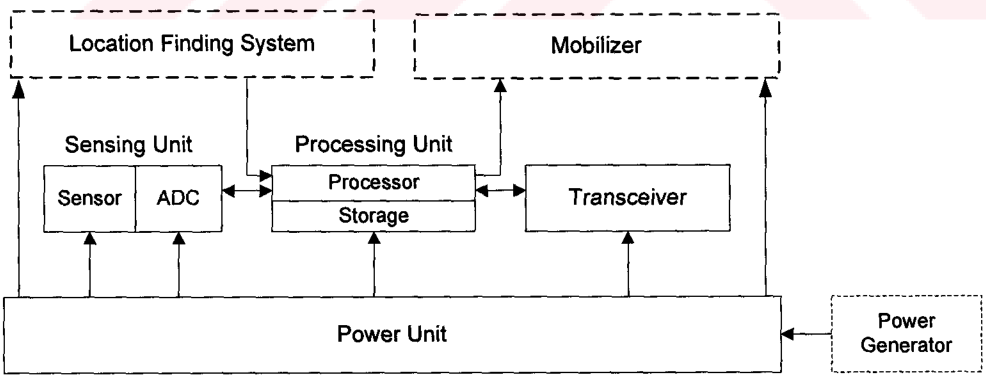


Figure 1. The components of a sensor node [1].

Modern sensors not only respond to physical signals to produce data, they also embed computing and communication capabilities. They are able to store, process locally and transfer the data they produce. A set of signal processing functions transform physical signals such as heat, light, sound, pressure, magnetism, or a particular motion into sensor data, i.e., measurements of physical phenomena as well as detection, classification or tracking of physical objects.

2. Application Examples of Wireless Sensor Networks

Sensor networks may consist of seismic, low sampling rate magnetic, thermal, visual, infrared, acoustic and radar sensors. These different types of sensors may use to monitor the ambient conditions which include the following [1]:

- Temperature,
- Humidity,
- Vehicular Movement,
- Lightening condition,
- Pressure,
- Soil makeup,
- Noise levels,
- The presence or absence of certain kinds of objects,
- Mechanical stress levels on attached objects,
- The current characteristics such as speed, direction, and size of an object.

Sensor network research is initially driven by military applications such as battlefield surveillance and enemy tracking because of their flexible, low cost, and self-organizing features. Existing applications can be classified into five categories as listed below[1].

a. Military Applications

- Monitoring friendly forces, equipment and ammunition,

- Battle-field surveillance,
- Reconnaissance of opposing forces and terrain,
- Targeting,
- Battle damage assessment, and
- Nuclear, biological and chemical (NBC) attack detection and reconnaissance.

b. Environmental Applications

- Forest fire detection,
- Flood detection,
- Biocomplexity mapping of the environment,
- Precision agriculture, and
- Habitat monitoring.

c. Health Applications

- Telemonitoring of human physiological data,
- Tracking and monitoring doctors and patients inside a hospital,
- Drug administration in hospitals.

d. Home Applications

- Home automation, and
- Smart environment.

e. Other Commercial Applications

- Environmental control in office buildings,
- Interactive museums,
- Detecting and monitoring car thefts,
- Managing inventory control,
- Vehicle tracking and detection.

B. PROBLEM DEFINITION AND PROPOSED SCHEME

One of the application areas that well fit the characteristics of sensor networks is tactical underwater surveillance systems. An example for the underwater sensor network architectures is introduced in [2] where sensors initially are in the surface buoys. After the deployment of these buoys, each sensor is lowered to appropriate depths such that the maximum coverage of the 3D sensor space is maintained [2]. In this architecture, although the sensors are underwater, the nodes can collaborate through the wireless medium over sea surface by using the antenna at the surface buoys as shown in Figure 2. In this thesis we introduce a classification mining based algorithm to detect submarines, small delivery vehicles, mines and divers for tactical underwater sensor networks that use this architecture.

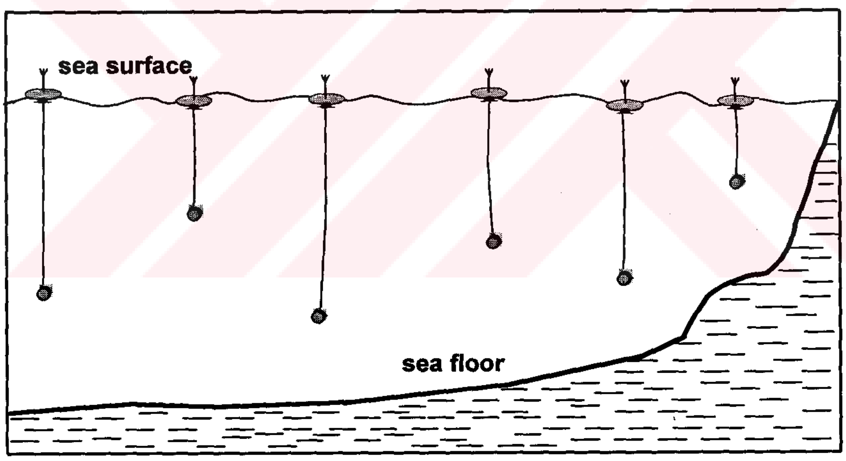


Figure 2. Underwater sensor array.

There are studies in applying data mining techniques to real time target detection, classification and tracking applications and habitat monitoring [3-6]. Data mining techniques can also be used to detect an underwater target based on the sensed data generated by a number of microsensors located close to the target. In this thesis we explore this idea to develop a classification based mining algorithm for tactical underwater sensor networks. We collect readings from multiple sensors of proximity, magnetic and acoustic type to detect the existence and type of an underwater target. We propose a decision tree classification mining based detection algorithm where decision trees are used to represent rules and adventageous for their fast execution time, ease in the interpretation of the rules and scalability for large multi-dimensional data sets [7-10].

C. CONTRIBUTION OF THE THESIS

Tactical underwater surveillance systems have three major tasks: detection, classification and tracking submerged targets. The scope of our study is to develop an efficient decision tree classification scheme for tactical underwater sensor networks. Our scheme first detects a target in the vicinity based on the readings of microsensors. And scheme classifies the target into one of the target classes based on the data coming from microsensors.

D. STRUCTURE OF THE THESIS

This thesis is organized as follows: Section II gives background information about microsensors and knowledge discovery in databases(data mining). Detection and Classification of targets by using decision tree are examined in Section III. The performance of the scheme is evaluated in Section IV, and the thesis is concluded in Section V.

II. BACKGROUND INFORMATION AND RELATED WORK

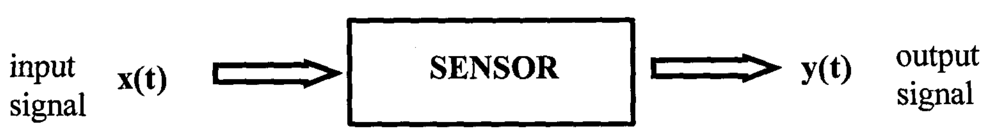
A. MICROSENSORS

Sensors can be defined as devices that convert a non-electrical physical or chemical quantity into an electrical signal [11]. The electrical signal from the sensor often needs modification before it can serve a useful function.

Some sensors are self-generating or self-exciting as opposed to modulating [12]. A self-generating sensor does not need an external power supply to work. In its simplest form a sensor is may be regarded as a system with an input x (t) and output y (t). A self-exciting sensor has its output energy supplied entirely by the input signal x (t). The general equation that describes a self-exciting sensor system is

$$y\,(t) = F(x(t)) \qquad (1)$$

where F(x(t)) is the characteristic relationship that describes the behavior of a self-exciting sensor. Figure 3 shows a system representation of (a) a self-exciting sensor and (b) a modulating sensor.



y(t)=F(x(t))

Figure 3a. Self-exciting sensor[11][12].

In the case of the modulating sensor the system equation can be written more explicitly as

$$y\,(t) = F(x(t) + x_d) \qquad (2)$$

where the external supply signal $x_d(t)$ should ideally be stationary and noise free.

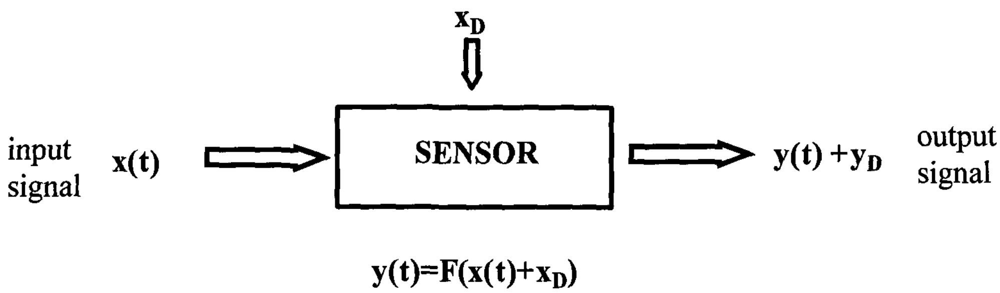


Figure 3b. Modulating Sensor[11][12].

x_d (t) could be a constant driving current or applied reference voltage. Figure 4 shows the ideal input-output characteristic of a sensor in which the input signal (measurement) is directly proportional to the output signal. The ideal sensor not only has a linear output signal y (t) but it should instantaneously follow the input signal x (t), where

$$y(t) = S \times x(t) \qquad (3)$$

The slope S of the input-output curve has a constant value for a linear sensor and is usually referred to as the *sensitivity*.

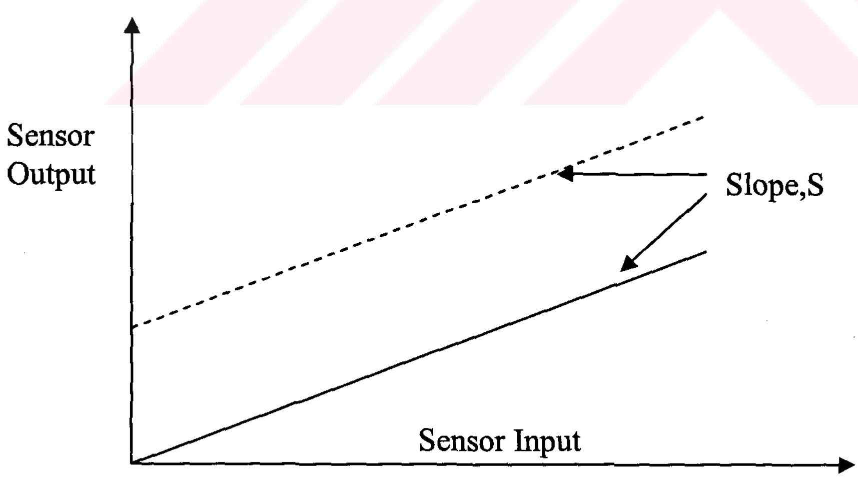


Figure 4. Ideal input-output relationship of self-exciting (solid-line, x_d=0) and modulating (dashed line, $x_d \neq 0$) sensors.

1. Why Microsensors?

Sensors have an important role to play in our everyday lives in which we have a need to gather information, process it and perform some task. A large sensor may have excellent operating characteristics but its marketability is severely limited simply by its size. A reduction in the size of a sensor often leads to an increase in its applicability through:

- lower weight (greater portability),
- lower manufacturing cost (less materials),
- wider range of applications.

The cost of a sensor is often the single most important factor. Clearly, fewer materials are needed to manufacture a small sensor but the cost of materials processing is often a more significant factor. Recent advance in processor technology has led to a considerable demand for small sensors or microsensors that can fully exploit the benefits of Integrated Circuits (IC) micro technology.

Information-processing systems may have many stages, but the basic components are; a sensor, a signal processor and an actuator. Figure 5 shows these basic components.

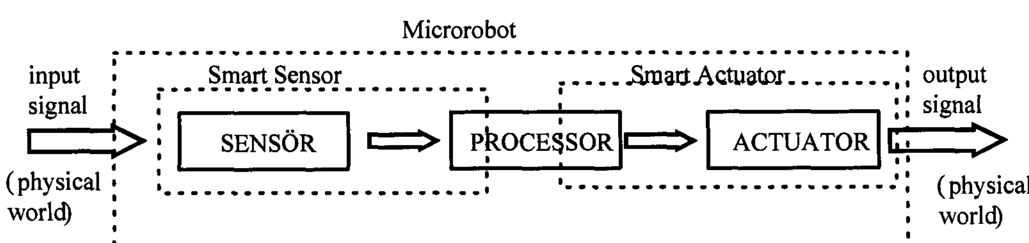


Figure 5. Basic components of a measurement or information-processing system[11].

A sensor with a partly or totally integrated processor is called an *integrated* or *smart* sensor. Similarly, an actuator with part or total integration of the processor could be called an integrated or smart actuator.

2. Microsensor Classes

Microsensors can be classified into seven categories according to their principle form of signal: thermal, radiation, mechanical, magnetic, chemical, biological and electrical [11]. Table 1 shows these forms of signals and their measurands.

Table 1. Classification of sensors by signal form.

Form of Signal	Measurands
Thermal	Temperature, heat, heat flow, entropy, heat capacity etc.
Radiation	Gamma rays, X-rays,ultra-violet, visible, infra-red, micro-waves, radio waves, etc.
Mechanical	Displacement, velocity, acceleration, force, torque, pressure, mass, flow, acoustic wavelength and amplitude etc.
Magnetic	Magnetic field, flux, magnetic moment, magnetisation, magnetic permeability etc.
Chemical	Humidity, pH level and ions, concentration of gases, vapours and odours, toxic and flammable materials, pollutants etc.
Biological	Sugar, proteins, hormones, antigens etc.
Electrical	Charge, current, voltage, resistance, capacitance, inductance, dielectric permittivity, polarisation, frequency etc.

2.1. Thermal Microsensors

Thermal sensors are used to measure various heat-related quantities, such as temperature, heat flux, and the heat capacity [11] [12]. Temperature is perhaps the most fundamental variable quantity and provides a measure of the thermal energy.

Thermal sensors are classified as contacting rather than non-contacting sensors. In contacting sensors the sensing element physically touches the heat source. And thermal signal is propagated from the heat source by conduction of heat into the sensing element which usually generates an electrical signal or modulates an electrical signal. Non-conducting temperature sensors are classified as radiation sensors. Radiation temperature sensors detect the electromagnetic waves emitted by a body.

2.2. Radiation Microsensors

Radiation is the emission of either particles or electromagnetic rays from a source. Particles may be generated by the decay of a radioactive material or by the interaction of a nucleus with another energy source. Electromagnetic particles have zero rest mass energy, and are usually described as waves rather than particles. As the energy of an electromagnetic particle is proportional to its frequency, it is usual to classify electromagnetic radiation according to its frequency as a fundamental property.

Figure 6 shows the entire electromagnetic spectrum from the highest energy cosmic ray particles (ER ~ GeV) down to the lowest energy radio waves (ER ~ neV). Many of these types of radiation are commonly used in our everyday lives. X-rays are used to image our internal bone structure in medicine. Ultra-violet radiation is used to produce an artificial sun-tan. Visible radiation (normally called light) is detected by the human eye but it occupies only a small range in wavelength (400 to 700 nm). However, visible radiation is of major importance in many applications (cameras, opto-electronics etc.) The utilization of radiation within other types of sensors is common. Infra-red radiation can be used to probe the properties of molecules or measure temperature. Microwaves and radar radiation are used in proximity sensors, whereas microwave radiation can also be used for heating purposes. Finally, radio waves are used in many communication systems.

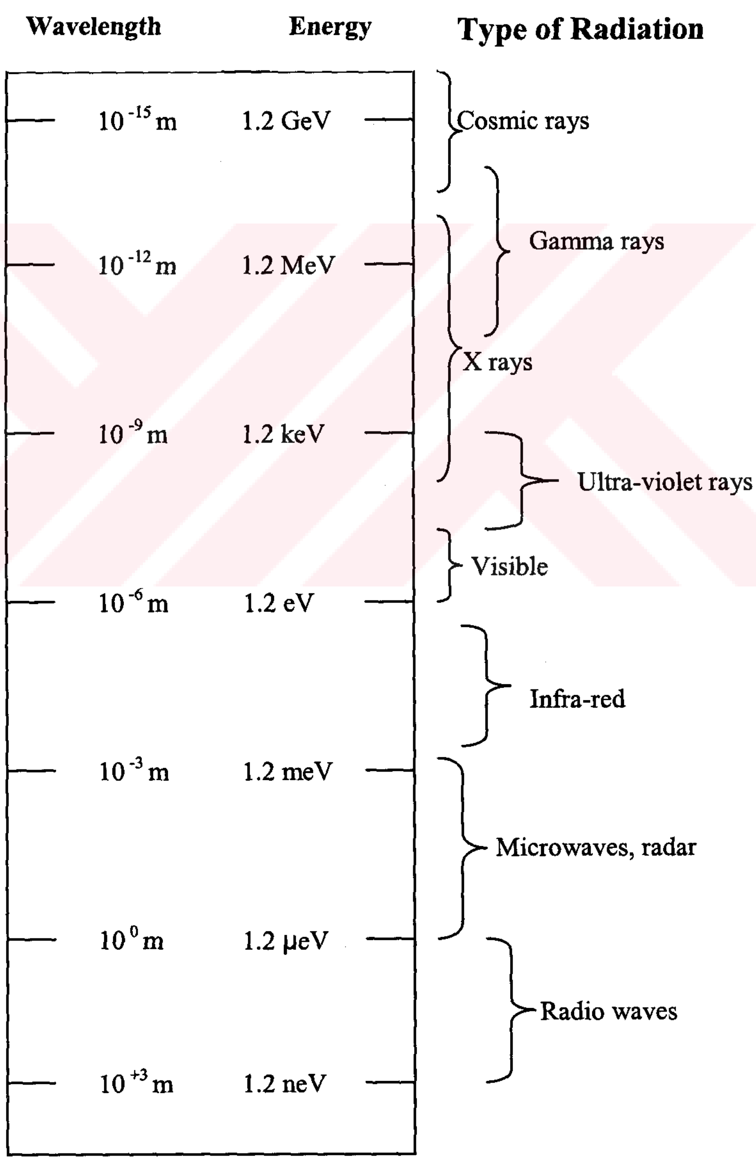


Figure 6. The spectrum of electromagnetic radiation[11].

Radiation sensors can be classified as non-conducting sensors because they detect remotely the emission of various particles or electromagnetic radiation, e.g. UV, IR.

2.3. Mechanical Microsensor

Mechanical sensors are the largest class of sensors because of their widespread applicability. There are a large number of mechanical measurands to consider, from acceleration to wavelength.

Table 2. Mechanical measurands.

Acceleration, angular	Flow, gas	Momentum, linear	Stiffness
Acceleration, linear	Flow, liquid	Movement	Strain
Acoustic energy	Flow-rate	Orientation	Stress
Altitude	Force, simple	Path length	Tension
Angle	Force, complex	Pitch	Thickness
Attraction	Frequency	Position	Torque
Compliance	Friction	Pressure	Touch
Contraction	Hardness	Proximity	Velocity, angular
Deflection	Immersion	Roll	Velocity, linear
Deformation	Inclination	Rotation	Vibration
Density	Length	Roughness	Viscosity
Diameter	Level	SAW	Volume
Dipole alignment	Mass	Shape	Wavelength
Displacement	Microbend	Shock Yaw	
Elastic properties	Momentum	Sound level	

Table 2 lists over 50 measurands that would normally be classified as mechanical. Clearly, there is a need to reduce this large of measurands into a smaller, more manageable scheme that could be used to define the genus of mechanical sensors. Table 3 shows a smaller number of mechanical measurands which are used to define the most important classes of mechanical sensors.

Table 3. Classification of main mechanical measurands.

1. Position, displacement	7. Stiffness, compliance
2. Velocity, speed	8. Mass, density
3. Acceleration	9. Flow, rate
4. Force, torque	10. Shape, roughness
5. Stress, pressure	11. Viscosity
6. Strain	12. Other (Acoustic/Ultrasonic)

2.4. Magnetic Microsensors

The discovery of various magnetic effects and the development of special magnetic materials have enabled the fabrication of a wide range of magnetic sensors. In direct applications, the magnetic sensor is essentially a magnetometer. Thus it is used to measure a magnetic field strength, such as coming from the earth, or the field modulated by magnetic material on magnetic tapes, discs etc. In indirect applications, the magnetic sensor is used as an intermediate sensor to detect non-magnetic signals. There are many examples of indirect applications such as:

- linear and angular position sensing using Hall effect devices;
- pressure sensing using Hall effect devices;
- control of brushless DC motors;
- non-contact current sensing.

In direct applications, magnetic sensors are commonly used to detect magnetic flux densities in the range of micro- to mili-tesla which are usually man-made.

2.5. Chemical Microsensors

A chemical sensor is a device which is capable of converting a chemical quantity into an electrical signal. Chemical species are usually present in mixtures where the sample matrix might be gas, liquid or solid. A chemical sensor which detects a biological quantity is normally referred to as a biochemical sensor or biosensor.

3. Microsensor Performance

Different applications may require fundamentally different sensor performances. For example, the temperature of a room does not normally need to be measured very precisely. In this case, a low, cost silicon thermometer, such as a thermo transistor, is fit for the desired purpose. However, a microsensor situated inside a heart pacemaker to measure an erratic heart-beat must be extremely reliable, because its failure could lead to the death of its user. In the second example, the microsensor is said to be safety-critical.

The overall performance of a microsensor or microsensor system can be described by a parameter called the (system) *effectiveness*. There are essentially three factors that contribute to the effectiveness of a microsensor, Figure 7.

First, the *capability* of a microsensor is its capacity to perform the desired function under predefined conditions.

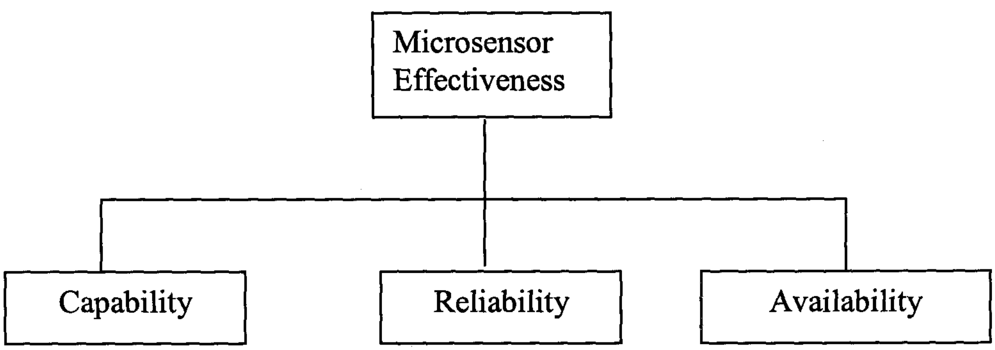


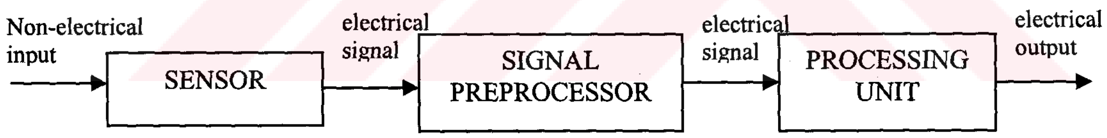
Figure 7. Effectiveness of a microsensor or microsensor system[11].

The second factor determining the effectiveness of a microsensor is its inherent *reliability.* Reliability is defined as the ability of an item to perform a required function under stated conditions for a stated period of time. Thus the reliability of a microsensor is its ability to stay within its technical specification over a prescribed period of time.

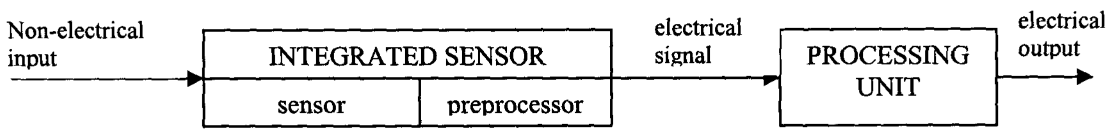
The *availability* is defined as the probability that an item is capable of performing a required function when required to do so. The availability of a microsensor is in the fact defined by its failure-rate and its repair-rate (e.g. its maintainability).

4. Smart Sensors

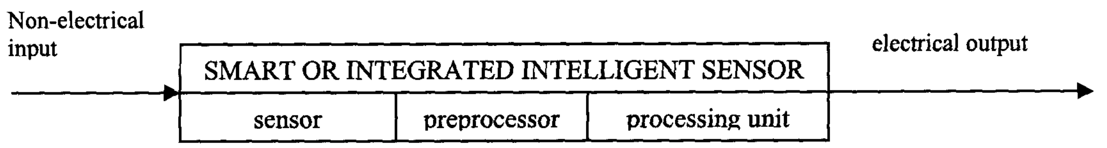
There are different views over the precise definition of a smart sensor. In [13], authors define smart sensor as a device onto which at least one sensing element and signal processing circuit has been integrated.



a. A sensor system



b. An integrated sensor in a sensor system



c. A smart sensor or integrated intelligent sensor

Figure 8. Sensor systems[11].

Instead the term *integrated sensor* to describe this type of low-level smart sensor where most of its preprocessor is integrated, Figure 8 (b). The revised definition is proposed more recently [14], term *smart* is reserved to denote the integration, in part of full, of the main processing unit, see Figure 8 (c), which adds intelligence.

The definition is proposed by [15] takes some account of work in artificial intelligence and as follows:

"The sensor itself has a data processing function and automatic calibration/automatic compensation function, in which the sensor detects and eliminates abnormal values or exceptional values. It incorporates an algorithm, which is capable of being altered, and has a certain degree of memory function. Further desirable characteristics are that the sensor is coupled to other sensors, adapts to changes in environmental conditions, and has a discrimination function".

This definition is rather long but does incorporate the essential functions required to define intelligence in a sensor. An intelligent sensor must possess one or more of the following three features:

- perform a logical function;
- communicate with one or more other devices;
- make a decision using crisp or fuzzy data.

The definition is not only consistent with that proposed by [16] for a smart sensor but also distinguishes between an integrated sensor and a hybrid intelligent sensor.

An intelligent sensor may be able to communicate with its operator and so provide valuable information about problems etc. This type of intelligence can, in its simplest form, provide a warning of abnormal operating conditions, or more clearly provide a feedback control mechanism. An intelligent sensor may have some form of high level adaptive control strategy which permits the control parameters to be automatically updated with time. The implementation of a sensor which can warn its user, or adapt to environmental conditions, requires some decision making capability. Traditionally sensors use parametric data to make a decision.

B. KNOWLEDGE DISCOVERY IN DATABASES AND DATA MINING

1. Data Mining

Advances in information technologies made possible for the scientific and industry communities to acquire, and store increased volumes of data in digital form, even more data than they can really process. This fact gave arise to a hybrid research field called Knowledge Discovery in Databases (KDD). Many people treat data mining as a synonym for Knowledge Discovery in Databases (KDD) [17].

KDD is concerned with the nontrivial identification and extraction of valid, novel, potentially useful, and ultimately understandable knowledge from large databases, and data mining is the central step in this process [18].

Data mining has become of great consequence for more issues, and the inevitability of data mining is becoming more apparent. Knowledge discovery as a process is depicted in Figure 9 and consists of an iterative sequence of the following steps [7]:

- *Data cleaning* (to remove noise and inconsistent data)
- *Data integration* (where multiple data sources may be combined)
- *Data selection* (where data relevant to the analysis task are retrieved from the database)

- *Data transformation* (where data are transformed or consolidated into forms appropriate for mining by performing summary or aggregation operations, for instance)
- *Data mining* (an essential process where intelligent methods are applied in order to extract data patterns)
- *Pattern evaluation* (to identify truly interesting patterns representing knowledge based on some interestingness measures)
- *Knowledge presentation* (where visualization and knowledge representation techniques are used to present mined knowledge to the user)

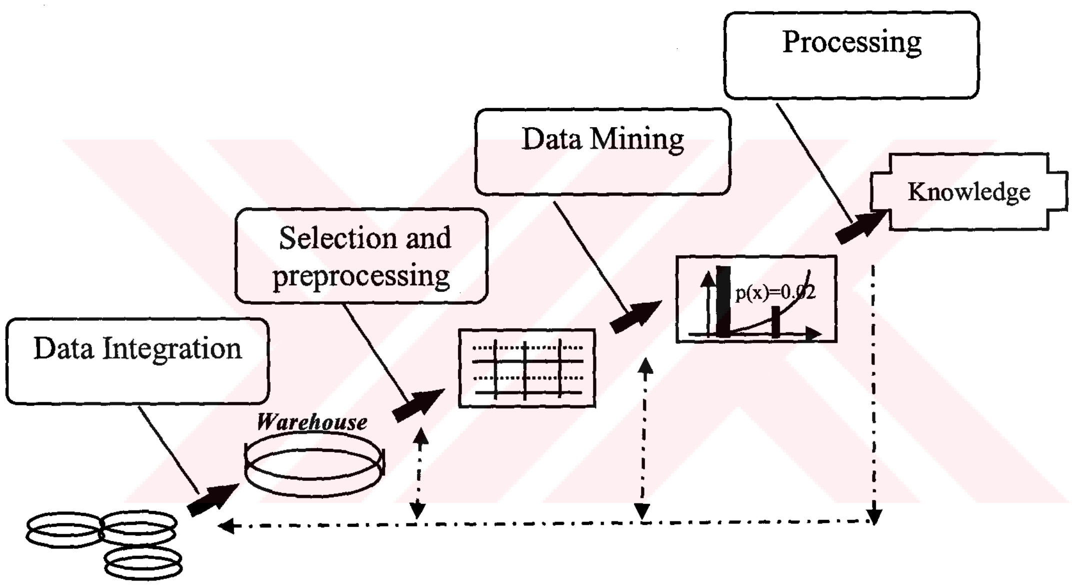


Figure 9. Data mining as a step in the process of knowledge discovery.

As mentioned before, the huge amount of data generated from our daily activities has prohibited traditional human labor to handle it, and thus, data mining becomes the inevitable solution. Basic techniques of data mining involve finding association rule, sequential pattern discovery, classification, and clustering. Finding association rule [19], [20] is the most extensively discussed technique, and a lot of literatures for mining association rules have already been proposed. Sequential pattern discovery [21], [22], [23] is a little similar to mining association rules while it involves the time factor in the mining association rule affairs.

Classification [24], [25] is a very useful methodology which has been widely used on various kinds of domains besides data mining, for example, artificial intelligence (AI), machine learning, and pattern reorganization, etc. Clustering [26], [27] is another methodology whose functionality is similar to the classification; however it differs from the classification with its essential nature.

Data Mining algorithms can be classified into two broad categories: predictive or descriptive approaches. In predictive data mining, the aim is to induce a hypothesis that correctly classify all the given examples and can be used for classification of future, yet unseen, instances. The most popular example of predictive technique is decision tree induction [28], [29]. On the other hand, the goal in descriptive data mining is to characterize as much as possible, by finding patterns regularities. Discovering of association rules is one example of this class of data mining techniques [30].

There exist other analytical tools that explore data interrelationships, for instance OLAP (On-Line Analytical Processing) [31]. But the difference between KDD and these tools is in the approach they use. Many of the analytical tools available support a verification-based approach, in which the user hypothesizes about specific data interrelationships and then uses the tools to verify or refute those hypotheses. The effectiveness of this approach relies mainly in the knowledge of the domain in question. On the other side, the goal of data mining is to find patterns of influence. Efforts have been made to combine existing analytical tools such as OLAP and Data Mining techniques.

2. Classification

Classification is a primary data mining task aimed at learning a function that classifies a database record into one of several predefined classes (e.g., classes of profitable versus nonprofitable customers) based on the values of the record attributes (age, occupation, etc.) [32]. Records are commonly gathered in *data sets*.

Each instance in a data set can be regarded as a vector with a given number of values. Each of the values represents an *attribute value* measured or chosen for the instance.

In order to make a classifier, we need to have a set of already classified instances to train the system. The set of already classified instances used for this purpose is called the *training set*. In addition to all the conditional attributes, these instances also have an attribute called the *decision attribute*, which represents the classification of each instance. The values of this attribute represent the possible classes the instances from this particular distribution can belong to. Thus, these values can either be *binary*, if there are only two classes.

When a classifier has been constructed, its performance can be tested by using a set of instances that have not yet been classified. The set of instances used to test the system's performance is called the *test set*. This set should be drawn from the same distribution as the training set. If the classifier performs adequately, we can classify future instances for which we do not know the class by running them through the classifier.

Common classification methods, like back propagation, Naive Bayes Classifier, and C4.5, are designed to optimize the predictive performance of the induced model, e.g., its ability to correctly classify. Other aspects of knowledge discovery, such as validity of discovered patterns, simplicity of representation, and identification of relevant features, are given only secondary consideration by most existing algorithms. Classification models induced from real-world data tend to be over complex, statistically insignificant, and wasteful in the number of used features. The information-theoretic classification method [33] is aimed at solving these problems by using three guiding principles:

maximizing the mutual information between a set of predictive attributes and the target (classification) attribute,

finding a minimal set of database attributes involved in the induced model, and verifying the statistical significance of the discovered patterns.

2.1 Information Theory and Classification

The data classification process is aimed at reducing the amount of uncertainty or gaining information about the target (classification) attribute. In Shannon's information theory [34], information is defined as that which removes or reduces uncertainty. For a classification task, more information means higher accuracy of a classification model since the predicted class of new instances is more likely to be identical to their actual class. A model that does not increase the amount of information is useless and its predictive accuracy is not expected to be better than just a random guess. Information theory [34] suggests a general modeling of conditional dependency between random variables. If nothing is known on the causes of a variable, its degree of uncertainty can be measured by the unconditional entropy. Entropy is different from statistical variance by its metric-free nature: It depends only on the probability distribution of a random variable rather than on its concrete values. Examples include the use of information gain in ID3 [28] and C4.5 [29] algorithms for finding the best feature to split a node of a decision tree.

However, most other methods of decision-tree learning, like CART [35] and C4.5 [29], have adopted the post pruning approach (grow a maximal tree and then prune it) for the sake of exploring a larger set of potentially valid patterns. The post pruning methods include cost complexity pruning [35] and pessimistic error pruning [29].

An information-theoretic algorithm for data-driven induction of classification models based on a minimal subset of available features is described in [36]. The relationship between input (predictive) features and the target (classification) attribute is modeled by a tree-like structure termed an information network. The algorithm produces much more compact models than other methods of decision-tree learning while preserving nearly the same level of classification accuracy.

3. Decision trees

The decision tree learning method is one of the methods that are used for classification. As for many other machine learning methods, the learning in decision trees is done by using a data set of already classified instances to build a decision tree. These training objects have a known classification before the building of the tree starts. The set of instances used to "train" the decision tree is called the *training set.*

A decision tree is a model that is both predictive and descriptive. It is called a decision tree because the resulting model is presented in the form of a tree structure. The visual presentation makes the decision tree easy to understand and assimilate. As a result, decision tree are a very popular data mining technique. Decision trees graphically display the relationships found in data. The tree can also be translated into rules such as

If ((Income = High) and (Years_on_job >threshold)) Then Credit_risk = Good

to improve human readability.

Decision trees classify instances by sorting them down the tree [37] from the root to some leaf node, which provides the classification of the instance. Each node in the tree specifies a test of some attribute of the instance, and each branch descending from that node corresponds to one of the possible values for this attribute. An instance is classified by starting from the root node of the tree, testing the attribute specified by this node, then moving down the tree branch corresponding to the value of the attribute in the given example. This process is repeated for the sub tree rooted at the new node.

Decision tree learning has several advantages. One of the advantages is that it gives a graphical representation of the classifier which makes it easier to understand. However, this is not the case for large trees, which tend to get over-complex and difficult to follow. Another advantage is that this method can handle missing attribute values in the training data.

3.1 Basic Concepts

In order to understand how decision trees are used for classification purposes, some basic notions have to be defined.

Decision trees consist of *nodes* and *branches*. The nodes can either be ordinary nodes with other descendant nodes, or *leaf nodes* which do not have descendants. The *root node* is the first node on top of the tree. Figure 10 illustrates these concepts.

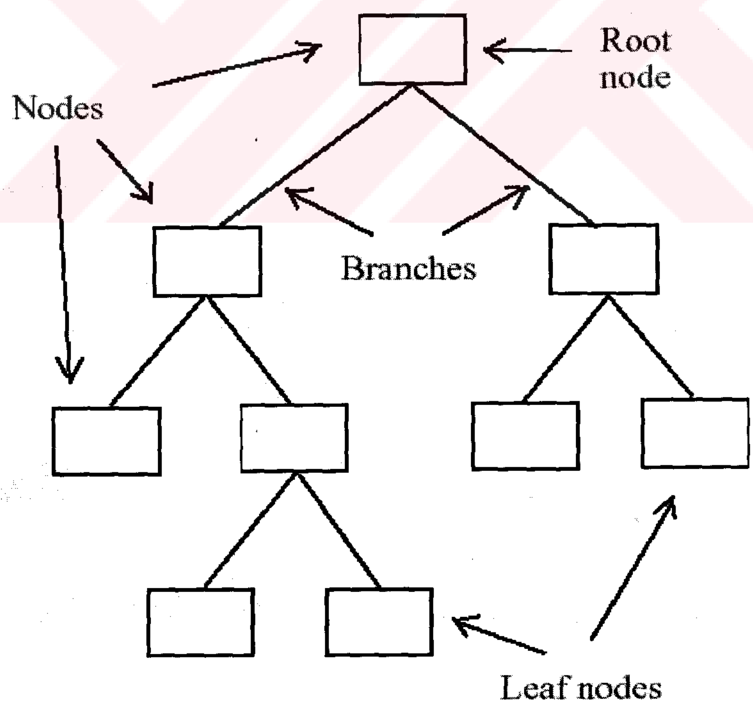


Figure 10. An example tree

Each of the ordinary nodes represents a test on one of the attributes that are measured for each object, and each of the leaf nodes represents a classification of the object. The branches descending from a node corresponds to the possible values for the attribute, and consequently there are as many branches from a node as there are attribute values for the attribute which is tested by the node. If the attribute values are continuous, there are usually two branches descending from the node, corresponding to a division of the objects according to some split $x \le x_i$ and $x > x_i$ where x is the attribute value for the object tested at the node, and x_i is some defined split value for the attribute.

To find the attribute that best separates the training set objects, the ID3 and C4.5 algorithms use a concept that is called *information gain*. Information gain is a measure on how well one can discern the remaining objects according to their classes by using a given attribute as a separator. Associated to the concept of information gain is the *entropy*.

Entropy is a measure of the "impurity" of the set of training objects. The entropy is defined as

$$E(S) = \sum_{i=1}^{c} - p_i \log_2 p_i \qquad (4)$$

where *S* is the set of training objects containing objects from *c* different classes, and p_i is the proportion of objects in the set *S* from class *i*. The entropy is 0 if all objects in *S* belong to the same decision class, and can be as large as $\log_2 c$.

For the special case where the objects are divided between only two decision classes, let us say a positive class and a negative class, the entropy reduces to

$$E(S) = - p_+ \log p_+ - p_- \log p_- \qquad (5)$$

If p_i is zero in formula (4) and (5) result of logarithm will be undefined. So we take zero for $\log_2 0$.

Note that the entropy is 0 if all members of S belong to the same class, for example, if all members are positive, then p - is 0, and

$$\text{Entropy}(S) = -(1 \times \log_2(1)) - (0 \times \log_2(0)) = 0$$

Note the entropy is 1 when the collection contains an equal number of positive and negative examples. If the collection contains unequal number of positive and negative examples, the entropy is between 0 and 1. Figure 11 shows the form of the entropy function relative to a Boolean classification as p+ varies between 0 and 1. Note that entropy is highest when proportion of positive examples is 0.5

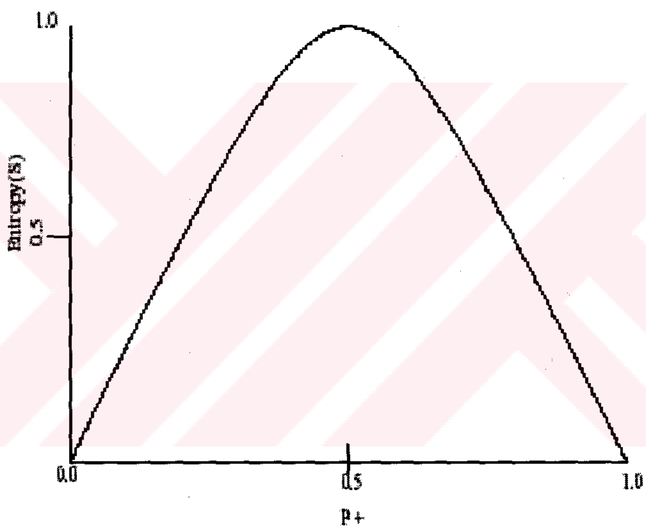


Figure 11. The entropy function relative to a Boolean classification.

3.2 Information Gain Measure

The *information gain* measure is based on the computed entropy for each attribute, and states the expected reduction in *entropy* if the training objects are separated by using the attribute in question. The *information gain* of an attribute *A* relative to a set of objects *S* is defined as

$$G(S,A) = E(S) - \sum_{v \in Values(A)} \frac{|Sv|}{|S|} E(Sv) \quad (6)$$

where Values(A) contains all possible values of the attribute *A* and *Sv* is the set of objects in *S* for which attribute *A* has value *v*. We see from the Equation 6 that the information gain is a measure on how much the entropy is expected to decrease if we partition the training set based on a given attribute. Because of this we choose as the "best" attribute the attribute which gives the highest information gain, since our goal is to decrease the entropy as we split the training objects.

3.3 Decision Tree Algorithms

Most algorithms that have been developed for learning decision trees are variations on a core algorithm that employs a top-down, greedy search through the space of possible decision trees. This approach is exemplified by the CART and ID3 algorithms and ID3's successor C4.5.

CART is an acronym for Classification and Regression Trees. CART uses strictly binary, or two-way, splits that divide each parent node into exactly two child nodes by posing questions with yes/no answers at each decision node. CART searches for questions that split nodes into relatively homogeneous child nodes, such as a group consisting largely of responders, or high credit risks, or people who bought sport-utility vehicles.

Other decision-tree approaches like ID3 [28] and C4.5 [29] use multi-way splits that fragment the data rapidly, making it difficult to detect rules that require broad ranges of data to discover. In the search for patterns in databases it is essential to avoid the trap of overfitting, or finding patterns that apply only to the training data. The testing and selection of the optimal tree are an integral part of the CART algorithm [35]. ID3 and C4.5 (J48) form the primary focus of our discussion. J48 which is Java implementation of C4.5 is the decision tree algorithm (Figure 12) that we use to build our trees in this work.

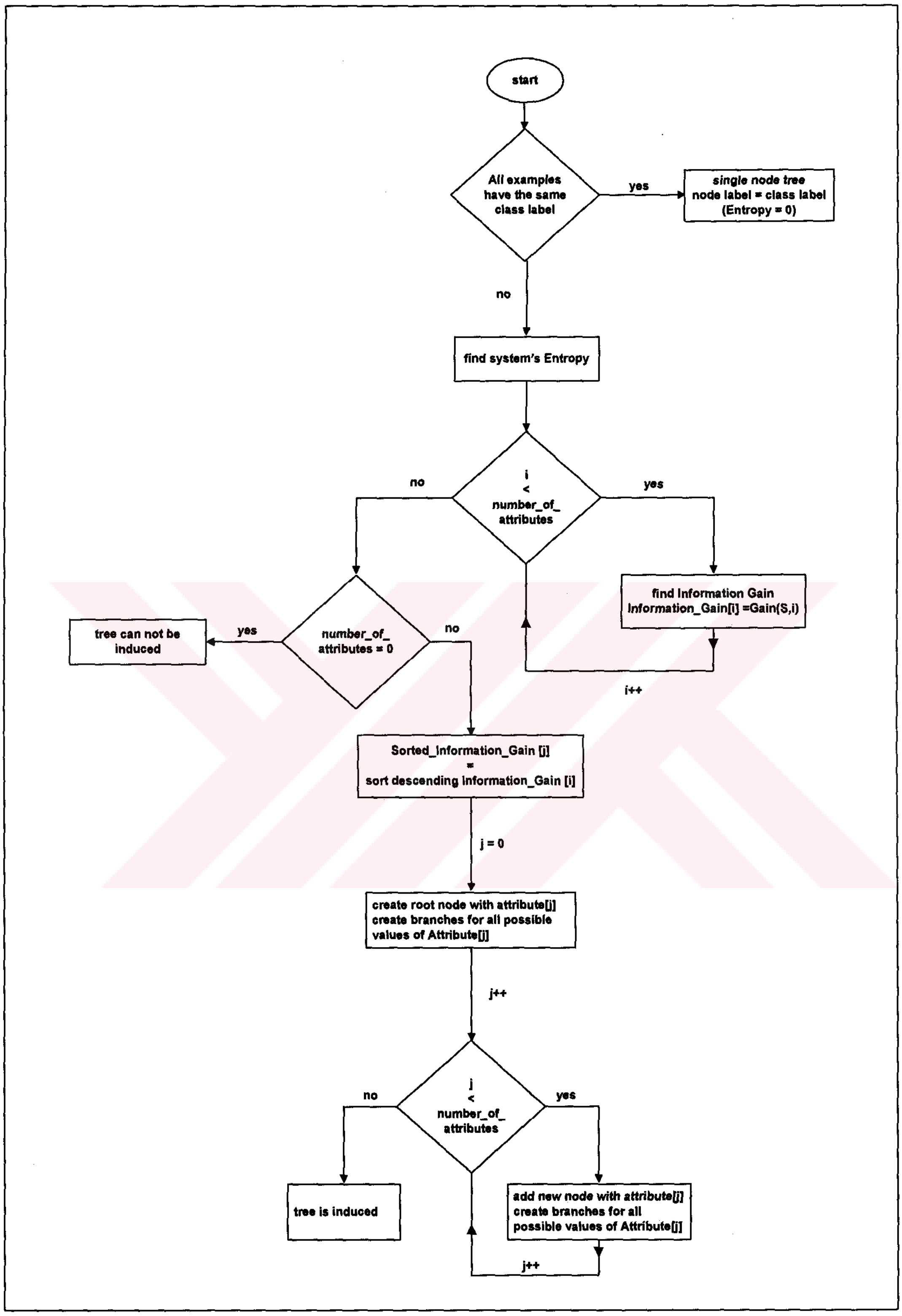


Figure 12. The flow diagram of the J48 (C4.5) algorithm.

3.4 Problems about Decision Trees and Studies on Decision Tree Performance.

Building a decision tree consists of several problems, for instance finding out how to consider which attribute is best for each node, and finding out when to stop building the tree, i.e. how deeply one should grow the tree.

The problem of finding out when to stop growing the tree is connected to a problem called overfitting the data. Overfitting means that if we grow a tree that classifies perfectly the objects in the training set, this tree may not classify other objects too well, because the tree is too specific [38]. The result of growing too big a tree may accordingly be that the tree is too specifically trained to handle exactly the details of the objects in the training set, and poorly trained to handle anything else. The classification of objects outside the training set will consequently be worsened, which is not a desired behavior.

The problem of overfitting is particularly present in the cases where the training objects contain noise or errors. Then a full grown tree will be too sensitive with respect to the noise or errors in the data, and the result is that the tree classifier performs badly on other data sets.

Figure 13 is taken from [39] and illustrates the consequence of overfitting. The results in this figure were obtained by applying the ID3-algorithm to the task of learning which medical patients have a form of diabetes. The numbers along the horizontal axis represents the size of the tree as it is grown. As the figure shows, the accuracy over the training objects increases as the tree grows, because the tree learns gradually more from the training objects as it grows. However, the accuracy on the independent test objects increases from a certain size of the tree, which indicates that when the tree grows larger than this size, it gets too specific in relation to the test data.

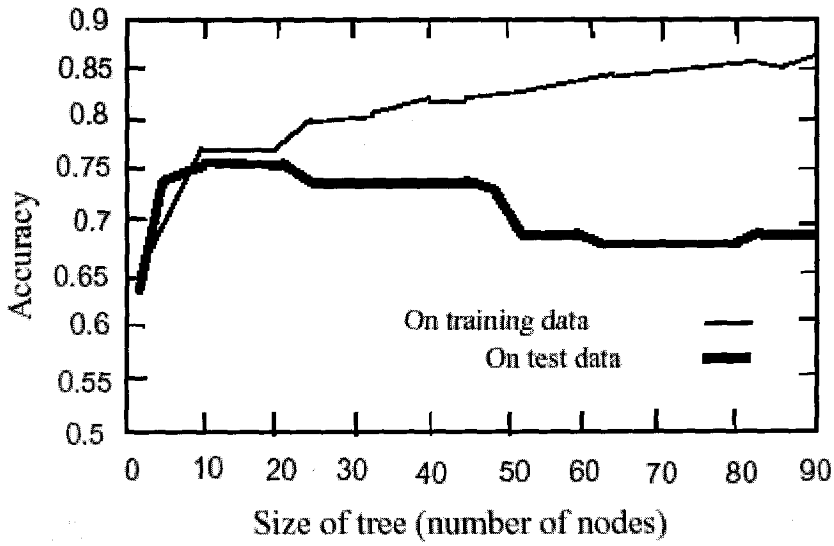


Figure 13. Overfitting in decision tree learning.[39]

The most common methods to avoid overfitting in decision trees are to either stop growing the tree before it is fully grown, or to grow a full tree and then prune it. To prune a node from a decision tree is to remove the subtree of that node, and to make the node a leaf node with decision value the most common value of the training objects connected to that node. In either case the problem is to find a criterion to use as an indication of which sized tree will be the tree to perform best. It is not easy to know exactly when to stop in order to get the best tree. Too large a tree will use too much of the detailed information contained in the training set, while too small a tree may not make use of all the helpful information contained in the training set. There are several approaches to finding the right sized tree. The most common of them is to use a validation set of objects, which should be distinct from the training objects, to evaluate how much post-pruning a node will improve the tree classifier.

In order to decide which nodes to prune, by using the validation set, we can use the concept of *error rate estimation.* The validation set can be run down trees of different sizes, and for each tree one can use the classification results to compute an estimated error rate. For instance, if one first grows a full tree based on the training set, and then prune the full grown tree node by node and run the validation set down the tree each time a node has been pruned, one will get an error rate estimation for each possible tree size. By comparing the computed values, one can see that there are tree sizes that are better than others. The typical results will be that as we prune the full grown tree, the error rate estimate will decrease, until we reach a size where continued pruning will result in increasing error rate estimates.

There are many studies to improve decision tree performance. Some of them are explained below:

In [39], authors extended the existing procedures for the generation of decision trees in order to allow the possibility of partial memberships in the nodes of the decision tree. A fuzzy decision tree has been generated by replacing the Boolean tests in a standard CART decision tree by fuzzy sigmoidal splits. The result of these splits is a function giving the degree of membership of an example in a node of the decision tree. In contrast with a CART tree, where a single leaf is responsible for the prediction, the classification is made by all the terminal nodes of the decision tree using the full set of membership values to the terminal nodes of the tree. The structure of the models allows to construct an elegant backpropagation algorithm. In classification problems, the fuzzy tree has an additional advantage that it is possible to give a measure of the proximity of a given point in the space of attributes and identify those points whose classification is more prone to errors.

An Elegant decision tree algorithm, *Gini* index, has been proposed in [40] which is an improvement over the SLIQ (Supervised Learning In Quest) algorithm. The *gini Index*, the measure of diversity of a tree, was introduced. With the help of the gini index it is decided which attribute is to be split and what is the splitting point of splitting of the attribute. The gini index is minimised at each split, so that the tree becomes less diverse. This algorithm increases the classification accuracy over the original SLIQ algorithm and the Neural Network classification technique.

In [41], an automated and simplified genetic programming (GP) based decision tree modeling technique is introduced. This technique uses two fitness functions: one for minimizing the average weighted cost of misclassification, and one for controlling the size of the decision tree. As compared to other decision tree-based methods GP-based decision trees are more flexible and can allow optimization of performance objectives other than accuracy. It provides a practical solution for building models in the presence of conflicting objectives.

C. TARGET DETECTION AND CLASSIFICATION

One of the important sensor network applications is target classification in battlefields identifying types of moving vehicles in a field. Tactical military software is required to have high reliability. Each software function is often considered mission-critical, and the lives of military personnel often depend on mission success.

We explored previously proposed target detection, classification and tracking methods described below.

In [42], authors used simple spectral features as the input to the neural-network classifier in order to distinguish a cylindrical target from a rock with similar shape. The method in [43] uses the resonant scattering property of underwater objects that is dependent on the object size, shape, structure, and composition. A signal processing scheme called "G-Transform," which consists of three sequential fast Fourier transform (FFT) of the backscattered signals, was developed [44] to represent the resonance or the modulation on the frequency spectrum of the backscattered signal. When used in conjunction with neural networks, the scheme was able to successfully detect and identify targets in shallow water.

In [45], a new wavelet-based classification scheme is developed to discriminate mine-like and nonmine-like objects from the acoustic backscattered signals. A two-layer backpropagation neural network (BPNN) is used for classification using the reduced feature vectors. The system consists of a feature extractor using wavelet packets in conjunction with linear predictive coding (LPC), a feature selection scheme, and a back propagation neural-network classifier. A back propagation neural-network classifier was used to perform the discrimination between targets and nontargets based upon a reduced set of features. The performance analysis of the neural-network classifier demonstrates the robustness and good generalization capability of the system.

The problem of detection of transient signals emitted by underwater objects and classification of these objects based on characteristics of transient signals, is addressed in [46] and a description of a detector is given. An estimator which estimates signal parameters that are suitable for classification is described. These estimated signal parameters are then used to classify a given signal set. The classifier based on "multiple discriminate analysis" is used for classification. Finally, the performances of the detector, the estimator and the classifier are verified using underwater acoustic data.

In [47], moving vehicle target classification performance using data obtained from sensor networks with collaboration both across nodes and within a node in terms of multimodal fusion is presented. [47] shows that single target classification results can be improved by collaboration between sensor nodes under limited data using Multimodal Fusion and Data Sharing techniques. The proposed collaborative classifiers were also applied to a challenging application under interference: classification when multiple targets are simultaneously present in the sensors' field of view.

In [48], CSP (Collaborative Signal Processing) method for target classification and tracking in distributed sensor networks is presented. The framework uses *location-aware data routing* that limits the scope of CSP to relevant subset of nodes conserving network resources, such as energy and bandwidth.

"To enable distributed tracking, the sensor field is divided dynamically into spatial cells. Within each cell, a manager node coordinates CSP tasks.
The approach presented has five basic steps for collaborative detection, classification, and tracking of a target moving through a sensor field.
1) Cells near potential target trajectories are put on alert. Nodes within cells collaborate to determine if a target is present.
2) When a target is detected, the cell becomes active. If classification finds a target of the desired type, tracking is initiated.
3) Tracking includes estimating target location, direction, and speed for predicting future target positions.
4) Based on the predictions, data from the active cell are sent to other cells, alerting them and facilitating CSP.
5) When the target is detected in an alerted cell, that cell becomes active and the process repeats.

CSP methods for target classification and tracking in distributed sensor networks are presented. These algorithms exploit multiple sensing modes gathered at different nodes. Significant savings are possible in power and bandwidth consumption by processing time series locally. Significant information can be distilled from the time series. Location-aware routing limits data distribution to regions directly affected by the data.. Further research is needed to determine the operational limitations of these approaches. As CSP techniques often rely on prior statistical information about the signals, an overriding challenge is to make CSP algorithms robust and/or adaptive to variations in environmental conditions that can significantly influence statistical signal characteristics. For example, the presence of a strong wind can radically influence acoustic measurements. Similarly, different vehicle operating conditions, such as gearshifts and acceleration, must also be taken into account. Finally, the effect of Doppler shifts can also be quite pronounced in acoustic and seismic measurements due to the relatively slow speed of wave propagation in such modalities. The choice between decision versus data fusion depends on the statistical correlation between different measurements. Thus, algorithms for determining the subset of nodes for data versus decision fusion could significantly enhance the efficiency of CSP algorithms. One simple approach may be based on the observation that feature vectors from different nodes provide snapshots of the target signal at different times. Thus, nodes in close proximity will be highly correlated, whereas sufficiently spaced nodes will be weakly correlated. A simple measure of the degree of correlation between nodes could be derived from the knowledge of the bandwidth of the target signal and the location of the nodes relative to the target (e.g., a stationary stochastic signal decorrelates after a time interval inversely proportional to its bandwidth)." [48]

Single-target classification algorithms can be extended to deal with multiple targets. A key problem is the interference between signals from different targets. In a multiple target classifier, each component classifier for a particular target class must also suppress interference from targets from other classes.

A new adaptive underwater target classification system to cope with environmental changes in acoustic backscattered data from targets and nontargets is introduced in [49]. A K-nearest neighbor (K -NN) system is used as a memory to provide the closest matches of an unknown pattern in the feature space. The classification decision is done by a back propagation neural network (BPNN). This scheme offers a very promising tool for underwater target classification in shallow water and under changing environmental conditions. It must be emphasized that the k-NN memory clearly has a significant impact on the performance of the entire adaptive system.

Classifying and tracking targets in sensor networks discussed in [50]. A distributed 2-dimensional sensor network based surveillance system using inexpensive sensor nodes is designed. Collaborative signal processing enables the system to simultaneously achieve better sensitivity and noise rejection, by averaging across time and space. Authors introduced a spatial statistic called the influence field, realize an estimator using a binary sensor field, and use it as the basis for a new type of classifier. Approach complements and improves upon existing unattended battlefield ground sensors by replacing the typically expensive, hand-emplaced, sparsely-deployed, non-networked, and transmit-only sensors with integrated collaborative sensing, computing, and communicating nodes. This will enable military forces to blanket a battlefield with easily deployable and low-cost sensors. Implementation provides a tunable level of classification quality based on the reliability of the network. The notion of an influence field is the spatial region surrounding an object in which the object causes fluctuations in one or more of the six energy domains. In other words, the influence field is the region surrounding the object in which the object can be sensed using some specific modality.

Although influence fields have been used in tracking, [50] first use it as the basis for classification. Influence field based classifier is highly distributed network. Each node can send out as little as one bit of information about the presence or absence of a target in its sensing range and only requires local detection and estimation, but no complex time-frequency domain signal processing. Magnetic, thermal, acoustic, chemical, electric, seismic, optical and ultrasonic microsensors are used for classify unarmed person, armed soldier, and vehicles.

“The classifier collects data received from the network and partitions it into windows of global time. Once the incoming data has been partitioned into windows based on global time, the classifier counts the number of nodes that have detected the presence of an target in that window. The classifier has to maintain a history of nodes that have started detecting an event but have not yet stopped detecting it. The classifier carries forward the count of such *active* motes from one window to the next. All detection events in a classification window need not belong to the same target. For instance, if multiple targets simultaneously in the network, each target will be detected by the nodes in the region surrounding it. The classifier distinguishes multiple targets and does not combine these simultaneous detections into a single larger target. Wind and other sources of noise can cause nodes to report false detections. The classifier identifies such outliers that could skew the classification. Consequently, the classifier uses localization information about the reporting nodes and knowledge of the target motion and phenomenological models. For example, if the classifier receives only two detection events and the influence field for the smallest target type, the soldier, is expected to be between 4 and 9 for the given density considering 50% network reliability, the classifier identifies these nodes as outliers and does not generate a classification. If, on the other hand, the influence field for a soldier is 9 while that for a car is 36, and if 4 soldiers walk through the network at the same time such that they are at sufficient distance from one another, the classifier identifies that the corresponding events belong to different targets and accurately classiffies the targets as 4 soldiers rather than a single car. Note that the data association problem is automatically addresses because of the fine-grained spatial locality of the detections. The output of the classifier at the end of each classification window is one or more classification decisions along with the supporting evidence, in the form of a set or sets of nodes, that are associated with the given target.” [50]. The estimator performance varies for different target classes and network reliability levels.

A data classification method which integrates attribute-oriented induction and the induction of decision trees is addressed in [51]. There are two algorithms, *MedGen* and *MedGenAdjust* proposed. Both MedGen and MedGenAdjust induce decision tree classifiers from the generalized relevant data. While MedGen directly applies a decision tree algorithm to the generalized data, MedGenAdjust allows for dynamic adjustment between different levels of abstraction during the tree building process. Each algorithm first generalizes the data to an intermediate abstraction level by attribute-oriented induction. Generalization of the training data and the use of a multidimensional data cube make the proposed algorithms scalable and efficient since the algorithms can operate on a smaller, compressed relation. The generalized data are stored in a multi-dimensional data cube, similar to that used in data warehousing. This is a multi-dimensional array structure. Each dimension represents a generalized attribute, and each cell stores the value of some aggregate attribute.

A tree-based modeling method for identifying fault-prone software modules, which has been applied to a subsystem of the Joint Surveillance Target Attach Radar System, JSTARS, a large tactical military system is presented in [52]. JSTARS performs real time detection, location, classification, and tracking of moving and fixed objects on the ground.

Much work has already been performed to implement target tracking algorithms in WSNs. Some algorithms which have been used previously are: Kalman filtering, extended Kalman filtering, and Particle filtering. Tracking algorithms need to be robust enough to track a target accurately even after node fallout occurs.

In [53], A scheme to track correlations in wireless microsensor networks is proposed. It uses a linear prediction model to find the initial correlation coefficients and uses a discrete Kalman filter model to track the correlations. The proposed scheme fits well to sensor networks due to its reduction on average per-frame energy consumption and high accuracy compared to other approaches.

[54] demonstrates the wireless sensor network based target tracking algorithm.

"When the sensing information exceeds a threshold, each sensor who is marked 'on' at the time and has range to detect the entity will vote (give a probability to the cluster head) on whether or not the entity is a threat. The cluster head gathers data from its surrounding nodes and fuses the information together to determine if the sensed object is a threat. If the object is indeed a threat, the WSN will classify the object by correlating its signature with previously known signatures. Then, the WSN will use an algorithm to track the object's movement in the monitored area. As the object moves outside of the initial cluster's range, a neighboring cluster will take on the responsibility of tracking the object's movement. Throughout this process, as the nodes are processing information and communicating with the cluster head, the cluster head is receiving information from its nodes, fusing the data, and sending it on to a neighboring cluster head to warn of the object's approach. The cluster head must also communicate with a data center which monitors all data received from the WSN. This process requires a lot of energy, which the sensors need to conserve for longer lifetimes. Therefore, in addition to sensing, communicating, and processing, the nodes must also be able to hibernate for a period of time, but be woken up when a local disturbance occurs. The goal of the algorithm used in WSNs is to track a target accurately but simply, without using a lot of power for computation and communication. A Kalman filter estimates the state of the system based on the system input, measurement of the system output, and the relation between the two. The Kalman filter is a linear model while the Extended Kalman filter is a non-linear model also used often in target tracking algorithms." [54]

III. DATA MINING BASED TARGET DETECTION and CLASSIFICATION

Tactical underwater surveillance systems have three major tasks: detection, classification and tracking submerged targets. This thesis focus on detection and classification of targets by using microsensors. Detection is to recognize the presence of a target in the proximity, and classification is to identify the target type.

A. TARGETS

The selection of sensors is an important task in the design of sensor networks. Choosing the right set of sensors for the job can improve dramatically the system's performance, lower its cost, and improve its lifetime. Firstly we need to know about targets and their specifications to decide which types of microsensors we can use. Our modal has four target classes, these are diver, mine, SDV and submarine. These targets and which sensor types detect them is described here briefly;

- **Diver.** It is expected that divers use special clothes to prevent the transfer of body heat in cold water. So it is hard to detect a diver with a thermal microsensor. Divers move and make waves underwater. Mechanical or acoustic microsensors can be used to sense these waves. Divers use diver tubes and other equipment that contains metal. Therefore magnetic microsensors can also be practical to detect divers.

- **Mine.** Mines are explosives planted under water to destruct ships and submarines. Most of them have metal components. We can only use magnetic microsensors to detect mines since they are stable and their heat is the same as the heat of water around them.

• **Submarine.** When submarines move, mechanical and acoustic microsensors can sense them. We can also expect large magnetic field around them because they are huge and made up of metal.

• **SDV (Small delivery vehicle).** SDVs are used for underwater personnel and diver transportation. They have almost the same characteristics as the submarines except for they are smaller. We can use all type of microsensors that we use to detect submarines also to detect SDVs.

B. SENSOR SELECTION

In Table 4 the microsensors that can be used to detect a given type of target are shown by a plus (+) sign. Please note that we use high number of randomly deployed sensor nodes. After deployment, our nodes adjust their depth such that the best coverage of the 3D sensor field is provided [2]. Therefore we assume that there is at least one microsensor group, i.e., every sensor node has several types of sensors and all of the sensors of a node are at the same depth, within the detection range of every point in the 3D sensor space.

Table 4.Sensor selection criteria.

		Target Type				
		Diver	**Mine**	**Submarine**	**SDV**	**Sea Animal**
Sensor Type	**Radiation**	+	+	+	+	+
	Mechanical	+	-	+	+	+
	Magnetic	+	+	+	+	-
	Acoustic	+	-	+	+	+

In our tactical underwater sensor network application we use mechanical and radiation microsensors to detect the *proximity* of the targets. Magnetic and acoustic microsensors are used for the *classification* of the detected target into one of the target classes as shown in Table 5.

Table 5. Sensors used for detection and classification.

Activity	**Sensor type**
Proximity detection	Radiation
	Mechanical
Classification	Magnetic
	Acoustic

Various types of **radiation microsensors** and their characteristics are shown in Table 6. Radiation sensors are non-conducting, because they detect emission. They can be classified in three categories: Ultraviolet (UV) wavelength is from 0.002 to 0.4 μm, Visible (Vis) wavelength is from 0.4 to 0.7 μm and Infrared (IR) wavelength is from 0.7 to 500 μm. Since IR thermal changes can be used to detect proximity we can use radiation microsensors as proximity sensors. Proximity detection sensors that we use in our model are photo-transistors (IR), photo-diodes (IR) and photo-conductive CdS photocells (Vis). Since the cost of photo-multiplier tubes and other photo-conductors can be as high as $100, and photo-multiplier-tubes are mechanically fragile, they do not fit the sensor networks. The cost of a photo-transistor or a photo-diode sensor is about 25 cents. A photo-conductive CdS is even cheaper, i.e., about 10 cents. Their operating temperature is between -55 to +100 °C. The size of a photo-transistor or a photo-diode is around 5 cm., and the size of a photo-conductive CdS cell is 0.5 cm. The characteristics of radiation microsensors are in Table 6.

The detection of a target that does not have thermal radiation such as mine would be difficult, so we can use visible light and NIR photo-transistors to detect the proximity of targets. Since this type of targets lie in visible area (about 30 meters depth from the sea surface) we can use these microsensors to detect the proximity of targets that do not have thermal radiation.

Table 6. Radiation microsensors [55].

		Photo-multiplier tubes	Photo-diodes	Photo-Transistors	Photo-conductive CdS	Other Photo-conductors
Electrical Characteristics	**Available wavelengths (μm)**	0.2 – 0.9	0.2 – 2.0	0.4 – 1.1	0.4 – 0.7	2 - 15
	Sensitivity	very high	high	high	high	high
	Cost	~ $100	~ 25 cents	~ 25 cents	~ 10 cents	~ $100
	Physical Size	> 20 cm	~ 5 cm	~ 5 cm	~ 0.5 cm	~ 5 – 10 cm

For proximity detection we can also use **mechanical microsensors** such as microswitches (touch sensor) shown in Table 7.

Table 7. Microswitches [56].

Mechanical Life	**Cost**	**Temp. Range**	**Operating Limits**	
			Operating force (max.)	**Release force (min.)**
$>10^5$ operations	< $1	0 °C to 70 °C	4.2 N	1.7 N

Since divers use specially designed clothes which prevents heat transfer and also the heat of a mine is equalized with the heat of water, it is hard to detect a diver or a mine with a thermal microsensor. So we use other types of microsensors to classify targets. One of these is **magnetic microsensors**. Magnetic microsensors are used to control underwater traffic[57], monitor magnetic properties of the volcanic rock of the seafloor [58] and find underwater wrecks [59]. We can use magnetic hall effect device which consists of attached wire to each side of a thin square or rectangular plate. The plate is often fixed to a ceramic substrate that provides mechanical support, thermal stability. Characteristics of hall device are shown in Table 8. Divers wear special clothes, use diver tube and carry special metallic weights. Mines have metallic components. SDVs and submarines also contain lots of metal. All of these cause difference in magnetic field that can be detected by a hall device.

Table 8. Magnetic microsensor [60].

	Size	Range	Resolution	Accuracy	Cost
Hall Device	0.13 mm diameter 0.25 mm thick	-/+ 100 mT	1mG(100nT)	-/+ 0.1 %	~ $1

Acoustic sensor is another type of sensor that can be used in underwater target detection. In acoustic detection we can use a SAW device given in Table 9. A SAW device has a polished surface which produces an electrical wave when an acoustic wave hits its surface.

Table 9. Acoustic microsensor [61].

	Temperature Range	Size	Cost
SAW (Surface Acoustic Wave Device)	-35 to 85°C	3.5 x 3.5 x 1.0 (mm^3)	~ $1

C. SENSOR MEASUREMENTS

We studied each target type in order to define the possible values that sensor measure when the target is within the nodes sensing range.

Figure 14 shows the values that magnetic sensors would measure for each target type. Since sea animals do not have any magnetic piece, sensors would not detect any magnetic difference in the water within 0 to 10 meters as shown with a stable line, starting from level *none* (0) and continues with the same value, i.e. with a zero slope.

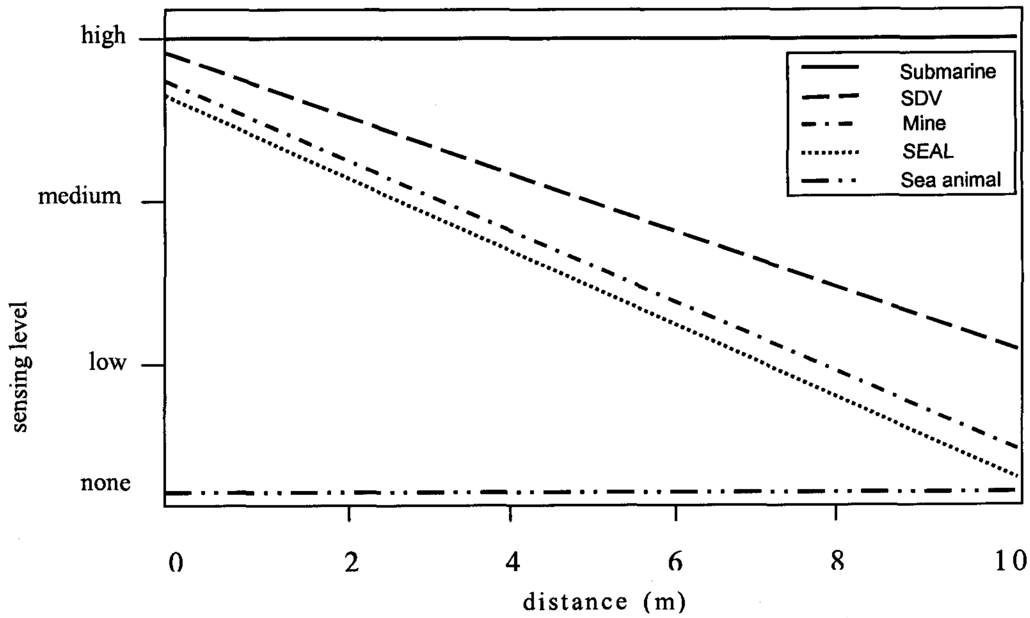


*Figure 14. **Magnetic** values for each target type.*

A different case is valid for SDV. Sensors would measure a *low* value when the target is 10 m. away from the sensor, the measurement value would increase linearly as the target gets closer to the sensor and *high* thermal value would be measured eventually, i.e. when the target is near the sensor, magnetic value would be *high* for the submarine for all differences and would change between *none* and *medium* for diver.

Expected acoustic detection values are shown in Figure 15.

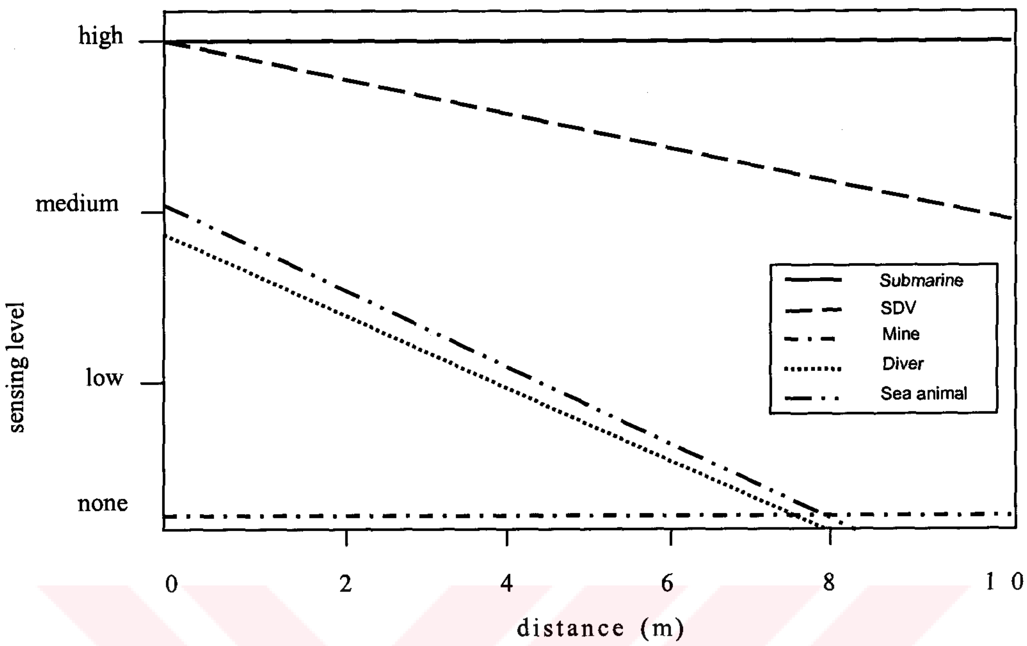


*Figure 15. **Acoustic** values for each target type.*

If we want to show the expected values measured by magnetic and acoustic sensors for each target type in different figures, we conclude by having Figure 16 for diver, Figure 17 for mine, Figure 18 for SDV, Figure 19 for submarine and Figure 20 for sea animals.

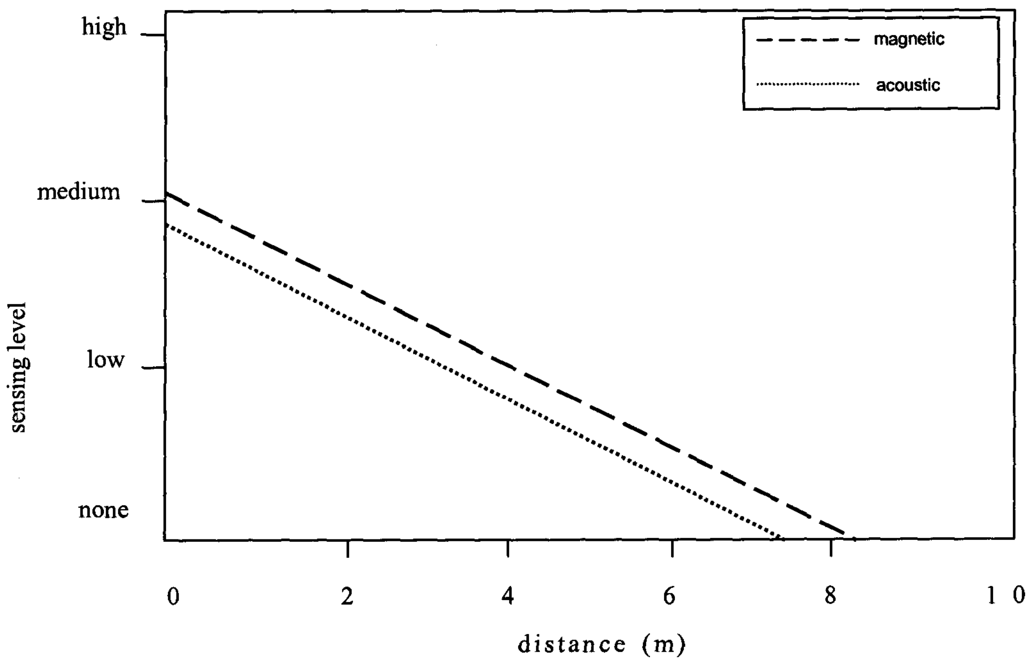


Figure 16. Sensor detection values for ***diver****.*

For each sensor type (magnetic, acoustic) for classification; difference made by a diver in the environment is shown in Figure 16. For example in 1 meter magnetic and acoustic sensors detect medium level difference in the environment. When distance is increase to 4 meters three of the sensors get low level difference. We got the sensing levels in every 2 meters and used them as tuples in training set;

In order to generate training data set we need to have tuples in format (magnetic value, acoustic value, object class).

From Figure 16 we generate the following tuples in order to use in creation of data set.

(medium, medium, Diver) in 0 meter

(medium, medium, Diver) in 2 meters

(low, low, Diver) in 4 meters

(low, low, Diver) in 6 meters

When we look at Table 10, we see that those tuples are inserted to the data set as 1, 6, 11, 16 th. lines respectively.

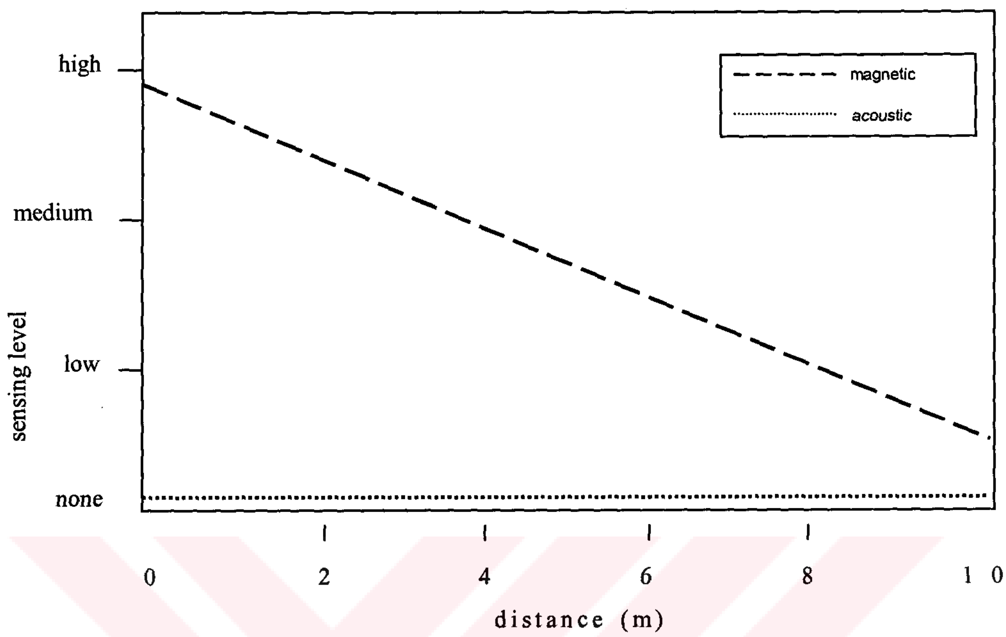


Figure 17. Sensor detection values for ***mine****.*

As explained above mines have metal components, we can only use magnetic microsensors to detect mines. In Figure 17 we can see mine's magnetic value is in high level and decreases with the distance. Lines 2, 7, 12, 17, 21, 24 of data set are generated by using Figure 17 and inserted to Table 10.

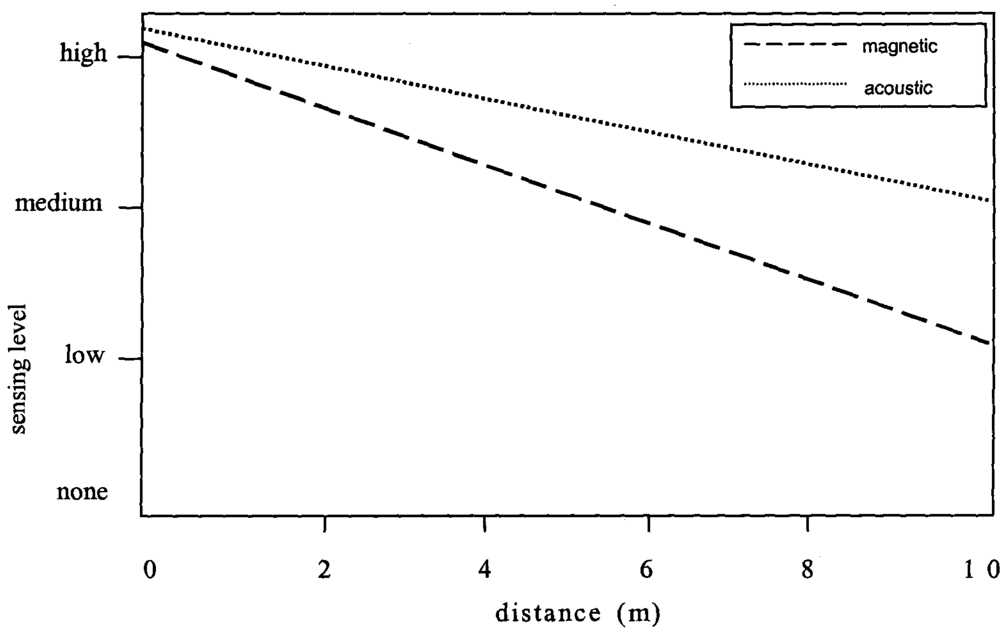


Figure 18. Sensor detection values for ***SDV****.*

Sensing levels of microsensors shown in Figure 18 for SDV. Lines 3, 8, 13, 18, 22, 25 of data set are generated by using Figure 18 and inserted to Table 10.

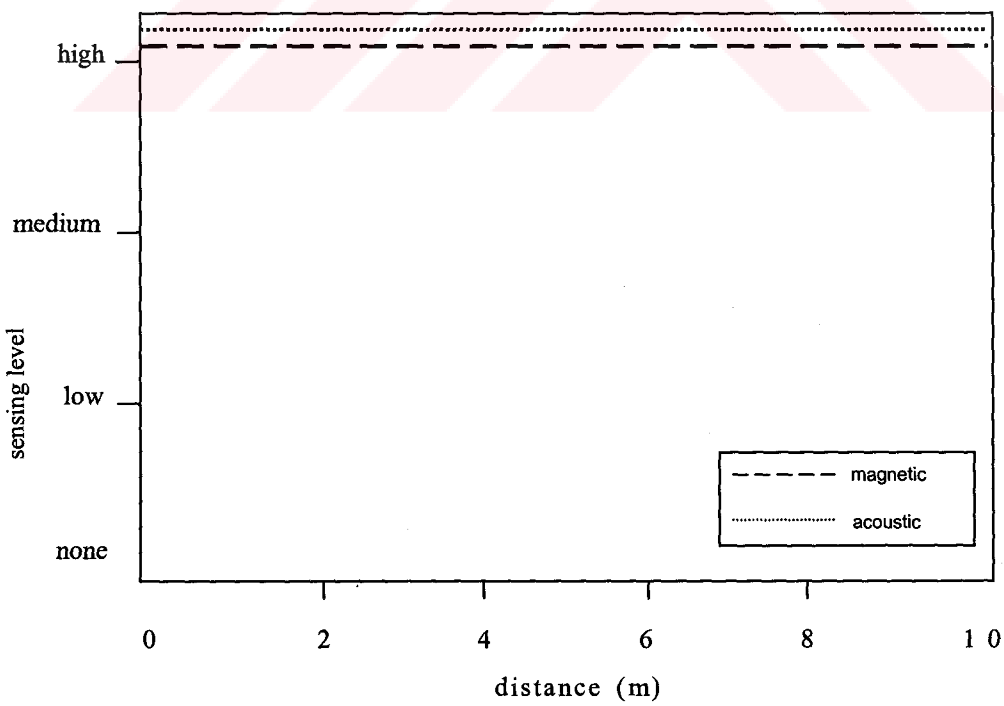


Figure 19. Sensor detection values for ***submarine****.*

Sensing levels of microsensors shown in Figure 19 for submarine. Lines 4, 9, 14, 19, 23, 26 of data set are generated by using Figure 19 and inserted to Table 10.

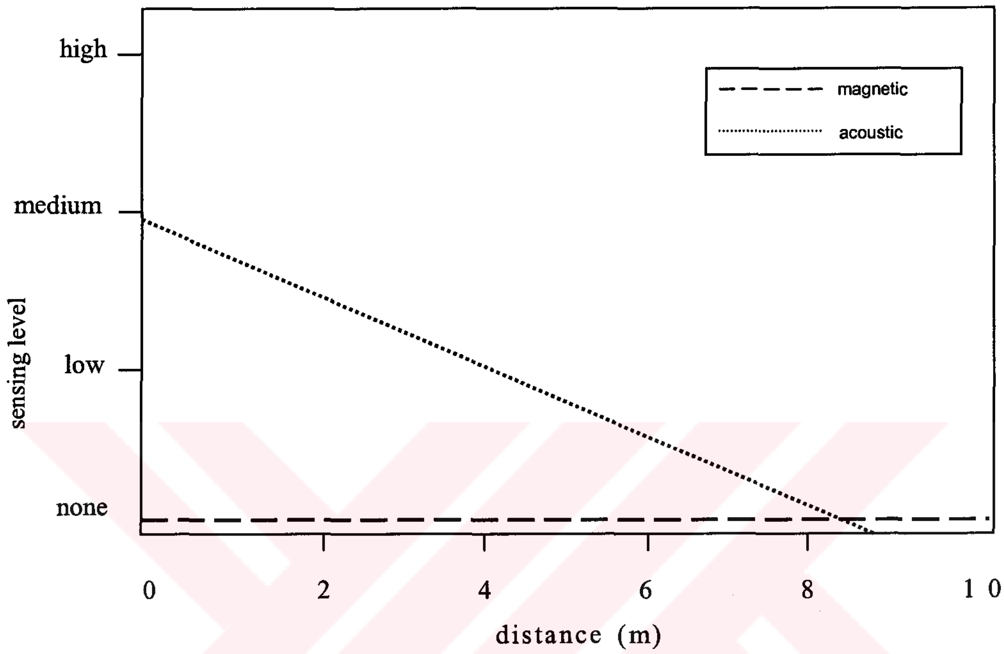


Figure 20. Sensor detection values for ***sea animal.***

If we try to classify an object that means we have proximity report, but it might not be a target, it can be a sea animal. For these type of objects difference levels are shown in Figure 20. Lines 5, 10, 15, 20 of data set are generated by using Figure 20 and inserted to Table 10.

D. TRAINING DATA SET

The attributes for our application are shown in Table 10 where the difference that they cause in the ambient thermal, magnetic and acoustic conditions of their vicinity. This is our training set (tuples). We can also use numerical data as our training data. As also shown in Table 4 mines don't have thermal and acoustic emissions, divers don't have thermal emission, and sea animals don't have magnetic emission.

Table 10. Training data set.

	Magnetic	Acoustic	Object Class
	near the node (0 meter)		
1	high	Medium	Diver
2	high	None	Mine
3	high	High	SDV
4	high	High	Submarine
5	none	Low	Sea animal
	distance of the target to the nodes is 2 meters		
6	high	Medium	Diver
7	high	None	Mine
8	high	High	SDV
9	high	High	Submarine
10	none	Low	Sea animal
	distance of the target to the nodes is 4 meters		
11	medium	Low	Diver
12	medium	None	Mine
13	medium	High	SDV
14	high	High	Submarine
15	none	Low	Sea animal
	distance of the target to the nodes is 6 meters		
16	medium	Low	Diver
17	medium	None	Mine
18	medium	High	SDV
19	high	High	Submarine
20	none	Low	Sea animal
	distance of the target to the nodes is 8 meters		
21	low	None	Mine
22	low	Medium	SDV
23	high	High	Submarine
	distance of the target to the nodes is 10 meters		
24	low	None	Mine
25	low	Medium	SDV
26	high	High	Submarine

There are some classification methods such as naïve bayes, neural networks and decision trees. We use decision tree because it is advantageous than other classification techniques for fast execution time, ease in the interpretation of the rules, and scalability for large multi-dimensional data sets [7-10]. Utilizing collaborative computing we make classification based data mining, and use a decision tree to detect and classify targets, i.e., submarines, SDVs, divers and mines.

Well known, top-down, greedy search algorithms for building decision trees are ID3 (Induction of Decision Tree) [28] and C4.5 [29] for inducing *Decision Trees* from data. These algorithms use entropy and information gain metrics to induce decision tree. Entropy and information gain are calculated for all attributes and the attribute with the greatest information gain is selected for the branch. Our scheme uses J48 algorithm [62] which is an implementation of the C4.5 algorithm [29]. Algorithm reduce overfitting by reducing the size of decision tree by pruning.

J48 algorithm needs training data to induce decision tree. For our scheme some assumptions are made by using target specifications explained above and sensor selection criteria in Table 4. We studied each target type in order to define the possible values that a sensor measure when the target is within the nodes sensing range and than we prepared the training data set for our classification mining based detection algorithm. The attributes for our application are shown in Table 10 where the difference that they cause in the ambient magnetic and acoustic conditions of their vicinity. This is our training data set (tuples).

E. CLASSIFICATION

1. Finding attribute which is best classifier

The terms entropy and information gain is explained in Section II. To choose the classifier attribute firstly we need to calculate *entropy* of the system.

Entropy S of the system can be calculated as;

diver (d) = 4 , mine (m) = 6 , SDV (s) = 6 , submarine (sub) = 6 ,

sea animal (sa) = 4

number of total data (total) = d + m + s + sub + sa

$$E = -(\frac{d}{total} \times \log_2 \frac{d}{total}) - (\frac{m}{total} \times \log_2 \frac{m}{total}) - (\frac{s}{total} \times \log_2 \frac{s}{total})$$
$$-(\frac{sub}{total} \times \log_2 \frac{sub}{total}) - (\frac{sa}{total} \times \log_2 \frac{sa}{total})$$

$E = -(4/26)\log_2(2/26) - (6/26)\log_2(6/26) - (6/26)\log_2(6/26) - (6/26)\log_2(6/26) - (4/26)\log_2(4/26)$
$= 2.29$

Information gain measures the reduction in entopy. We choose as the "best" attribute the attribute which gives the highest information gain, since our goal is to decrease the entropy as we split the training objects.

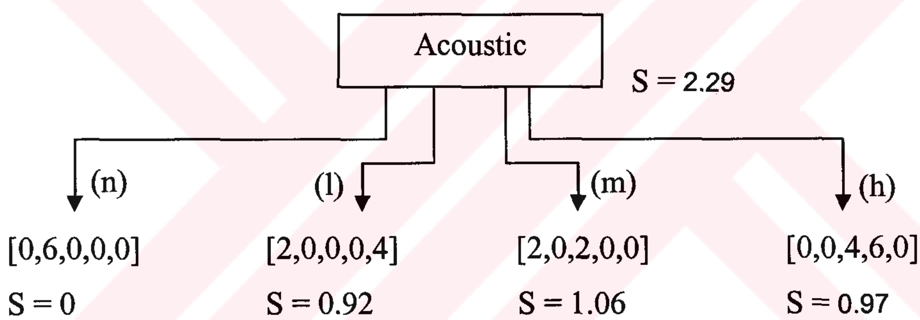


Figure 21. Entropy of Acoustic for each split.

Values(Acoustic) = none, low, medium, high
S = [4 diver, 6 mine, 6 SDV, 6 submarine, 4 sea animal]
$S_{acoustic=none} \leftarrow [0,6,0,0,0]$
$S_{acoustic=low} \leftarrow [2,0,0,0,4]$
$S_{acoustic=medium} \leftarrow [2,0,2,0,0]$
$S_{acoustic=high} \leftarrow [0,0,4,6,0]$

number of 'none' value for acoustic 'an' is 6
number of 'low' value for acoustic 'al' is 6

number of ‘medium’ value for acoustic ‘am’ is 4

number of ‘high’ value for acoustic ‘ah’ is 10

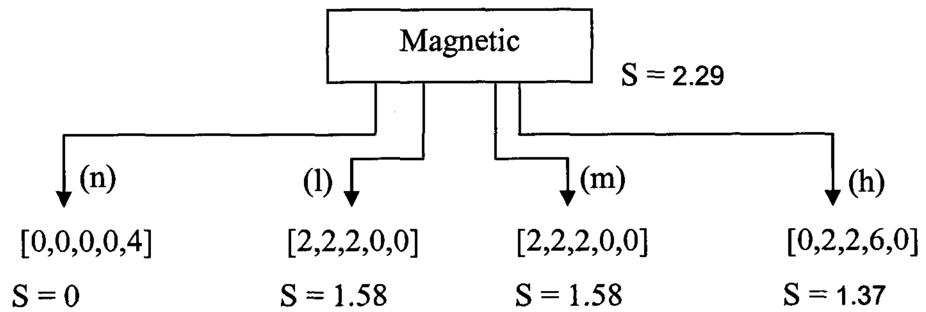


Figure 22. Entropy of Magnetic for each split

Values(Magnetic) = none, low, medium, high

S = [4 diver, 6 mine, 6 SDV, 6 submarine, 4 sea animal]

$S_{magnetic=none} \leftarrow [0,0,0,0,4]$

$S_{magnetic=low} \leftarrow [2,2,2,0,0]$

$S_{magnetic=medium} \leftarrow [2,2,2,0,0]$

$S_{magnetic=high} \leftarrow [0,2,2,6,0]$

number of ‘none’ value for magnetic ‘mn’ is 4

number of ‘low’ value for magnetic ‘ml’ is 6

number of ‘medium’ value for magnetic ‘mm’ is 6

number of ‘high’ value for magnetic ‘mh’ is 10

The information gain due to sorting the original 26 examples by the attribute acoustic may be calculated as

$$\text{Gain}(S,A) = \text{Entropy}(S) - \left(\frac{an}{total} \times \text{Entropy}(S_{Acoustic=none})\right) - \left(\frac{al}{total} \times \text{Entropy}(S_{Acoustic=low})\right)$$

$$-\left(\frac{am}{total} \times \text{Entropy}(S_{Acoustic=medium})\right) - \left(\frac{ah}{total} \times \text{Entropy}(S_{Acoustic=high})\right)$$

Gain (S,Acoustic)

= 2.29 – (6/26) x 0 – (6/26) x 0.92 – (4/26) x 1.06 – (10/26) x 0.97

= 1.55

The information gain of attribute magnetic may be calculated as

$$\mathrm{Gain}(S,A) = \mathrm{Entropy}(S) - \left(\frac{mn}{total} \times \mathrm{Entropy}(S_{Magnetic=none})\right) - \left(\frac{ml}{total} \times \mathrm{Entropy}(S_{Magnetic=low})\right)$$
$$- \left(\frac{mm}{total} \times \mathrm{Entropy}(S_{Magnetic=medium})\right) - \left(\frac{mh}{total} \times \mathrm{Entropy}(S_{Magnetic=high})\right)$$

Gain (S,Magnetic)

= 2.29 – (4/26) x 0 – (6/26) x 1.58 – (6/26) x 1.58 – (10/26) x 1.37

= 1.04

information gain is used by C4.5 to select the best attribute at the each step in growing the tree. The use of information gain to evaluate the relevance of attributes is summarized in Figure 21 and Figure 22. In these Figures the information gain of two different attributes, Acoustic and Magnetic, is computed in order to determine which is the better attribute for classifying the training examples.

Acoustic provides greater information gain than *magnetic,* so tree starts with the attribute *acoustic.*

2. Decision Tree for Target Classification

When we apply our training data set which has 26 tuples to classify an object into one of five different classes to J48 (java implementation of C4.5) decision tree algorithm in WEKA (Waikato Environment for Knowledge Analysis) package [62] we get the pruned tree in Figure 23.

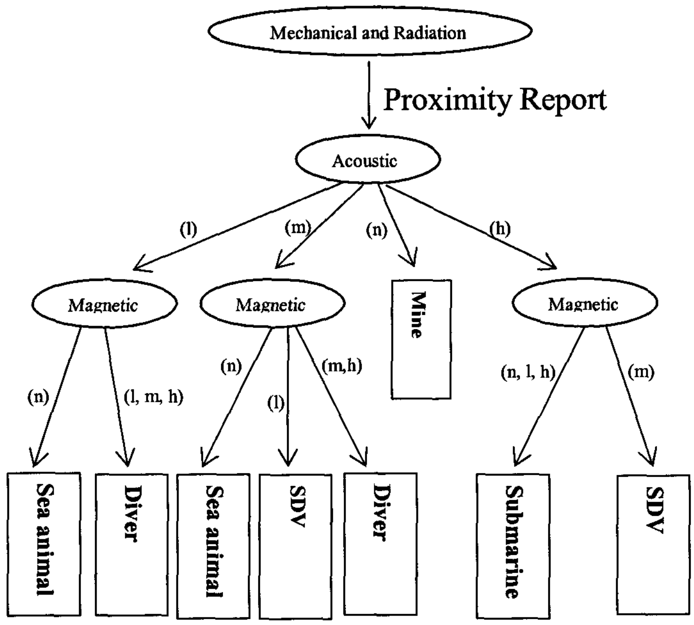


Figure 23. The decision tree for the target classification (n : none, l : low, m : medium, h : high)

J48 [62] calculates the information gain metric for all attributes and choose the highest one for the node. It continues recursively to the end of the tree. Please note that the first node after detection by a mechanical or radiation sensor is acoustic. For our training data set acoustic has the highest information gain value so tree for the classification starts with the acoustic attribute. Mine can be detected with just acoustic microsensor after detection of proximity with the radiation and mechanical microsensors. If we have proximity report and the acoustic value is in 'none' level we can say that it is 'mine'.

J48 algorithm gives the confusion matrix for our case as follows:

Table 11. The confusion matrix for the target classification.

		Target Classified As				
		Diver	**Mine**	**SDV**	**Submarine**	**Sea animal**
Target	**Diver**	4				
	Mine		6			
	SDV			4	2	
	Submarine				6	
	Sea animal					4

As shown in Table 11 all targets except SDVs are correctly classified. 2 of 6 SDVs are classified as submarine because SDVs and submarines make the same magnetic and acoustic difference in close distance. As you can see in Table 10 tuple 3 for SDV and tuple 4 for submarine is the same. So there can be misclassification. We have used 26 data in classification, 24 of objects are correctly classified, 2 of the objects are incorrectly classified.

2. The Classification Probabilities

We have two metrics for classification. Our first classification metric is the classification probability P_{ik} that the target type i is classified as the target type k. In Table 12 we show the classification probabilities for all possible target type pairs in our target domain. For example the probability that an SDV is classified as an SDV is 66%, and an SDV is classified as a submarine is 34%. These probabilities are derived by WEKA [62] based on our training data and confusion matrix.

Table 12. The classification probabilities.

		Target Classified As				
		Diver	**Mine**	**SDV**	**Submarine**	**Sea animal**
Target	**Diver**	100 %				
	Mine		100 %			
	SDV			66 %	34 %	
	Submarine				100 %	
	Sea animal					100 %

4. Precisions for Targets

Another metric is precision ε_i, which is the probability that a target classified as type *i* is of type *i*. Precision ε_i values for our application is shown in Table 13. Precision for a mine is 1. This means if a target is classified as mine, the probability that the target is a mine is 1. This is different from the values in Table 12 which shows the probability that a given type of target is classified as a certain type. For example the probability that an SDV is classified as a submarine is 0.34. However, when a target is classified as an SDV, we are sure that it is an SDV. Precision for a submarine is 0.75 because if an object classified as a submarine it is % 75 (100/134) a submarine and 25 % (34/134) an SDV. The precision for other types of targets is 1.

Table 13. The precision table for targets.

Class	**Precision**
Diver	1
Mine	1
SDV	1
Submarine	0.75
Sea animal	1

If the nodes are time synchronized and location aware, proximity reports can be used to derive the speed and the vectoral displacement of a target which may increase the precision of the target classification. The vectoral displacement parameter indicates the variations in the speed and the movement direction of a tracked target. For this a node that reports a proximity attaches also the location and time tags to the proximity report. The measurements made by magnetic and acoustic sensors are used for data association which is to associate the data reported by multiple sensors to a specific target. When we have a time series about the location of a tracked target, then its speed and vectoral displacement can easily be computed. Since we focus on the basic detection and classification scheme in this paper, we do not explain the details about our solution for speed and vectoral displacement calculation.

IV. PERFORMANCE EVALUATION

A. SIMULATION TOOL

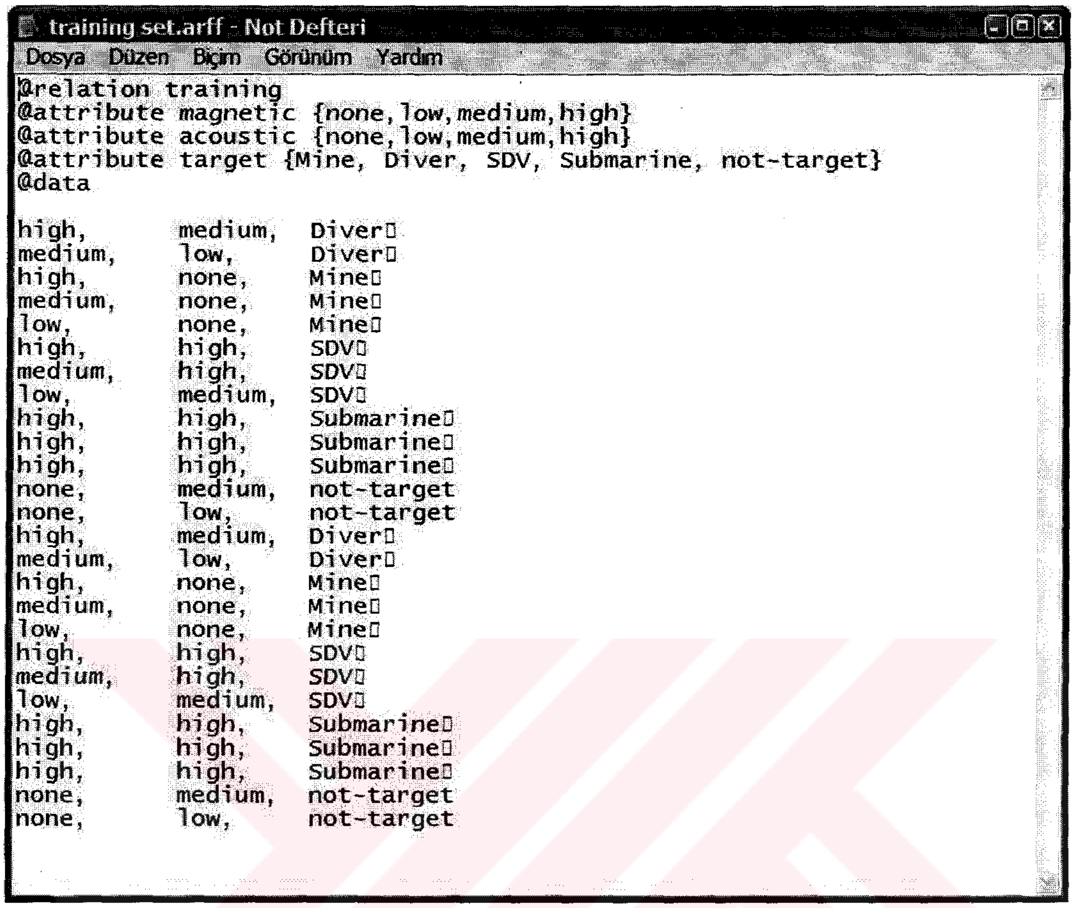
The experiments are performed on the WEKA (Waikato Environment for Knowledge Analysis) system [62] based on our training data set given in Table 10. The system is written in Java, an object oriented programming languange that is widely available for all major computer platforms, and WEKA has been tested under Linux, Windows, and Macintosh operating systems[7]. Java allows us to provide a uniform interface to many different learning algorithms, along with methods for pre- and postprocessing and for evaluating results of learning schemes on any given dataset.

The main focus of WEKA is on classifier and filter algorithms. However, it also includes implementations of algorithms for learning association rules and for clustering data for which no class value is specified.

WEKA expects database to be in ARFF format, because it is necessary to have type information about each attribute which can not be automatically deduced from the attribute values.

Figure 24 shows an ARFF file for the training data in Table 10. @data line signals the the start of the instances in the dataset. Instances are written one per line, with values for each attribute in turn, seperated by commas. If a value is missing it is represented by a single question mark (there is no missing values in this dataset). The attriute specifications in ARFF files allow the dataset to be checked to ensure that it contains legal values for all attributes. We can use also boolean and numeric values for attributes as;

@attribute magnetic numeric
@attribute acoustic {yes, no}



training set.arff - Not Defteri

Dosya Düzen Biçim Görünüm Yardım

```
@relation training
@attribute magnetic {none,low,medium,high}
@attribute acoustic {none,low,medium,high}
@attribute target {Mine, Diver, SDV, Submarine, not-target}
@data

high,      medium,   Diver□
medium,    low,      Diver□
high,      none,     Mine□
medium,    none,     Mine□
low,       none,     Mine□
high,      high,     SDV□
medium,    high,     SDV□
low,       medium,   SDV□
high,      high,     Submarine□
high,      high,     Submarine□
high,      high,     Submarine□
none,      medium,   not-target
none,      low,      not-target
high,      medium,   Diver□
medium,    low,      Diver□
high,      none,     Mine□
medium,    none,     Mine□
low,       none,     Mine□
high,      high,     SDV□
medium,    high,     SDV□
low,       medium,   SDV□
high,      high,     Submarine□
high,      high,     Submarine□
high,      high,     Submarine□
none,      medium,   not-target
none,      low,      not-target
```

Figure 24. An ARFF file for the training data in Table 10.

When we apply this file to J48 decision tree learner which is an implementation of C4.5 we get the output shown in Figure 25.

```
=== Run information ===

Scheme:       weka.classifiers.j48.J48 -C 0.25 -M 2
Relation:     training
Instances:    26
Attributes:   3
              magnetic
              acoustic
              target
Test mode:    evaluate on training data

=== Classifier model (full training set) ===

J48 pruned tree
------------------

acoustic = none: Mine (6.0)
acoustic = low
|   magnetic = none: not-target (2.0)
|   magnetic = low: Diver (2.0)
|   magnetic = medium: Diver (0.0)
|   magnetic = high: Diver (0.0)
acoustic = medium
|   magnetic = none: not-target (2.0)
|   magnetic = low: SDV (2.0)
|   magnetic = medium: Diver (2.0)
|   magnetic = high: Diver (0.0)
acoustic = high
|   magnetic = none: Submarine (0.0)
|   magnetic = low: Submarine (0.0)
|   magnetic = medium: SDV (2.0)
|   magnetic = high: Submarine (8.0/2.0)

Number of Leaves  :     13

Size of the tree :      17


Time taken to build model: 0.03 seconds

=== Evaluation on training set ===
=== Summary ===

Correctly Classified Instances          24               92.3077 %
Incorrectly Classified Instances         2                7.6923 %
Kappa statistic                          0.903
Mean absolute error                      0.0462
Root mean squared error                  0.1519
Relative absolute error                 14.5312 %
Root relative squared error             38.143  %
Total Number of Instances               26

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  1         0          1         1         1        Mine
  1         0          1         1         1        Diver
  0.667     0          1         0.667     0.8      SDV
  1         0.1        0.75      1         0.857    Submarine
  1         0          1         1         1        not-target

=== Confusion Matrix ===

 a b c d e   <-- classified as
 6 0 0 0 0 | a = Mine
 0 4 0 0 0 | b = Diver
 0 0 4 2 0 | c = SDV
 0 0 0 6 0 | d = Submarine
 0 0 0 0 4 | e = not-target
```

Figure 25. Output from the J48 decision tree learner for Table 10.

B. PERFORMANCE OF CLASSIFICATION MINING BASED DETECTION AND CLASSIFICATION

In this section we evaluate the performance of our classification mining based detection and classification (CMDC) scheme for various metrics. The first two of these metrics are correct classification rate β and precision ε_i. Correct classification rate β is the ratio of the number of correctly classified targets to the total number of targets.

$$\beta = \frac{\text{number of correctly classified targets}}{\text{total number of targets}} \qquad (7)$$

We compare the performance of three mostly used different learning algorithms in terms of classification rate β and precision ε_i. These algorithms are OneR, Naïve Bayes and J48.

Figure 26 depicts the correct classification ratio for each target type separately. The aggregated correct classification ratio for all types of targets in our domain is shown in Figure 27.

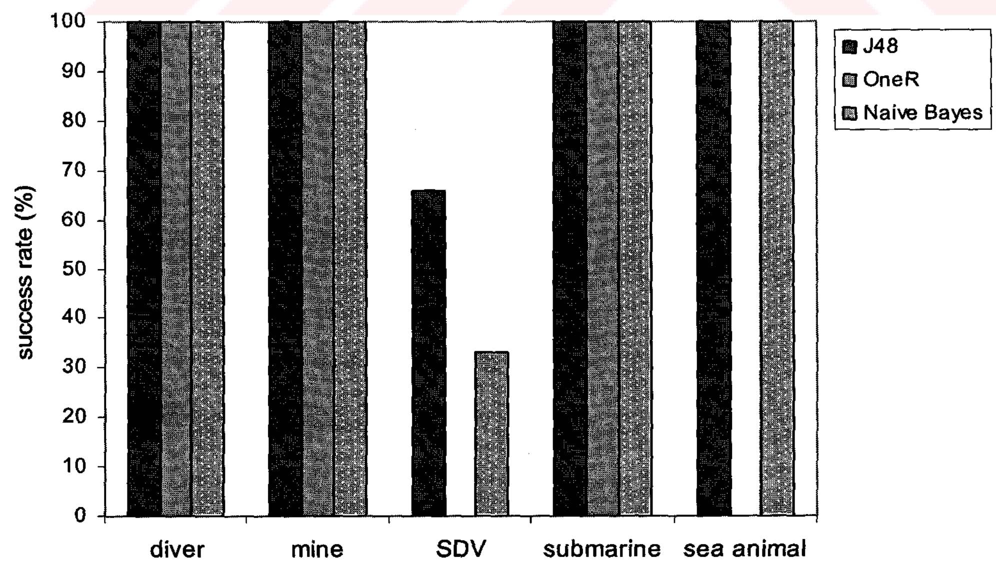


Figure 26. The probability that a target is correctly classified for a given target type.

As shown in Figure 26 J48 and Naïve Bayes algorithms correctly classify all of the targets except SDVs. The correct classification rate of J48 for SDV is 66%. The probability that an SDV can be incorrectly classified as submarine by J48 algorithm is 0.34. OneR algorithm classifies all divers, mines and submarines correctly. Correct classification rate of OneR for SDV and sea animals is 0. All of SDVs and sea animals are incorrectly classified by OneR algorithm.

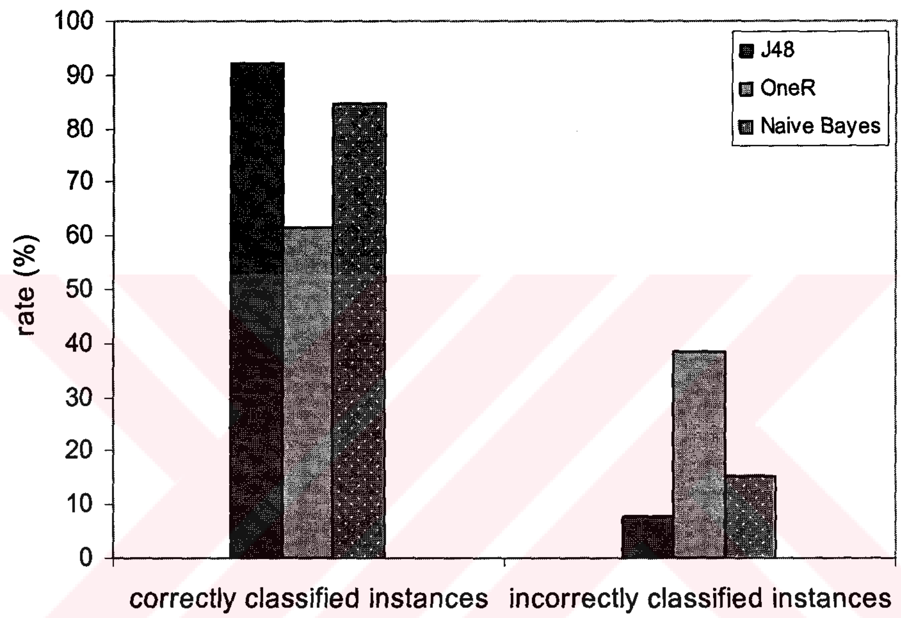


Figure 27. The probability that a target is correctly classified.

Aggregated correct classification rates of algorithms are shown in Figure 27. J48 algorithm has 92%, OneR has 61% and Naïve Bayes has 84% correct classification rate (β). As shown in Figure 26 and Figure 27 J48 algorithm has highest correct classification rate.

Precision is different from correct classification rate. It gives the probability that a target is of the type found out by our CMDC scheme. For example precision for a mine type target is the probability that the target is a mine when it is classified as a mine by our scheme. In Figure 28 we show the precisions for J48, OneR, Naïve Bayes algorithms.

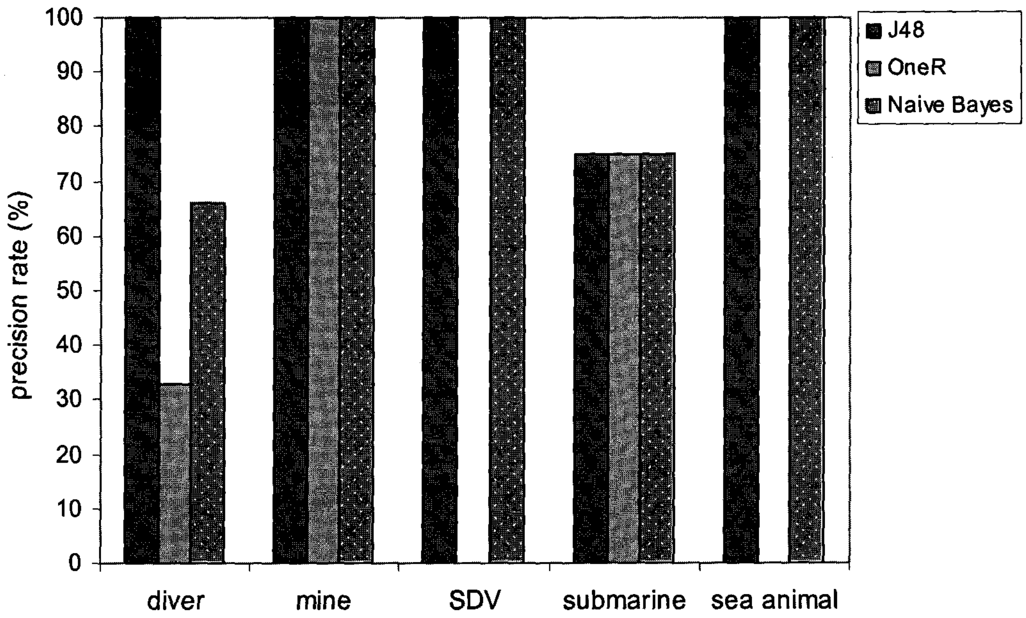


Figure 28. The precision for a specific target type.

The precision for mine is 1 in all three algorithms. If an object is classified as a mine that means the probability that it is a mine is 1. The precision of diver for OneR is 0.33 and precision of diver for Naïve Bayes is 0.66 . These two algorithms always correctly classify the divers but may classify other objects as diver. Therefore we prefer J48 algorithm.

Except from correct classification rate and precision, accuracy α of measurement is another performance metric which is given by

$$\alpha = \frac{1}{d^n} \times \theta \qquad (8)$$

In Equation 8, n is the slope that determines the effect of the distance between the target and the sensor on the accuracy. θ is the factor that aggregates all the other parameters that can change the accuracy of the sensing process. Accuracy is inversely proportional with the distance d. Figure 29 shows the accuracy for varying distances and slope values when factor θ is 1.

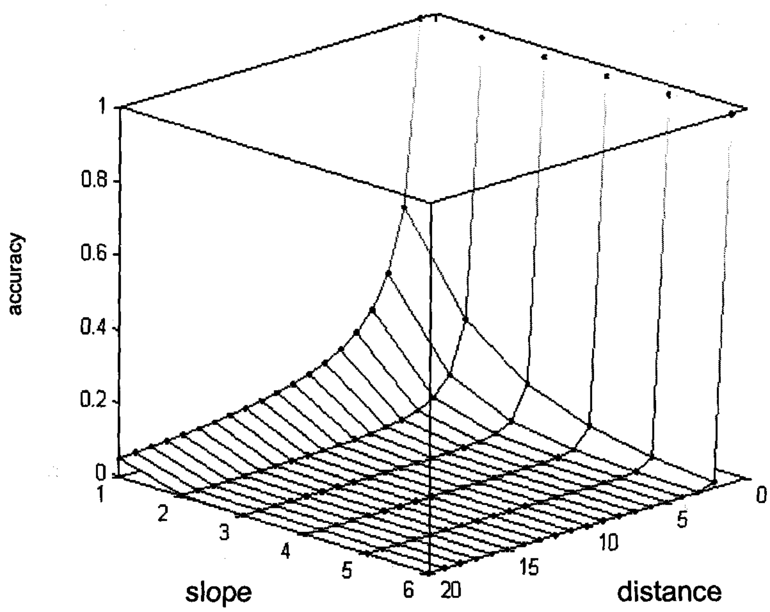


Figure 29. Accuracy of measurement for $\theta = 1$.

The probability of detection of a target is also affected by the deployment density. Deployment density of nodes in the sensor field is defined by λ and given by the following formula :

$$\lambda = \frac{\rho \times N}{V} \qquad (9)$$

where ρ is the sensor node coverage which is the volume covered by a node, N is the total number of sensor nodes in the field and V is the volume of the sensor field. Please note that our sensor field is three dimensional, and the volume V of the sensor field is given by

$$V = w \times l \times d \qquad (10)$$

where w is the width, l is the length and d is the depth of the sensor field.

The sensor node coverage ρ for a node that has the sensing range r is given in Equation 11.

$$\rho = \frac{4}{3} \times \pi \times r^3 \qquad (11)$$

Target size also influences the probability of its detection because it affects the number of nodes that can detect the target. The number of nodes *n* that can detect a target is given by

$$n = \lambda \times s_i \qquad (12)$$

where s_i is the size of the target. By using this equation we evaluate the number of nodes that can detect a given target for varying sensor node ranges in Figure 30 when the parameter values in Table 14 are used.

Table 14. Parameters for the results in Figure 30.

Parameter	Value
N	500
V	1.000.000
S_{diver}	2
S_{mine}	5
S_{SDV}	15

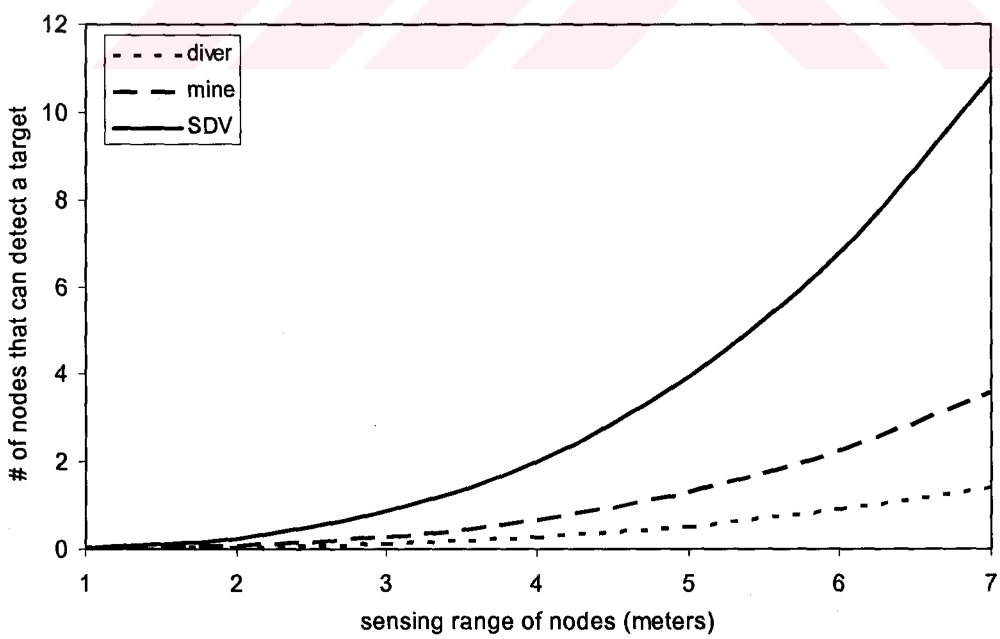


Figure 30. The number of nodes that can detect a target for varying sensor ranges

An SDV can be detected by the 4 sensor nodes where the sensing range of nodes is 5 meters and the other parameters as in Table 14. With the same parameters 1 sensor node will detect a mine and a diver cannot be detected by any sensor. When sensing range is 7 meters 11 sensor nodes will detect an SDV, 3 nodes will detect mine, 1 sensor can detect a diver.

Increasing the total number of nodes in Equation 9 we get higher deployment density. Higher deployment density increases the number of nodes that detect can detect a target. If we use parameters in Table 14 except N in Equation 9 and sensing range is 5 meters in Equation 11 we get results in Figure 31.

SDVs can be detected by the 15 sensor nodes when the deployment density is 1. With the same parameters 5 nodes can detect a mine and a diver is detected by 2 sensors.

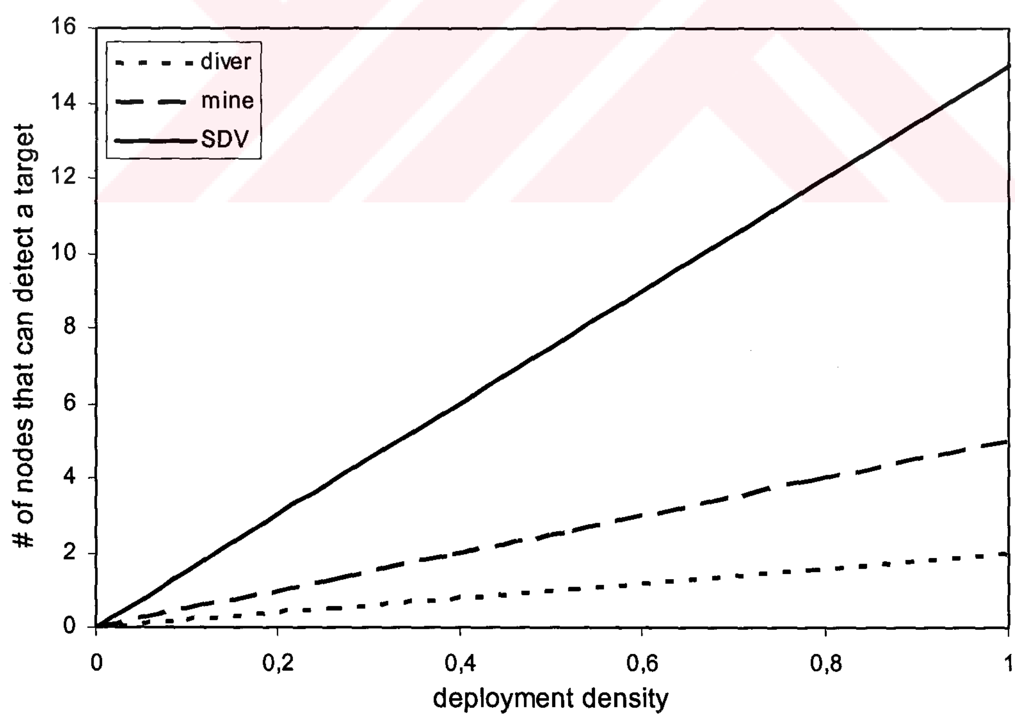


Figure 31. The number of sensor nodes that can detect a target for varying node densities.

These values can also be used to classify targets. If deployment density is 1 and 15 sensor nodes detect a target at a time we can say that it is an SDV.

V. CONCLUSIONS

Terrorist and guerilla warfare courtermeasures require distributed networks of sensors that can be deployed easily, and have self-organizing capabilities. In such applications, cabling is usually impractical. Sensor networks must be fast and easy to install and maintain.

Detection, classification and tracking of targets is a basic surveillance or military application, and has received a considerable amount of attention in the literature. Recent developments in the miniaturization of sensing, computing, and communications technology have made it possible to use multi sensor within a single device or sensor network node. Their low cost makes it feasible to deploy them in significant numbers across large areas.

We proposed a classification mining based detection and classification scheme for tactical underwater sensor networks. Mechanical, radiation, magnetic and acoustic microsensors, which cost only a couple of dollars, are used to tackle the challenge of detecting and classifying submarines, SDV (small delivery vehicles), underwater mines and divers in open, shallow and very shallow water. Our scheme first detects a target in the vicinity based on the readings of radiation and mechanical sensors. Then the detected target is classified into one of the target types based on the data coming from acoustic and magnetic microsensors. Decision tree algorithms are used as a classification mining technique. Our studies show that decision tree classification is faster and has higher accuracy than other classification methods.

As a future work, data mining based target tracking can be examined. Also new data schemes can be designed by using association rule mining and sequential pattern analysis.

LIST OF REFERENCES

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, "Wireless Sensor Networks: A Survey", Computer Networks Journal (Elsevier), Vol. 38, No.4, pp. 393-422, March 2002.

[2] Hakan Tezcan, Erdal Cayirci, Vedat Coskun "A distributed scheme for 3D space coverage in tactical underwater sensor networks" (submitted to MILCOM'2004)

[3] W. Lee, S. Stolfo, P. Chan, E. Eskin, W. Fan, M. Miller, S. Hershkop, J. Zhang. "Real Time Data Mining-based Intrusion Detection" In DARPA Information Survivability Conference and Exposition II. June 2001.

[4] Lilian Harada "An Efficient Sliding Window Algorithm for Detection of Sequential Patterns" Proceedings of the Eighth International Conference on Database Systems for Advanced Applications (DASFAA'03) 2003 IEEE

[5] R. R. Brooks, P. Ramanathan, A. M. Sayeed "Distributed Target Classification and Tracking in Sensor Networks" Special Issue on Sensor Networks, Revision, January 2003.

[6] H. Wang, J.Elson, L.Girod, D.Estrin, K.Yao "Target Classification and Localization in Habitat Monitoring" Proceedings of IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP 2003) Hong Kong, China. April 2003.

[7] I.H.Witten, E.Frank "Data Mining : Practical machine learning tools and techniques with java implementations" , Morgan Kauffmann, San Francisco, 2000.

[8] Cabena, Hadjinian "Discovering data mining from concept to implementation" Prentice Hall 1998.

[9] A.K.H.Tung, H.Lu, J.Han, L Feng "Efficient Mining of Intersaction Association Rules" IEEE Transactions on Knowledge and Data Engineering Vol.15, No.1, 2003

[10] J.Han, M.Kamber "Data Mining : concepts and techniques" Morgan Kauffman Publishers 2001.

[11] Julian W. Gardner "Microsensors Principles and Applications." John Wiley & Sons Ltd. December 1994.

[12] www.signal.uu.se/Courses/CourseDirs/Sensors/

[13] Middelhoek S and Hoogerwerf AC (1985) "Smart Sensors: when and where? Sensors and Actuators", 8, 39-48.

[14] Hauptmann P (1991) "Sensors: Principles and Applications", Prentice Hall International (UK) Ltd. p.7

[15] Breckenbridge RA and Husson C "Smart Sensors in spacecraft: the impact and trends". Proc. AIAA/NASA Conf. on Smart Sensors, Hampton, USA, pp.1-5.

[16] J.M.Giachino, "Smart Sensors", Sensors and Actuators, n.10, pp 239-248, Elsevier Science,1986.

[17] Jiawei Han, Micheline Kamber.: "Data Mining: Concepts and Techniques", Simon Fraser University, 2001.

[18] http://www-ai.ijs.si/NadaLavrac/ Dzeroski and Lavrac, 2001

[19] R. Agrawal, R. Srikant.: Fast Algorithm foe Mining Association Rules. In Proc. Of the 20th VLDB Conference, pages 487-400, Santiago, Chile, 1994.

[20] J.S. Park, M.S. Chen, and P.S.Yu. "Using a Hash-Based Method with Transaction Trimming for Mining Association Rules" IEEE Trans. On Knowledge and Data Eng.,Vol. 9, no. 5, pp. 813-825, Sept./Oct. 1997.

[21] Rakesh Agrawal, Ramakrishnan. Srikant. "Mining Sequential Patterns", ICDE 1995, Taipei, Taiwan.

[22] H. Mannila, H. Toivonen, and I. Verkamo, "Discovery of Frequent Episodes in Event Sequences," The DMKD Journal, 1(3): pp.259-289, 1997.

[23] R. Srikant and R. Agrawal, "Mining Sequential Patterns: Generalizations and Performance Improvements," Proceedings of the 5th EDBT, 1996.

[24] John C. Shafer, Rakesh Agrawal, Manish Mehta.: SPRINT: A Scalable Parallel Classifier for Data Mining. VLDB 1996, p. 544-555.

[25] M. Mehta, R. Agrawal, J. Rissanen.: SLIQ: A fast scalable classifier for data mining.In Proc. 1996 Intl. Conf. on Extending Database Technology (EDBT'96), Avignon, France, March 1996.

[26] G.D. Ramkumar and Arun Awami. "Clustering Data without Distance Functions", IEEE bulletin of technical committee on Data Engineering, March 1998, Vol. 21 No. 1

[27] R. C. Dubes and A. K. Jain. "Algorithms for Clustering Data", Prentice Hall,1998.

[28] J.R. Quinlan, "Induction of Decision Trees," Machine Learning, vol. 1, no. 11, pp. 81-106, 1986.

[29] J.R. Quinlan, "C4.5: Programs for Machine Learning", Morgan Kaufmann, 1993.

[30] De Raedt *et al.*, 2001 www.kdl.cs.umass.edu/srl2003_upload/files/kersting2-paper.pdf

[31] www.olapreport.com/fasmi.htm

[32] M. Last, O.Maimon "A Compact and Accurate Model for Classification" IEEE Transactions on Knowledge and Data Engineering Vol.16, No.2, 2004

[33] Juan Liu, P. Moulin, "Information theoritic classification: Information-theoritic analysis of interscale and intrascale dependencies between image wavelet coefficients", IEEE Trans. Image Process. 2001

[34] T.M. Cover and J.A. Thomas, "Elements of Information Theory". Wiley, 1991.

[35] L. Breiman, J.H. Friedman, R.A. Olshen, and P.J. Stone, "Classification and Regression Trees", Wadsworth, 1984.

[36] M. Last, O. Maimon "A Compact and Accurate Model for Classification" IEEE Transactions on Knowledge and Data Engineering, vol.16 no.2 February 2004.

[37] T. M. Mitchell. Machine Learning. McGraw-Hill Companies, Inc., New York, 1997.

[38] www.cs.rutgers.edu/~mlittman/courses/

[39] A. Suarez, J. F. Lutsko "Globally Optimal Fuzzy Decision Trees for Classification and Regression" IEEE Transactions on pattern analysis and machine intelligence, vol.21, no.12, December 1999.

[40] B. Chandra, S. Mazumdar, V. Arena, N. Parimi "Elegant Decision Tree Algorithm for Classification in Data Mining" Proceedings of the Third International Conferance on Web Information Systems Engineering 2002 IEEE.

[41] T. M. Khoshgoftaar , Y. Liu, N. Seliya "Genetic Programming-Based Decision Trees for Software Quality Classification" ICTAI 2003 374-383

[42] R. P. Gorman and T. J. Sejnowski, "Learned classification of sonar targets using a massively parallel network," IEEE Trans. Acoust., Speech, Signal Processing, vol. 36, no. 7, July 1988.

[43] B. A. Telfter, H. H. Szu, and G. Dobeck, "Adaptive time-frequency classification of acoustic backscatter," in Proc. SPIE Int. Symp. Aerospace/Defense Sensing Contr., Orlando, FL, Apr. 1995.

[44] G. Goo andW.W. L. Au, "Detection and identification of buried objects in shallowwater," in Proc. SPIE Inter. Symp. Aerospace/Defense Sensing Contr., vol. 2765, Orlando, FL, Apr 1996, pp. 201–214.

[45] M.R. Azimi-Sadjadi, D.Yao, Q. Huang, G.J.Dobeck "Underwater Target Classification Using Wavelet Packets and Neural Networks" IEEE Transactions on Neural Networks, vol.11. no.3 May 2000.

[46] S. Kadambe, R. Orr "Application of Gabor Analysis for Detection, Estimation and Classification of Underwater Acoustic Data." 1996 IEEE.

[47] C.Meesookho, S.Narayanan, C.S.Raghavendra "Collaborative Classification Applications in Sensor Networks" SAM2002, Aug. 2002.

[48] R.R. Brooks, P.Ramanathan, A.M. Sayeed "Distributed Target Classification and Tracking in Sensor Networks" Proceedings of the IEEE, vol.91, no.8, August 2003.

[49] M.R.Azimi-Sadjadi, D.Yao, A.A. Jamshidi, G.J.Dobeck "Underwater Target Classification in Changing Environments Using an Adaptive Feature Mapping" IEEE Transactions on Neural Networks, vol.13. no.5 September 2002.

[50] A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Gouda, Y. Choi, T. Herman, S. Kulkarni, U. Arumugam, M. Nesterenko, and A. Vora "A Line in the Sand : A Wireless Sensor Network for Target Detection, Classification and Tracking" Computer Networks (Elsevier), to appear.

[51] M.Kamber, L.Windstone, W. Gong, S. Cheng, J. Han "Generalization and Decision Tree Induction: Efficient Classification in Data Mining" Proceedings of the 7th International Workshop on Research Issues in Data Engineering (RIDE'97) 1997 IEEE.

[52] T.M.Khoshgoftaar, E.B.Allen, L.A.Bullard, R.Halstead, G.P.Trio "A Tree-Based Classification Model for Analysis of a Military Software System" Proceedings of the 1996 High-Assurance System Engineering 1996 IEEE.

[53] C.Tang, C.S.Raghavendra "Correlation Analysis and Applications in Wireless Microsensor Networks" MobiQuitous 2004 The First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services. August 22-26, 2004 - Boston, Massachusetts, USA

[54] L.Mayer "On the Robustness of Wireless Sensor Network based Target Tracking Algorithms" University of Vermont October 2003

[55] Larry Godfrey "Choosing the detector for your unique light sensing application" PerkinElmer Precisely. Published: January 1, 2000.

[56] Peter Elgar "Sensors for Measurement and Control" Prentice Hall 1998.

[57] http://www.spie.org/app/Publications/magazines/oerarchive/may/underwater.html

[58] http://deeptow.whoi.edu/primer.html

[59] http://www.farasia.com.cn/products/prod95.htm

[60] http://www.sensorsmag.com/articles/1297/he1297/main.shtml

[61] http://www.jrc.co.jp/product/comm/device/saw/saw_top_e.html

[62] WEKA (Waikato Environment for Knowledge Analysis), University of Waikato, Department of Computer Science, New Zealand. [http://www.cs. waikato.ac.nz/~ml].