Software system for mining spatio-temporal association rules

Abraham N. Konda
The University of Montana
The University of Montana

Permission is granted by the author to reproduce this material in its entirety, provided that this material is used for scholarly purposes and is properly cited in published works and reports.

**Please check "Yes" or "No" and provide signature**

Yes, I grant permission  

No, I do not grant permission

Author's Signature: [Signature]

Date: Aug 29, 2009

Any copying for commercial purposes or financial gain may be undertaken only with the author's explicit consent.
A SOFTWARE SYSTEM FOR MINING SPATIO-TEMPORAL ASSOCIATION RULES

by

Abraham N Konda


presented in partial fulfillment of the requirements

for the degree of

Master of Science

The University of Montana

August 2004

Approved by:

Chairperson:

Dean, Graduate school

Date
Massive amounts of spatial, temporal and spatio-temporal data are generated everyday. The resulting information explosion has given rise to the problem of extracting interesting patterns from the data efficiently and automatically. The recent advent of Geographic Information Systems (GIS) partially solves the above problem by providing software tools that handle either spatial or temporal data but rarely both. By developing a software prototype that handles spatio-temporal data, we try to develop a more complete tool for analyzing GIS databases. The prototype exploits appropriate machine learning algorithms to accomplish the task of discovering spatio-temporal trends implicit in the data. It has been integrated with ArcGIS, a popular GIS software tool, and utilizes some of the software components it provides. As a case study, we use it to mine spatio-temporal patterns in average monthly earth temperatures. Results from experiments are promising.
# TABLE OF CONTENTS

1.0 Introduction .................................................................................................................. 1

2.0 Background Information .............................................................................................. 3

2.1 Data mining ................................................................................................................... 3

2.2 Evolutionary computation ............................................................................................. 6

2.2.1 Genetic Algorithms ............................................................................................... 7

2.2.2 Genetic Programming ......................................................................................... 9

2.3 Unsupervised Classification: K-means clustering algorithm ...................................... 12

2.4 Cycles and Anomalies ................................................................................................. 14

2.5 Geographic information systems .................................................................................. 15

3.0 System Description ....................................................................................................... 16

3.1 System Block Diagram ............................................................................................... 16

3.2 User Interface Description .......................................................................................... 18

4.0 Experiments .................................................................................................................. 24

4.1 Description of datasets ............................................................................................... 24

4.2 Methodology ............................................................................................................... 24

4.3 Results ......................................................................................................................... 26

4.3.1 Mining anomalous data ..................................................................................... 26

4.3.2 Mining raw data .................................................................................................. 29

4.3.3 Mining with user feedback ................................................................................ 33

4.4.4 Mining in an area of interest .............................................................................. 37

5.0 Discussion & Future work ............................................................................................ 39

6.0 Conclusions .................................................................................................................. 40

7.0 References ................................................................................................................... 42
LIST OF FIGURES

Figure 1: An outline of a genetic algorithm................................................................. 8
Figure 2: A parse tree for the expression ‘A + (B – C)’ ............................................. 9
Figure 3: A parse tree for a spatio-temporal association rule ................................. 10
Figure 4: Crossover between two trees resulting in two children ......................... 10
Figure 5: Different types of mutation operators on parse trees .............................. 11
Figure 6: An outline of K - Means clustering algorithm ......................................... 12
Figure 7: Depiction of K-Means clustering process ............................................... 13
Figure 8: K-Means algorithm applied to monthly earth temperatures with K = 200 .... 13
Figure 9: Block diagram of spatio-temporal data mining system ............................ 17
Figure 10: Cluster Viewer, the starting step of the data mining process where data is clustered and viewed ............................................................... 19
Figure 11: Parameter Settings panel, the second step in the data mining process where the user does additional preprocessing and sets the genetic parameters .......... 20
Figure 12: Panel that keeps track of current population and user feedback ............... 23
Figure 13: Depiction of relationship among regions off the coast of South America and Africa .......................................................... 27
Figure 14: Plot of the standardized Z values of the average temperatures of clusters 167 and 0 .......................................................... 27
Figure 15: Depiction of relationship among regions in Atlantic, Pacific and South America, Africa .......................................................... 28
Figure 16: Plot of the standardized Z values of the average temperatures of clusters 76 and 175 .......................................................... 29
Figure 17: Depiction of relationship among regions in Africa, Saudi Arabia, Iran, Afghanistan and Northern Spain, Southern France ............................................ 30
Figure 18: Plot of average values of clusters 47 and 120 ........................................... 31
Figure 19: Depiction of relationship among regions in Northern Canada, Alaska, Russia and Pacific, Atlantic and Indian oceans

Figure 20: Plot of average values of clusters 57 and 174

Figure 21: Depiction of relationship among regions in Indian Ocean and Pacific, Atlantic and Indian Ocean

Figure 22: Plot of the standardized Z values of the average temperatures of clusters 140 and 27

Figure 23: Depiction of relationship among regions in Pacific, Atlantic and northern parts of Russia

Figure 24: Plot of the average values of clusters 20 and 197

Figure 25: Depiction of relationship among regions in Northern USA, Southern Canada, Western Europe, Northern China and Pacific and Atlantic off the coast of Panama

Figure 26: Depiction of relationship among regions in Canada, Europe, Asia and Pacific, Atlantic off the coast of Panama
LIST OF TABLES

Table 1: A sample database with four records ............................................................... 5

Table 2: Parameter values used for experiments ........................................................... 25
1.0 Introduction
Massive amounts of digital data such as aerial images, time series data, remote sensing data, environmental data etc; are being generated everyday. The resulting information explosion created an urgent need for discovering hidden patterns in the data with as little human assistance as possible. Data Mining is the discovery of hidden predictive information in data. It is a challenging field with many practical applications. Much research has been done and many software tools, such as CART, CHAID, DARWIN etc., were created to mine data [1]. However, research in spatio-temporal data mining and software tools for the same have been lacking. As a result, this thesis deals with the design and implementation of a flexible system for spatio-temporal data mining.

The recent advent and popularity of Geographic Information Systems (GIS) gave rise to many software tools that can handle spatial and temporal data. Yet, very few software tools exist today that work with spatio-temporal data. Hence, many GIS systems are incomplete. GIS systems that implement spatio-temporal data mining must be able to predict, analyze and model spatio-temporal events. El Nino, El-Nina, the monsoon, earthquakes, floods, accidents etc; are some examples of spatio-temporal events. They can be determined by their previous occurrences in the spatial and temporal domain. Valuable information can be discovered using existing data mining software tools. However, the functionality of most of these software tools as mentioned above is restricted to either the spatial or the temporal domain. We therefore develop a software tool for spatio-temporal data mining. The software prototype uses evolutionary algorithms, since they have already been implemented successfully on similar problems in the spatial and temporal domain. Another reason for using heuristic methods such as
evolutionary computation is that they tend to outperform algorithms as the search space grows exponentially. Key components for successful implementation are data preprocessing and proper input representation of the problem. By developing it as an extension that handles spatio-temporal data, this thesis tries to make existing GIS systems complete.

One important feature of the developed system is its friendly user interface. The user interface allows the user to easily interact with the evolutionary computation process without him/her ever being exposed to the complexities of the underlying procedure. The user starts with the data preprocessing stage. Here the data is clustered, well-known cycles are removed and anomalies are found. In the next stage, the user sets the various parameters needed for the evolutionary computation process. The user then proceeds to the data mining stage. During the data mining process the user can interact with the system by giving feedback. The system tries to adapt according to the user interests and finds patterns the user is interested in. The output is stored in an easily readable format for the user to peruse later.

As a case study, this system has been used to mine for interesting patterns occurring in monthly earth temperatures from 1981-2000. It has been successful in finding spatio-temporal relationships among anomalies. It has also been successful in discovering highly correlated regions. User feedback played a vital role in accelerating the computation process.

The rest of this thesis is organized as follows. A brief introduction to data mining is given first. This is followed by a discussion of evolutionary computation, genetic algorithms and genetic programming. An outline of the clustering algorithm used
is given next. The significance of cycles and anomalies in the data is then explained, followed by a brief description of ArcGIS, the GIS system used in our prototype. Finally, we explain the methodology, the experiments and the results, and conclude with a discussion of future work.

2.0 Background Information

2.1 Data mining

As the cost of digital storages devices reduces every day more and more data is being stored in digital format. These days, it is not unusual to hear about databases of the order of tera and even penta bytes. GIS systems, national medical records, business transactions, weather images, etc; are all stored in huge databases. Databases, which store huge amounts of data, are known as data warehouses. Analysis of these data warehouses is beyond man’s ability. As a result, there arose “an urgent need for a new generation of computational theories and tools to assist humans in extracting useful information” [2]. This automated process of extracting hidden predictive information from databases is known as data mining [3,4,5,6].

The drop in the cost of computing has steered data mining research to adopt computationally expensive pattern recognition algorithms found in the field of AI and image processing. Data mining also utilizes techniques from a variety of other fields such as statistics, visualization, database management systems (DBMS), etc. It is a powerful technology that helps end users discover implicit patterns in the data. Armed with these powerful tools, corporations are able to save millions of dollars in their daily business

---

transactions. One famous example is the discovery of a statistically significant correlation between purchases of beer and purchases of diapers in the evenings, in the USA []. Other examples include detection of fraud, identity thefts and identifying tax defaulters using data mining.

Data mining techniques can be broadly classified into two main categories:

a) data predictive methods, and

b) data descriptive methods.

Classification, regression and time series deal with building predictive models. The various algorithms used for prediction are neural networks, decision trees, genetic algorithms etc. Visualizing, clustering and link analysis try to describe the data. Link analysis tries to describe the data by discovering relationships inherent in the data. The two popular approaches to link analysis are association discovery and sequential discovery. Association discovery finds relationships/associations among frequently occurring items in a database [12,13,14,15]. Associations are nothing but if-then rules. Associations are written as \( A = \rightarrow B \). A is called the antecedent or left hand side (LHS) and B the consequent or right hand side (RHS). Sequence discovery deals with temporal association rules. Confidence and support are two important terms in link analysis. The frequency with which a particular association appears in the database is known as support. Confidence is the relative frequency of the occurrence of items and their combinations. The confidence for the rule \( A = \rightarrow B \) is given by the formula

\[ \text{Confidence} = \frac{\text{Frequency}(A \text{ and } B)}{\text{Frequency}(A)} \]

For example, if we have a database as shown in table 1 and an association rule \( A = \rightarrow B \) then the values for support and confidence are
Table 1: A sample database with four records

<table>
<thead>
<tr>
<th>Record</th>
<th>A, B, C, D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Record 0</td>
<td>A, B, C, D</td>
</tr>
<tr>
<td>Record 1</td>
<td>A, B, F, E</td>
</tr>
<tr>
<td>Record 2</td>
<td>A, C, D, E</td>
</tr>
<tr>
<td>Record 3</td>
<td>A, B, F, D</td>
</tr>
</tbody>
</table>

Support = (Number of occurrence of A, B in a record) / (Total number of records)

\[ = \frac{3}{4} = 0.75 \]

Confidence = (Number of occurrence of A, B in a record) / (Number of occurrence of A)

\[ = \frac{3}{4} = 0.75 \]

Association rules with high confidence beyond some minimum support levels are generally considered to be interesting.

Describing spatio-temporal trends through link analysis is a very difficult task [7,9,10,11]. This is mainly because of the spatial and temporal nature of the data. Some spatio-temporal trends occur in conjunction with other trends in a different spatial and temporal location. For example, an anomaly occurring in Missoula at time ‘t’ and another anomaly occurring in the Atlantic ocean at time ‘t + k’ may cause an anomaly to occur in south America at time ‘t + k1’. If any of the clauses in the LHS doesn’t occur then the RHS may not occur at all. Association rules that have spatial and temporal features have been designed in this thesis so they can mine for spatio-temporal trends. They are described in detail in section 3. The spatio-temporal data is envisioned as a table where
each record is an instance in time and holds values taken by different items at that time. The ‘interestingness’ of a rule that has minimum support is given by the weighted average of its fitness (this term is explained in section 2.2) and user feedback.

The design of spatio-temporal association rules facilitates users to mine for spatial and temporal trends as well. In order to mine spatial data the temporal parameter of the clauses in the LHS and RHS of an association rule should always take the same values. In order to mine for temporal patterns the clauses in the LHS and RHS should contain the same spatial parameter. Though spatial and temporal mining are well explored fields [9,15], we will give some discovered spatial rules later to highlight the flexibility of our developed system in mining spatial, temporal and spatio-temporal rules.

2.2 Evolutionary computation
Evolutionary computation is an umbrella term used to describe computational models based on the principles of evolution. Computer programs that incorporate evolutionary computation in problem solving are known as evolutionary algorithms (EA). A wide variety of evolutionary algorithms exist. Some of the popular ones are

- Genetic algorithms
- Evolutionary programming
- Evolution strategies
- Classifier systems
- Genetic programming

Conceptually, all EA are similar in that they use evolutionary processes or genetic operators such as crossover, mutation and selection to simulate the evolution of

---

individuals in a diverse population with the goal of finding highly fit individuals. Evolutionary processes can either explore the search space or exploit the available fitness information. Mutation is a genetic operator, which perturbs individual structures. It is used for exploring the search space. Crossover is an operator, which exploits the available fitness information by focusing on highly fit individuals. By crossing highly fit individuals, it tries to generate fitter individuals. Highly fit individuals are good representatives of the solution space. ‘Natural selection’, a fundamental principle of evolution, simply put, states that the stronger individuals of a population will eventually replace the weaker ones. Artificial selection mimics ‘natural selection’ by replacing weaker individuals with stronger ones in an artificial environment. A fitness function is used to determine the performance of individual entities within the problem domain.

EA provide robust and powerful adaptive search mechanisms. They can, in principle, compute any computable function but are not suited for problems where efficient ways of solving them are already known. They have been successfully applied to many problems in AI, biology, engineering, game playing, evolvable hardware and scheduling. One of their disadvantages is that they are computationally expensive and hence take considerable amount of time to execute.

2.2.1 Genetic Algorithms

Genetic algorithms (GAs) are a type of machine learning algorithms. They use computer programs to simulate the evolution of individuals, also known as chromosomes, in a diverse population. Each individual represents alternative solutions to the problem at hand. Crossover, mutation, evaluation and selection are the operators used to explore and exploit the search space. An outline of a general GA is given in Table 2.
A diverse population is essential for evolution to work. Line 1 in the above program ensures that we start with a diverse population. Usually, a random population is generated as the first step. The next step evaluates the performance of every individual within the problem domain using a fitness function. The fitness function is designed such that individuals with higher fitness have higher performance value than others. Line 3 ensures that evolution across generations is simulated until the stopping criterion is met. Execution of a single while loop is said to be a generation. The role of crossover, mutation and selection are similar to the ones discussed in EC. ‘Natural selection’ is modeled by imposing a selection distribution such that there is a high probability of passing a fit individual to the next generation. Alternatively, we can use this distribution to weed out weaker individuals in the population. In theory, if a solution exists, any GAs should converge to the solution given ‘sufficient’ amount of time to execute on a machine.

<table>
<thead>
<tr>
<th>Procedure GA</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. create a diverse population</td>
</tr>
<tr>
<td>2. evaluation</td>
</tr>
<tr>
<td>3. while not stopping criteria</td>
</tr>
<tr>
<td>4. crossover</td>
</tr>
<tr>
<td>5. mutation</td>
</tr>
<tr>
<td>6. evaluation</td>
</tr>
<tr>
<td>7. selection</td>
</tr>
<tr>
<td>8. end while</td>
</tr>
</tbody>
</table>

**Figure 1: An outline of a genetic algorithm**
GAs employ a randomized beam search method to seek a maximally fit hypothesis. The GA search process is more abrupt than other machine learning algorithms and is hence less likely to fall into local minima [19]. It is therefore more robust than other machine learning algorithms.

2.2.2 Genetic Programming

“GENETIC PROGRAMMING (GPs) is an extension of the genetic model of learning into the space of programs”. Each individual is not a linear string but is a program which when executed represents a candidate solution to the problem at hand. The most suitable way of expressing these ‘genetic programs’ is in the form of parse trees. For example, the expression ‘A+B−C’ would be parsed into a tree as shown in figure 1.

![Figure 2: A parse tree for the expression ‘A + (B − C)’](image)

Association rules in this paper are modeled as trees. For example, the tree given in figure 2 represents the association rule

\[
\text{IF} (\text{Anomaly[Cluster1,t,+]} \text{And Anomaly[Cluster2,t+k1,-]}) \text{ THEN Anomaly[Cluster3,t+k2,+]}
\]
Figure 3: A parse tree for a spatio-temporal association rule

A depiction of the genetic operators ‘Crossover’ and ‘Mutation’ is given below.

Crossover: Crossover operator on two trees is depicted in figure 3.

Figure 4: Crossover between two trees resulting in two children
Mutation: Many types of mutation operators exist. Some of the operators we used are shown below in figure 4.

![Mutation operators on parse trees]

**Figure 5: Different types of mutation operators on parse trees**

Selection: Different types of selection such as proportional selection, tournament selection, truncation selection, \((\mu,\lambda)\) selection, \((\mu + \lambda)\) selection etc; exist. In this paper, we used \((\mu + \lambda)\) selection where \(\mu\) parents produce \(\lambda\) children and the best \(\mu\) of the entire \((\mu + \lambda)\) population passes onto the next generation.

GAs and GPs being a part of EAs have been applied to solve a wide variety of problems successfully. Though they are computationally expensive, they do lend themselves to parallel computation.
2.3 Unsupervised Classification: K-means clustering algorithm

Grouping of unlabeled data into clusters that behave similarly is known as unsupervised classification. Points belonging to a cluster behave similarly than points belonging to other clusters. Data clustering techniques are used to get a high level visual description of the data distribution [17,18]. It is usually the first step in solving data-mining problems. There are many types of clustering techniques such as k-means, hierarchical clustering, fuzzy clustering, etc. Because of its simplicity, we have used K-means algorithm in this thesis. The ‘K’ stands for the number of clusters in the data and is specified by the user. An outline of a K-means algorithm is given in table 3.

<table>
<thead>
<tr>
<th>K-Means</th>
</tr>
</thead>
<tbody>
<tr>
<td>Choose k random centers in the data</td>
</tr>
<tr>
<td>While stopping criteria is not met</td>
</tr>
<tr>
<td>For all points in the data</td>
</tr>
<tr>
<td>Assign current point to the closest center</td>
</tr>
<tr>
<td>Recalculate the k centers</td>
</tr>
<tr>
<td>End K-Means</td>
</tr>
</tbody>
</table>

Figure 6: An outline of K – Means clustering algorithm

The algorithm starts by labeling k distinct random points of the data as cluster centers. It then labels each data point by the label of the cluster center closest to it. The cluster centers are recomputed. This process runs iteratively until no points move to a different cluster center in two consecutive iterations. The algorithm is then said to have converged. We have used Euclidean distance as a measure of closeness/similarity. The initial and final step of K-means algorithm appears in Figure 5.
Figure 7: Depiction of K-Means clustering process

K – means clustering for k = 200 on monthly average temperatures of the earth appears in figure 6.

Figure 8: K-Means algorithm applied to monthly earth temperatures with K = 200
Observe that k-means clustering algorithm was successful in separating land and water regions.
2.4 Cycles and Anomalies
Patterns that represent deviations from the norm are more interesting than patterns that model well known cyclic variations. For example, it would be of no interest to anyone, if they found a pattern, which states that the average temperature of a particular place is almost the same twelve months from now. To solve the problem of evaluating the 'interestingness' of a spatio-temporal pattern we can either get the subjective opinion of an expert or we can mine interesting data such as data without cycles, anomalous data etc, thereby removing the need for an expert during the computation process. Researchers at the University of Minnesota have come up with various methods for removing cycles [6]. Some of the methods are filtering based on discrete Fourier transforms, singular value decompositions, moving average of monthly Z scores, etc. In this thesis, the monthly Z scores method was used. Values for each month are taken and their corresponding Z scores are computed. For example, if the data is in the form

\[
\text{jan}_1, \text{feb}_1, \text{mar}_1, ..., \text{jan}_2, \text{feb}_2, \text{mar}_2, ..., \text{jan}_n, \text{feb}_n, \text{mar}_n, ...
\]

Then the values for January (\(\text{jan}_1, \text{jan}_2, ..., \text{jan}_n\)) are taken and their Z score are calculated using the formula

\[
Z_i = (\text{Jan}_i - \text{Jan}_{\text{Mean}}) / \text{Jan}_{\text{stddev}}
\]

**NOTE:** Well-known cycles in earth science data are also known as seasonalities. The twelve-month cycle is predominant in most earth science data and is a well-known cycle. Since the data we have is average monthly earth temperatures, we used a moving average with a window of 12 to remove the cycles of period 12.

Anomalies are huge variations in input data. Modeling anomalies is of particular interest to scientists. Anomalies can be obtained easily from Z scores. Assuming that we have valid data, normalized data beyond the 95% is treated as an
anomaly. Normalized data beyond 95% corresponds to data beyond Mean ± 1.96. In this thesis, a positive anomaly is said to have occurred in a cluster if the percentage of positive anomalous points, for example, normalized points whose value is above 1.96, is more than 0.85 times the number of anomalous points in the cluster. Similarly a negative anomaly is said to have occurred in a cluster if the percentage of negative anomalous points, for example, normalized points whose value is below -1.96, is more than 0.85 times the number of anomalous points in the cluster.

2.5 Geographic information systems
Geographic information systems (GIS) are a coordination of computer software, hardware, data, and personnel to help manipulate and present information tied to a spatial location. Simply put, a GIS combines layers of information about a spatial location to give users a better understanding of that place [10]. They can handle data of different sizes and formats simultaneously. Spatial features are stored in either raster or vector format. These spatial features can be associated to a database containing descriptive information about them. Spatial features along with their descriptive attributes can be used for data querying, data modeling, and data mining. Many GIS systems provide users with the capability to reuse already existing components while implementing their own. GIS have been used for modeling, analysis, and prediction in both the academic and commercial worlds.

One GIS system that has been widely used is ArcGIS. Environmental Systems Research Institute, Inc (ESRI) develops it. It is essentially a suite of software tools for cartographic applications. The tools can be used for various tasks such as to create, edit, visualize, model, manage and browse geographic data. It also provides users with a built
in Visual Basic editor (VBA) that can be used to build customized tools. Other capabilities include Internet services, advanced analysis for databases, networks, images etc. ArcGIS is the market leader in GIS software both in the educational and commercial world. For these reasons, we developed our system as an extension to ArcGIS.

3.0 System Description

3.1 System Block Diagram
The block diagram of the software prototype appears below in Figure 7. It consists of the data preprocessing stage and the data mining stage. Data preprocessing is a crucial step whenever machine-learning algorithms are involved. It allows the learning algorithm to learn relevant features while ignoring well-known cycles and spurious inputs. As part of the data preprocessing stage the input data is clustered using the K means algorithm with K = 200. Well-known cycles are then removed and anomalies in the input data are found using the techniques described in section 2.4. The preprocessed data is then fed to the data mining stage.
The data mining stage mines raw and anomalous data. If mining raw data, statistical parameters for each cluster such as mean are calculated and used to find correlations among clusters. If mining anomalies, the clusters are divided into positively or negatively anomalous clusters using techniques described in section 2.4. This data is then used to find causalities in anomalous data that occur frequently in the spatio-temporal domain. The data mining stage is further enhanced by the presence of a user feedback system. As the user interacts with the system, it generates rules the user is most likely interested in. The output is in the form of easy to read spatio-temporal association rules. The rules occur in two forms. When mining raw data the rules are of the form

\[
\text{IF } (\text{AVG} \ (\text{cluster}_1, \ t + t_1)) \ \text{THEN} \ (\text{AVG} \ (\text{cluster}_2, \ t + t_2))
\]
This rule is to be interpreted as “the sequence of averages of cluster1 is highly correlated to the translated sequence of averages of cluster2”. The second sequence is translated by $t_2 - t_1$ in the temporal domain. The fitness is given by the correlation of these two sequences.

If mining for anomalies the rules are of the form

IF [ANOMALY[cluster1, $t_1$, +] AND ANOMALY[cluster2, $t_2$, -]]

THEN [ANOMALY[cluster3, $t_3$, +]]

The above rule is interpreted as ‘a positive anomaly occurs in cluster3 at time $t_3$ whenever a positive anomaly occurs in cluster1 at time $t_1$ and a negative anomaly occurs in cluster2 at time $t_2$’. The times ‘$t_1$’, ‘$t_2$’ and ‘$t_3$’ are variables that are varied incrementally across all possible time values such that the initial time gaps among variables are constant. Instead of ‘AND’ other Boolean operators such as ‘OR’ and ‘XOR’ are also used. The ‘+’ in the rule indicates that a positive anomaly occurred in the cluster at a particular time and the ‘-’ indicates a negative anomaly. The fitness is given by the confidence of the rule as described in section 2.1. The rest of this section describes the GUI of software system in detail.

3.2 User Interface Description
The cluster viewer tab shown in figure 8 below is used to cluster the data into 200 regions.
Figure 10: Cluster Viewer, the starting step of the data mining process where data is clustered and viewed

The user then clicks the ‘Goto Datamining’ button at the bottom of the tab to go to the spatio-temporal datamining software system. The data mining software system starts with the ‘Parameter Settings’ panel. It is divided into three sections: ‘Clusters’; ‘Cycles and anomalies’; and ‘Data mining stage’ as shown below in figure 9.
Figure 11: Parameter Settings panel, the second step in the data mining process where the user does additional preprocessing and sets the genetic parameters.

The ‘Clusters’ section displays all the different clusters. The ‘Cluster Membership’ is the path to the file, which contains information pertaining to the label of each data point.

The ‘Cycles and Anomalies’ section is a data preprocessing stage, which the user, if he decides to mine anomalies, needs to go through. The various parameters that need to be set are

- Input File Name: The file path for the input files.
- Output File Name: The file path where the output is stored.
- Number of Rows: The number of rows of data.
• Number of Columns: The number of columns of data

• Number of Time Instances: The number of snapshots of the data across time.

• Percentile for anomaly: The percentile beyond which data is considered anomalous.

Once the user clicks the ‘Remove Cycles and find Anomalies’ button the system removes the well known cycles and finds the anomalies using methods described in section 2.4.

By clicking the ‘Continue to Datamining stage’ button the user proceeds to the ‘data mining stage’.

The ‘data mining stage’ is further divided into ‘Tree construction information’ and ‘Genetic Process Parameters’ sections. The ‘Tree construction information’ section deals with setting the required parameters for the parse tree construction. The ‘Select clusters interested in’ list box allows users to select the clusters he/she is interested in working with. The other parameters that need to be set are

• Members: The number of trees in a population

• Temp. Members: The maximum number of additional trees that can be produced as a result of crossover.

• Max. Nodes in Tree: This parameter controls the size of the tree that is randomly constructed initially.

• Time gap: The time gap specifies minimum time gap between the LHS and RHS of an association rule.

• Num Operators: This parameter specifies the number of operators to be used in the tree construction phase. If it is 3 then ‘and’, ‘or’ and ‘xor’ operators will be used. If it is 2 then ‘and’ and ‘or’ will be used and if it is 1 only ‘and’ is used.
The ‘Genetic Process parameters’ section handles information needed for the genetic program to execute. The various parameters that need to be set are

- **Num Iterations**: The number of iterations the genetic program will run before completion.
- **Iteration increment**: Increments at which the genetic program prompts user for feedback. If its value is the same as the number of iterations, then the user is not prompted for feedback.
- **Percentage Anomalous points**: The minimum number of points within a cluster of same anomalous type (’+’ / ’-’) for the cluster to be considered to be anomalous.
- **Min. fitness to write to file**: Since the program runs till the number of iterations mentioned, there is a very small chance for highly fit solutions that are produced in the middle of the computation process to be lost. This parameter specifies the minimum fitness of a tree to be written into a file in order to prevent it from being lost.
- **Crossover Rate**: The probability for two trees to cross.
- **% Tree to Cross**: The genetic program orders the trees in descending order of their fitness. This parameter tells us how many trees from the top to be considered for crossover. The program therefore only crosses highly fit trees.
- **Mutation Rate**: The probability for two trees to mutate.
- **% Confidence**: The percentage of the confidence to be taken into account for calculating fitness.
- **Min. Support**: The minimum support of an association rule for it to be considered interesting.
The percentage of the User feedback to be taken into account while evaluating fitness.

The user then clicks either the 'Mine Raw Data' button or 'Mine for Anomalies' button at the bottom of the 'Parameter Settings' panel as shown in figure 9.

Figure 12: Panel that keeps track of current population and user feedback.

The 'Mine for Anomalies' and 'Mine Raw Data' display the panel shown above in figure 10. Initially, the 'Start' button is enabled and the 'Exit' button is disabled. The genetic program executes once the user clicks the 'Start' button. The current iteration number appears at the top left. After an iteration increment, the current population appears in the 'Association rules in current iteration' frame. The rule number along with the spatio-temporal association rule appears. This is followed by the fitness, confidence,
confidence count (frequency of LHS), support, and a textbox to enter user feedback for that particular rule. User feedback reduces the time taken for a genetic program to arrive at solutions. It also produces association rules the user is interested in. The program is run until the next iteration increment by clicking on the ‘Next Iteration’ button. In case the user wants to quit, clicking the ‘Cancel’ button enables the ‘Exit’ button and disables the ‘Start button’. The ‘Association rules with high fitness’ frame displays those trees that have high fitness values while the genetic program is in execution. The results are stored in an html format.

4.0 Experiments

4.1 Description of datasets
As a case study, we conducted several experiments to mine average monthly earth temperature from 1982 Jan to 2000 Dec. In total, we have 228 different files corresponding to every month during this period. The dimensions of each file are 720x1440. They are stored in 2-bytes/pixel big-endian format.

4.2 Methodology
The data was clustered into 200 clusters using the K-Means algorithm described in section 2.3. Well-known cycles were removed and data beyond the 95% were considered to be anomalies. Different experiments were conducted to test the efficiency and accuracy of the data mining system. Since the input spatial-temporal data is the average monthly earth temperatures, we decided to find regions that are highly correlated to each other. Other experiments were conducted to find out if the system was capable of mining anomalies. The user feedback system was tested to check if the system outputs rules the
user is interested in. Finally, we conducted some experiments to confirm if the system can mine rules in a user-defined area of interest. The accuracy of the output rules was verified by comparing it with the raw input data. The standardized $Z$ values of average cluster temperatures across all possible time instances is taken to represent the behavior of points belonging a cluster. The parameters used for all these experiments are given below in table 4.

Table 2: Parameter values used for experiments

<table>
<thead>
<tr>
<th>Clusters</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Members</td>
<td>50</td>
</tr>
<tr>
<td>Temp. Members</td>
<td>8</td>
</tr>
<tr>
<td>Max nodes in Tree</td>
<td>3</td>
</tr>
<tr>
<td>Time Gap</td>
<td>Varied in multiples of 6 months</td>
</tr>
<tr>
<td>Num operators</td>
<td>3</td>
</tr>
<tr>
<td>Num Iterations</td>
<td>30</td>
</tr>
<tr>
<td>Iteration increment</td>
<td>30*</td>
</tr>
<tr>
<td>% anomalous points</td>
<td>0.85</td>
</tr>
<tr>
<td>Min fitness to write to file</td>
<td>0.75</td>
</tr>
<tr>
<td>Crossover rate</td>
<td>0.333</td>
</tr>
<tr>
<td>% trees to cross</td>
<td>0.333</td>
</tr>
<tr>
<td>Mutation rate</td>
<td>0.333</td>
</tr>
<tr>
<td>% Confidence</td>
<td>0.5</td>
</tr>
<tr>
<td>% User interest</td>
<td>0.5</td>
</tr>
<tr>
<td>Min support</td>
<td>$\sim 1%$ (rule occurs $\geq 2$ times)</td>
</tr>
</tbody>
</table>
While mining with user feedback, the iteration increment was changed from its usual value of 30 to 1 so that the program prompts the user after each iteration. Though the user, in principle, can be prompted for a maximum of 30 times, this rarely happened. At the most, the user was prompted 3-4 times before a highly fit individual was discovered.

The next subsections describe some of the results obtained.

4.3 Results
The aim of these experiments is to prove that the system does mine raw and anomalous data to output interesting association rules. Some results appear below. The output rules are interpreted as described in section 3.1.

4.3.1 Mining anomalous data
The aim of this experiment is to check if the system is capable of mining anomalies accurately. The results obtained are then compared to the raw data.

Some of the spatio-temporal rules obtained while mining anomalies are

- IF [ANOMALY[167, t, -]] THEN [ANOMALY[0, t, -]]

  CON:1 CNT:8

This rule is depicted in figure 11 and can be interpreted as ‘A negative anomaly occurs in cluster 0 if a negative anomaly occurs in cluster 167’. The black cluster in figure 11 is the cluster 167 and the cluster in white is the cluster 0. The LHS of the rule occurred 8 times, of which the RHS occurred 8 times giving the rule a confidence of 1. The above rule is a spatial rule and gives the spatial relationship among sea clusters.
The Z values of the average temperatures of clusters 167 and 0 are plotted below in figure 12. It does indicate that a negative anomaly occurs in cluster 0 if a negative anomaly occurs in cluster 167.

Figure 14: Plot of the standardized Z values of the average temperatures of clusters 167 and 0
Another rule obtained is

- IF [(ANOMALY[76,t,-] AND ANOMALY[193,t,+]) ] THEN [ANOMALY[175,t,-]]
  
  CON: 1 CNT: 4

This rule is depicted in figure 13 and can be interpreted as ‘If a negative anomaly occurs in cluster 76 and a positive anomaly in 193 then a negative anomaly occurs in cluster 175’. The black clusters in figure 13 are the clusters 76 and 193 and the cluster in white is the cluster 175. The LHS of the rule occurred 4 times of which the RHS occurred 4 times giving the rule a confidence of 1. The above rule is a spatial rule and gives the spatial relationship between clusters in the oceans and clusters on land.

Figure 15: Depiction of relationship among regions in Atlantic, Pacific and South America, Africa

The Z values of the average temperatures of clusters 76 and 175 are plotted below in figure 14. It indicates the occurrence of negative anomalies in cluster 175 whenever a negative anomaly occurs in cluster 76.
4.3.2 Mining raw data

The aim of this experiment is to check whether the system is capable of mining raw data accurately and discovering highly correlated clusters. The results obtained are then compared to the raw data and are plotted to verify the accuracy of the results obtained.

Some of the rules obtained are

- IF AVG[47,t+14] THEN AVG[120,t+32]
  
  fitness: -0.882813

This rule is depicted in figure 15 and can be interpreted as 'the average temperature of cluster 47 at time t+14 is negatively correlated to the average temperature of cluster 120 at time t + 32'. The fitness of this rule, which is given by the correlation of the two sequences of the average temperatures is -0.882813. The black cluster in figure 15 represents the cluster 47 whereas the white cluster represents the cluster 120. The plot of the average cluster temperatures is given below in Figure 16.
The graph in Figure 16 is a plot of the average temperatures of the clusters 47 and 120 as obtained from the raw input data. The blue line represents the average temperatures of cluster 47 and the pink line represents the average temperatures of cluster 120 shifted by 18 (= 32 - 14). Clearly, these two sequences are negatively correlated.
Another such obtained rule is

- IF AVG[57,t+3] THEN AVG[174,t+11]

  \textit{fitness: -0.889224}

This rule is depicted in figure 17 and can be interpreted as 'the average temperature of cluster 57 at time t+3 is negatively correlated to the average temperature of cluster 174 at time t + 11'. The fitness of this rule, which is given by the correlation of the average temperatures, is \textit{-0.889224}. The black cluster in figure 17 represents the cluster 57 whereas the white cluster represents the cluster 174. The plot of the average cluster temperatures is given below in Figure 18.
Figure 19: Depiction of relationship among regions in Northern Canada, Alaska, Russia and Pacific, Atlantic and Indian oceans

Figure 18 below is a plot of the average temperatures of the clusters 57 and 174 as obtained from the raw input data. The blue line represents the average temperatures of cluster 57 and the pink line represents the sequence of average temperatures of cluster 174 that is shifted by 8 (= 11 − 3). From the chart, it is evident that these two sequences are negatively correlated.
4.3.3 Mining with user feedback

The purpose of this experiment is to check if the user feedback makes any difference in the mining process. The user feedback is used to calculate the ‘interestingness’ of a rule. The feedback values range from 0 to 5 with 5 being the highest. High feedback increases the ‘interestingness’/fitness of a rule. The clusters 0, 20, 27, 37, 40, 77, 100, 133, 140 and 197 were given a high user feedback of 5 whenever they occurred in a rule. All other rules that did not contain the above clusters got a user feedback of 1. As expected the output did contain regions the user was highly interested in. Some of the discovered rules are given below.

- IF [ANOMALY[140,t, -]] THEN [ANOMALY[27,t, -]]
  
  CON:0.875 CNT:8

This rule appears in figure 19 below. It can be interpreted as ‘If a negative anomaly occurs in cluster 140 then a negative anomaly occurs in cluster 27’. The black cluster in figure 19 is the cluster 140 and the cluster in white is the cluster 27. The LHS of the rule
occurred 8 time of which the RHS occurred 7 times giving the rule a confidence of 0.875.

The above rule is a spatial rule.

**Figure 21: Depiction of relationship among regions in Indian Ocean and Pacific, Atlantic and Indian Ocean**

The $Z$ values of the average temperatures of the clusters 140 and 27 are plotted below in figure 20. The blue line represents the $Z$ values of the average temperatures of cluster 140 and the pink line represents the $Z$ values of the average temperatures of cluster 27. It does indicate that a negative anomaly occurs in cluster 27 whenever a negative anomaly occurs in cluster 140.
Anomaly occurrence in clusters 140 and 27

Figure 22: Plot of the standardized Z values of the average temperatures of clusters 140 and 27.

Another such rule obtained is

- IF AVG[20,t+5] THEN AVG[197,t+23]

fitness: -0.856869

This rule is depicted in figure 21 and can be interpreted as ‘the average temperature of cluster 20 at time t+5 is negatively correlated to the average temperature of cluster 193 at time t + 23’. The fitness of this rule, which is given by the correlation of the average temperatures is −0.856869. The black cluster represents the cluster 20 whereas the white cluster represents the cluster 197.
Figure 23: Depiction of relationship among regions in Pacific, Atlantic and northern parts of Russia

The graph in figure 22 below is a plot of the average temperatures of the clusters 20 and 197 as obtained from the raw input data. The blue line represents the Z values of the average temperatures of clusters 20 and the pink line represents the Z values of the average temperatures of cluster 197 shifted to left by 18 (= 23 – 5). Clearly, the two sequences are negatively correlated.
4.4.4 Mining in an area of interest
The white dotted rectangles shown in the figures 23, 24, and 25 are areas of interest. Clusters occurring in the region bounded by the white dotted rectangles are clusters of interest. The goal of this experiment is to test if the system can find any rules pertinent to a user-defined area of interest.

Some of the discovered rules are

- IF [ANOMALY[75,t+1, +]] THEN [ANOMALY[114,t+6, +]]
  CON:1 CNT:3

This rule is depicted in figure 23 and can be interpreted as 'If a positive anomaly occurs in cluster 75 at time t+1 then a positive anomaly occurs in cluster 114 at time t+6'. The black cluster in figure 23 is the cluster 75 and the cluster in white is the cluster 114. The LHS of the rule occurred 3 time of which the RHS occurred 3 times giving the rule a confidence of 1. The above rule is a spatio-temporal rule.
Another such rule is

- IF [ANOMALY[195,t+4, +]] THEN [ANOMALY[114,t+6, +]]

CON:0.75 CNT:8

This rule is depicted in figure 24 below. It can be interpreted as ‘If a positive anomaly occurs in cluster 195 at time t+4 then an positive anomaly occurs in cluster 114 at time t+6’. The black cluster in figure 24 is the cluster 195 and the cluster in white is the cluster 114. The LHS of the rule occurred 8 time of which the RHS occurred 6 times giving the rule a confidence of 0.75. The above rule is a spatio-temporal rule.
The results from the above experiments proves that the data mining system does what it is supposed to do and it is capable of mining both raw as well as anomalous data to discover some interesting spatio-temporal relationships.

5.0 Discussion & Future work
The results obtained depict the climatic relationship among regions on land and sea. We discovered relationships between land-land, land-sea, and sea-sea regions. For example, there seems to be a significant relationship among sea regions of the coast of South America and Africa (figure 11). This is a sea-sea relationship. Another discovered rule is the relationship among land regions in the northern part of Africa, regions in Saudi Arabia, Iran, Afghanistan, and parts of Spain, France (figure 15). This is a land-land relationship. A relationship among regions in Alaska, northern parts of Canada and Russia and sea regions close to Antarctica has been found (figure 17). This is a land-sea relationship.

Figure 26: Depiction of relationship among regions in Canada, Europe, Asia and Pacific, Atlantic off the coast of Panama
relationship. It also depicts relationship among regions in different hemispheres. Experiments to discover rules for El Nino and El Nina have produced some results, though their usefulness needs to be assessed by an expert. Figure 24 depicts the relationship among regions in Canada, parts of Europe along the same latitude and sea regions off the coast of Panama. Empirical results also show that regions close to each other exhibit similar anomalous trends. All these demonstrate the potential of the developed system.

However, there is room for improvement. The system has to be modified so it can work with databases. The output is in the form of association rules. It needs to be automatically converted into visual images for the user and expert to understand easily. The genetic process is slow and there are opportunities for improving the running time of the algorithm. The data, as mentioned above, ranges from 1982 to 2000. In order to mine for spatio-temporal dependencies of anomalies more data is needed. Other techniques for data mining mentioned above need to be implemented to assess the value of link analysis in spatio-temporal data mining. We also need to collaborate with an expert to assess the ‘interestingness’ and usefulness of the obtained rules.

Another avenue for future work would be to find relationships between land and sea clusters and discover the time lags and leads in their spatio-temporal trends. By obtaining and working with other data pertaining to wind speeds, ocean currents, digital elevation models (DEM) etc; we should be able to get better results.

6.0 Conclusions

The focus of this thesis has been to design and implement a flexible system for spatio-temporal data mining. We have successfully implemented the above system as an
extension to ArcGIS. The K-Means clustering algorithm proved to be an important step in the data mining process. It was able to provide a high level visual description of the entire data sets. Evolutionary computation algorithms, though a bit slow, were able to find many spatio-temporal rules automatically. User feedback, being a significant process in this thesis, was able to direct the computation process towards the clusters in which he/she was interested. The results we have obtained signify the feasibility of applying evolutionary computation to the task of spatio-temporal data mining as well.
7.0 References


