Intelligent Systems Lab
Computer Science and Engineering
University of Colorado at Denver

Computer Science
University of Colorado at Boulder

The 2002 Computer Science Seminars
University of Colorado at Denver

Data Mining and Knowledge Discovery

Lukasz Kurgan

Data Mining and Knowledge Discovery

- **Knowledge Discovery (KD) is a nontrivial process of identifying**
  - **valid**
  - **novel**
  - **potentially useful**
  - **and ultimately understandable**

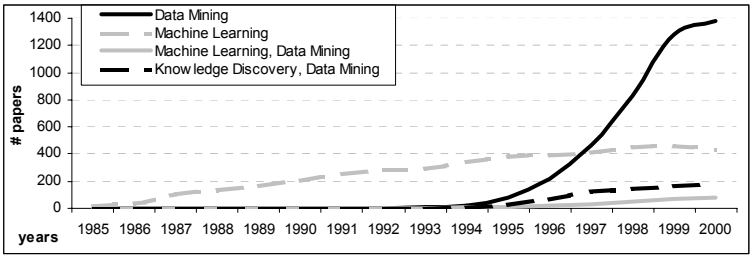
  patterns from large collections of data*

- **One of the KD steps is Data Mining (DM)**
  - **concerned with the actual extraction of knowledge from data**

* Fayyad, U.M., Piatesky-Shapiro, G., Smyth, P., and Uthurusamy, R., *Advances in Knowledge Discovery and Data Mining*, AAAi/MIT Press, 1996

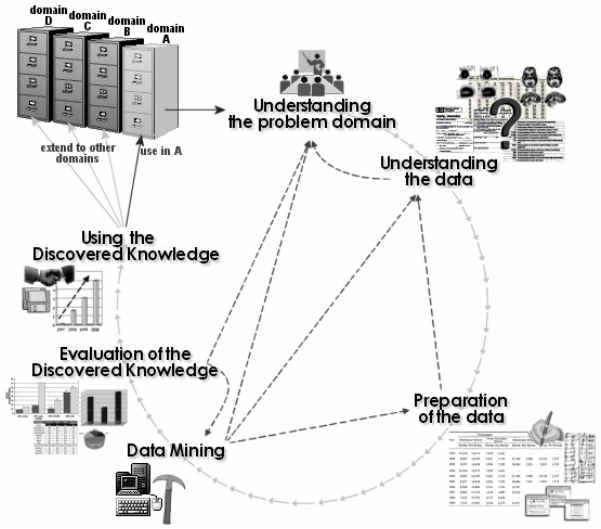
Data Mining and Knowledge Discovery

- Evolution of Data Mining and Knowledge Discovery
  - study using online research service Axiom®
  - exponentially growing field, with a strong emphasis on applications
    - incorporation of existing tools and algorithms
    - trends include
      - machine learning
      - temporal and spatial data analysis
      - XML-related technology
      - data warehousing
      - high performance systems
      - visualization



Data Mining and Knowledge Discovery

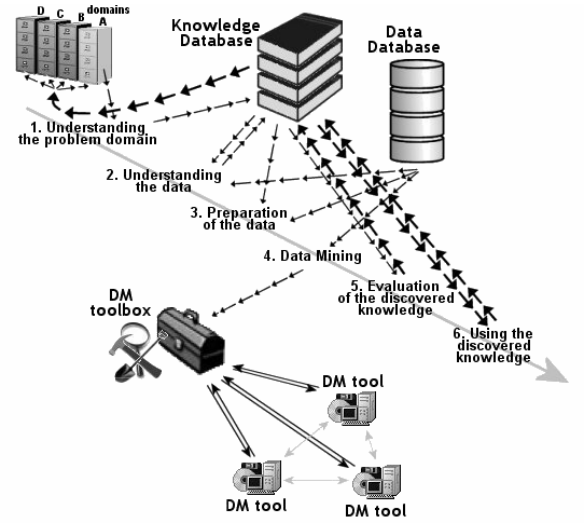
- DMKD process model
  - Understanding the problem domain
  - Understanding the data
  - Preparation of the data
  - Data mining
  - Evaluation of the discovered knowledge
  - Using the discovered knowledge



Cios, K. J., & Kurgan, L., Trends in Data Mining and Knowledge Discovery, In: Pal N.R., Jain, L.C. and Teoderesku, N. (Eds.), Knowledge Discovery in Advanced Information Systems, Springer, to appear, 2002

Data Mining and Knowledge Discovery

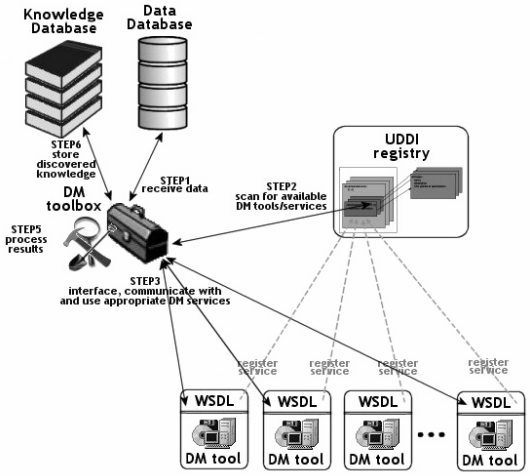
- **DMKD process model and XML**
  - XML for data transportation and storage
    - can be stored using XML enabled DBMS or native XML DBMS
  - Simple Object Access Protocol (SOAP): XML/HTTP based communication protocol
  - Predictive Model Markup Language (PMML): XML–based language used to define predictive data models
  - Universal Description Discovery and Integration (UDDI): XML based, platform–independent framework for describing, discovering and integrating web services



Cios, K. J., & Kurgan, L., Trends in Data Mining and Knowledge Discovery, In: Pal N.R., Jain, L.C. and Teoderesku, N. (Eds.), Knowledge Discovery in Advanced Information Systems, Springer, to appear, 2002

Data Mining and Knowledge Discovery

- **DM Toolbox Architecture**
  - No single DM tool performs well on different all types of data
  - Uses XML based technologies like XML-RPC, SOAP, PMML, WSDL, and UDDI
  - Execution model
    1. accepts the data from a user
    2. dynamically checks availability and description of online-enabled DM tools using UDDI
    3. invokes the tools that can provide meaningful results for currently processed data
    4. serves data to the chosen DM tools for processing
    5. receives the results
    6. analyses and integrates the results, and serves them back to the user



Kurgan, L., Cios, K.J., & Trombley, M., The WWW Based Data Mining Toolbox Architecture, submitted to the 6th International Conference on Neural Networks and Soft Computing, 2002

Preparation of the Data

- The key step in the DMKD process
  - success of the DMKD process depends on this step
  - usually consumes between 25 and 65% of the project time
  - deciding which data will be used by data mining tools in the next step
  - preparation of the data
    - sampling of the data
    - correlation and significance tests
    - data cleaning
      - removing or correcting incomplete records, noise, missing values, etc
    - derivation of new attributes
      - discretization
    - data integration
    - feature selection and extraction algorithms
      - reducing dimensionality

Data

- attributes in columns
- examples in rows
- class labels define the concept described by the data
- heart disease data

Class:	1 (present), 2 (absent)
Age:	continuous
Sex:	0,1
Chest Pain Type:	1,2,3,4
Resting Blood Pressure:	continuous
Serum Cholesterol:	continuous
Fasting Blood Sugar:	0,1
Resting Electr Results:	0,1,2
Max Heart Rate:	continuous
Exercise Induced Angina:	0,1
Old peak:	continuous
Slope Exercise ST:	0,1,2,3
Number Major Vessels:	continuous
Thallium:	3,6,7

2	70	1	4	130	322	0	2	109	0	2.4	2	3	3
1	67	0	3	115	564	0	2	160	0	1.6	2	0	7
2	57	1	2	124	261	0	0	141	0	0.3	1	0	7
1	64	1	4	128	263	0	0	105	1	0.2	2	1	7
1	74	0	2	120	269	0	2	121	1	0.2	1	1	3
1	65	1	4	120	177	0	0	140	0	0.4	1	0	7
2	56	1	3	130	256	1	2	142	1	0.6	2	1	6
2	59	1	4	110	239	0	2	142	1	1.2	2	1	7

Discretization

- Discretization transforms a continuous attribute values into a finite number of intervals and associates with each interval a numerical, discrete value

- Supervised discretization
  - discretizes attributes by taking into account the class labels assigned to examples

Discretization

- CAIM (Class-Attribute Interdependency Maximization) discretization algorithm
  - maximizes mutual class-attribute interdependence
  - generates possibly the smallest number of intervals for a given continuous attribute
  - tested on several well-know benchmarking datasets
  - compared with six other state-of-the-art discretization algorithms

Kurgan L. & Cios K.J., Discretization Algorithm that Uses Class-Attribute Interdependence Maximization, *Proceedings of the 2001 International Conference on Artificial Intelligence (IC-AI 2001)*, pp.980-987, Las Vegas, Nevada, 2001
Kurgan, L., & Cios, K.J., CAIM Discretization Algorithm, submitted to IEEE Transactions on Data and Knowledge Engineering, 2001

Discretization

- The algorithm consists of these two steps:
  - initialization of the candidate interval boundaries and the initial discretization scheme
  - consecutive additions of a new boundary that results in the locally highest value of the CAIM criterion
  - uses greedy approach, which searches for the approximate optimal value of the CAIM criterion by finding its local maximum values
  - computationally inexpensive and well approximates finding the optimal discretization scheme
    - shown by the results

The 2002 Computer Science Seminars, University of Colorado at Denver Lukasz Kurgan

Discretization

- CAIM discretization criterion

$$CAIM(C,D \mid F) = \frac{\sum_{i=1}^{n} \frac{\max_i^2}{M_{ir}}}{n}$$

where:

n is the number of intervals

i iterates through all intervals, i.e. i=1,2,...,n

max_i is the maximum value among all q_{ir} values (maximum value within the i^{th} column of the quanta matrix), r=1,2,...,*S*

M_{ir} is the total number of continuous values of attribute *F* that are within the interval $(d_{r-1}, d_r]$

Quanta matrix:

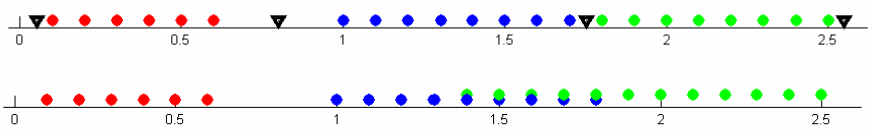
Class	Interval					Class Total
	$[d_0, d_1]$	...	$(d_{r-1}, d_r]$	...	$(d_{n-1}, d_n]$	
C_1	q_{11}	...	q_{1r}	...	q_{1n}	M_{1+}
:	:	...	:	...	:	:
C_i	q_{i1}	...	q_{ir}	...	q_{in}	M_{i+}
:	:	...	:	...	:	:
C_S	q_{S1}	...	q_{Sr}	...	q_{Sn}	M_{S+}
Interval Total	M_{+1}	...	M_{+r}	...	M_{+n}	M

Discretization

Algorithm	#intervals	CAIR value
Equal Width	4	0.59
Equal Freq.	4	0.66
Paterson-Niblett	12	0.53
Max. Entropy	4	0.47
IEM	4	0.74
CADD	3	0.79
CAIM	3	0.82

Iris plants data

- red = Iris-setosa
- blue = Iris-versicolor
- green = Iris-virginica



Discretization

- test performed using 8 datasets
  - about 1300 experiments
  - average rank used to show the results

Criterion	Discretization Method	RANK mean
CAIR mean value through all intervals	Equal Width	4.3
	Equal Frequency	4.8
	Paterson-Niblett	3.6
	Maximum Entropy	6.1
	CADD	3.3
	IEM	3.1
	CAIM	2.0
total # of intervals	Equal Width	4.6
	Equal Frequency	4.6
	Paterson-Niblett	3.9
	Maximum Entropy	4.4
	CADD	3.6
	IEM	2.3
	CAIM	1.3
time [s]	Equal Width	1.0
	Equal Frequency	1.9
	Paterson-Niblett	6.4
	Maximum Entropy	3.5
	CADD	6.6
	IEM	4.3
	CAIM	4.3

Discretization Method	ML algorithm	RANK mean	ML algorithm	RANK mean
Equal Width	CLIP4 accuracy	4.6	CLIP4 # rules	3.8
Equal Frequency		4.8		3.5
Paterson-Niblett		4.3		2.6
Maximum Entropy		5.3		3.6
CADD		3.9		3.5
IEM		2.9		3.0
CAIM		1.8		2.1
Equal Width	C5.0 accuracy	5.3	C5.0 # rules	4.9
Equal Frequency		6.0		5.8
Paterson-Niblett		4.3		3.3
Maximum Entropy		5.6		5.8
CADD		5.4		4.9
IEM		3.3		2.5
CAIM		2.1		1.9
Built-in		3.3		3.1

Discretization

- Summary
  - can be used with any class-labeled data
  - maximizes interdependence between class labels and discrete intervals
  - generates the smallest number of intervals for a given continuous attribute
  - automatically selects the number of intervals in contrast to many other discretization algorithms
  - works quickly enough to be applicable to real-life problems
  - the tests show that when the proposed algorithm is applied as a front-end tool, it improves the performance of supervised ML algorithm

Data Integration

- Provide unified access to semantically and structurally diverse information sources
- XML data
  - content
    - numbers, character strings, images, etc.
  - context
    - describes what role the content plays
    - defines a standard to add markup (tags) to identify structure of a documents
    - e.g. a rule is built out of selectors, a selector is a pair of attributes (name and value))

Data Integration

- XMapper system
  - provides semantic mapping that enables integration of information between two XML data sources

```
<hea1>
  <example>
    <class>1</class>
    <Age>35</Age>
    <Sex>0</Sex>
    <Chest Pain Type>4</Chest Pain Type>
    <Resting Blood Pressure>138</Resting Blood Pressure>
    <Serum Cholesterol>183</Serum Cholesterol>
    <Fasting Blood Sugar>0</Fasting Blood Sugar>
    <Resting Electr Results>0</Resting Electr Results>
    <Max Heart Rate>182</Max Heart Rate>
    <Exercise Induced Angina>0</Exercise Induced Angina>
    <Old peak>1.4</Old peak>
    <Slope Exercise ST>1</Slope Exercise ST>
    <Number Major Vessels>0</Number Major Vessels>
    < Thallium >3</ Thallium >
  </example>
</hea1>
```

```
<hea3>
  <example>
    <class>2</class>
    <FBSugar>0</FBSugar>
    <REResults>0</REResults>
    <SlopePESTS>1</SlopePESTS>
    <S>1</S>
    <CPT>2</CPT>
    <MaxHR>141</MaxHR>
    <EIA>0</EIA>
    <OP>0.3</OP>
    <MajVesselsNo>0</MajVesselsNo>
    <Years>57</Years>
    <RBPress>124</RBPress>
    <SChol>261</SChol>
  </example>
</hea3>
```

Kurgan, L., Swiercz, W., & Cios, K.J., Semantic Mapping of XML Tags using Inductive Machine Learning, submitted to the 2002 International Conference on Machine Learning and Applications, Las Vegas, 2002

The 2002 Computer Science Seminars, University of Colorado at Denver Lukasz Kurgan

Data Integration

- mapping discovered by the XMapper system
  - 1-to-1 mappings
  - unmatchable tags

XML1	XML2
class	class
Sex	S
Example	example
Resting Blood Pressure	RBPress
Serum Cholestoral	SChol
Max Heart Rate	MaxHR
Resting Electr Results	REResults
ChestPainType	CPT
FastingBloodSugar	FBSugar
Slope Exercise ST	SlopePESTS
Exercise Induced Angina	EIA
Number Major Vessels	MajVesselsNo
Age	Years
Old peak	OP
Thallium	

Data Integration

- **XMapper**
  - **every tag from the two XML documents extracts a vector of features that describes its properties**
  - **distance between the vectors is calculated for every pair of tags, which belong to different sources**
  - **1-to-1 mappings are generated by sequentially finding pairs of tags with the minimum distance**

Data Integration

- **XMapper consists of two modules**
  - ***constraints analysis* module that extracts**
    - **properties of data stored in XML sources, like data types, length, number of null values etc.,**
    - **structural information, like number of children nodes, data types of children nodes etc.**
  - ***learning module***
    - **extract information about relationship between attributes used in both data sources**
    - **uses inductive ML algorithm DataSqueezer**

Data Integration

- Tested
  - 7 artificial and 3 real-life domains
  - XML documents within a domain differed in tag names, tag order and structure

domain	# of sources	# experiments (source pairs)	mean accuracy [%]
cmc	3	3	100.0
hea	3	3	88.1
iris	2	1	100.0
mush	3	3	85.5
pid	3	3	85.1
spect	2	1	65.2
thy	2	1	60.0
mean for artificial domains			83.4
course	5	10	85.2
faculty	5	10	100.0
realest	5	10	60.0
mean for real-life domains			81.7
total mean			82.6

comparison between the LSD and XMapper
- XMapper's average accuracy 81.7%
- LSD's average accuracy 79.6%

	course	faculty	realest
XMapper	85%	100%	60%
LSD	76%	92%	71%

The 2002 Computer Science Seminars, University of Colorado at Denver Lukasz Kurgan

Data Integration

- XMapper
  - fully automated
  - uses standalone XML only
    - no need for creating DTD or Schema files that describe the XML sources
  - generates mappings between all, including non-leaf, tags
    - in contrast to the LDS system
  - returns ordered, in terms of confidence, mappings
    - significant help the user to discover incorrect mappings
  - high degree of accuracy
  - returns both matched and unmatched tags

Data Mining

- Another key step in the DMKD process
  - applies DM tools to discover new information
  - involves
    - selection of data modeling tools
    - deciding on training and test procedures
    - building the model itself
    - assessing model quality
  - DM tools include many types of algorithms, such as machine learning, rough and fuzzy sets, Bayesian methods, evolutionary computing, neural networks, clustering, association rules, etc.

Inductive Machine Learning

- Machine learning (ML)
  - the ability of a computer program to improve its own performance, based on the past experience, by generation of a new data structure that is different from an old one
    - e.g. generation of production rules from numerical or nominal data
    - generated description is explicit
      - e.g. in the form of rules or decision trees
      - it be analyzed, learned from, or modified by the user
- Induction infers generalized information, or knowledge, by searching for regularities among the data

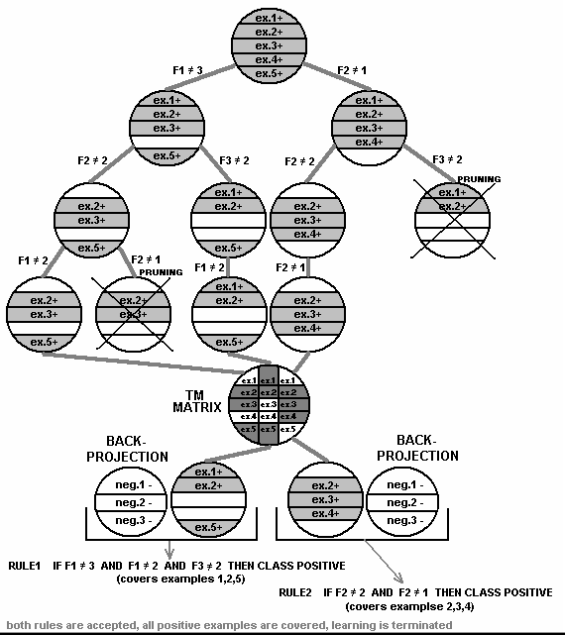
ML Algorithms

- Two inductive ML algorithms
  - both used to perform classification
    1. An algorithm generates a data model using historical data
    2. The model is used to classify unseen data into predefined categories
  - CLIP4
    - generates inequality rules
      - IF A!=B THEN C
    - uses set covering problem to generate rules
    - hybrid of decision tree and rule algorithms
  - DataSqueezer
    - generates equality rules
      - IF A=B THEN C
    - uses data generalization mechanisms

Cios K. J. & Kurgan L. (2001), Hybrid Inductive Machine Learning: An Overview of CLIP Algorithms, In: Jain L.C., and Kacprzyk J. (Eds.) *New Learning Paradigms in Soft Computing* pp. 276-322, Physica-Verlag (Springer)
Cios, K.J., & Kurgan, L., Hybrid Inductive Machine Learning Algorithm, submitted, 2001

CLIP4

- data is partitioned into subsets using a tree structure
- rules are generated only from subsets stored at the leaf nodes
- improved performance
  - accuracy
  - speed



CLIP4

- CLIP4's benchmarking tests
  - compared with 33 other ML algorithms
  - CLIP4 algorithm is not statistically significantly different from the algorithm that achieved the smallest error rates

error rates for the 33 ML algorithms		CLIP4 error rates	median # of leaves/rules for the 21 alg.	CLIP4 # of rules	CPU time for the 33 ML algorithms		CLIP4 CPU time [min]	CLIP4 # of selectors
min	max				min [s]	max [h]		
17.1	45.4	25.1	17.8	16.8	6.9 s	46.8 h	5.8 min	598.5

The 2002 Computer Science Seminars, University of Colorado at Denver Lukasz Kurgan

CLIP4

- The main advantages of the CLIP4 algorithm are
  - generates inequality rules
  - flexibility
    - works with highly dimensional data
      - high number of examples
      - high number of attributes
  - provides solutions to multiple learning tasks
    - generates classification rules
    - performs feature selection
    - generates feature and selector ranking

DataSqueezer

- **Intuitively easy to understand**
  - **equality production rules**

Class: home, treatment
Temperature: normal, low, high
Number Major Vessels: normal, low, high
Chest Pain Type: 1,2,3,4

home	low	normal	2
home	low	normal	3
home	normal	normal	3
home	normal	low	2
home	normal	low	1
treatment	low	high	4
treatment	low	low	4
treatment	high	normal	4

for home:
- **low, normal, * (2)**
- **normal, normal, 3 (1)**
- **normal, low, * (2)**

for treatment
- **low, * , 4 (2)**
- **high, normal , 4 (2)**

RULES for CLASS : home (2 rules)
1. **IF Temperature = normal THEN Class = home**
2. **IF Temperature = low AND Number Major Vessels = normal THEN Class = home**

RULES for CLASS : treatment (1 rule)
1. **IF Chest Pain Type = 4 THEN Class = treatment**

The 2002 Computer Science Seminars, University of Colorado at Denver Lukasz Kurgan

DataSqueezer

- **DataSqeezer's benchmarking tests**
  - **compared with 33 other ML algorithms**
  - **DataSqeezer algorithm is not statistically significantly different from the algorithm that achieved the smallest error rates**
  - **only 7 out of 33 algorithms achieved smaller error rates and were faster then the DataSqeezer at the same time**
  - **generates very compact rules**
    - **only 3.5 selectors / rule while generating similar number of rules**

error rates for the 33 ML algorithms		DS error rates	median # of leaves/rules for the 21 alg.	DS # of rules	CPU time for the 33 ML algorithms		DS CPU time [s]	DS # of selectors	DS # of selectors/ rule
min	max				min [s]	max [h]			
17.1	45.4	25.4	17.8	22.0	6.9 s	46.8 h	38.4 s	80.5	3.5

DataSqueezer

- Generates very compact rules
- Very easy to implement
  - the only one data structure needed is a table
- Can be windowed
  - generates rules from pockets of data
  - results in speed-up and scalability

Applications

- The DMKD process model was used in several of ours projects:
  - "Data Mining and Knowledge Discovery" project sponsored by Ohio Aerospace Institute and GE Aircraft Engines
    - development of a software for engine life time prediction
  - Design of an automated diagnostic system developed in cooperation with Medical College of Ohio
    - system for computerized diagnosing of Single Proton Emission Computed Tomography (SPECT) images of myocardial perfusion
  - Design of a system for intelligent analysis of temporal data developed in cooperation with Denver Children's Hospital
    - analysis of the cystic fibrosis
    - currently in progress

Cios, K.J., Kurgan, L., Mitchell, S, Bailey, M., Duckett, D., & Gau, K., Report for the OAI Phase I Collaborative Core Research Project on Data Mining and Knowledge Discovery, the 2000 Ohio Aerospace Institute (OAI) Collaborations Forum, Cleveland, OH, 2000
Kurgan, L.A., Cios, K.J., Tadeusiewicz, R., Ogiela, M. & Goodenday, L.S., Knowledge Discovery Approach to Automated Cardiac SPECT Diagnosis, *Artificial Intelligence in Medicine*, vol. 23/2, pp.149-169, 2001

Applications

- **Our algorithms were used in several applications**
  - **Design of an automated diagnostic system for SPECT heart images**
    - **CLIP4** used to develop a set of rules for computing the diagnosis
  - **"ML for record linkage" project sponsored by US Air Force**
    - XML, RDF (resource description framework), and ML based approach to intelligent record linking and searching of the World Wide Web of the future for the most relevant information to the user
    - **CLIP4** used as the ML technology
  - **Design of a system for intelligent analysis of temporal data**
    - **CAIM** used to discretize continuous attributes
    - **DataSqueezer** used to generate rules for predefined temporal intervals
      - rule tables
      - attribute and selector ranking tables

References

Cios, K.J., Kurgan, L., Mitchell, S, Bailey, M., Duckett, D., & Gau, K., Report for the OAI Phase I Collaborative Core Research Project on Data Mining and Knowledge Discovery, the 2000 Ohio Aerospace Institute (OAI) Collaborations Forum, Cleveland, OH, 2000

Cios K. J. & Kurgan L., Hybrid Inductive Machine Learning: An Overview of CLIP Algorithms, In: Jain L.C., and Kacprzyk J. (Eds.) *New Learning Paradigms in Soft Computing* pp. 276-322, Physica-Verlag (Springer), 2001

Cios, K.J., & Kurgan, L., Hybrid Inductive Machine Learning Algorithm, submitted, 2001

Cios, K. J., & Kurgan, L., Trends in Data Mining and Knowledge Discovery, In: Pal N.R., Jain, L.C. and Teoderesku, N. (Eds.), Knowledge Discovery in Advanced Information Systems, Springer, to appear, 2002

Kurgan, L., Cios, K.J., Tadeusiewicz, R., Ogiela, M. & Goodenday, L.S., Knowledge Discovery Approach to Automated Cardiac SPECT Diagnosis, *Artificial Intelligence in Medicine*, vol. 23/2, pp.149-169, 2001

Kurgan L. & Cios K.J., Discretization Algorithm that Uses Class-Attribute Interdependence Maximization, *Proceedings of the 2001 International Conference on Artificial Intelligence (IC-AI 2001),* pp. 980-987, Las Vegas, Nevada, 2001

Kurgan, L., & Cios, K.J., CAIM Discretization Algorithm, submitted to IEEE Transactions on Data and Knowledge Engineering, 2001

Kurgan, L., Swiercz, W., & Cios, K.J., Semantic Mapping of XML Tags using Inductive Machine Learning, submitted to the 2002 International Conference on Machine Learning and Applications, Las Vegas, 2002

Kurgan, L., Cios, K.J., & Trombley, M., The WWW Based Data Mining Toolbox Architecture, submitted to the 6th International Conference on Neural Networks and Soft Computing, 2002