Report from KDD 2004

Johannes Gehrke

Department of Computer Science
Cornell University
http://www.cs.cornell.edu/johannes

Cornell University

The SIGKDD Conference

- Started as a workshop in 1989
- Became a conference in 1995
- Became an ACM Conference in 1999

- KDD 2002 (Edmonton, AB)
- KDD 2003 (Washington, DC)
- KDD 2004 (Seattle, WA)

Cornell University

SIGKDD 2004 Chairs

- General chair: Ronny Kohavi
- Program co-chairs:
  - William DuMouchel, Johannes Gehrke
- Industrial/Government co-chairs:
  - John Elder, Bharat Rao

Cornell University

KDD 2004: Statistics

- 337 research track submissions
  - Accepts: 40 full (12%), 44 poster (13%)
- 47 industrial/government track submissions
  - Accepts: 14 full (30%), 13 poster (28%)

Cornell University

KDD 2004: Eight Workshops

- BIOKDD 2004: Data Mining in Bioinformatics
- Mining Temporal and Sequential Data
- MRDM 2004: Multi-Relational Data Mining
- MDM/KDD 2004: Multimedia Data Mining
- DM-SSP 2004: Data Mining Standards
- LinkKDD 2004: “Link Discovery” Workshop
- WebKDD 2004: Web Mining and Web Analysis
- MSW 2004: Mining for and from the Semantic Web

Cornell University

KDD 2004: Tutorials

- Online Mining Data Streams: Problems, Applications and Progress (Jian Pei, Haixun Wang, Philip S. Yu)
- Data Quality and Data Cleaning: An Overview (Tamraparni Dasu, Theodore Johnson)
- Graph Structures in Data Mining (Soumen Chakrabarti, Christos Faloutsos)
- Mining Unstructured Data (Ronen Feldman)
- Junk E-mail Filtering (Joshua Goodman, Geoff Hulten)
- Data Mining and Machine Learning in Time Series Databases (Eamonn Keogh)

Cornell University

SIGKDD Innovation Award

- 2004 SIGKDD Innovation Award Winner: Jiawei Han (UIUC)

- 2004 SIGKDD Service Award Winner: Xindong Wu (U of Vermont)

Cornell University

Keynotes

- Eric Haseltine (NSA)
  - User-oriented approach to creating KDD solutions
- David Heckerman (Microsoft)
  - Graphical models for data mining

Cornell University

Panels

- Can Natural Language Processing Help Text Mining? (Anne Kao, Boeing)

- Data Mining: Good, Bad, or Just a Tool? (Raghu Ramakrishnan, University of Wisconsin, Madison)

Cornell University

SIGKDD Cup

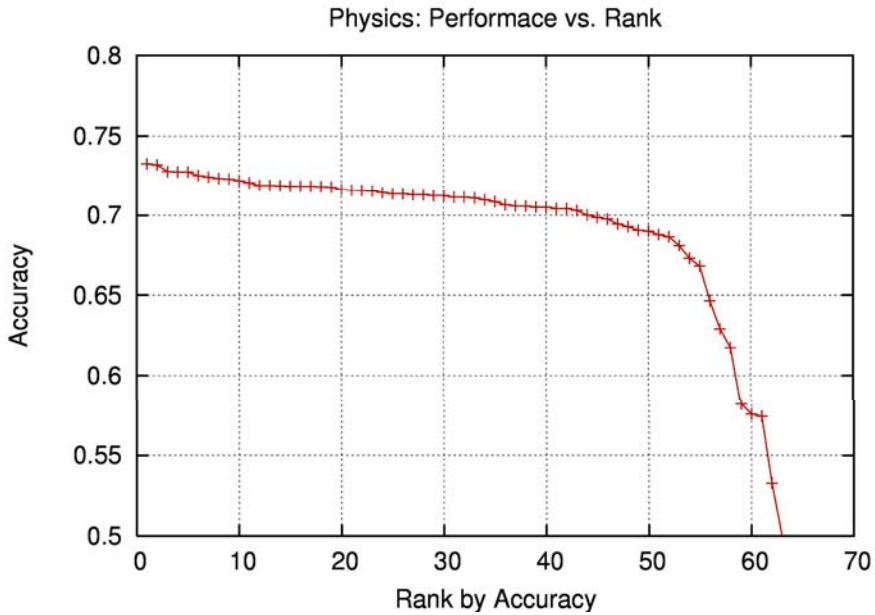
- SIGKDD Cup Overview (Rich Caruana, Thorsten Joachims)
  - Classification problems that require optimization of a specific performance metric
- Two tasks: Particle physics, protein homology
- http://kodiak.cs.cornell.edu/kddcup/

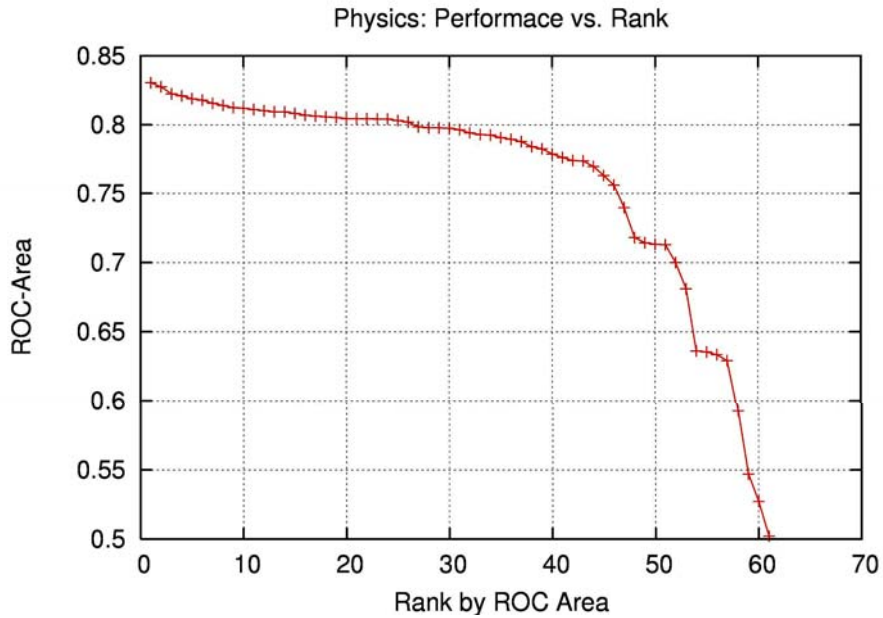
Cornell University

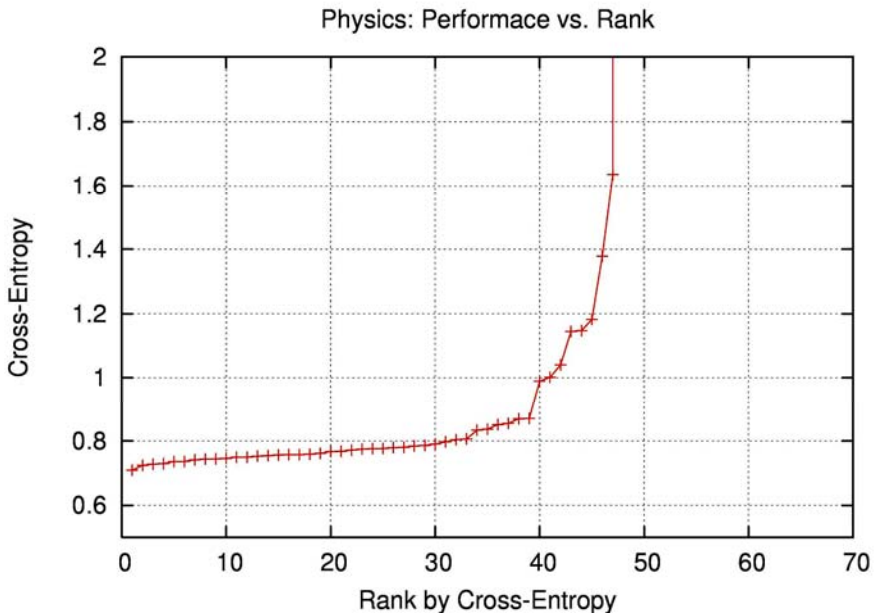
Task 1: Particle Physics Metrics

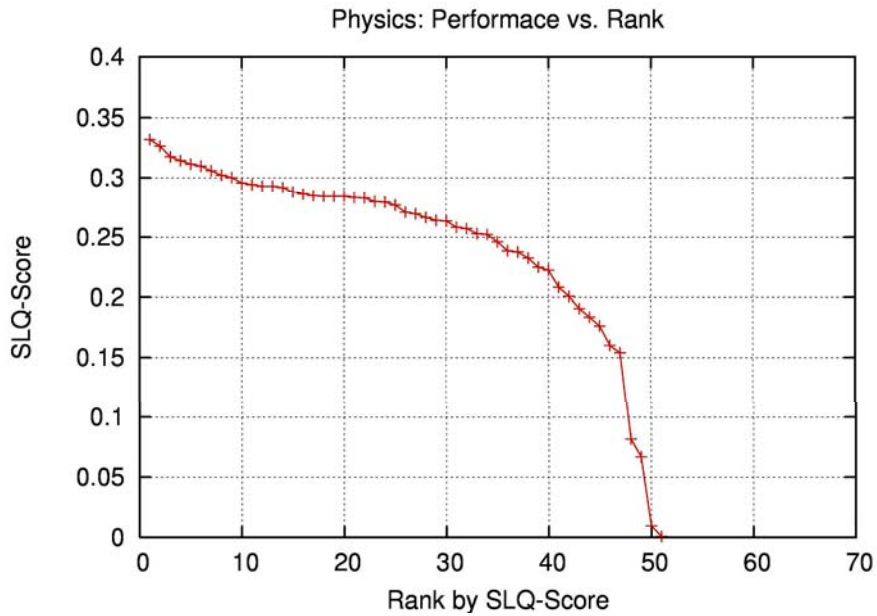
- 4 performance metrics:
  - Accuracy: had to specify threshold
  - Cross-Entropy: probabilistic predictions
  - ROC Area: only ordering is important
  - SLAC Q-Score: domain-specific performance metric from SLAC
- Participants submit separate predictions for each metric
  - About half of participants submitted different predictions for different tasks
  - Winner submitted four sets of predictions, one for each task
- Calculate performance using PERF software provided to participants

Cornell University


Physics: Performace vs. Rank
Accuracy
Rank by Accuracy
0.8
0.75
0.7
0.65
0.6
0.55
0.5
0
10
20
30
40
50
60
70


Physics: Performace vs. Rank
ROC-Area
Rank by ROC Area
0.85
0.8
0.75
0.7
0.65
0.6
0.55
0.5
0
10
20
30
40
50
60
70


Physics: Performace vs. Rank
Cross-Entropy
Rank by Cross-Entropy
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0
10
20
30
40
50
60
70


Physics: Performace vs. Rank
SLQ-Score
Rank by SLQ-Score
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
10
20
30
40
50
60
70

Determining the Winners

- For each performance metric
  - Calculate performance using same PERF software available to participants
  - Rank participants by performance
  - Honorable mention for participant ranked first
- Overall winner is participant with best average rank across all metrics

Winners

- Particle physics
  - Winner: David S. Vogel, Eric Gottschalk, and Morgan C. Wang; MEDai (Neural network with special feature construction)
- Protein homology prediction
  - Winner: Bernhard Pfahringer; University of Waikato (Weka with model ensemble: SVM+log regression, boosted unpruned trees, random rules)

Does Optimizing to Each Metric Help?

- About half of participants submitted different predictions for each metric
- Among winners:

Physics Task	1st	4 sets
	2nd	1 set
	3rd	1 set
Protein Task	1st	1 set
	1st	2 sets
	1st	4 sets

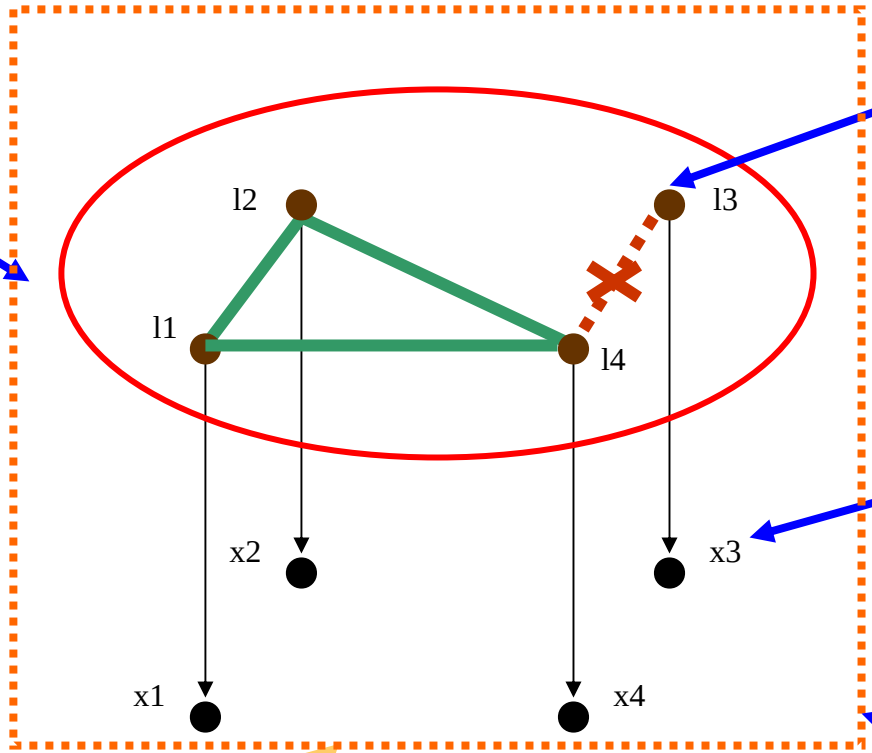
- Some evidence that top performers benefit from optimizing to each metric

Cornell University

Award Papers

- BEST RESEARCH PAPER AWARD
  - A Probabilistic Framework for Semi-Supervised Clustering *(Sugato Basu, Mikhail Bilenko, Raymond Mooney; UT Austin)*
- BEST INDUSTRIAL PAPER AWARD
  - Learning to Detect Malicious Executables in the Wild *(Jeremy Kolter, Marcus A. Maloof; Georgetown)*

Probabilistic Model: HMRF



Markov Random Field (MRF)

Hidden RVs of cluster labels: *L*

P(L): Prior over constraints

Observed data values: *X*

P(X|L): Data Likelihood

Hidden Markov Random Field (HMRF)

Goal of semi-supervised clustering: MAP estimation of P(L|X) on HMRF

Cornell University

MAP estimation on HMRF

Constraint potentials

$$\Pr(L) \propto \exp[-\sum_{i,j} V(x_i, x_j, l_i, l_j)]$$

Cluster distortion

$$\Pr(X \mid L) \propto \exp[-\sum_{x_i} D(x_i, \mu_{l_i})]$$

Posterior Probability

$$\Downarrow$$

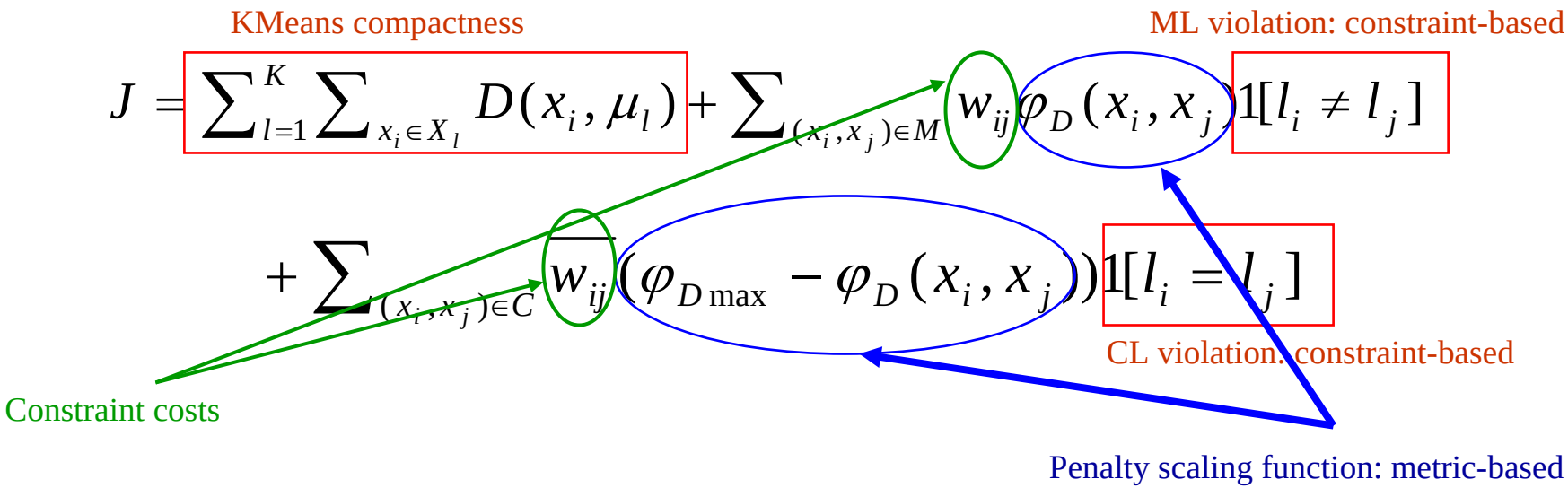
$$\Pr(L \mid X) \propto \Pr(X \mid L) \Pr(L)$$

Semi-supervised clustering objective

$$-\log \Pr(L \mid X) \propto \left(\sum_{x_i} D(x_i, \mu_{l_i}) + \sum_{i,j} V(x_i, x_j, l_i, l_j) \right)$$

Cornell University

HMRF-KMeans Objective Function



- The joint objective function allows:
  - Integrated framework for metric-learning and constrained clustering
  - K-Means-type algorithm for any Bregman divergence *D* (e.g., KL divergence, Euclidean distance) or directional distance (cosine)

Cornell University

HMRF-KMeans Algorithm

Initialization:

- Use connected neighborhoods derived from constraints to initialize clusters

Till *convergence*:

1. **Point assignment:**

- Assign each point x to cluster h^* to minimize both distance and constraint violations

2. **Mean re-estimation:**

- Estimate cluster centroids as means of each cluster
- Re-estimate metric parameters to minimize constraint violations

Cornell University

Award Papers

- BEST RESEARCH PAPER AWARD
  - A Probabilistic Framework for Semi-Supervised Clustering *(Sugato Basu, Mikhail Bilenko, Raymond Mooney; UT Austin)*
- BEST INDUSTRIAL PAPER AWARD
  - Learning to Detect Malicious Executables in the Wild *(Jeremy Kolter, Marcus A. Maloof; Georgetown)*

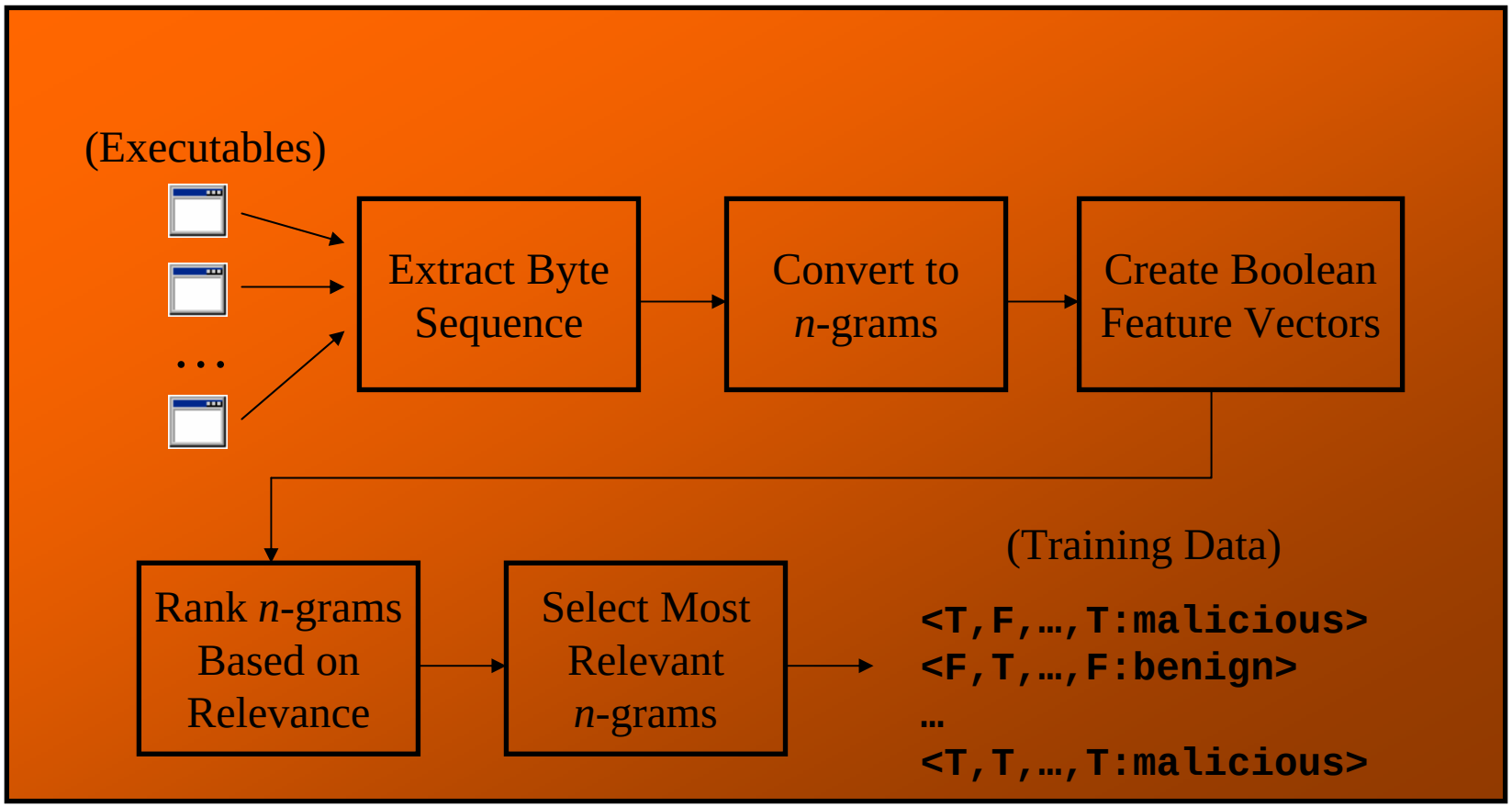
Cornell University

Using Byte Sequences as Features

- Rather than extract higher-level data, we treat executables as byte sequences.
  - Simple to extract.
  - Potentially capture information from all parts of the executable.
- How do we convert a byte sequence into a feature vector?

Cornell University

Extracting Features from Executables



Cornell University

Converting to n-grams

- Standard technique from information retrieval.
- Extract every possible overlapping group of n consecutive bytes (a “sliding window” of n bytes).
- For n-grams of size 2, the byte sequence “01 23 ab dc” translates into the n-grams 0123, 23ab, and abdc.
- We use n-grams of size 4, determined by pilot studies.

Cornell University

Creating Boolean Feature Vectors

- Convert an executables list of *n*-grams into a Boolean feature vector signifying the presence or absence of any given *n*-gram.

malicious.exe	benign.exe
0123	23ab
23ab	abcd

0123	23ab	abcd	Class
T	T	F	Malicious
F	T	T	Benign

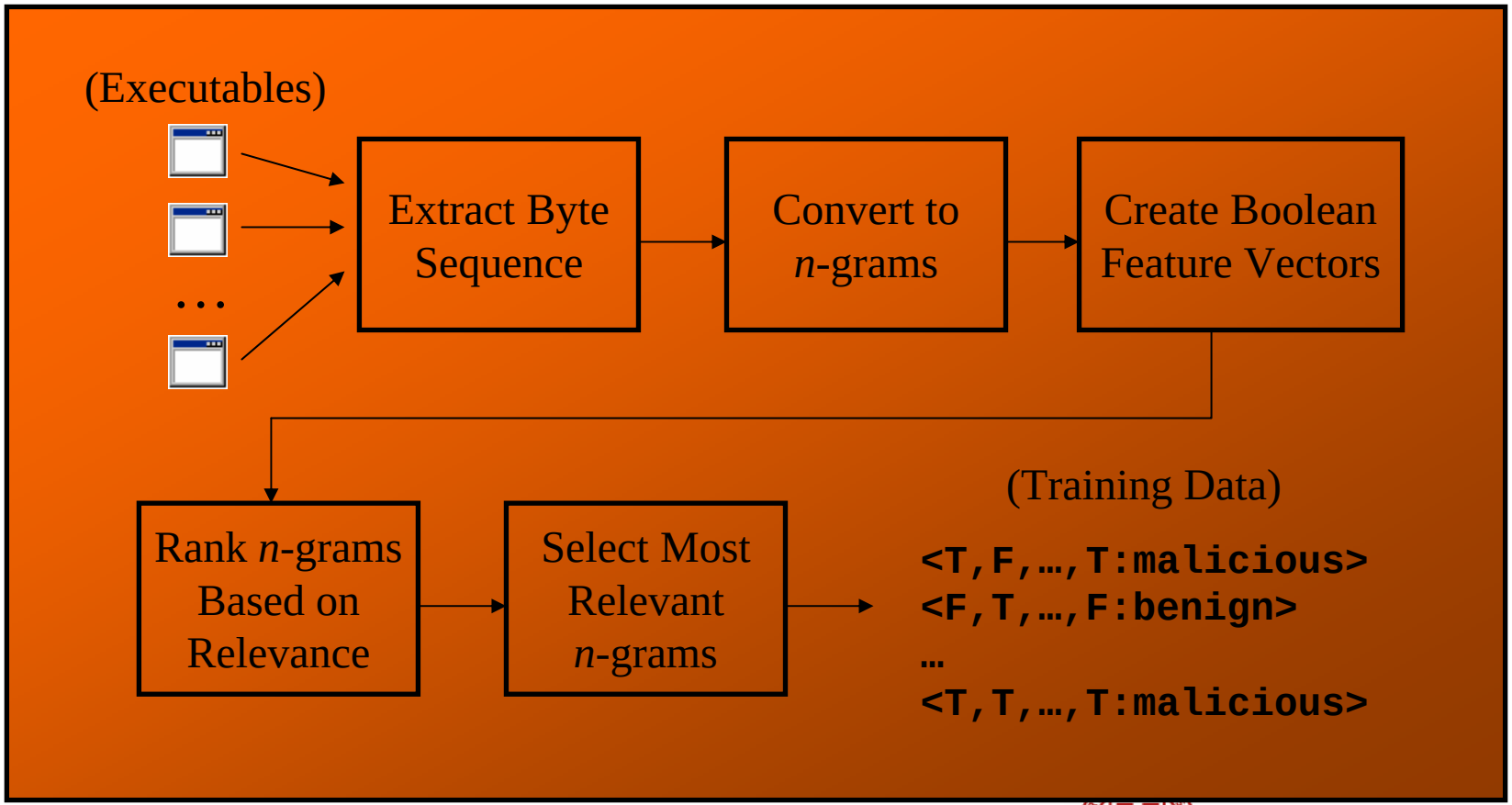
Cornell University

Feature Selection

- Using *n*-grams of size 4, all executables in our data set generated 255,904,403 distinct *n*-grams.
- Reduce to improve efficiency and performance.
- Use *information gain* to measure relevance of each *n*-gram (ranking from 0 to 1).
- Use only the 500 most relevant *n*-grams in feature vector, as determined by pilot studies.

Cornell University

Extracting Features from Executables



Classification Methods

- Naïve Bayes
- J48, implementation of C4.5
- Support Vector Machines
- IB*k*, instance based learner
- TFIDF classifier, based on information retrieval techniques
- Boosted first three methods using AdaBoost.M1.
- All algorithms except TFIDF implemented in WEKA.

Cornell University

Collection of Executables

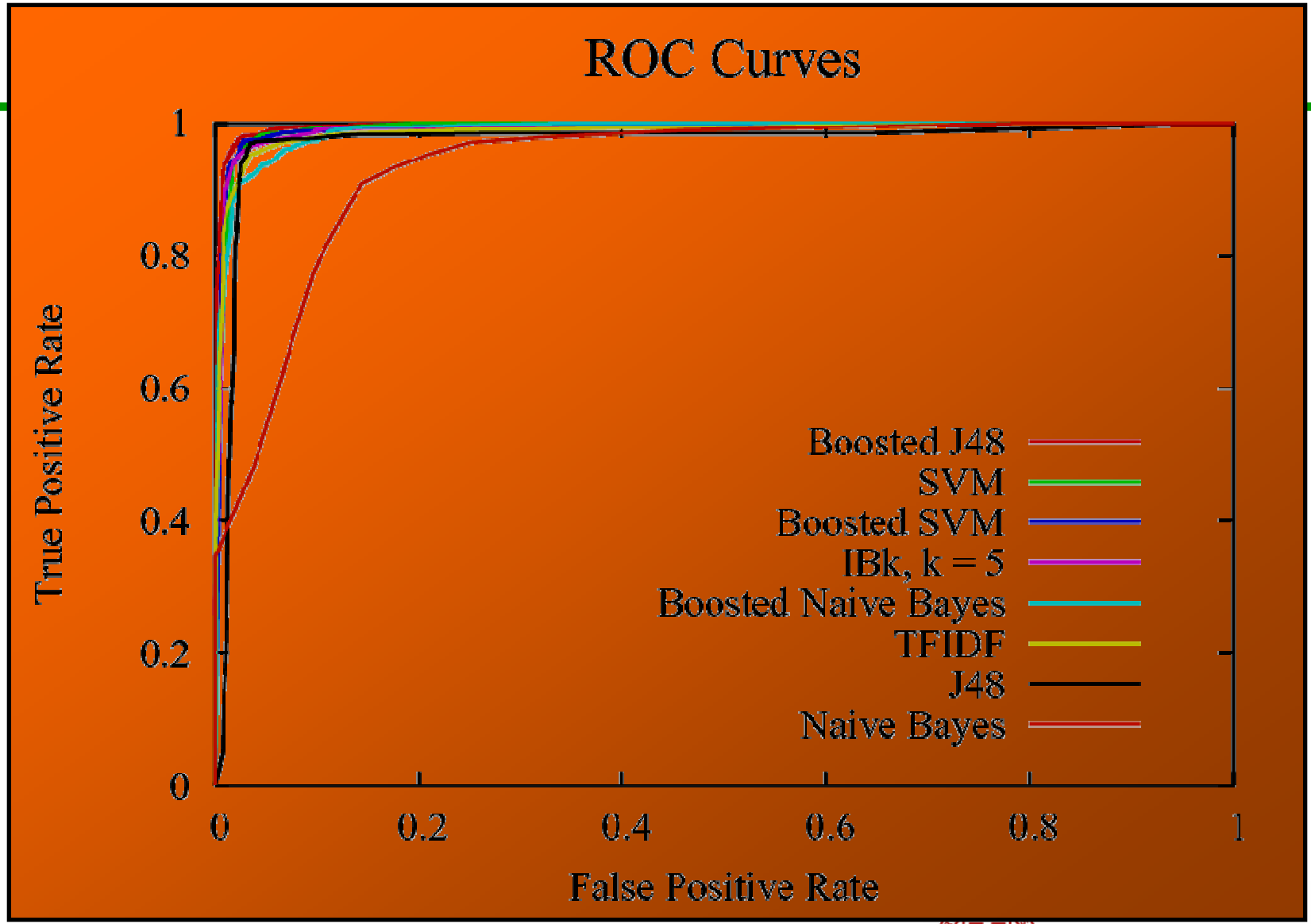
- Obtained malicious and benign executables for the Windows operating system, all in PE format.
- 1651 malicious executables
  - Obtained from MITRE and VX Heavens (http://vx.netlux.org). All in public domain.
  - Commerical program failed to detect 50 programs.
- 1971 benign executables
  - Obtained from Windows 2000/XP machines, SourceForge, and download.com.

Cornell University

Evaluation Methodology

- Evaluated performance of classification methods using ROC analysis.
  - Costs associated with false positives or false negatives are unknown, and most likely different.
- Used area under the curve as performance metric.
- Performed 10-fold stratified cross-validation.
  - Generated average ROC curves by pooling results from all 10 folds.

Cornell University


ROC Curves
True Positive Rate
False Positive Rate
0
0.2
0.4
0.6
0.8
1
Boosted J48
SVM
Boosted SVM
IBk, k = 5
Boosted Naive Bayes
TFIDF
J48
Naive Bayes

Other Research Papers

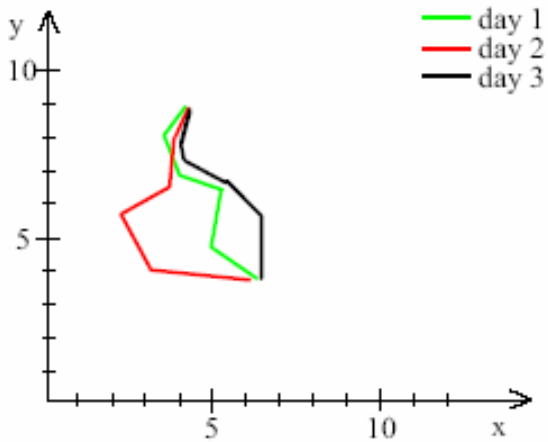
Time Series

- Recovering Latent Time-Series from their Observed Sums: Networked Tomography with Particle Filters (*Airoldi, Faloutsos)*
  - Given: Link loads Y, traffic matrix A, estimate traffic flow Y=A X
  - Idea: Use log-Normal distribution and EM Algorithm
- Clustering Time Series from ARMA Models with Clipped Data (*Bagnall, Janacek)*

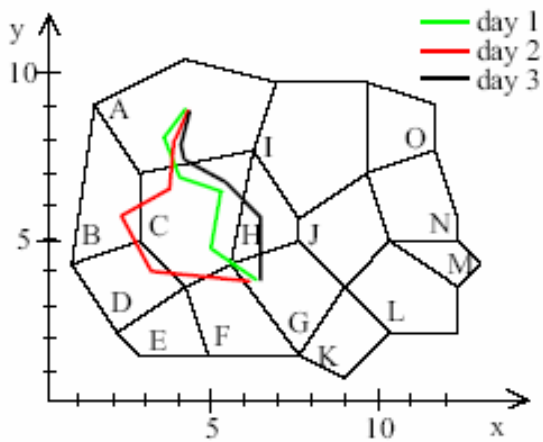
Cornell University

Time Series (Contd.)

- Mining, Indexing, and Querying Historical Spatiotemporal Data (*Mamoulis, Cao, Kollios, Hadjieleftheriou, Tao, Cheung)*
  - Problem: Find sequences in spatial patterns. Pattern: sequence of regions in space. Issue: What is a region?
  - Idea: Region is a density-based cluster, use level-wise pattern mining to find frequent patterns



(a) an object's movement



(b) a set of predefined regions

Cornell University

Multiple Objectives

- Regularized Multi-Task Learning (*Evgeniou, Pontil)*
- Turning CARTwheels: An Alternating Algorithm for Mining Redescriptions (*Ramakrishnan, Kumar, Mishra, Potts, Helm)*
  - Problem: Find redescriptions. Examples:
    - Countries with > 200 Nobel prize winners ⇔ Countries with > 150 billionaires
    - Countries with defense budget > $30B intersect Countries with declared nuclear arsenals ⇔ Permanent member of UN Security Council – Countries with history of communism
- Toward Parameter-Free Data Mining (*Keogh, Lonardi, Ratanamahatana)*

Cornell University

Latent Models

- Web Usage Mining Based on Probabilistic Latent Semantic Analysis (*Jin, Zhou, Mobasher)*
- Probabilistic Author-Topic Models for Information Discovery (*Steyvers, Smyth, Rosen-Zvi, Griffiths)*

Cornell University

Anomaly and Fraud Detection

- Selection, Combination, and Evaluation of Effective Software Sensors for Detecting Abnormal Computer Usage *(Shavlik, Shavlik)*
- Adversarial Classification *(Dalvi, Domingos, Mausam, Sanghai, Verma)*
  - In many domains, adversary manipulates data to defeat data miner
  - Examples: Spam, fraud detection, intrusion detection, terrorism, aerial surveillance, comparison shopping, file sharing, search engine optimization, etc.
  - Model: Game between two players. Adversary tries to make CLASSIFIER classify positive instances as negative (Adversary cannot modify negative instances)
  - CLASSIFIER: Has cost to measure feature Xi, has utility to classify instances
  - Adversary: Has cost of changing features, has utility to change instance classification
  - Goal: Create Classifier that maximize expected utility
  - Theorem: Nash equilibrium exists under special conditions
  - Algorithm: Adversary-aware Naïve Bayes

Cornell University

Spatial Clustering

- Rapid Detection of Significant Spatial Clusters *(Neill, Moore)*
- Fast Mining of Spatial Collocations *(Zhang, Mamoulis, Cheung, Shou)*

Cornell University

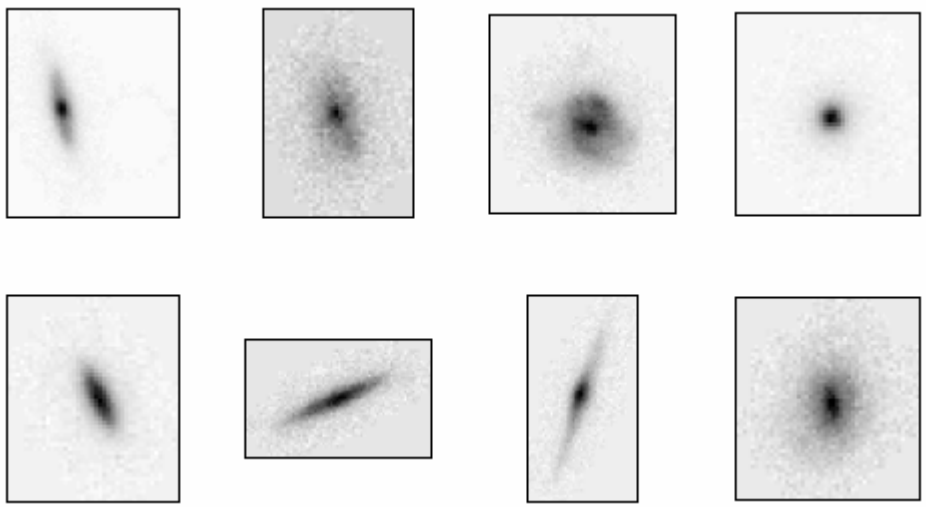
Dimensionality Reduction

- GPCA: An Efficient Dimension Reduction Scheme for Image Compression and Retrieval (*Ye, Janardan, Li)*
- IDR/QR: An Incremental Dimension Reduction Algorithm via QR Decomposition (*Ye, Li, Xiong, Haesun, Janardan, Kumar)*

Cornell University

Dimensionality Reduction (Contd.)

- Fast Galactic Morphology via Eigenimages (*Anderson, Moore, Connolly, Nichol)*
  - Problem: Classify images into type of galaxy
  - Issue: noise in the images (distortion by lens imperfections, atmosphere)



Supervised Learning

- A Bayesian Network Framework for Reject Inference (*Smith, Elkan)*
- An Iterative Method for Multi-Class Cost-Sensitive Learning (*Abe, Zadrozny)*
  - Problem: Cost-sensitive learning
  - Types of approaches: Make the learner cost-sensitive, apply risk theory when assigning examples, modify distribution of training examples
  - Algorithm based on gradient boosting; great performance improvements
- Data Mining in Metric Space: An Empirical Analysis of Supervised Learning Performance Criteria (*Caruana, Niculescu-Mizil)*

Cornell University

Constraints and Prior Knowledge

- Interestingness of Frequent Itemsets Using Bayesian Networks as Background Knowledge (*Jaroszewsicz, Simovici)*
- Incorporating Prior Knowledge with Weighted Margin Support Vector Machines (*Wu, Srihari)*

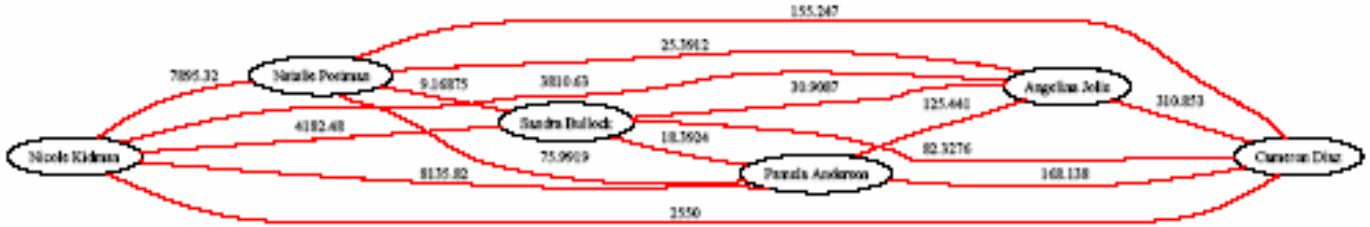
Cornell University

Analyzing Graphs

- Fast Discovery of ‘Connection Subgraphs’ (*Faloutsos, McCurley, Tomkins)*
  - Problem:
    - Given an undirected, weighted graph G, vertices s and t, and integer budget b
    - Find: Connected subgraph H that contains s,t and <= b other vertices that maximizes a goodness function

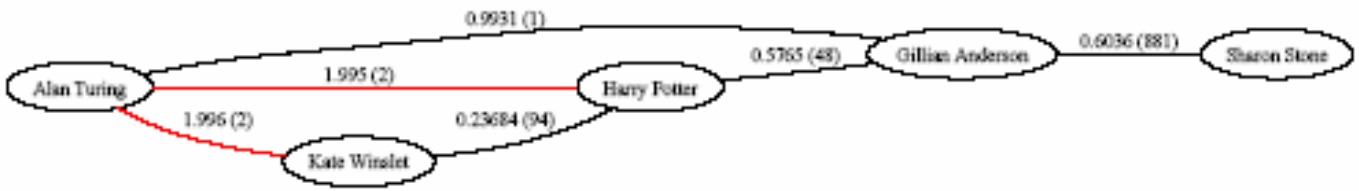
Cornell University

Analyzing Graphs (Contd.)



(a) Nicole Kidman to Cameron Diaz

(b) Nicholas Negroponte to Sam Palmisano



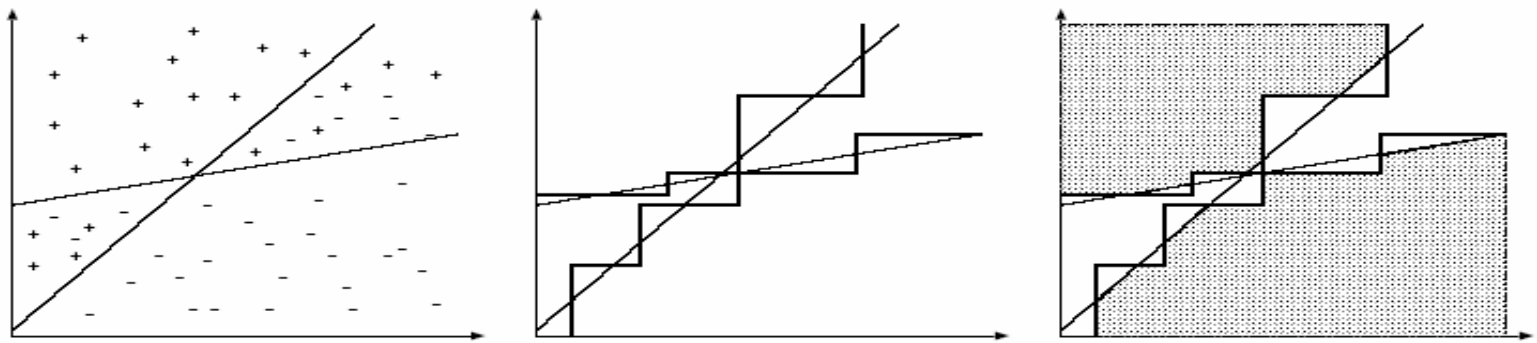
Cornell University

Analyzing Graphs (Contd.)

- Mining the Space of Graph Properties (*Jeh, Widom)*
- Scalable Mining Large Disk-Based Graph Databases (*Wang, Wang, Pei, Zhu, Shi)*
- Cyclic Pattern Kernels for Predictive Graph Mining (*Horvath, Gärtner, Wrobel)*
  - Problem: Prediction problem with a training set of (graph, label) pairs.
  - Approach: Use novel cyclic graph kernels

Data Streams

- Systematic Data Selection to Mine Concept-Drifting Data Streams (*Fan)*
  - Problem: Lots of work on mining data streams; most make ad-hoc decisions about what “old” data to use
  - Idea: Use old data if it comes from the same distribution
  - Implementation: Decision tree ensemble



(a) Evolving hyperplane (b) Their optimal model (c) Those data that help

Data Streams (Contd.)

- Incremental Maintenance of Quotient Cube for Median (*Li, Cong, Tung, Wang)*
- Machine Learning for Online Query Relaxation (*Muslea)*
- A Graph-Theoretic Approach to Extract Storylines from Search Results (*Kumar, Mahadevan, Sivakumar)*

Cornell University

Frequent Itemsets and Association Rules

Abstract:

- A set of items {1,2,...,k}
- A dabase of transactions (itemsets) D={t_1, t_2, ..., t_n}, t_j subset {1,2,...,k}

GOAL:

Find all itemsets that appear in at least s_{min} transactions

("appear in" == "are subsets of")

$I \subseteq t$: t supports I

For an itemset I, the number of transactions it appears in is called the support of I.

s_{min} is called the minimum support.

Concrete:

- I = {milk, bread, cheese, ...}
- D = { {milk,bread,cheese}, {bread,cheese,juice}, ...}

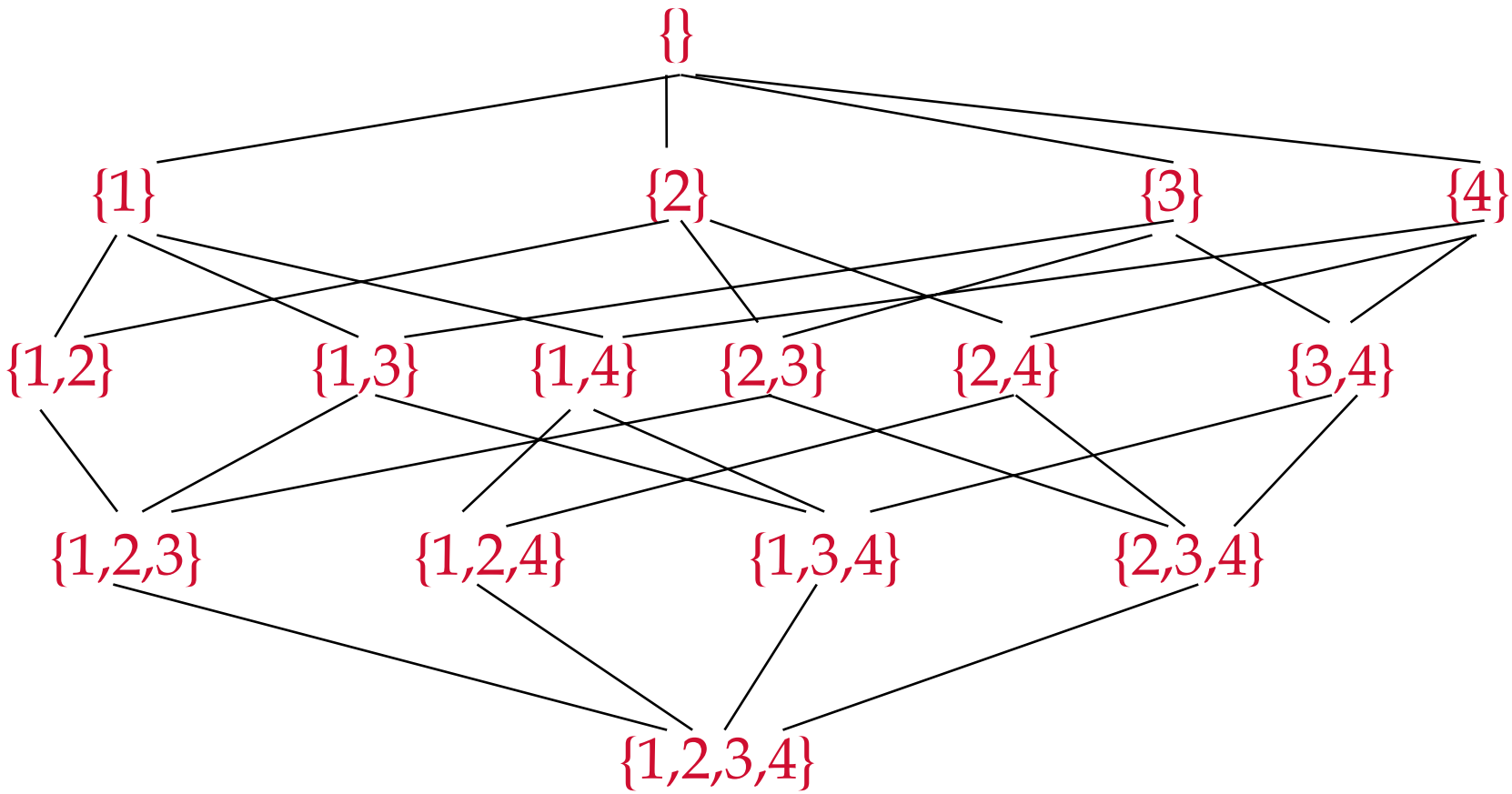
GOAL:

Find all itemsets that appear in at least 1000 transactions

Transaction {milk,bread,cheese} supports itemset {milk,bread}

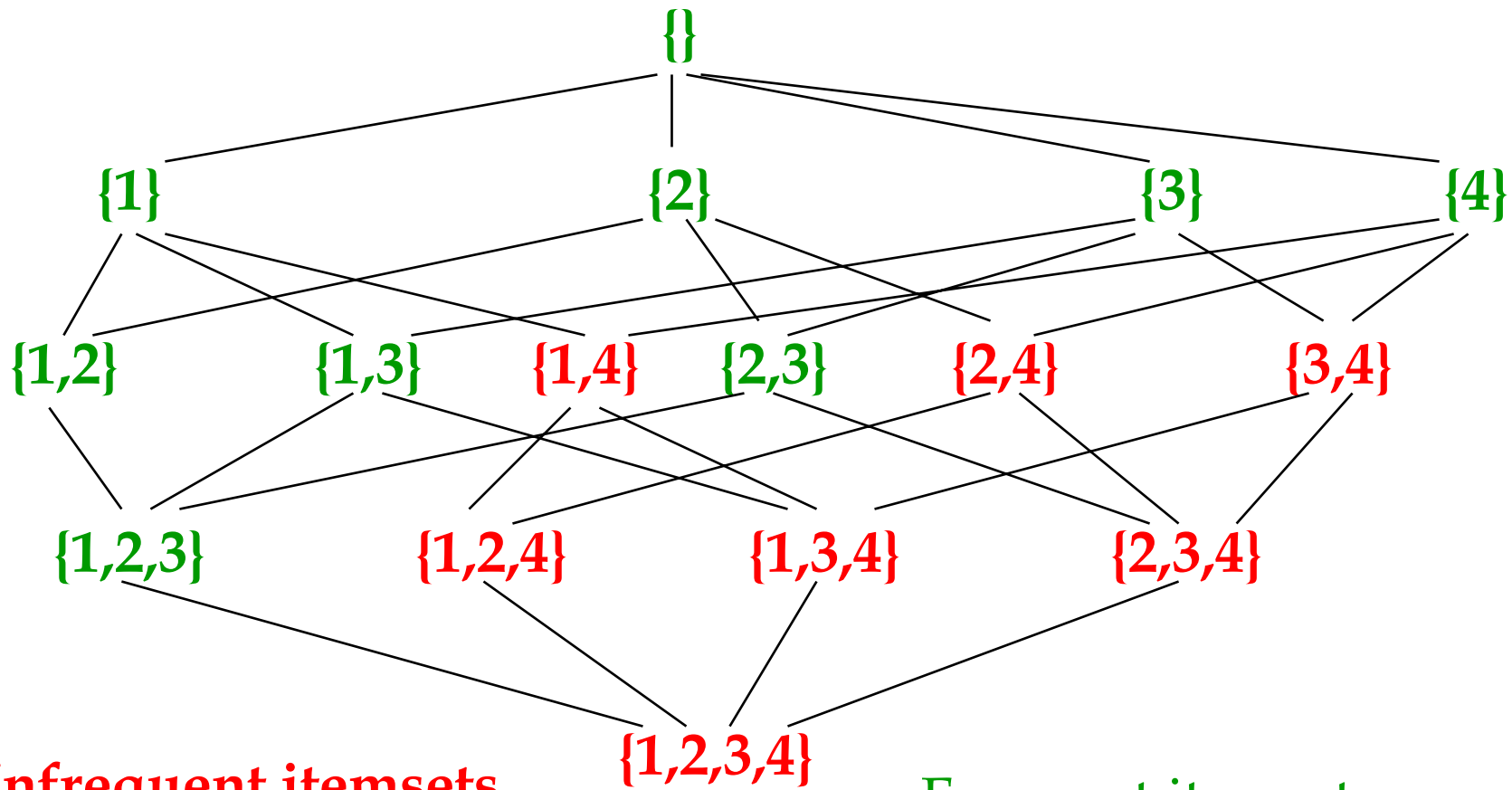
Cornell University

The Itemset Lattice



Cornell University

Frequent Itemsets



Infrequent itemsets

Frequent itemsets

Cornell University

Frequent Sets and Association Rules

- The Complexity of Mining Maximal Frequent Itemsets and Maximal Frequent Patterns *(Yang)*
  - Shows that finding all maximal frequent itemsets is P#-Complete
- Approximating a Collection of Frequent Sets *(Afrati, Gionis, Mannila)*
  - Problem: Given a collection of frequent sets S, find a collection of sets D, |D|=k, such that D approximates S as well as possible
  - Metric: Maximize size of intersection subject to
    - Keeping false positives below user-defined ratio
    - Taking D a subset of S
  - Algorithms: Approximation algorithms that achieve constant-factor approximations with respect to intersection size

Cornell University

Frequent Sets and Association Rules

- Support Envelopes: A Technique for Exploring the Structure of Association Patterns
  (Steinbach, Tan, Kumar)
- On the Discovery of Significant Statistical Quantitative Rules
  (Zhang, Padmanabhan, Tuzhilin)
- Efficient Closed Pattern Mining in the Presence of Tough Block Constraints
  (Gade, Wang, Karypis)

Cornell University

Unsupervised Learning

- Mining Reference Tables for Automatic Text Segmentation (*Agichtein, Ganti)*
  - Problem: Text segmentation
  - Example: “Segmenting text into structure records V. Borkar, Deshmukh and Sarawagi, SIGMOD”
  - Idea: Exploit reference relations (data warehouses with clean tuples)
  - Two-phase approach: (1) Build attribute recognition model from reference data, (2) segment input string
  - Results: On the average, more than 50% accuracy gain

Cornell University

Unsupervised Learning (Contd.)

- Exploiting Dictionaries in Named Entity Extraction: Combining SemiMarkov Extraction Processes and Data Integration Methods (*Cohen, Sarawagi)*
- Mining and Summarizing Customer Reviews (*Hu, Liu)*

Correlation Analysis

- Discovering Complex Matchings across Web Query Interfaces: A Correlation Mining Approach (*He, Chang, Han)*
- Fully Automatic Cross-Associations (*Chakrabarti, Papadimitriou, Modha, Faloutsos)*
- Exploiting a Support-Based Upper Bound of Pearson's Correlation Coefficient for Efficiently Identifying Strongly Correlated Pairs (*Xiong, Shekhar, Tan, Kumar)*

Cornell University

Thank you!

Authors who contributed slides to this talk:

Sugato Basu, Mikhail Bilenko, Rich Caruana, Jeremy Kolter, Marcus A. Maloof, Raymond Mooney, Thorsten Joachims

Report from KDD 2004

Johannes Gehrke

Department of Computer Science
Cornell University
http://www.cs.cornell.edu/johannes

The SIGKDD Conference

- Started as a workshop in 1989
- Became a conference in 1995
- Became an ACM Conference in 1999

- KDD 2002 (Edmonton, AB)
- KDD 2003 (Washington, DC)
- KDD 2004 (Seattle, WA)

SIGKDD 2004 Chairs

- General chair: Ronny Kohavi
- Program co-chairs:
  - William DuMouchel, Johannes Gehrke
- Industrial/Government co-chairs:
  - John Elder, Bharat Rao

KDD 2004: Statistics

- 337 research track submissions
  - Accepts: 40 full (12%), 44 poster (13%)
- 47 industrial/government track submissions
  - Accepts: 14 full (30%), 13 poster (28%)

KDD 2004: Eight Workshops

- BIOKDD 2004: Data Mining in Bioinformatics
- Mining Temporal and Sequential Data
- MRDM 2004: Multi-Relational Data Mining
- MDM/KDD 2004: Multimedia Data Mining
- DM-SSP 2004: Data Mining Standards
- LinkKDD 2004: “Link Discovery” Workshop
- WebKDD 2004: Web Mining and Web Analysis
- MSW 2004: Mining for and from the Semantic Web

KDD 2004: Tutorials

- Online Mining Data Streams: Problems, Applications and Progress (Jian Pei, Haixun Wang, Philip S. Yu)
- Data Quality and Data Cleaning: An Overview (Tamraparni Dasu, Theodore Johnson)
- Graph Structures in Data Mining (Soumen Chakrabarti, Christos Faloutsos)
- Mining Unstructured Data (Ronen Feldman)
- Junk E-mail Filtering (Joshua Goodman, Geoff Hulten)
- Data Mining and Machine Learning in Time Series Databases (Eamonn Keogh)

SIGKDD Innovation Award

- 2004 SIGKDD Innovation Award Winner: Jiawei Han (UIUC)
- 2004 SIGKDD Service Award Winner: Xindong Wu (U of Vermont)

Keynotes

- Eric Haseltine (NSA)
  - User-oriented approach to creating KDD solutions
- David Heckerman (Microsoft)
  - Graphical models for data mining

Panels

- Can Natural Language Processing Help Text Mining? (Anne Kao, Boeing)
- Data Mining: Good, Bad, or Just a Tool? (Raghu Ramakrishnan, University of Wisconsin, Madison)

SIGKDD Cup

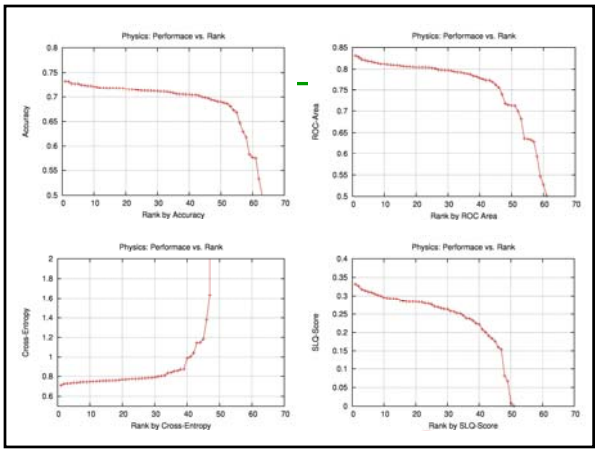
- SIGKDD Cup Overview (Rich Caruana, Thorsten Joachims)
  - Classification problems that require optimization of a specific performance metric
- Two tasks: Particle physics, protein homology
- http://kodiak.cs.cornell.edu/kddcup/

Cornell University

Task 1: Particle Physics Metrics

- 4 performance metrics:
  - Accuracy: had to specify threshold
  - Cross-Entropy: probabilistic predictions
  - ROC Area: only ordering is important
  - SLAC Q-Score: domain-specific performance metric from SLAC
- Participants submit separate predictions for each metric
  - About half of participants submitted different predictions for different tasks
  - Winner submitted four sets of predictions, one for each task
- Calculate performance using PERF software provided to participants

Cornell University



Determining the Winners

- For each performance metric
  - Calculate performance using same PERF software available to participants
  - Rank participants by performance
  - Honorable mention for participant ranked first
- Overall winner is participant with best average rank across all metrics

Cornell University

Winners

- Particle physics
  - Winner: David S. Vogel, Eric Gottschalk, and Morgan C. Wang; MEDai (Neural network with special feature construction)
- Protein homology prediction
  - Winner: Bernhard Pfahringer; University of Waikato (Weka with model ensemble: SVM+log regression, boosted unpruned trees, random rules)

Cornell University

Does Optimizing to Each Metric Help?

- About half of participants submitted different predictions for each metric
- Among winners:

Physics Task	1st	4 sets
	2nd	1 set
	3rd	1 set
Protein Task	1st	1 set
	1st	2 sets
	1st	4 sets

- Some evidence that top performers benefit from optimizing to each metric

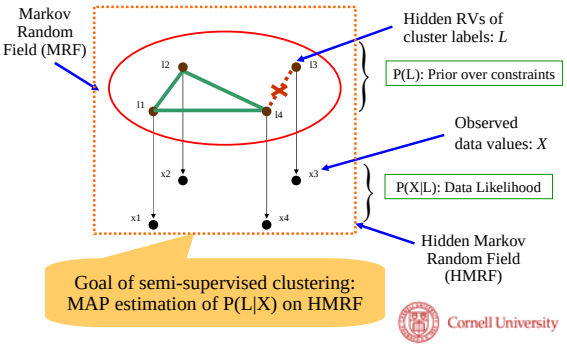
Cornell University

Award Papers

- BEST RESEARCH PAPER AWARD
  - A Probabilistic Framework for Semi-Supervised Clustering *(Sugato Basu, Mikhail Bilenko, Raymond Mooney; UT Austin)*
- BEST INDUSTRIAL PAPER AWARD
  - Learning to Detect Malicious Executables in the Wild *(Jeremy Kolter, Marcus A. Maloof; Georgetown)*

Cornell University

Probabilistic Model: HMRF



Goal of semi-supervised clustering: MAP estimation of P(L|X) on HMRF

Cornell University

MAP estimation on HMRF

Constraint potentials

$$\Pr(L) \propto \exp[-\sum_{i,j} V(x_i, x_j, l_i, l_j)]$$

Cluster distortion

$$\Pr(X \mid L) \propto \exp[-\sum_{x_i} D(x_i, \mu_{l_i})]$$

Posterior Probability

$$\Downarrow$$

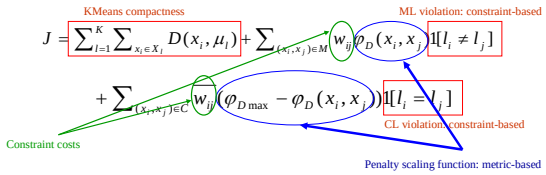
Semi-supervised clustering objective

$$\Pr(L \mid X) \propto \Pr(X \mid L) \Pr(L)$$

$$-\log \Pr(L \mid X) \propto \left(\sum_{x_i} D(x_i, \mu_{l_i}) + \sum_{i,j} V(x_i, x_j, l_i, l_j) \right)$$

Cornell University

HMRF-KMeans Objective Function



- The joint objective function allows:
  - Integrated framework for metric-learning and constrained clustering
  - K-Means-type algorithm for any Bregman divergence *D* (e.g., KL divergence, Euclidean distance) or directional distance (cosine)

Cornell University

HMRF-KMeans Algorithm

Initialization:

- Use connected neighborhoods derived from constraints to initialize clusters

Till *convergence*:

1. **Point assignment:**
   - Assign each point *x* to cluster *h** to minimize both distance and constraint violations
2. **Mean re-estimation:**
   - Estimate cluster centroids as means of each cluster
   - Re-estimate metric parameters to minimize constraint violations

Cornell University

Award Papers

- BEST RESEARCH PAPER AWARD
  - A Probabilistic Framework for Semi-Supervised Clustering *(Sugato Basu, Mikhail Bilenko, Raymond Mooney; UT Austin)*
- BEST INDUSTRIAL PAPER AWARD
  - Learning to Detect Malicious Executables in the Wild *(Jeremy Kolter, Marcus A. Maloof; Georgetown)*

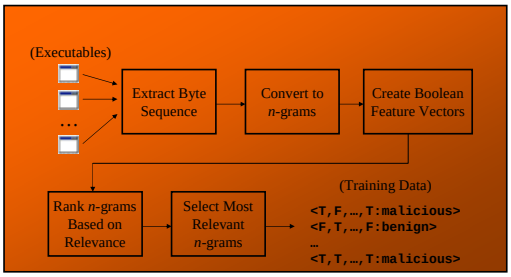
Cornell University

Using Byte Sequences as Features

- Rather than extract higher-level data, we treat executables as byte sequences.
  - Simple to extract.
  - Potentially capture information from all parts of the executable.
- How do we convert a byte sequence into a feature vector?

Cornell University

Extracting Features from Executables



Cornell University

Converting to n-grams

- Standard technique from information retrieval.
- Extract every possible overlapping group of n consecutive bytes (a “sliding window” of n bytes).
- For n-grams of size 2, the byte sequence “01 23 ab dc” translates into the n-grams 0123, 23ab, and abdc.
- We use n-grams of size 4, determined by pilot studies.

Cornell University

Creating Boolean Feature Vectors

- Convert an executables list of *n*-grams into a Boolean feature vector signifying the presence or absence of any given *n*-gram.

malicious.exe	benign.exe
0123	23ab
23ab	abcd

↓

0123	23ab	abcd	Class
T	T	F	Malicious
F	T	T	Benign

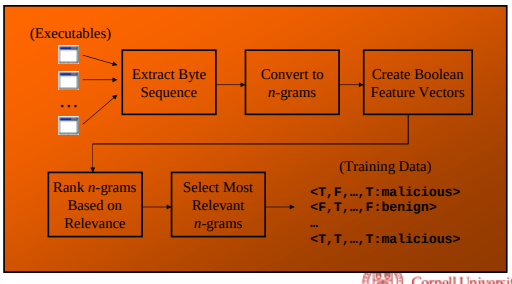
Cornell University

Feature Selection

- Using *n*-grams of size 4, all executables in our data set generated 255,904,403 distinct *n*-grams.
- Reduce to improve efficiency and performance.
- Use *information gain* to measure relevance of each *n*-gram (ranking from 0 to 1).
- Use only the 500 most relevant *n*-grams in feature vector, as determined by pilot studies.

Cornell University

Extracting Features from Executables



Cornell University

Classification Methods

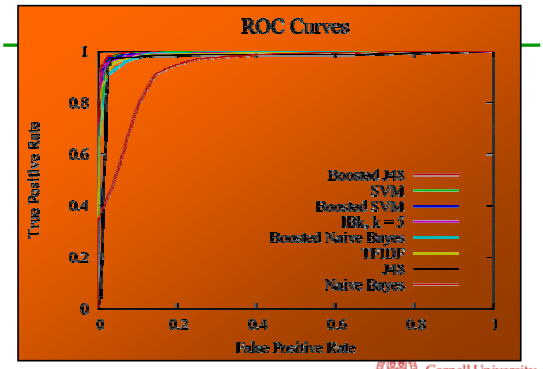
- Naïve Bayes
- J48, implementation of C4.5
- Support Vector Machines
- IB*k,* instance based learner
- TFIDF classifier, based on information retrieval techniques
- Boosted first three methods using AdaBoost.M1.
- All algorithms except TFIDF implemented in WEKA.

Collection of Executables

- Obtained malicious and benign executables for the Windows operating system, all in PE format.
- 1651 malicious executables
  - Obtained from MITRE and VX Heavens (http://vx.netlux.org). All in public domain.
  - Commerical program failed to detect 50 programs.
- 1971 benign executables
  - Obtained from Windows 2000/XP machines, SourceForge, and download.com.

Evaluation Methodology

- Evaluated performance of classification methods using ROC analysis.
  - Costs associated with false positives or false negatives are unknown, and most likely different.
- Used area under the curve as performance metric.
- Performed 10-fold stratified cross-validation.
  - Generated average ROC curves by pooling results from all 10 folds.



Cornell University

Thank you!

Authors who contributed slides to this talk:

Sugato Basu, Mikhail Bilenko, Rich Caruana, Jeremy Kolter, Marcus A. Maloof, Raymond Mooney, Thorsten Joachims