Direct Mining of Discriminative Patterns for Classifying Uncertain Data

Chuancong Gao, Jianyong Wang

Database Laboratory
Department of Computer Science and Technology
Tsinghua University, Beijing, China

Classification on Certain Data

Example:

A toy example about certain categorical dataset containing 4 classes.

Evaluation	Price	Looking	Tech. Spec.	Quality
Unacceptable	+	-	/	-
Acceptable	/	-	/	/
Good	-	+	/	/
Very Good	/	+	+	+

(+: Good, /: Medium, -: Bad)

Classification on Certain Data

Example:

A toy example about certain categorical dataset containing 4 classes.

Evaluation	Price	Looking	Tech. Spec.	Quality
Unacceptable	+	-	/	-
Acceptable	/	-	/	/
Good	-	+	/	/
Very Good	/	+	+	+

(+: Good, /: Medium, -: Bad)

A Lot of Methods:

- Decision Tree - C4.5, etc.

Classification on Certain Data

Example:

A toy example about certain categorical dataset containing 4 classes.

Evaluation	Price	Looking	Tech. Spec.	Quality
Unacceptable	+	-	/	-
Acceptable	/	-	/	/
Good	-	+	/	/
Very Good	/	+	+	+

(+: Good, /: Medium, -: Bad)

A Lot of Methods:

- Decision Tree - C4.5, etc.
- Rule-based Classifier - Ripper, etc.

Classification on Certain Data

Example:

A toy example about certain categorical dataset containing 4 classes.

Evaluation	Price	Looking	Tech. Spec.	Quality
Unacceptable	+	-	/	-
Acceptable	/	-	/	/
Good	-	+	/	/
Very Good	/	+	+	+

(+: Good, /: Medium, -: Bad)

A Lot of Methods:

- Decision Tree - C4.5, etc.
- Rule-based Classifier - Ripper, etc.
- Associative Classification (Pattern-based Classification) - CBA, RCBT, HARMONY, DDPMine, MbT, etc. (Better Performance)

Classification on Certain Data

Example:

A toy example about certain categorical dataset containing 4 classes.

Evaluation	Price	Looking	Tech. Spec.	Quality
Unacceptable	+	-	/	-
Acceptable	/	-	/	/
Good	-	+	/	/
Very Good	/	+	+	+

(+: Good, /: Medium, -: Bad)

A Lot of Methods:

- Decision Tree - C4.5, etc.
- Rule-based Classifier - Ripper, etc.
- Associative Classification (Pattern-based Classification) - CBA, RCBT, HARMONY, DDPMine, MbT, etc. (Better Performance)
- etc.

Associative Classification

Two-Step Framework:

- Mine a set of frequent patterns.

Associative Classification

Two-Step Framework:

- Mine a set of frequent patterns.
- Select a subset of most discriminative patterns from the mined patterns.

Associative Classification

Two-Step Framework:

- Mine a set of frequent patterns.
- Select a subset of most discriminative patterns from the mined patterns.

One-Step Framework:

- Directly mine a set of most discriminative frequent patterns.

Associative Classification

Two-Step Framework:

- Mine a set of frequent patterns.
- Select a subset of most discriminative patterns from the mined patterns.

One-Step Framework:

- Directly mine a set of most discriminative frequent patterns.

After Having Discriminative Patterns:

- Convert each pattern to a binary feature: Whether the instance contains the pattern.

Associative Classification

Two-Step Framework:

- Mine a set of frequent patterns.
- Select a subset of most discriminative patterns from the mined patterns.

One-Step Framework:

- Directly mine a set of most discriminative frequent patterns.

After Having Discriminative Patterns:

- Convert each pattern to a binary feature: Whether the instance contains the pattern.
- Train a classifier using the feature data converted from training data.

Associative Classification

Mainly Differences between Different Algorithms:

- Different types of mined pattern - All, Closed, Generator, etc. (Two-Step Framework)

Associative Classification

Mainly Differences between Different Algorithms:

- Different types of mined pattern - All, Closed, Generator, etc. (Two-Step Framework)
- Different discriminative measures - Confidence (CBA, HARMONY), Information Gain (DDPMine) (Better Performance), Fisher Score, etc.

Associative Classification

Mainly Differences between Different Algorithms:

- Different types of mined pattern - All, Closed, Generator, etc. (Two-Step Framework)
- Different discriminative measures - Confidence (CBA, HARMONY), Information Gain (DDPMine) (Better Performance), Fisher Score, etc.
- Different instance covering strategies - Sequential Covering (DDPMine), Top-K (RCBT), Search Tree-based Partition (MbT), etc.

Associative Classification

Mainly Differences between Different Algorithms:

- Different types of mined pattern - All, Closed, Generator, etc. (Two-Step Framework)
- Different discriminative measures - Confidence (CBA, HARMONY), Information Gain (DDPMine) (Better Performance), Fisher Score, etc.
- Different instance covering strategies - Sequential Covering (DDPMine), Top-K (RCBT), Search Tree-based Partition (MbT), etc.
- Different classification models - Rule-based (RCBT, HARMONY), SVM (DDPMine, MbT) (Better Performance), Naïve Bayes, etc.

Associative Classification

Mainly Differences between Different Algorithms:

- Different types of mined pattern - All, Closed, Generator, etc. (Two-Step Framework)
- Different discriminative measures - Confidence (CBA, HARMONY), Information Gain (DDPMine) (Better Performance), Fisher Score, etc.
- Different instance covering strategies - Sequential Covering (DDPMine), Top-K (RCBT), Search Tree-based Partition (MbT), etc.
- Different classification models - Rule-based (RCBT, HARMONY), SVM (DDPMine, MbT) (Better Performance), Naïve Bayes, etc.
- Different feature types - Binary, Numeric (New, NDPMine)

Classification on Uncertain Data

Example:

A toy example about uncertain categorical dataset. The uncertainty usually is caused by noise, measurement precisions, etc.

Evaluation	Price	Looking	Tech. Spec.	Quality
Unacceptable	+	-	/	{-: 0.8, /: 0.1, +: 0.1}
Acceptable	/	-	/	{-: 0.1, /: 0.8, +: 0.1}
Good	-	+	/	{-: 0.1, /: 0.8, +: 0.1}
Very Good	/	+	+	{-: 0.1, /: 0.1, +: 0.8}

(+: Good, /: Medium, -: Bad)

Classification on Uncertain Data

Example:

A toy example about uncertain categorical dataset. The uncertainty usually is caused by noise, measurement precisions, etc.

Evaluation	Price	Looking	Tech. Spec.	Quality
Unacceptable	+	-	/	{-: 0.8, /: 0.1, +: 0.1}
Acceptable	/	-	/	{-: 0.1, /: 0.8, +: 0.1}
Good	-	+	/	{-: 0.1, /: 0.8, +: 0.1}
Very Good	/	+	+	{-: 0.1, /: 0.1, +: 0.8}

(+: Good, /: Medium, -: Bad)

Very Few Methods:

- Uncertain Decision Tree - C4.5-based DTU.

Classification on Uncertain Data

Example:

A toy example about uncertain categorical dataset. The uncertainty usually is caused by noise, measurement precisions, etc.

Evaluation	Price	Looking	Tech. Spec.	Quality
Unacceptable	+	-	/	{-: 0.8, /: 0.1, +: 0.1}
Acceptable	/	-	/	{-: 0.1, /: 0.8, +: 0.1}
Good	-	+	/	{-: 0.1, /: 0.8, +: 0.1}
Very Good	/	+	+	{-: 0.1, /: 0.1, +: 0.8}

(+: Good, /: Medium, -: Bad)

Very Few Methods:

- Uncertain Decision Tree - C4.5-based DTU.
- Uncertain Rule-based Classifier - Ripper-based uRule

Our Solution

A new associative classification algorithm working on uncertain data.

Our Solution

A new associative classification algorithm working on uncertain data.

Difference to Certain Dataset:

Patterns involving uncertain attributes have probabilities to appear in instances.

Our Solution

A new associative classification algorithm working on uncertain data.

Difference to Certain Dataset:

Patterns involving uncertain attributes have probabilities to appear in instances.

Challenges:

- How to represent frequentness information? - Using expected value of support. (Easy to calculate - Sum all the probabilities appearing in different instances)

Our Solution

A new associative classification algorithm working on uncertain data.

Difference to Certain Dataset:

Patterns involving uncertain attributes have probabilities to appear in instances.

Challenges:

- How to represent frequentness information? - Using expected value of support. (Easy to calculate - Sum all the probabilities appearing in different instances)
- How to represent discriminative information?

Our Solution

A new associative classification algorithm working on uncertain data.

Difference to Certain Dataset:

Patterns involving uncertain attributes have probabilities to appear in instances.

Challenges:

- How to represent frequentness information? - Using expected value of support. (Easy to calculate - Sum all the probabilities appearing in different instances)
- How to represent discriminative information?
- How to cover instances?

Discriminative Measures on Uncertain Data

Choose to use expected value of confidence. Unlike expected support, expected confidence is hard to calculate.

Discriminative Measures on Uncertain Data

Choose to use expected value of confidence. Unlike expected support, expected confidence is hard to calculate.

Definition of Expected Confidence:

Given a set of transactions T and the set of possible worlds W w.r.t. T, the expected confidence of an itemset x on class c is

$$E(conf_x{}^c) = \sum_{w_i \in W} conf_{x,w_i}{}^c \times P(w_i) = \sum_{w_i \in W} \frac{sup_{x,w_i}{}^c}{sup_{x,w_i}} \times P(w_i)$$

where $P(w_i)$ is the probability of world w_i. $conf_{x,w_i}{}^c$ is the respected confidence of x on class c in world w_i, while sup_{x,w_i} ($sup_{x,w_i}{}^c$) is the respected support of x (on class c) in world w_i.

Discriminative Measures on Uncertain Data

Choose to use expected value of confidence. Unlike expected support, expected confidence is hard to calculate.

Definition of Expected Confidence:

Given a set of transactions T and the set of possible worlds W w.r.t. T, the expected confidence of an itemset x on class c is

$$E(conf_x{}^c) = \sum_{w_i \in W} conf_{x,w_i}{}^c \times P(w_i) = \sum_{w_i \in W} \frac{sup_{x,w_i}{}^c}{sup_{x,w_i}} \times P(w_i)$$

where $P(w_i)$ is the probability of world w_i. $conf_{x,w_i}{}^c$ is the respected confidence of x on class c in world w_i, while sup_{x,w_i} ($sup_{x,w_i}{}^c$) is the respected support of x (on class c) in world w_i.

$O(\prod_{A_k \in \mathbb{A}^u} |dom_{A_k}|^{|T|})$ possible worlds.

Efficient Computation of Expected Confidence

Lemma:

Since $0 \leq {sup_x}^c \leq sup_x \leq |T|$, we have:

$$
\begin{aligned}
E({conf_x}^c) &= \sum_{w_i \in W} {conf_{x,w_i}}^c \times P(w_i) \\
&= \sum_{i=0}^{|T|} \sum_{j=0}^{i} \frac{j}{i} \times P(sup_x = i \wedge {sup_x}^c = j) \\
&= \sum_{i=0}^{|T|} \frac{E_i({sup_x}^c)}{i} = \sum_{i=0}^{|T|} E_i({conf_x}^c)
\end{aligned}
$$

, where $E_i({sup_x}^c)$ and $E_i({conf_x}^c)$ denote the part of expected support and confidence of itemset x on class c when $sup_x = i$.

Efficient Computation of Expected Confidence

Given $0 \leq n \leq |T|$, define $E_n(sup_x{}^c) = \sum_{i=0}^{|T|} E_{i,n}(sup_x{}^c)$ as the expected support of x on class c on the first n transactions of T, and $E_{i,n}(sup_x{}^c)$ as the expected support of x on class c with support of i on the first n transactions of T.

Efficient Computation of Expected Confidence

Given $0 \leq n \leq |T|$, define $E_n(sup_x{}^c) = \sum_{i=0}^{|T|} E_{i,n}(sup_x{}^c)$ as the expected support of x on class c on the first n transactions of T, and $E_{i,n}(sup_x{}^c)$ as the expected support of x on class c with support of i on the first n transactions of T.

Denoting $P(x \subseteq t_i)$ as p_i for each transaction $t_i \in T$, we have

$$E_{i,n}(sup_x{}^c) = p_n \times E_{i-1,n-1}(sup_x{}^c) + (1 - p_n) \times E_{i,n-1}(sup_x{}^c)$$

when $c_n \neq c$

Efficient Computation of Expected Confidence

Given $0 \leq n \leq |T|$, define $E_n(sup_x{}^c) = \sum_{i=0}^{|T|} E_{i,n}(sup_x{}^c)$ as the expected support of x on class c on the first n transactions of T, and $E_{i,n}(sup_x{}^c)$ as the expected support of x on class c with support of i on the first n transactions of T.

Denoting $P(x \subseteq t_i)$ as p_i for each transaction $t_i \in T$, we have

$$E_{i,n}(sup_x{}^c) = p_n \times E_{i-1,n-1}(sup_x{}^c) + (1 - p_n) \times E_{i,n-1}(sup_x{}^c)$$

when $c_n \neq c$, and

$$E_{i,n}(sup_x{}^c) = p_n \times E_{i-1,n-1}(sup_x{}^c + 1) + (1 - p_n) \times E_{i,n-1}(sup_x{}^c)$$

when $c_n = c$, where $1 \leq i \leq n \leq |T|$.

Efficient Computation of Expected Confidence

Given $0 \leq n \leq |T|$, define $E_n(sup_x{}^c) = \sum_{i=0}^{|T|} E_{i,n}(sup_x{}^c)$ as the expected support of x on class c on the first n transactions of T, and $E_{i,n}(sup_x{}^c)$ as the expected support of x on class c with support of i on the first n transactions of T.

Denoting $P(x \subseteq t_i)$ as p_i for each transaction $t_i \in T$, we have

$$E_{i,n}(sup_x{}^c) = p_n \times E_{i-1,n-1}(sup_x{}^c) + (1 - p_n) \times E_{i,n-1}(sup_x{}^c)$$

when $c_n \neq c$, and

$$E_{i,n}(sup_x{}^c) = p_n \times E_{i-1,n-1}(sup_x{}^c + 1) + (1 - p_n) \times E_{i,n-1}(sup_x{}^c)$$

when $c_n = c$, where $1 \leq i \leq n \leq |T|$.

$$E_{i,n}(sup_x{}^c) = 0$$

for $\forall n$ where $i = 0$, or where $n < i$.

Efficient Computation of Expected Confidence

Defining $P_{i,n}$ as the probability of x having support of i on the first n transactions of T, we have

$$\begin{aligned} E_{i,n}(sup_x{}^c) &= p_n \times (E_{i-1,n-1}(sup_x{}^c + 1)) \\ &+ (1 - p_n) \times E_{i,n-1}(sup_x{}^c) \\ &= p_n \times (E_{i-1,n-1}(sup_x{}^c) + P_{i-1,n-1}) \\ &+ (1 - p_n) \times E_{i,n-1}(sup_x{}^c) \end{aligned}$$

when $c_n = c$

Efficient Computation of Expected Confidence

Defining $P_{i,n}$ as the probability of x having support of i on the first n transactions of T, we have

$$\begin{aligned} E_{i,n}(sup_x{}^c) &= p_n \times (E_{i-1,n-1}(sup_x{}^c + 1)) \\ &+ (1 - p_n) \times E_{i,n-1}(sup_x{}^c) \\ &= p_n \times (E_{i-1,n-1}(sup_x{}^c) + P_{i-1,n-1}) \\ &+ (1 - p_n) \times E_{i,n-1}(sup_x{}^c) \end{aligned}$$

when $c_n = c$, since we have:

$$E_{i-1,n-1}(sup_x{}^c + 1) = E_{i-1,n-1}(sup_x{}^c) + P_{i-1,n-1}$$

Efficient Computation of Expected Confidence

Denoting $P(x \subseteq t_i)$ as p_i for each transaction $t_i \in T$, we have

$$P_{i,n} = p_n \times P_{i-1,n-1} + (1 - p_n) \times P_{i,n-1}$$

, where $1 \leq i \leq n \leq |T|$.

Efficient Computation of Expected Confidence

Denoting $P(x \subseteq t_i)$ as p_i for each transaction $t_i \in T$, we have

$$P_{i,n} = p_n \times P_{i-1,n-1} + (1 - p_n) \times P_{i,n-1}$$

, where $1 \leq i \leq n \leq |T|$.

$$P_{i,n} = \begin{cases} 1 & \textit{for } n = 0 \\ P_{i,n-1} \times (1 - p_n) & \textit{for } 1 \leq n \leq |T| \end{cases}$$

where $i = 0$.

Efficient Computation of Expected Confidence

Denoting $P(x \subseteq t_i)$ as p_i for each transaction $t_i \in T$, we have

$$P_{i,n} = p_n \times P_{i-1,n-1} + (1 - p_n) \times P_{i,n-1}$$

, where $1 \leq i \leq n \leq |T|$.

$$P_{i,n} = \begin{cases} 1 & \mathit{for} \;\; n = 0 \\ P_{i,n-1} \times (1 - p_n) & \mathit{for} \; 1 \leq n \leq |T| \end{cases}$$

where $i = 0$.

$$P_{i,n} = 0$$

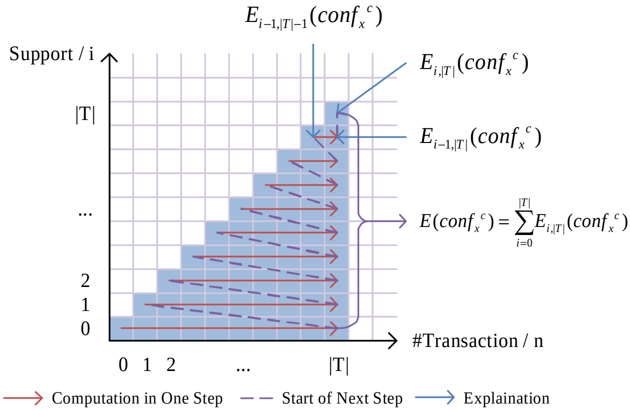
where $n < i$.

Efficient Computation of Expected Confidence

Since $E(conf_x{}^c) = E_{|T|}(conf_x{}^c) = \sum_{i=0}^{|T|} E_{i,|T|}(conf_x{}^c)$. The computation is divided into $|T| + 1$ steps with $E_{i,|T|}(conf_x{}^c) = E_{i,|T|}(sup_x{}^c)/i$ $(0 \leq i \leq |T|)$ computed in ith step.

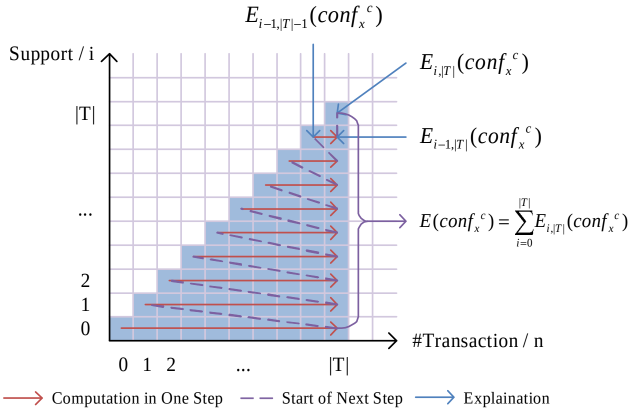
Efficient Computation of Expected Confidence

Since $E(conf_x{}^c) = E_{|T|}(conf_x{}^c) = \sum_{i=0}^{|T|} E_{i,|T|}(conf_x{}^c)$. The computation is divided into $|T| + 1$ steps with $E_{i,|T|}(conf_x{}^c) = E_{i,|T|}(sup_x{}^c)/i$ $(0 \le i \le |T|)$ computed in ith step.



Efficient Computation of Expected Confidence

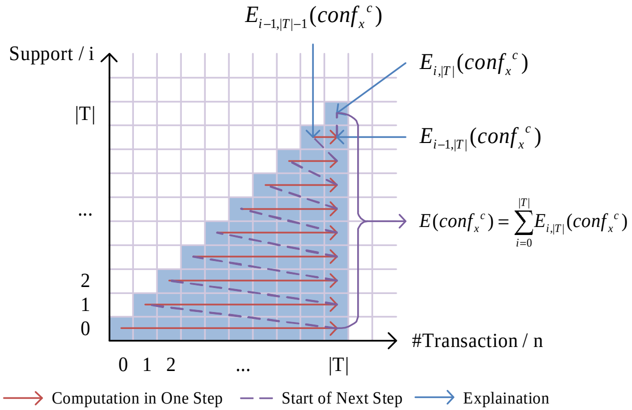
Since $E(conf_x{}^c) = E_{|T|}(conf_x{}^c) = \sum_{i=0}^{|T|} E_{i,|T|}(conf_x{}^c)$. The computation is divided into $|T|+1$ steps with $E_{i,|T|}(conf_x{}^c) = E_{i,|T|}(sup_x{}^c)/i$ $(0 \le i \le |T|)$ computed in ith step.



Time Complexity: $O(|T|^2)$

Efficient Computation of Expected Confidence

Since $E(conf_x{}^c) = E_{|T|}(conf_x{}^c) = \sum_{i=0}^{|T|} E_{i,|T|}(conf_x{}^c)$. The computation is divided into $|T|+1$ steps with $E_{i,|T|}(conf_x{}^c) = E_{i,|T|}(sup_x{}^c)/i$ $(0 \le i \le |T|)$ computed in ith step.



Time Complexity:

$O(|T|^2)$

Space Complexity:

$O(|T|)$

Upper Bounds of Expected Confidence

For $\forall i (1 \leq i \leq |T|)$, we have

$$
\begin{aligned}
& E(conf_x{}^c) = E_{|T|}(conf_x{}^c) \\
= & \sum_{k=0}^{i-1} \frac{E_{k,|T|}(sup_x{}^c)}{k} + \sum_{k=i}^{|T|} \frac{E_{k,|T|}(sup_x{}^c)}{k} \\
\leq & \sum_{k=0}^{i-1} \frac{E_{k,|T|}(sup_x{}^c)}{k} + \sum_{k=i}^{|T|} \frac{E_{k,|T|}(sup_x{}^c)}{i} \\
= & \sum_{k=0}^{i-1} \frac{E_{k,|T|}(sup_x{}^c)}{k} + \sum_{k=0}^{|T|} \frac{E_{k,|T|}(sup_x{}^c)}{i} - \sum_{k=0}^{i-1} \frac{E_{k,|T|}(sup_x{}^c)}{i} \\
= & \sum_{k=0}^{i-1} E_{k,|T|}(sup_x{}^c) \times (\frac{1}{k} - \frac{1}{i}) + \frac{E(sup_x{}^c)}{i} \\
= & bound_i(conf_x{}^c)
\end{aligned}
$$

Upper Bounds of Expected Confidence

For $1 \leq i \leq |T|$, we have:

$$\begin{aligned} E({sup_x}^c) &= bound_1({conf_x}^c) \\ &\geq \cdots \geq bound_i({conf_x}^c) \geq \cdots \\ &\geq bound_{|T|}({conf_x}^c) = E({conf_x}^c) \end{aligned}$$

Upper Bounds of Expected Confidence

For $1 \le i \le |T|$, we have:

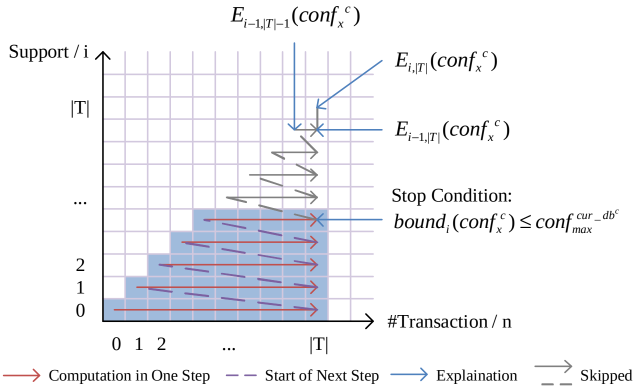
$$\begin{aligned} E(sup_x{}^c) &= bound_1(conf_x{}^c) \\ &\ge \cdots \ge bound_i(conf_x{}^c) \ge \cdots \\ &\ge bound_{|T|}(conf_x{}^c) = E(conf_x{}^c) \end{aligned}$$

Since

$$\begin{aligned} bound_i(conf_x{}^c) &= bound_{i-1}(conf_x{}^c) \\ &\quad - \left(\frac{1}{i-1} - \frac{1}{i}\right) \times \left(E(sup_x{}^c) - \sum_{k=0}^{i-1} E_{k,|T|}(sup_x{}^c)\right) \end{aligned}$$

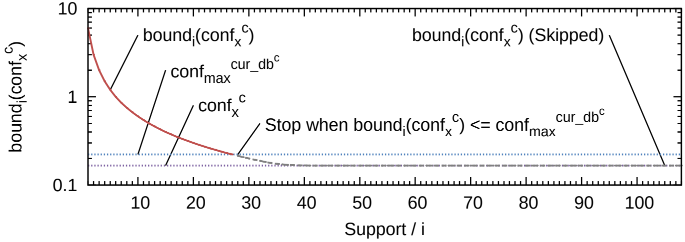
, can compute $bound_i(conf_x{}^c)$ with $bound_{i-1}(conf_x{}^c)$.

Upper Bounds of Expected Confidence



Upper Bounds of Expected Confidence

Running Example:



Algorithm Framework

1. Calculate the (expected) confidence of current prefix pattern.

Algorithm Framework

1. Calculate the (expected) confidence of current prefix pattern.
2. If the confidence value is larger than previous maximal value, update covered instances.

Algorithm Framework

1. Calculate the (expected) confidence of current prefix pattern.
2. If the confidence value is larger than previous maximal value, update covered instances.
3. If at least one instance covered, select the prefix.

Algorithm Framework

1. Calculate the (expected) confidence of current prefix pattern.
2. If the confidence value is larger than previous maximal value, update covered instances.
3. If at least one instance covered, select the prefix.
4. Continue growing current prefix, and go to Step 1.

Algorithm Framework

1. Calculate the (expected) confidence of current prefix pattern.
2. If the confidence value is larger than previous maximal value, update covered instances.
3. If at least one instance covered, select the prefix.
4. Continue growing current prefix, and go to Step 1.

Need to sort all the uncertain attributes after certain attributes, to help shrink current projected database.

Instance Covering Strategy

Previous Strategy in HARMONY:

Just find one most discriminative covering pattern with the highest confidence for each instance. On uncertain data, the probability of the instance being covered could be very low.

Instance Covering Strategy

Previous Strategy in HARMONY:

Just find one most discriminative covering pattern with the highest confidence for each instance. On uncertain data, the probability of the instance being covered could be very low.

Our method:

Apply a threshold of minimum cover probability $coverProb_{min}$. Assure that the probability of each instance not covered by any pattern is less than $1 - coverProb_{min}$, by maintaining a list storing confidence values of covering patterns on class c in descending order.

Used Classifiers

SVM Classifier

Convert each pattern to a binary feature by whether it is contained by the instance.

Used Classifiers

SVM Classifier

Convert each pattern to a binary feature by whether it is contained by the instance.

Rule-based Classifier (From HARMONY)

For each test instance we just sum up the product of the confidence of each pattern on each class and the probability of the instance containing the pattern. The class with the largest value is the predicted class of the instance.

Used Datasets

Dataset	#Instance	#Attribute	#Class	Area
australian	690	14	2	Financial
balance	635	4	3	Social
bands	539	38	2	Physical
breast	699	9	2	Life
bridges-v1	106	11	6	N/A
bridges-v2	106	10	6	N/A
car	1728	6	4	N/A
contraceptive	1473	9	3	Life
credit	690	15	2	Financial
echocardiogram	131	12	2	Life
flag	194	28	8	N/A
german	1000	19	2	Financial
heart	920	13	5	Life
hepatitis	155	19	2	Life
horse	368	27	2	Life
monks-1	556	6	2	N/A
monks-2	601	6	2	N/A
monks-3	554	6	2	N/A
mushroom	8124	22	2	Life
pima	768	8	2	Life
postoperative	90	8	3	Life
promoters	106	57	2	Life
spect	267	22	2	Life
survival	306	3	2	Life
ta_eval	151	5	3	N/A
tic-tac-toe	958	9	2	Game
vehicle	846	18	4	N/A
voting	435	16	2	Social
wine	178	13	3	Physical
zoo	101	16	7	Life

30 Public UCI Certain Datasets

Used Datasets

Dataset	#Instance	#Attribute	#Class	Area
australian	690	14	2	Financial
balance	635	4	3	Social
bands	539	38	2	Physical
breast	699	9	2	Life
bridges-v1	106	11	6	N/A
bridges-v2	106	10	6	N/A
car	1728	6	4	N/A
contraceptive	1473	9	3	Life
credit	690	15	2	Financial
echocardiogram	131	12	2	Life
flag	194	28	8	N/A
german	1000	19	2	Financial
heart	920	13	5	Life
hepatitis	155	19	2	Life
horse	368	27	2	Life
monks-1	556	6	2	N/A
monks-2	601	6	2	N/A
monks-3	554	6	2	N/A
mushroom	8124	22	2	Life
pima	768	8	2	Life
postoperative	90	8	3	Life
promoters	106	57	2	Life
spect	267	22	2	Life
survival	306	3	2	Life
ta_eval	151	5	3	N/A
tic-tac-toe	958	9	2	Game
vehicle	846	18	4	N/A
voting	435	16	2	Social
wine	178	13	3	Physical
zoo	101	16	7	Life

30 Public UCI Certain Datasets

Real values have been discretizated.

Convert to Uncertain Datasets

Two parameters:

Uncertain Attribute Number:

Number of attributes selected to converted to uncertain. Those with highest information gain values are selected.

Convert to Uncertain Datasets

Two parameters:

Uncertain Attribute Number:

Number of attributes selected to converted to uncertain. Those with highest information gain values are selected.

Uncertain Degree (0 - 1):

The probability the attribute value taking values other than the original value.

Convert to Uncertain Datasets

Two parameters:

Uncertain Attribute Number:

Number of attributes selected to converted to uncertain. Those with highest information gain values are selected.

Uncertain Degree (0 - 1):

The probability the attribute value taking values other than the original value.

Represented by Ux@y, where x is uncertain degree and y is uncertain attribute number.

Accuracy Evaluation

Average accuracies on 30 datasets. For accuracy on each dataset, refer to our paper.

Using SVM Classifier:

Dataset	uHARMONY	DTU	uRule
U10@1	**79.0138**	74.8738	75.2111
U10@2	**78.6970**	73.1629	73.4107
U10@4	**77.9657**	72.2670	69.4649
U20@1	**78.9537**	74.6577	74.6287
U20@2	**78.6073**	72.5642	72.5460
U20@4	**77.8352**	69.9157	68.2066

Accuracy Evaluation

Average accuracies on 30 datasets. For accuracy on each dataset, refer to our paper.

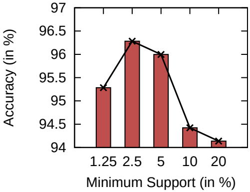
Using SVM Classifier:

Dataset	uHARMONY	DTU	uRule
U10@1	**79.0138**	74.8738	75.2111
U10@2	**78.6970**	73.1629	73.4107
U10@4	**77.9657**	72.2670	69.4649
U20@1	**78.9537**	74.6577	74.6287
U20@2	**78.6073**	72.5642	72.5460
U20@4	**77.8352**	69.9157	68.2066

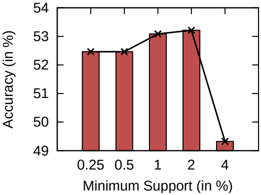
Using Rule-based Classifier:

Dataset	uHARMONY^{rule}	DTU	uRule
U10@4	**73.2517**	72.2670	69.4649

Sensitivity Test



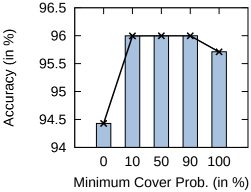
(a) breast



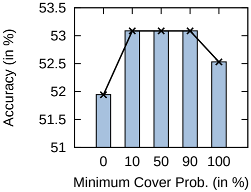
(b) wine

Figure: Accuracy Evaluation of U10@1 w.r.t. Minimum Support

Sensitivity Test



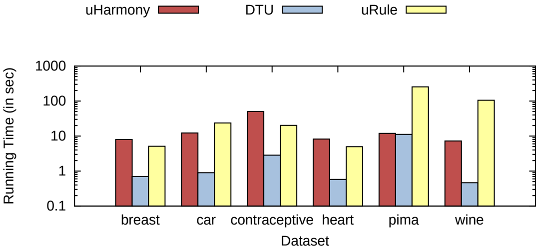
(a) breast



(b) wine

Figure: Accuracy Evaluation of U10@1 w.r.t. Minimum Cover Prob.

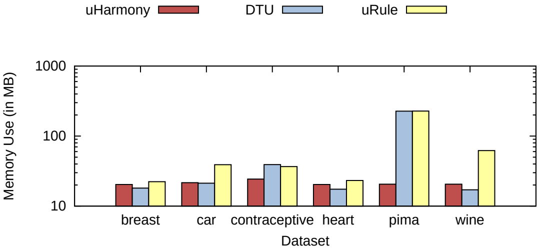
Runtime Efficiency



(a) Running Time (in sec)

Figure: Classification Efficiency Evaluation of U10@1

Runtime Efficiency



(b) Memory Use (in MB)

Figure: Classification Efficiency Evaluation of U10@1

Effectiveness of the Expected Confidence Upper Bound

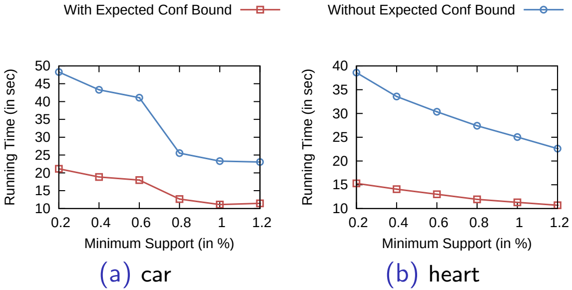
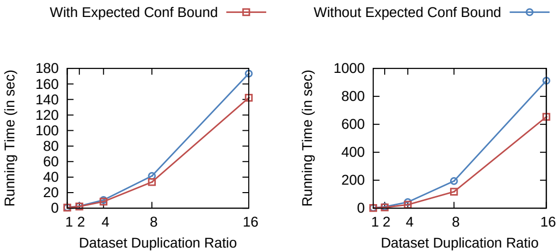


Figure: Running Time Evaluation of U10@4

Scalability Test



(a) car ($sup_{min} = 0.01$) (b) heart ($sup_{min} = 0.01$)

Figure: Scalability Evaluation (U10@1, Running Time)

Conclusions

- Proposed the first associative classification algorithm on uncertain data.

Conclusions

- Proposed the first associative classification algorithm on uncertain data.
- Proposed an efficient computation of expected confidence value, together with the computation of upper bounds.

Conclusions

- Proposed the first associative classification algorithm on uncertain data.
- Proposed an efficient computation of expected confidence value, together with the computation of upper bounds.
- New instance covering strategy has been proposed and tested to be effective.

Conclusions

- Proposed the first associative classification algorithm on uncertain data.
- Proposed an efficient computation of expected confidence value, together with the computation of upper bounds.
- New instance covering strategy has been proposed and tested to be effective.
- Conducted an extensive evaluation on 30 public real data, under varying uncertain parameters. With significant improvements on accuracy, comparing with two other state-of-the-art alrotihms.

Conclusions

- Proposed the first associative classification algorithm on uncertain data.
- Proposed an efficient computation of expected confidence value, together with the computation of upper bounds.
- New instance covering strategy has been proposed and tested to be effective.
- Conducted an extensive evaluation on 30 public real data, under varying uncertain parameters. With significant improvements on accuracy, comparing with two other state-of-the-art alrotihms.
- Evaluated the runtime efficiency, proved the effectiveness of using upper bounds.

The End

Thank you for Listening!

Questions or Comments?