

# Maintenance of case bases: current algorithms after fifty years

**J. M. Juarez**<sup>1</sup>, S. Craw<sup>2</sup>, J. Ricardo Lopez-Delgado<sup>1</sup> , M. Campos<sup>1</sup>

<sup>1</sup> University of Murcia

<sup>2</sup> Robert Gordon University

<https://www.ijcai.org/proceedings/2018/0770.pdf>



UNIVERSIDAD DE  
MURCIA



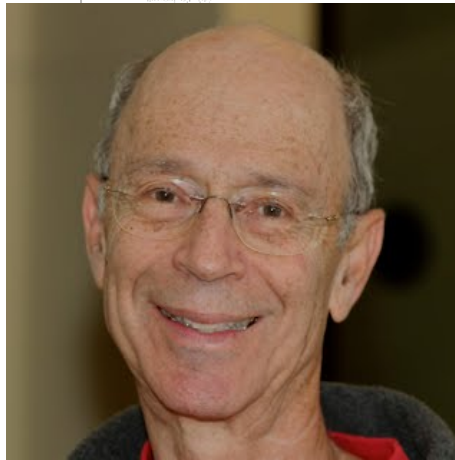
July 16 IJCAI-ECAI 2018 , Stockholm  
<http://www.ijcai-18.org/>

FUNDED BY MINECO WITH ID  
PROJECT TIN2013-45491-R.

# 50 years



"The Condensed Nearest Neighbor Rule," IEEE Trans. on Information Theory, Vol. IT-14, No. 3, pp 515-516 (May 1968)



Peter E. Hart CNN algorithm

CORRESPONDENCE:

Since, by (8)

(11)

(12)

(13)

(14)

> 1.

hold,

Then,

region on for

the

question then arises of how actually to implement these

desirable expression, i.e., of how to store a function such as

with a finite number of bits. The obvious implementation, using finite

precision, is to store the function in a table. Perhaps the storage

of a function in a table is not the best. For example, if the function is

of a finite set of points, the function values are the probabilities, and the

integrals are replaced by probabilities, and the integrals by finite sums.

C. G. HEBRON, JR., UNIVERSITY OF TEXAS AT AUSTIN, TEXAS

REFERENCES

1. S. G. Priddy, "Technique for reducing patterns without a loss of information," Stanford Electronics Laboratories, Stanford, Calif., Tech. Rept. 8105-0, March 1965, (A Stanford approach to IEEE Trans. Information Theory, vol. IT-13, pp. 57-64, February 1967).

2. C. G. Hebron, Jr., and D. G. Lathrop, "Typical nearest-neighbor pattern recognition," Proc. 4th Annual Princeton Conf. Information Sciences and Systems, March 30-31, 1967.

\* The technique is used in Hebron and Lathrop.

The Condensed Nearest Neighbor Rule

The purpose of this note is to introduce the condensed nearest neighbor decision rule (CNN rule) and to pose some unsolved theoretical questions which it raises. The CNN rule, one of a class of AOC decision rules which have appeared in the literature in the past few years, was motivated by statistical considerations

Manuscript received November 21, 1966; revised October 8, 1967.

515

pertaining to the nearest neighbor decision rule (NN rule). We briefly review the NN rule and then describe the CNN rule.

The NN rule assigns an unclassified sample to the same class as the nearest of a stored, correctly classified sample. In other words, given a collection of n reference points, each classified by some external source, a new point is assigned to the same class as its nearest neighbor. The most interesting theoretical property of the NN rule is that under very mild regularity assumptions on the underlying statistics, for any metric, and for a variety of loss functions, the large-sample risk incurred is less than twice the Bayes risk. (The Bayes decision rule achieves minimum risk but requires complete knowledge of the underlying statistics.) From a practical point of view, however, the NN rule is not a prime candidate for many applications because of the storage requirements it imposes. The CNN rule is suggested as a rule which retains the basic approach of the NN rule without imposing unduly stringent storage requirements.

Before describing the CNN rule we first define the notion of a consistent subset of a sample set. This is a subset which, when used as a stored reference set for the NN rule, correctly classifies all of the remaining points in the sample set. A minimal consistent subset is a consistent subset with a minimum number of elements. Every set has a consistent subset, since every set is trivially a consistent subset of itself. Obviously, every finite set has a minimal consistent subset, although the minimum size is not, in general, achieved uniquely. The CNN rule uses the following algorithm to determine a consistent subset of the original sample set. In general, however, the algorithm will not find a minimal consistent subset. We assume that the original sample set is arranged in some order; then we set up bins called zones and operate as follows:

- 1) The first sample is placed in zone 1.
2) The second sample is classified by the NN rule, using as a reference set the current contents of zones. (Since zones has only one point, the classification is trivial at this stage.) If the second sample is classified correctly it is placed in zone 1; otherwise it is placed in zone 2.
3) Proceeding inductively, the ith sample is classified by the current contents of zones. If classified correctly it is placed in zone i; otherwise it is placed in zone i+1.
4) After one pass through the original sample set, the procedure continues to loop through GRABAG until termination, which can occur in one of two ways:

- a) The GRABAG is exhausted, with all its members now transferred to zones (in which case, the consistent subset found is the entire original set).
b) One complete pass is made through GRABAG with no transfers to zones. (If this happens, all subsequent passes through GRABAG will result in no transfers, since the underlying decision surface has not been changed.)
5) The final contents of zones are used as reference points for the NN rule; the contents of GRABAG are discarded.

Qualitatively, the rule behaves as follows: If the Bayes risk is small, i.e., if the underlying densities of the various classes have small overlap, then the algorithm will tend to pick out points near the (perhaps fuzzy) boundary between the classes. Typically, points deeply imbedded within a class will not be transferred to zones, since they will be correctly classified. If the Bayes risk is high, then zones will contain essentially all the points in the original sample set, and no important reduction in sample size will have been achieved. No theoretical properties of the CNN rule have been established.

The CNN rule has been tried on a number of problems, both real and artificial. In order to investigate the behavior of the rule when the classes are (essentially) disjoint—the case in which the CNN rule is of greatest interest—several experiments similar to the following were run. The underlying probability structure for a two-class problem was assumed to consist of two probability densities, each a uniform distribution on the supports shown in Fig. 1. The set of all vectors with integer components lying within each

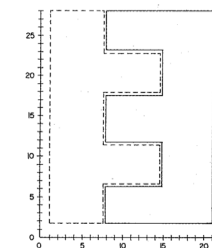


Fig. 1. Class boundaries.

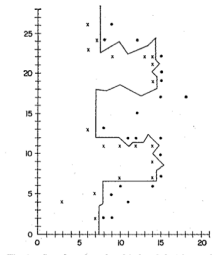


Fig. 2. Samples selected and induced decision surface.

ort was taken to simulate a random sampling from each class. The 482 points thus obtained were ordered by a random number and processed using the algorithm described above. The algorithm terminated after four iterations through GRABAG, with time zones contained 40 samples. Fig. 2 shows the final points and the decision surface induced by the NN rule using 40 samples as a stored reference set. Note that all samples had integer-valued components, ties occurred with nonzero probability, and these were broken arbitrarily. This is due to the fact that occasionally the decision surface lies exactly within one or the other of the supports rather than between them. The points most deeply imbedded within each class were the two points in the random ordering. A more realistic experiment was performed using data supplied by IBM. This data consisted of approximately 12 000 dimensional binary vectors drawn from 25 different statistical distributions. (The data represent upper-case typewritten characters, using "I," typed with nine different styles of fonts.) The 80 samples were divided into a training set and a testing set of approximately equal size, and the CNN algorithm was used on the testing set. The algorithm terminated after four iterations through GRABAG, at which time zones contained 197 of the original 625 samples. An error rate of 1.28 percent was obtained on the independent test set. This was somewhat disappointing in view of the fact a number of simpler classifiers (the ternary reference classifier, linear machine, and piecewise-linear machine) using ideally less computer time, achieved error rates on the order

IEEE TRANSACTIONS ON INFORMATION THEORY, MAY 1968

of 0.3-0.5 percent.<sup>10,11</sup> It was also a little surprising, since (necessarily) the 197 stored points correctly classified all the 625 samples in the training set.

These and similar experiments have persuaded us that the CNN rule offers interesting possibilities, but that a great deal more work of both a theoretical and experimental nature will be needed before the rule is thoroughly understood. For example, under suitably restrictive assumptions on the underlying statistics:

- 1) What is the expected number of iterations before termination?
2) What is the expected reduction in the size of the stored sample set?
3) What is the expected increase in CNN risk over NN risk for a sample set of given size?

In view of the desirable theoretical properties of the k-NN rule,<sup>12-15</sup>—the rule that makes a decision on the basis of votes cast by each of the k nearest neighbors—we pose a final obvious question which should, perhaps, be answered experimentally: How would the CNN rule perform if the vote of, say, the three nearest neighbors was substituted for the decision of the single nearest neighbor everywhere in the algorithm?

PETER E. HART Applied Physics Lab. Stanford Research Institute Menlo Park, Calif. 94025

REFERENCES

1. P. E. Hart, "An asymptotic analysis of the nearest-neighbor decision rule," Stanford Electronics Lab., Stanford, Calif., Tech. Rept. 8209-2 (SRL-66-016), May 1965.
2. R. M. Cover and P. E. Hart, "Nearest-neighbor pattern classification," IEEE Trans. Information Theory, vol. IT-13, pp. 21-27, January 1967.
3. R. M. Cover, "Estimation by the nearest-neighbor rule," IEEE Trans. Information Theory, vol. IT-14, pp. 50-55, January 1968.
4. W. H. White and S. J. Dwyer, III, "Performance and implementation of the k-nearest neighbor decision rule with incorrectly identified training samples," IEEE Proc. 4th Allerton Conf. Circuit and System Theory, vol. EC-15, pp. 914-924, December 1966.
5. C. N. Liu and G. L. Shelton, Jr., "An experimental investigation of a modified fast pattern recognition system," IEEE Trans. Electronic Computers, vol. EC-15, pp. 914-924, December 1966.
6. W. J. Nilsson, Learning Machines—Foundations of Trainable Pattern Classification Systems. New York: McGraw-Hill, 1965.
7. G. G. Casey et al., "An experimental comparison of several design algorithms used in pattern recognition," IBM Corp., Research Rept. RC1500, November 1965.
8. R. S. Neel, "Multifont character-recognition experiments using trainable classifiers," Stanford Research Institute, Menlo Park, Calif., Tech. Note 1, Contract AF 30(602)-3945, August 1966.

Uncertainty and the Probability of Error

Let X and Y be discrete random variables which can be thought of as the input and output, respectively, of a communication channel. Let X and Y take on the values {x\_i: i = 1, ..., m} and {y\_j: j = 1, ..., n}, respectively, where m ≥ n. A decision rule for X in terms of Y can be considered as a partition {A\_i: i = 1, ..., m} such that A\_i ∩ A\_j = φ, i ≠ j, and ∪\_{i=1}^m A\_i = {y\_j: j = 1, ..., n} where the decision is x\_i if Y ∈ A\_i. This also defines a "post-decision" random variable Z, where Z is defined by Z = x\_i if Y ∈ A\_i, i = 1, ..., m. Two putative measures of the efficiency of this system are uncertainty (or equivocation) and probability of error. It is desirable to determine the relationship between these two measures. In particular, we can compare H(X|Y) with the minimum probability of error P\_e(e) if we want to evaluate the channel independent of the decision rule. Otherwise we can compare, given a particular decision rule, H(X|Z) with the probability of error P\_e(e). The purpose of the paper is to demonstrate the exact relationship between H(X|Y) and P\_e(e).

First, we relate H(X|y\_k) to P\_e(e|y\_k) for each k. Now P\_e(e|y\_k) = 1 - max\_j P(x\_j|y\_k), and letting y\_k be fixed, we denote P\_i = P(x\_i|y\_k), i = 1, ..., m, such that P\_i ≥ P\_0, i = 2, ..., m. Then P\_e(e|y\_k) =

Manuscript received September 13, 1966; revised October 16, 1967. The work of D. L. Tabbé was supported by NASA training; the work of S. J. Dwyer, III, was supported in part by the Missouri Regional Medical Program grant of U. S. Public Health.

“The Condensed Nearest Neighbor Rule,” IEEE Trans. on Information Theory, Vol. IT-14, No. 3, pp 515-516 (May 1968)

### CNN Algorithm

**Input:** original case-base C

**Output:** maintained case-base CM

CM  $\leftarrow$  empty

c  $\leftarrow$  first case of C

C  $\leftarrow$  C - {c}

CM  $\leftarrow$  CM  $\cup$  {c}

**Repeat**

**For all** c in C

    C  $\leftarrow$  C - {c}

**if not** CorrecClassifyNN(c,CM)

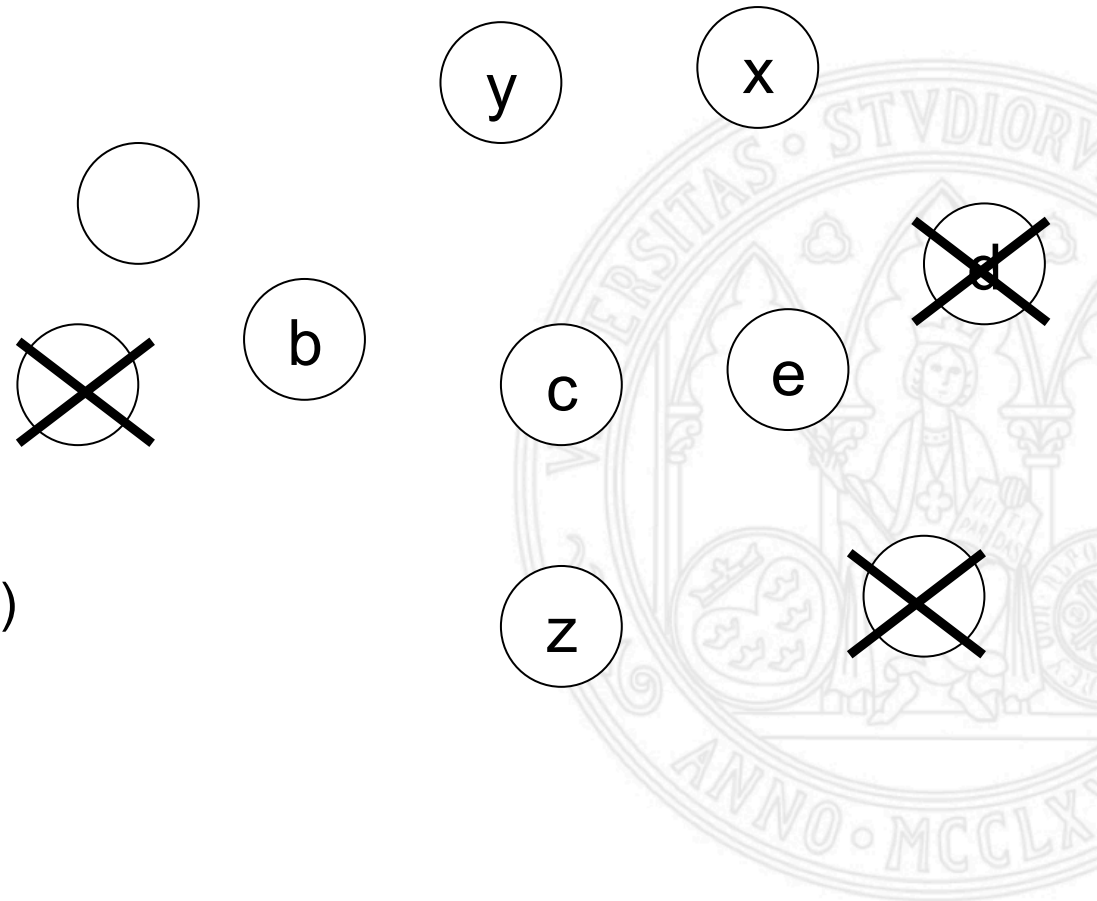
      CM  $\leftarrow$  CM  $\cup$  {c}

**endif**

**enfor**

**Until** C without changes

**Return** CM



- **OUTLINE:**

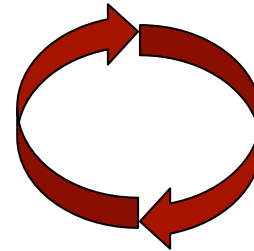
1. 50 years map of CBM algorithms
2. Advances last 5 years
3. Conclusions



- **CASE-BASED REASONING (CBR)**

- Solves by retrieving similar problems (cases)

- Retrieve, Adapt, Learn



- $\neq$  model-based systems

- No structure: atomic **case** = (problem, solution)

- **Case Base**: Knowledge incremented **dynamically**

- Challenge: number of cases

- Approach: Case-Base **Maintenance** (CBM) alg.

# • 50 years CBM map

NN period

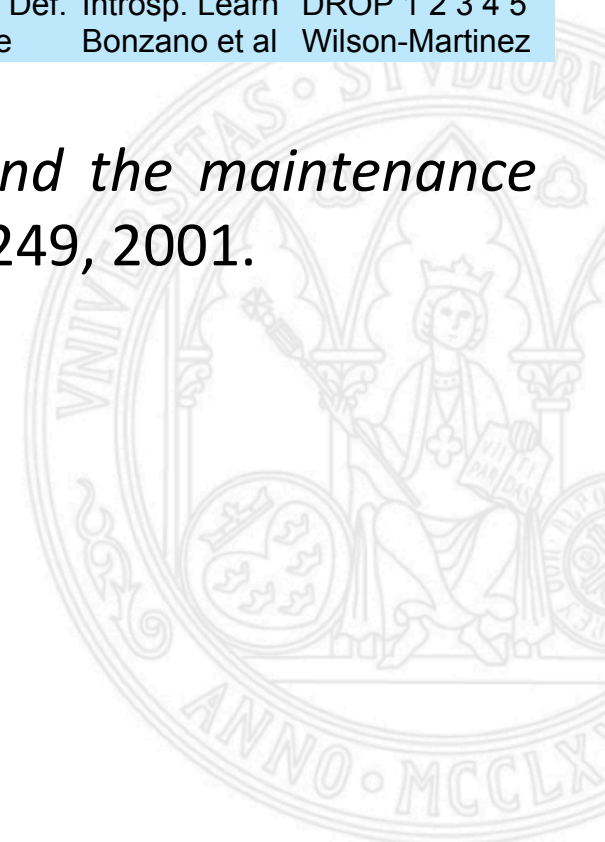
1968	1972	1972	1975	1976	1976	1987	1991	1995	1997	1997-2000
CNN	RNN	ENN	SNN	RENN	All-KNN	SHRINK	IB1 IB2 IB3 IB4	Competence Def.	Introsop. Learn	DROP 1 2 3 4 5
Hart	Gates	Gates	Ritter et al	Tomek	Tomek	Kibler-Aha	Aha-Kibler	Smyth-Keane	Bonzano et al	Wilson-Martinez

B. Smyth and E. McKenna. *Competence models and the maintenance problem*. Computational Intelligence, 17(2): 235-249, 2001.

!= Machine Learning: purpose of each case in CBR

COMPETENCE: capacity of each case to solve

- Coverage
- Reachability
- Competence Groups



# • 50 years CBM map

NN period

1968	1972	1972	1975	1976	1976	1987	1991	<b>1995</b>	1997	1997-2000
CNN	RNN	ENN	SNN	RENN	All-KNN	SHRINK	IB1 IB2 IB3 IB4	<b>Competence Def.</b>	Introspect. Learn	DROP 1 2 3 4 5
Hart	Gates	Gates	Ritter et al	Tomek	Tomek	Kibler-Aha	Aha-Kibler	<b>Smyth-Keane</b>	Bonzano et al	Wilson-Martinez

## Competence period

Meta-CB	IBL-DS	CTE	CBE	<b>COV</b>	<b>RFD</b>	<b>RC</b>	Maint. Retrieval	KeepItSimple	ICF	<b>Competence Model</b>
CumminsDerek	Beringer-Hüllermeier	Craw et al	Delany-Cunningham	<b>McKenna-Smyth</b>			Craw-Jarmulak	Yang-Wu	Brighton-Mellis	<b>Smyth-McKenna</b>
2011	2007	2007	2004	<b>2001-2000</b>			2001	2000	1999	<b>1998</b>



- **OUTLINE:**
  1. 50 years map of CBM algorithms
  - 2. Advances last 5 years**
  3. Conclusions



- **Last 5 years**
  - Intense research
  - Novel (theoretical) models
  - New problems (non existing)
  - Real life solutions



# Last 5 years: CBM map

NN period

1968	1972	1972	1975	1976	1976	1987	1991	1995	1997	1997-2000
CNN	RNN	ENN	SNN	RENN	All-KNN	SHRINK	IB1 IB2 IB3 IB4	Competence Def.	Introsop. Learn	DROP 1 2 3 4 5
Hart	Gates	Gates	Ritter et al	Tomek	Tomek	Kibler-Aha	Aha-Kibler	Smyth-Keane	Bonzano et al	Wilson-Martinez

## Competence period

Meta-CB	IBL-DS	CTE	CBE	COV	RFD	RC	Maint. Retrieval	KeptItSimple	ICF	Competence Model
CumminsDerek	Beringer-Hüllermeier	Craw et al	Delany-Cunningham	McKenna-Smyth	Craw-Jarmulak	Yang-Wu	Brighton-Mellis	Smyth-McKenna		
2011	2007	2007	2004	2001-2000	2001	2000	1999	1998		
2014	2014	2015	2017	2015	2015	2017	2014	2016	2015	2016
WCOID-GM	Closure	CM	SCBM	FFD	Pref-CBM	FP-CA	T-CBM	NEFCS/SSR	CBNI	MOE-CBM
Smiti-Elouedi	Lu et al	Chebel et al	Smiti-Elouedi	Leake-Schack	Abdel-Hüllermeier	Mathew-Chakraborti	Lupiani etal	Lu etal	Yamamoto et al	Lupiani etal

Competence Improvements

Re-structuring case-bases

Time

Complexity

Period 2014-2018

- **Last 5 years: complex problems**

- **Multi-objective optimization**

- **Search Space: Case-Base**

- **Size vs. Accuracy**

- **Noise vs. Redundancy**

- **MOEA: NSGA-II**

- **Optimization:**

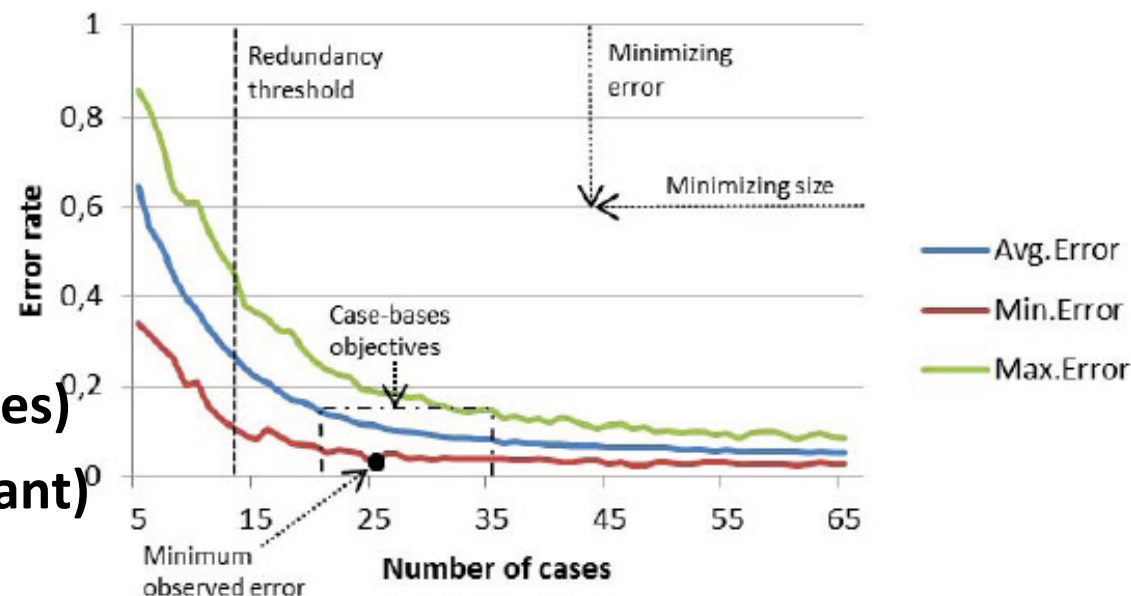
- **Min(no redundant cases)**

- **Min(dist. Non-redundant)**

- **Max(accuracy system)**

MOE-CBM Lupiani-Massie-Craw et al 2016

Case-bases that fitness function is searching for



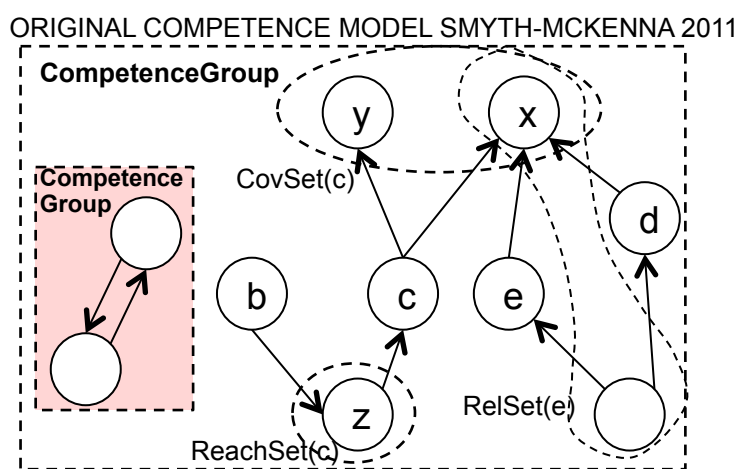
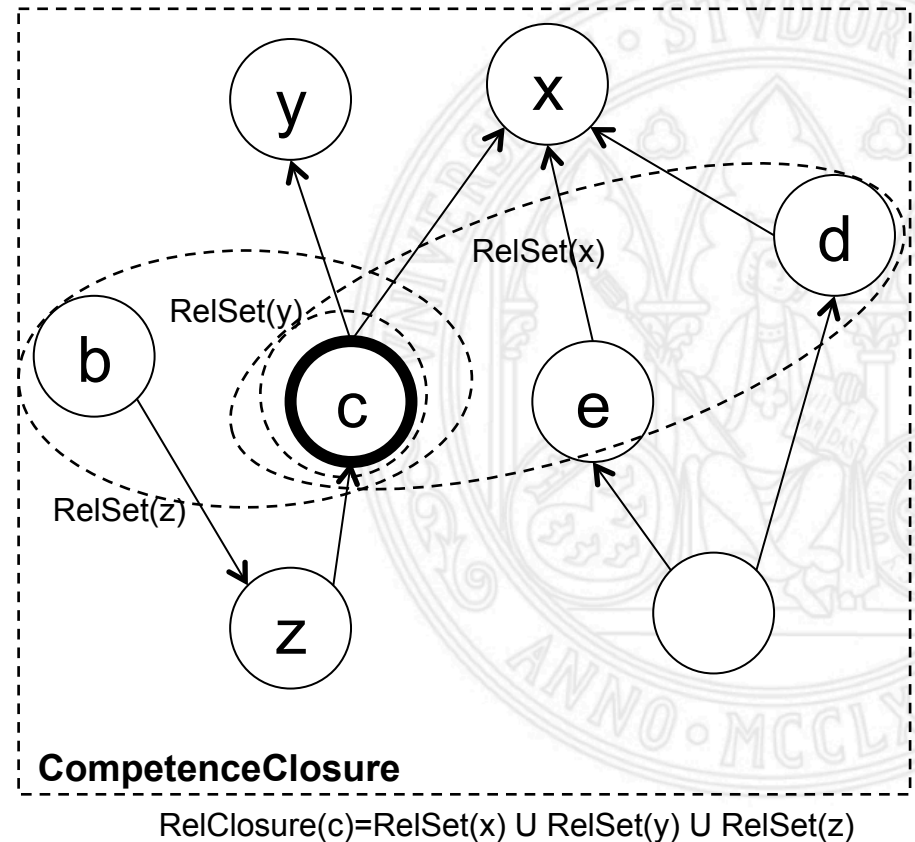
- **Last 5 years: improving competence**

- **Closure-Competence Model**

- **Inadequate Comp. Model**
- **Disjoint partitions (consistent)**

$$CompetenceClosure(G) \iff \forall c, c' \in G, \exists SharedCoveragePath(c, c') \wedge \forall c_k \in C - G, \nexists c \in G : SharedCoverage(c_k, c)$$

Lu-Zhang-Lu Competence Model Proposal 2014



Competence closure and Related closure example

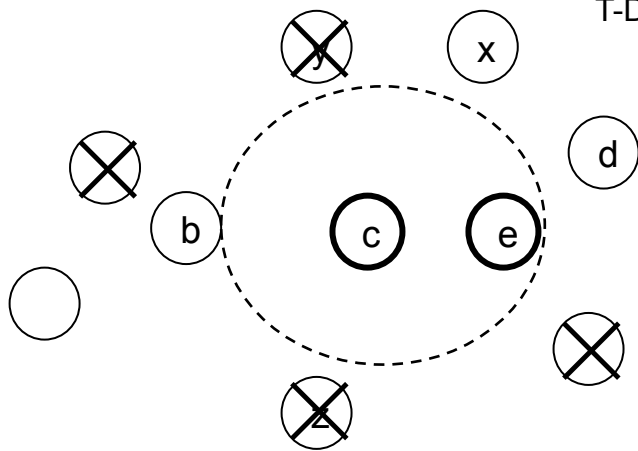
• Last 5 years: temporal dimension (real life problems)

– Temporal CBM

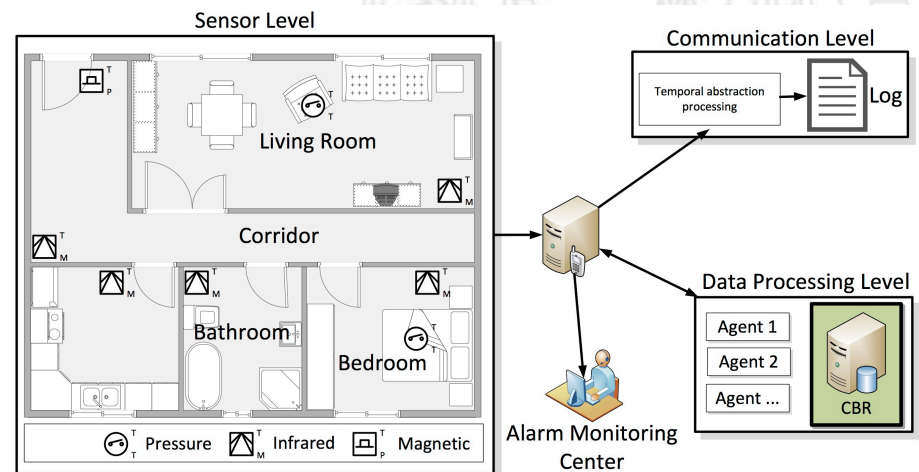
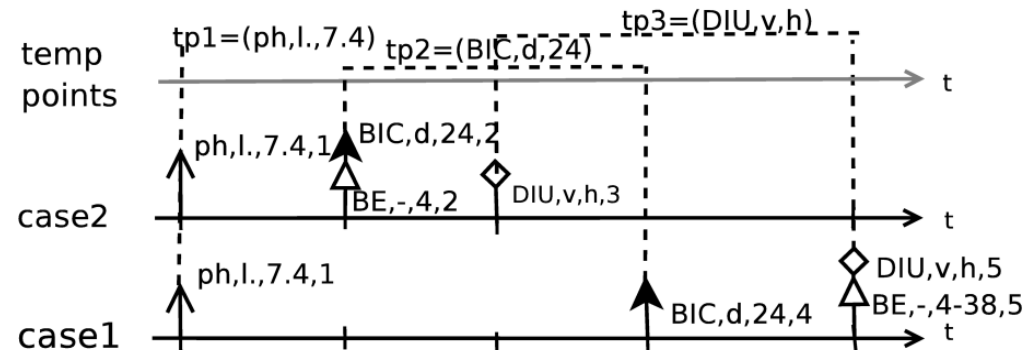
- Case: sequence of events
- Adapt CBM approaches

TEMPORAL MAINTENANCE:

T-CNN  
T-RENN  
T-DROP1-3  
T-ICF  
Etc.



T-CBM Lupiani-Juarez-Palma 2014



E. Lupiani, J.M. Juarez, J. Palma, and Roque Marin. Monitoring elderly people at home with temporal case-based reasoning. Knowledge-Based Systems, 134:116 – 134, 2017.

E.Lupiani, J.M. Juarez, and J. Palma. A proposal of temporal case-base maintenance algorithms. In Luc Lamontagne and Enric Plaza, editors, Case-Based Reasoning Research and Development, pages 260–273. Springer, 2014.

- **Last 5 years: re-structuring case-bases**

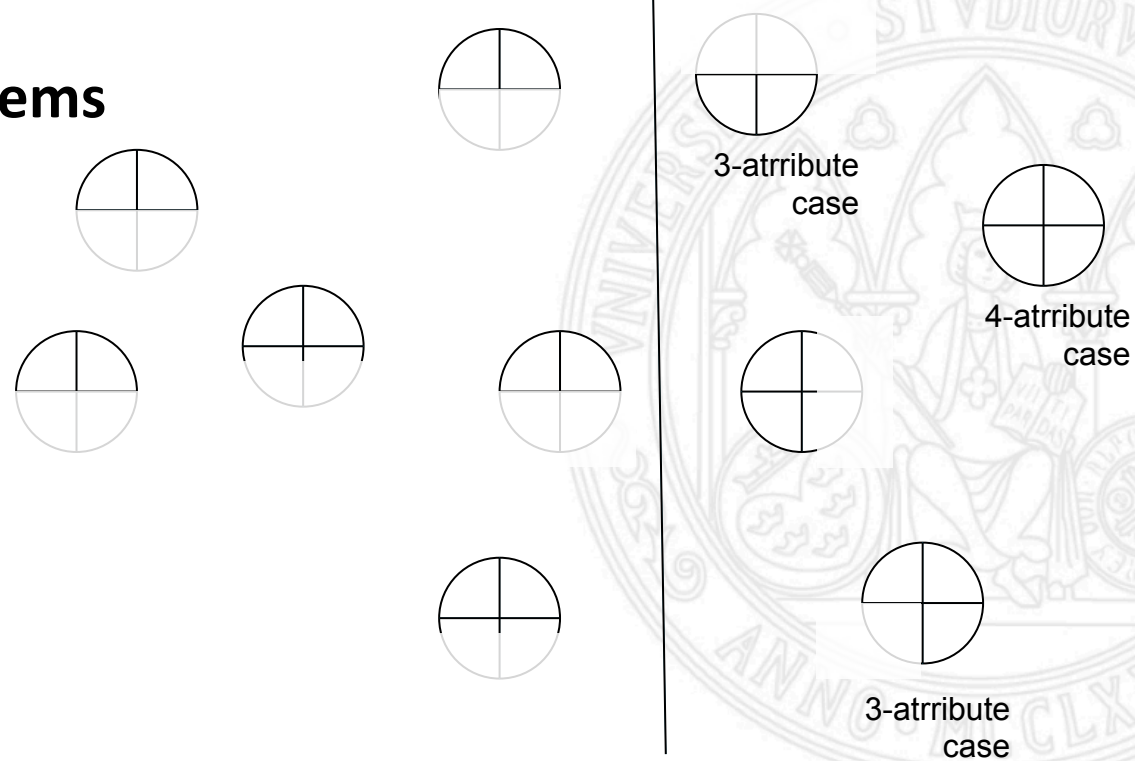
- **Flexible Feature Deletion**

- **Remove part case**
- **Less competence loss**
- **High dimensional problems**

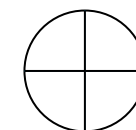
FFD Leak-Schack, 2015

Orthogonal Deletion 2 features

Local Feature Deletion



**Example 4-feature case**

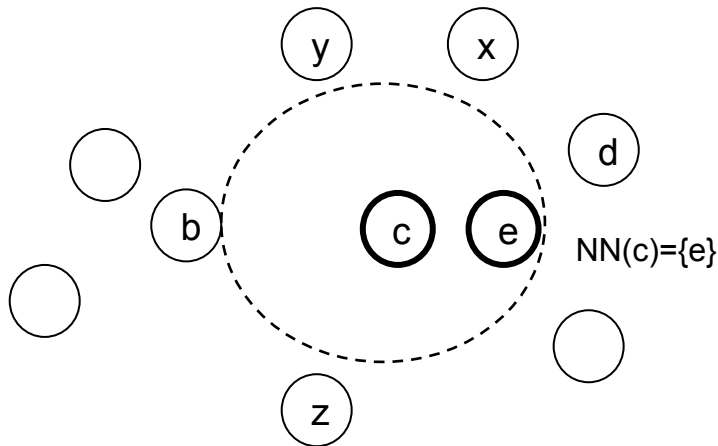


- **Last 5 years: re-structuring case-bases**

- **Preference CBR Model**

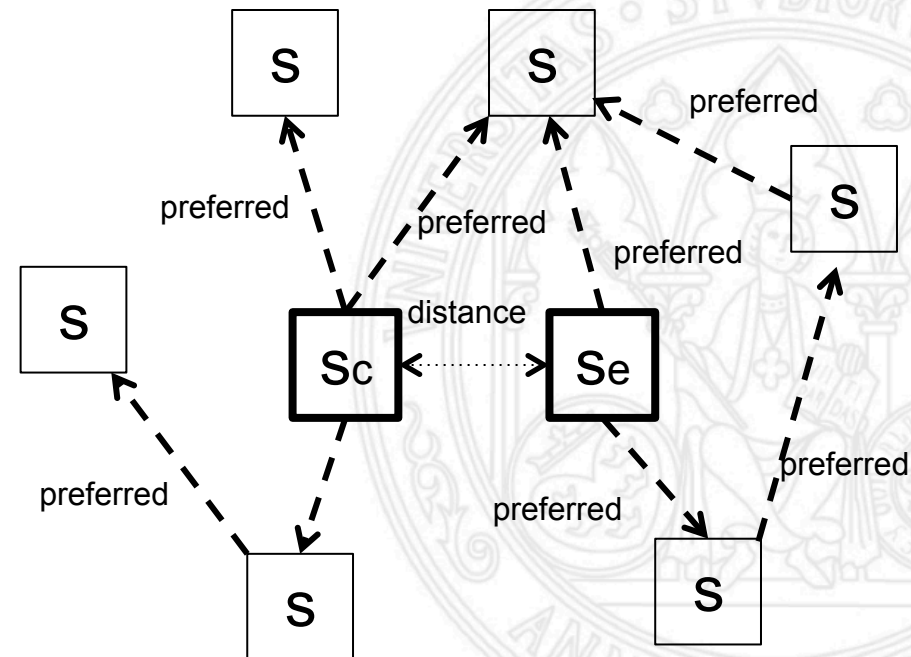
- **Solution  $S_i$  preferred to  $S_j$**
- **Decompose case structure**
- **$c=(prob,sol,PrefSet)$**

PROBLEM DIMENSION:



Pref-CBR approach Abdel-Aziz & Hüllermeier, 2015

SOLUTION DIMENSION (PREFERENCES)



Example Pref-CBM approach



- **OUTLINE:**

1. 50 years map of CBM algorithms

2. Advances last 5 years

- 3. Conclusions**



- **Conclusions**

- **Deep Impact of Competence Model Smyth-McKenna-Keane**
- **New potential CBR applications (not explored in depth):**
  - **Interpretable reduction of high dimensional problems: e.g. Flexible Feature Deletion[Leake&Schack,2015]**
  - **Social network datasets: e.g. Compositional Adaptation [Mathew&Chakraborti,2017]**
- **Applications in the Industry:**
  - **Monitoring: T-CBM [Lupiani et al 2015]**
  - **Long term use of intelligent systems: Drift-CBM [Lu et al 2016]**
- **Future directions:**
  - **Fair comparisons (now limited to classical CBM alg.)**

## Acknowledgements (CBM algorithms authors)

A. Abdel-Aziz	M. K. Haouchine	R. Rowe
D. Aha	T. Kawabe	M. Salamó
M.K. Albert	M.T. Keane	B. Schack
J. Beringer	D. Kibler	A. Smiti
A. Bonzano	R. Knauf	B. Smyth
D. Bridge	Y. Kobayashi	Y. Sukarai
H. Brighton	D. Leake	I. Tomek
S. Chakraborti	R. Lopez de Mantaras	S. Tsuruta
B. Chebel-Morello	S.R. Lowry	D.R. Wilson
S. Craw	J. Lu	N. Wiratunga
L. Cummins	N. Lu	H.B. Woodruff
P. Cunningham	E. Lupiani	J. Wu
Z. Elouedi	R. Marin	Y. Yamamoto
G. Gates	T.R. Martinez	Q. Yang
E. Golobardes	S. Massie	N. Zerhouni
P. Hart	D. Mathew	G. Zhang
E. Hüllermeier	E. McKenna	
T.L. Isenhour	C. Mellish	
S. J. Delany	J. Palma	
J. Jarmulak	G.L. Ritter	
J.M. Juarez		



## Acknowledgements (CBM algorithms authors)

A. Abdel-Aziz

M. K. Haouchine

R. Rowe

D. Aha

T. Kawabe

M. Salamó

M.K. Albert

M.T. Keane

B. Schack

J. Brington

R. Chaffin

B. Shoykhet

A. Borra

R. Chaffin

B. Shoykhet

D. Bridge

Y. Kobayashi

Y. Sukarai

H. Brighton

D. Leake

I. Tomek

S. Chakraborti

R. Lopez de Mantaras

S. Tsutsui

B. Chebel-Morello

S.R. Lowry

D.R. Wilson

S. Craw

J. Lu

**jmjuarez@um.es**

L. Cummins

N. Lu

M. M. Veloso

P. Cunningham

E. Lupiani

J. Wu

Z. Elouedi

R. Marin

Y. Yamamoto

G. Gates

T.P. Martinez

Q. Yang

E. Golobardes

S. Massie

N. Zerhouni

P. Hart

D. Mathew

G. Zhang

T.L. Isenhour

C. Mellish

S. J. Delany

J. Palma

J. Jarmulak

G.L. Ritter

J.M. Juarez

**Jose M. Juarez**

DOWNLOAD PAPER (OPEN ACCESS) :

<https://www.ijcai.org/proceedings/2018/0770.pdf>