

Sequences classification by least general generalisations

Fabien TORRE
joint work with F. TANTINI and A. TERLUTTE

INRIA LNE and CNRS LIFL (Mostrare) – LORIA (Parole)

ICGI 2010, Valencia

Outline of the talk

- 1 VOLATA: a generic classification system
- 2 Sequence classification by combining...
 - Automata
 - Balls of words
- 3 Experiments and discussion

Supervised classification with VOLATA

A supervised classification problem

needs to define:

- 1 an input space: \mathcal{X} ;
- 2 a finite set of discrete classes: \mathcal{Y} ;
- 3 an hypothesis language: $\mathcal{H} \supseteq \mathcal{X}$ and $h \in \mathcal{H} : \mathcal{X} \rightarrow \mathcal{Y}$;
- 4 a subsumption relation \succeq between hypotheses.

VOLATA

- requires a generalisation operator called *least general generalisation* (LGG);
- then provides several learning methods, especially ensemble methods.

Least General Generalisations

Definition: *lgg*

Given a set of examples $E \subseteq \mathcal{X}$, an hypothesis $h \in \mathcal{H}$ is a *least general generalisation* of E iff:

- $\forall e \in E : h \succeq e$;
- there exists no h' such that $\forall e \in E : h' \succeq e$ and $h \succ h'$.

... and if the *lgg* is unique

Two possible definitions of the LGG operator:

- $\text{LGG}(e_1, e_2, \dots, e_n \in \mathcal{X})$ returns $h \in \mathcal{H}$;
- $\text{LGG}(h_{n-1}, e_n)$ returns $h \in \mathcal{H}$.

We prefer the second one, more efficient for learning.

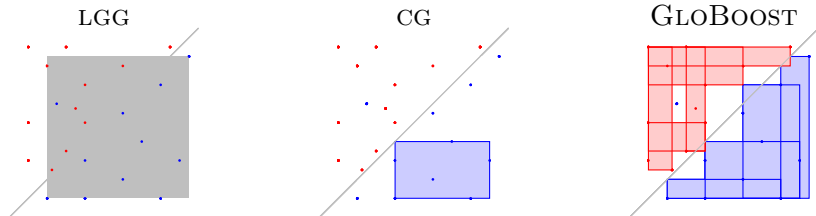
VOLATA: a three-levels architecture

- (level 1) LGG operator computes the least general generalisation of a set of examples. Follows from (\mathcal{H}, \succeq) .
- (level 2) examples generalisation using LGG and classes. For a given class, CG (*correct generalisation*) generalises examples one by one and checks correction of each generalisation wrt other classes. Depends on the presentation order of examples.
- (level 3) full classifiers learning. GLOBBOOST is an ensemble method that uses the order dependency of CG to obtain random correct hypotheses and combine them with a *one hypothesis, one vote* principle. Importance of diversity.

Only the first level depends on hypothesis language \mathcal{H} , the two others are generics.

LGG, CG and GLOBBOOST in action

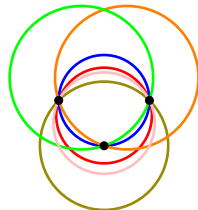
In the plane with examples/points and hypotheses/rectangles:



works only because least general rectangles are unique...

The case of disks in the plane

In the plane,
with examples/points and hypotheses/disks,
there is an infinity of least general disks
that include three points.



Requirements to classify with VOLATA

- ① choose appropriate (\mathcal{H}, \succeq) ;
- ② guarantee that least general hypotheses are unique;
- ③ define the corresponding LGG operator.

Application to grammatical inference and sequence classification?

Outline of the talk

- 1 VOLATA: a generic classification system
- 2 Sequence classification by combining...
 - Automata
 - Balls of words
- 3 Experiments and discussion

Least general generalisations and grammatical inference

Comparison

Learning in the limit. When a positive example arrives, the learner must propose a language that contains seen examples, and finally the target language.

LGG operators learn in the limit!

LGG operator in GI context.

Given a language L and a word w , it provides the smallest language that contains L and w .

LGG operators in learnability proofs

Available for:

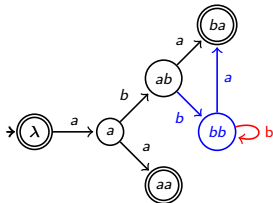
- k -TSS automata [García and Vidal, 1990]
- 0-reversible automata [Angluin, 1982]

but not for balls of words [de la Higuera et al., 2008].

LGG operators for k -TSS and 0-reversible automata

LGG-TSSI:

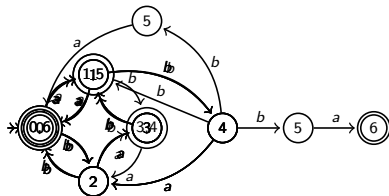
- 1 $S = \{\lambda, aa, aba\}$ and $k = 3$;
- 2 learned automaton:



- 3 adding *abba* and *abbba*.

LGG-ZR:

- 1 $S = \{\lambda, aa, bb, abab, baba\}$
- 2 learned automaton:



- 3 adding *abba*.

Now able to combine such automata and predict class sequences!

Balls of words: another representation for languages

Balls of words [de la Higuera et al., 2008]

- three unit cost edit operations:
 - insertion : $aab \rightarrow aab\bar{b}$
 - deletion : $a\bar{a}b \rightarrow ab$
 - substitution: $a\bar{a}b \rightarrow a\bar{b}b$
- edit distance: $d(w_1, w_2)$ is the minimum number of operations needed to transform w_1 into w_2 ;
- a language is defined by a centre (a word) and a radius:
 $e \in B_2(bab)$ (or $B_2(bab) \succeq e$) iff $d(bab, e) \leq 2$;
- a ball of words represents a finite language;
- learnability results, by positive examples, with noisy data.

LGG operator for balls of words ?

Multiple least general generalisations for balls of words

A sample with multiple *lgg*

- Let $E = [a, b, ab]$;
- $h = B_1(a)$ subsumes all examples in E ;
- $h' = B_1(b)$ subsumes all examples in E ;
- h contains aa , h' contains bb : h and h' are not comparable;
- both h and h' are *lgg* of E because all balls included in both h and h' do not subsume E (like $B_1(\lambda)$ that does not subsume $ab \in E$).

The *unique lgg assumption* is not true for balls of words.
We have to define a new generalisation operator for balls!

An algorithm for generalising balls of words

Algorithm GBALL

Require: $e \in \mathcal{X}$ an example, $h = B_r(o) \in \mathcal{H}$ an hypothesis.

Ensure: $g \in \mathcal{H}$ a generalisation of e and h ($g \succeq h$ and $g \succeq e$).

- 1: $c = o \xrightarrow{*} e$ {a shortest path}
- 2: let x, y integers and o' a word such that $o \xrightarrow{x} o' \xrightarrow{y} e$
- 3: $x = d(o, o')$, $y = d(o', e)$, $x + y = d(o, e)$
- 4: $r' = \max(r + x, y)$
- 5: **return** $B_{r'}(o')$

About GBALL

- no experimental difference between strategies to chose o' ;
- properties of this algorithm?

Order dependency



Dependency on the presentation order of examples,
interesting property for ensemble methods!

Monotonic generalisation...

If $g = \text{GBALL}(e, h)$ then $g \succeq e$ and $g \succeq h$.

Proof.

Recall of the algorithm: $x + y = d(o, e)$ and $r' = \max(r + x, y)$.

- $g \succeq e$ because $d(o', e) = y \leq r'$ and therefore $e \in B_{r'}(o')$;
- $g \succeq h$ because for each word $w \in B_r(o)$ and by triangular inequality:

$$d(o', w) \leq d(o', o) + d(o, w)$$

therefore $d(o', w) \leq x + r \leq r'$ and $w \in B_{r'}(o')$.



... but not least general

Counterexample

- Let $E = [a, b]$;
- first hypothesis = first example = $h = B_0(a)$;
- two possibilities to choose σ' on the path $a \xrightarrow{1} b$;
- either $\text{GBALL}(h, b) = B_1(a)$, or $\text{GBALL}(h, b) = B_1(b)$;
- but the ball $B_1(\lambda)$ contains E ;
- and is more specific than the computed hypotheses:
 $B_1(a) \succeq B_1(\lambda)$ and $B_1(b) \succeq B_1(\lambda)$.

our method generalises a little too much...

Non monotonic correct generalisation

Example

Recall: CG generalises using GBALL and checks correction.

- Let $E^+ = [\lambda, b, a]$ and $E^- = [bb]$
- CG with GBALL and $x = 1$ provide successively:
 - $B_0(\lambda)$ (initial hypothesis);
 - $B_1(b)$ (rejected because $B_1(b)$ accepts bb);
 - $B_1(a)$ (validated).
- $B_1(a) \succeq b$ while the adding of b has been previously rejected.

Implies less efficient ADABOOST implementation...

Summary

Balls have multiple *lgg*. GBALL is order dependant and monotonic but does not give a *lgg* ball and leads to a non monotonic CG.

Outline of the talk

- 1 VOLATA: a generic classification system
- 2 Sequence classification by combining...
 - Automata
 - Balls of words
- 3 Experiments and discussion

Experiments (1): UCI sequential datasets [Asuncion and Newman, 2007]

Classical GI methods against ensemble methods.

Datasets and protocol

- tic-tac-toe, badges, us-first-names, promoters, splice;
- 10-fold cross validation, 90% of data for learning.

Competitors

- Majority, RPNI, TRAXBAR, RED-BLUE;
- GLOBOOST + LGG-TSSI and GLOBOOST + LGG-ZR
- GLOBOOST + GBALL and random strategy.

Results on UCI datasets

	tic-tac-toe	badges	promoters	first-name	splice
Majority	65.34 %	71.43 %	50.00 %	81.62 %	50.26 %
RPNI	91.13 %	62.24 %	-	81.42 %	-
TRAXBAR	90.81 %	57.48 %	56.60 %	81.37 %	58.33 %
RED-BLUE	93.89 %	61.09 %	63.02 %	82.83 %	54.65 %
GLGG-TSSI ₁₀₀₀	91.47 %	72.69 %	61.13 %	89.50 %	78.07 %
GLGG-ZR ₁₀₀₀	98.36 %	71.43 %	50.00 %	83.07 %	-
GGBALL ₁₀₀₀	92.95 %	80.41 %	87.63 %	87.10 %	93.76 %
GGBALL ₁₀₀₀₀	94.69 %	81.39 %	88.43 %	88.80 %	95.63 %
GGBALL ₁₀₀₀₀₀	94.96 %	81.36 %	89.08 %	89.06 %	95.62 %

Observations: combinations are better than classical methods

- reversible: too specific;
- automata: too slow.
- balls: fastness+diversity;
- balls: good in genomic.

Experiments (2): handwritten digit classification

VOLATA and balls on a real problem.

Datasets and protocol

- *Nist special database 3*;
- 10 classes, 10 568 examples;
- 10-fold cross validation, 10% of data for learning;

Competitors

- competitor: SeDiL [Boyer et al., 2008].
- GLOBBOOST + GBALL and random strategy.

Results

SEDiL	95.86 %
GGBALL ₁₀₀₀	93.81 %
GGBALL ₁₀₀₀₀	95.93 %
GGBALL ₁₀₀₀₀₀	96.32 %

Good results
for VOLATA and balls!

Summary and perspectives

Automata

- unique *lgg* for two classes;
- corresponding LGG available and embedded in VOLATA;
- not fast, very specific;
- other language classes with or without unique least general generalisations?
- classes that reach regular languages by union.

Balls

- GBALL usable by VOLATA;
- fast generalisations and fast classifications;
- great diversity, many balls;
- good experimental results;
- real applications;
- study hollow balls.

Thank you for your attention. Any questions?

Bonus

- 4 Why GBALL?
- 5 Automata
- 6 Balls
 - Learned balls
 - VC-dim
- 7 Laplace
- 8 Algorithms
- 9 Bibliography

Bonus

- how to select a specific ball generalisation?
- what are the differences between LGG and GBALL?
- about LGG-TSSI? about LGG-ZR?
- how do the learned balls seem? digits, tic-tac-toe, us-first-name, badges;
- how to manage noisy classifications?;
- what are the full learners? DLG, GLOBo, GLOBOOST, BAGGING-LGG, ADABOOST-LGG;
- how to prove infinite VC-dim for balls?
- how to mix weak learners?
- how to positionate VOLATA wrt GI challenges?
- what is the bibliography of this work?

Distinguish a peculiar ball

Choose a *lgg* ball

Which one? smallest ball?

Theorem from [de la Higuera and Casacuberta, 2000] says that finding the centre of the smallest ball containing a finite set of words is a NP-hard problem.

Choose a non-*lgg* ball

Which one? the ball with the first example as center ($x = 0$)?

We loose the diversity of balls:

- the result only depends on the first example;
- very few balls: the number possible balls is the number of training examples.

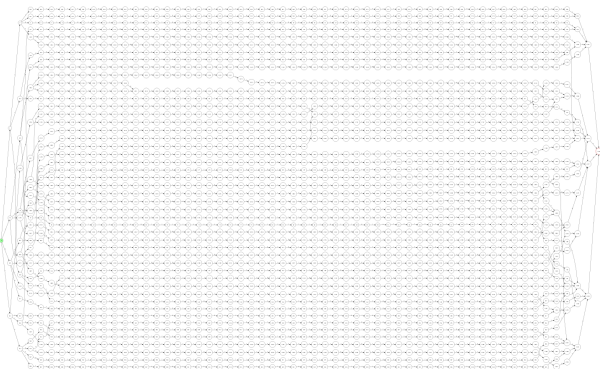
what are the differences between LGG and GBALL?

Comparison between a unique LGG and our GBALL:

	LGG	GBALL
least general generalisation	ok	ko
order dependency	ko	ok
monotony of G	ok	ok
monotony of CG	ok	ko

About LGG-TSSI algorithm

About LGG-ZR algorithm



6 Balls

- Learned balls
- VC-dim

A ball of digits

A learned ball for the zero class

$B_{19}(2233323434444455665666766000000012111702311)$

visualisation of the centre:



Covering: 36 zero-examples



etc.

all with distance 19 to the centre.

Balls of tic-tac-toe endgames

A learned ball

$B_5^+(xxxobxxbxxx)$

subsumes 80 positive examples (distance 5 to centre), 0 negative example.

The distance between an example and the nearest example is always 2. The average distance between two examples is 4.1.

Covered examples

xxxobboob	xxxobobobx
xxxobobox	xxxobboxo
xxobbbxoo	xxoobobx
xxoobbox	xxobxboxo
xoxobxob	xoxobobx
xxxooboxb	xxxobobxo
xxxobbxoo	xxxobboox
xxoobboxb	xxoobbxo
xxobxobox	xoxobxbo
xoxobxbox	etc.

In the plane

We can put 16 points on a circle of radius 5, with a distance 2 between a point and the next one.

Then, the average distance between two points is 6.8.

Balls of first-names

[▶ back to conclusion](#)

A learned ball: $B_7(LRLRTSVKCA)$
contains 346 female first-names (all at distance 7) and 0 male:

- ALBERTHA
- BERTA
- DRUSILLA
- ELSA
- FRANCESCA
- HORTENSIA
- JESSIKA
- KRYSTINA
- LORENZA
- MIRTA
- NERISSA
- OCTAVIA
- PARTICIA
- REBBECA
- SYLVIA
- TERESSA
- URSULA
- VERONICA
- etc.
- average distance between examples : 4.9
- average min distance between examples : 1.2
- maximal min distance between examples : 3.0

Two balls of badges

COLT'94 : + if vowel at the second letter of the badge, else -.

B_{12}^+ (*MICHAELI.JORDAFFER*)

- 1 MICHAEL W. BARLEY (8)
- 2 RICHARD A. CARUANA (11)
- 3 MARK W. CRAVEN (12)
- 4 PAUL W. GOLDBERG (12)
- 5 MICHAEL I. JORDAN (4)
- 6 MICHAEL KEARNS (11)
- 7 MICHAEL LITTMAN (11)
- 8 MICHAEL REDMOND (10)
- 9 CULLEN SCHAFFER (12)
- 10 MICHELE SEBAG (12)
- 11 KIMMEN SJOLANDER (12)

B_{14}^- (*L.THORNEMCCAMMULAS*)

- 1 ANTOINE CORNUEJOLS (14)
- 2 THOMAS ELLMAN (14)
- 3 THOMAS E. KAMMEYER (14)
- 4 GRIGORIS KARAKOULAS (12)
- 5 SRIDHAR MAHADEVAN (14)
- 6 R. ANDREW MCCALLUM (11)
- 7 L. THORNE MCCARTY (6)
- 8 J. R. QUINLAN (14)
- 9 R. BHARAT RAO (14)
- 10 CLAUDE SAMMUT (14)

Dimension VC des boules

Theorem ([Janodet, 2010])

The VC-dimension of balls of words, with $|\Sigma| = 2$, is infinite.

Sketch of the proof for $n = 5$

- five words: *baaaa*, *abaaa*, *aabaa*, *aaaba*, *aaaab* ;
- labeling with two positive examples: *baaaa*(+), *abaaa*(+), *aabaa*(-), *aaaba*(-), *aaaab*(-);
- the ball $B_2(\textit{bbaaa})$ contains + examples and not the - ones.

Remarks

- can be generalise for all n and for all labeling;
- the proof uses a hollow ball;
- a hollow ball *looks like* a *lgg*.

Noisy classes, Laplace accuracy [Clark and Niblett, 1989]

Idée

Let t the total number of examples subsumed by an hypothesis and g the number of examples well predicted by this hypothesis,

$$\text{LaplaceAccuracy} = \frac{b + 1}{t + k}$$

We want this accuracy to grow (Volata second level).

Measures

$$\begin{aligned} \text{Accuracy} &= \frac{b}{t} = \frac{1}{1} = 100\% = \frac{2}{2} = 100\% = \frac{7}{8} = 87.5\% = \frac{8}{9} = 88.8\% \\ \text{Laplace accuracy} &= \frac{b+1}{t+k} = \frac{1+1}{1+2} = 66.67\% = \frac{2+1}{2+2} = 75\% = \frac{7+1}{8+2} = 80\% \end{aligned}$$

Algorithme MGC

Require: $E = [e_1, \dots, e_n] \subseteq \mathcal{X}$ an ordered set of examples from the same class, N a set of counter examples.

Ensure: $h \in \mathcal{H}$ a generalisation of E , **maximally** correct wrt E and N .

1: $h = e_1$

2: **for** $i = 2$ to n **do**

3: $h' = \text{MG}(h, e_i)$ {Generalisation between two hypotheses.}

4: **if** $(\forall e \in N : h' \not\subseteq e)$ **then**

5: $h = h'$ {(correct hypothesis) h' becomes the current generalisation.}

6: **end if**

7: **end for**

8: **return** h

GLOBBOOST algorithm [Torre, 2005]

Require: n examples (x_i, y_i) and T the number of iterations.

Ensure: H the final classifier.

- 1: **for** $t = 1$ to T **do**
- 2: $target =$ randomly chosen class
- 3: $P = \{x_i | y_i = target\}$
- 4: $N = \{x_i | y_i \neq target\}$
- 5: randomly suffle P
- 6: $h_t = CG(P, N)$ {call to CG algorithm}
- 7: **end for**
- 8: **return** $H(x) = \text{sign} \left(\sum_{t=1}^T h_t(x) \right)$

Bibliographie I



Angluin, D. (1982).

Inference of reversible languages.

Journal of the ACM, 29(3):741–765.



Asuncion, A. and Newman, D. (2007).

UCI machine learning repository.



Boyer, L., Esposito, Y., Habrard, A., Oncina, J., and Sebban, J. (2008).

Sedil: Software for edit distance learning.

In Daelemans, W., Goethals, B., and Morik, K., editors,
Proceedings of the 19th European Conference on Machine Learning, pages 672–677. Springer.

Bibliographie II



Clark, P. and Niblett, T. (1989).

The cn2 induction algorithm.

Machine Learning, 3(4):261–283.



de la Higuera, C. and Casacuberta, F. (2000).

Topology of strings: median string is NP-complete.

Theoretical Computer Science, 230:39–48.



de la Higuera, C., Janodet, J.-C., and Tantini, F. (2008).

Learning languages from bounded resources: The case of the dfa and the balls of strings.

In Clark, A., Coste, F., and Miclet, L., editors, *Proceedings of the 9th International Conference in Grammatical Inference*,

Bibliographie III

volume 5278 of *Lecture Notes in Artificial Intelligence*, pages 43–56. Springer.



García, P. and Vidal, E. (1990).

Inference of k-testable languages in the strict sense and application to syntactic pattern recognition.

IEEE Trans. Pattern Anal. Mach. Intell., 12(9):920–925.



Janodet, J.-C. (2010).

The vapnik-chervonenkis dimension of balls of strings is infinite.

Personal Communication.

Bibliographie IV



Torre, F. (2005).

Globoost : Combinaisons de moindres généralisés.

Revue d'Intelligence Artificielle, 19(4-5):769–797.