Learning the areas of expertise of classifiers in an ensemble

Esma Kilic, Them Alpaydin

Procedia Computer Science 2011 (special issue: WCIT 2010)

January 26, 2012

Resumen

1. Introduction
2. Proposed system
   - Referees
   - Gating
3. Experimental results
   - First experiment

Problem Statement: Base Classifiers Fusion

- Using several base-classifiers in an ensemble introduces a new problem: how to fusion their outputs.
- Typical approaches include:
  - Fixed rules. Examples: averaging or voting
  - Stacked generalization. How to fusion labels is learnt by a second-level classifier, which takes base-classifiers' outputs as its inputs.
- All base-classifiers' output are used to produce the final output
- Maybe not all classifiers are equally accurate in all parts of the input space?

Base-classifier selection

- Static selection: based on accuracy/diversity, a subset of base-classifiers is selected for all test instances
- Dynamic selection: online selection, during classification
  - Woods et al. propose to select the base-classifiers with the nearest training sets to the test instance. Too costly.
  - Kuncheva (2000) proposes to cluster inputs into high density regions and then measure the accuracy of base-classifiers in each cluster. Areas of expertise might not match clusters.
  - Jacob et al. propose the use of gating system. As the base-classifiers learn, the gating network learns how to divide the input space.
  - Ortega et al. propose the use of a referee for each base-classifier. Referees learn to discriminate in which areas the associated classifier classifies properly.

Proposals

- Base classifiers $D_j, j = 1 \ldots L$ already trained with the training set
- Validation set Val composed of $x^t, t = 1 \ldots N$
- Two selection methods:
  - Referees: two-class classifier with output in $[0, 1]$
  - Gating: L-class classifier with L classifiers
- $p_j(x)$ indicates in both cases the confidence that base classifier D_j is the most accurate for test instance $x \in Val$

Linear referees

- Assumes that the expertise region is linearly separable
- Sigmoid function. Referee parameters v_j are optimized to reduce the squared error

$$p_j(x) = \frac{1}{1+\exp\left[-\left(v_j^T x + v_{j0}\right)\right]}$$

Tree referees

- More flexible than linear referees

$$p_j(x) = \frac{\sum_{t=1}^{N} d_{jc}(x^t) 1(x^t \in Leaf)}{\sum_{t=1}^{N} 1(x^t \in Leaf)}$$

where d_{jc} is the posterior probability predicted by D_j for the true class of $x'(c)$, and $1(a)$ is a boolean function which is 1 if a is *true*, 0 otherwise.

Decision Aggregation (DA)

- $$O_i(x) = \sum_{j=1}^{L} w_j(x) d_{ji}(x)$$
- $$y = \arg\max_i O_i(x)$$

, where $O_i(x)$ is the overall output for class i, $w_j(x)$ is 1 if D_j is among the n selected base classifiers, 0 otherwise. y is the final output.

DA Scheme

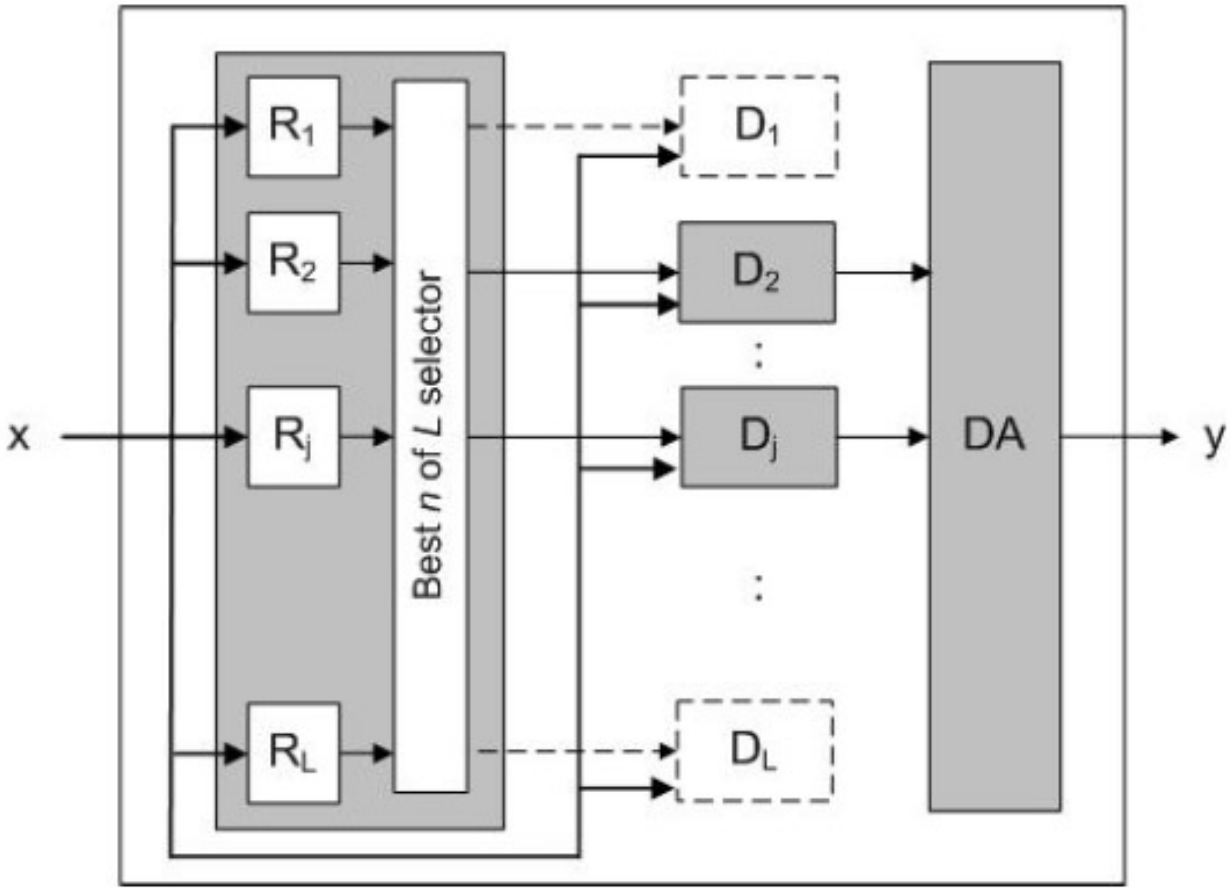

Figure: Testing phase of the referee system

Gating

- Similar to *mixture of experts* but:
  - base classifiers and gating system are trained with different sets of instances
  - base classifiers are general and could be completely different
- Softmax function

$$p_j(x) = \frac{\exp\left(v_j^T x + v_{j0}\right)}{\sum_{l=1}^{L} \exp\left(v_l^T x + v_{l0}\right)}$$

DA Scheme

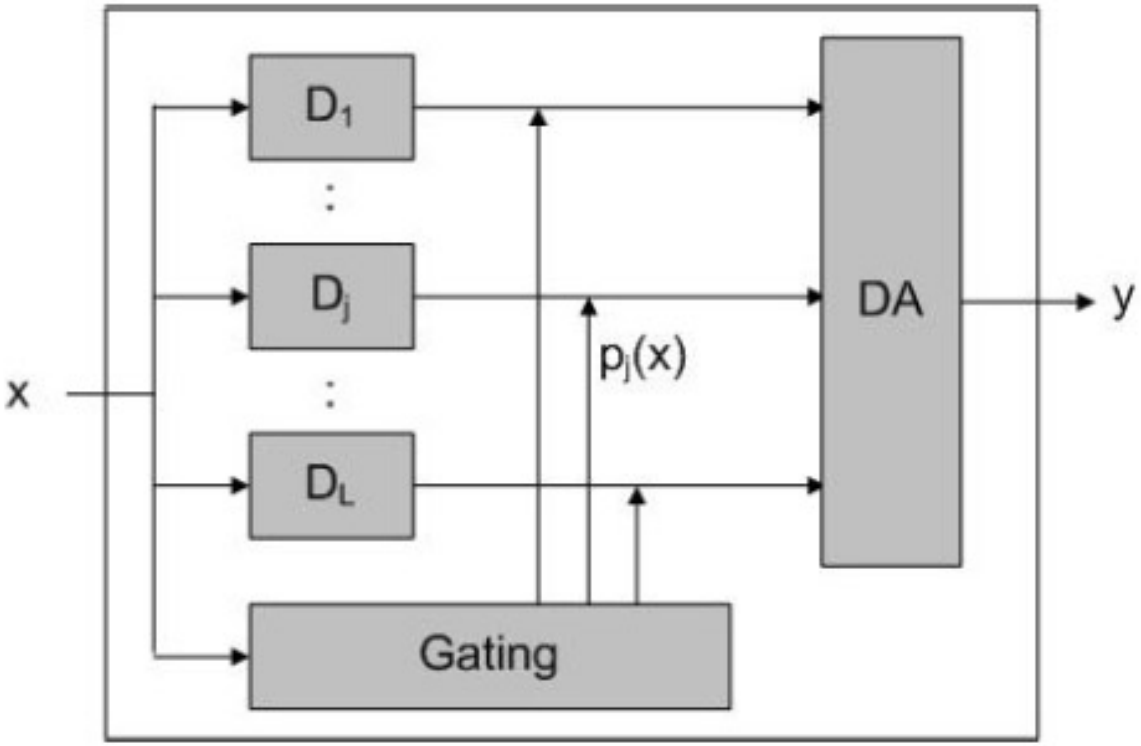

Figure: Testing phase of the gating-based systems

Gating II

- Another option is to weight each base classifier
- *Classes* correspond to classifiers and *class posteriors* to weights in voting

$$p_j(x) = \frac{\sum_{t=1}^{N} 1(class(x^t) = j) 1(x^t \in Leaf)}{\sum_{t=1}^{N} 1(x^t \in Leaf)}$$

Settings

- Databases: 20 from UCI and Delve: australian, balance, breast, car, cmc, credit, mushroom, nursery, optdigits, pageblock, pendigits, pima, ringnorm, segment, spambase, thyrioid, tictactoe, titanic, twonorm, yeast.
- Validation: 1/3 as test set and 2/3 as training set. Then, training set is resampled using 5x2 cross-validation to generate ten training/validation folds. *Tra* is used to train the base classifiers and *Val* is randomly divided in $val - A$ and $val - B$, which are respectively used to train the combiners and finetune the subset size n in the combiners.(???)

Settings II

- 21 base classifiers are used: c45, gau, 1nn-7nn, ldt, log, ml0-ml5, mlt, sm and sv0-sv4
- Five different selection methods are compared: linear referee networks (*rlp*), referee trees (*rdt*), linear gating netwok (*glp*), gating tree (*gdt*), class-independent version of the local competence algorithm from Woods et al (*cin*) and a simple vote over the L classifiers (*vote*)

Results I

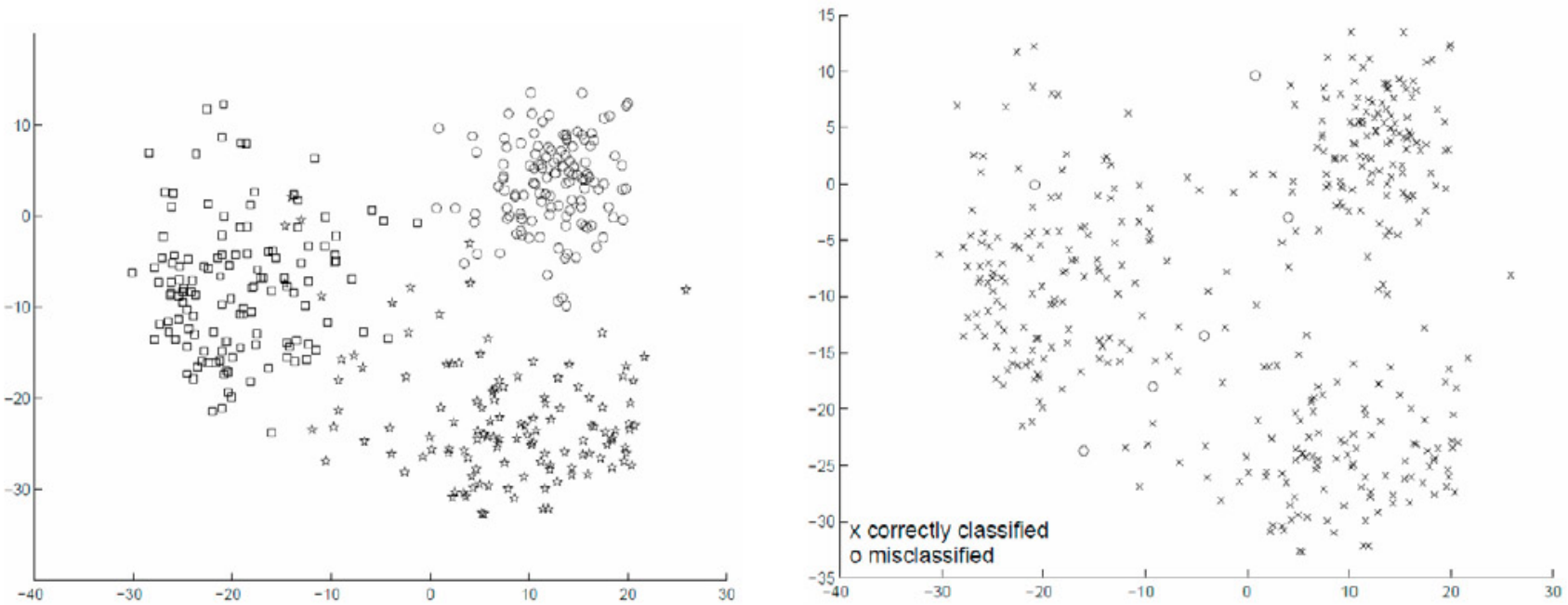

Figure: optdigits database: (a) labeled classes 0,1 and 2 projected using PCA and (b) correctly/incorrectly classified instances using sv2

Results II

- Statistically significant differences are determined using the 5x2 cv F test
- Results are shown as number of databases in which an algorithm obtained a win/ties/loss compared to any other with 95% confidence

Results II-b

algorithm	rlp-n	rdt-n	cin-n	glp	gdt	vote
rlp-n	0/20/0	0/19/1	3/17/0	3/17/0	3/15/2	4/14/2
rdt-n	1/19/0	0/20/0	4/16/0	3/16/1	3/16/1	2/17/1
cin-n	0/17/3	0/16/4	0/20/0	0/17/3	0/18/2	2/16/2
glp	0/17/3	1/16/3	3/17/0	0/20/0	2/15/3	2/16/2
gdt	2/15/3	1/16/3	2/18/0	3/15/2	0/20/0	3/17/0
vote	2/14/4	1/17/2	2/16/2	2/16/2	0/17/3	0/20/0

Results III

- The next table shows the frequency each of the 21 base classifiers are selected with each of the selection algorithms: frequently ($[5.76, \infty]\%$), occasionaly ($(3.76, 5.76)\%$) and rarely ($[0, 3.76\%]$)
- Also, the missclassificaction rate: *correctly* ($[0, 0.5]\%$), *intermediately* ($(0.5, 1)\%$) and *incorrectly* ($[1, \infty]\%$)

Results III-b

algorithm	frequency \ correctness	correctly	intermediately	incorrectly
rlp	7 frequently	0	3	4
	8 occasionally	0	8	0
	6 rarely	5	1	0
rdt	7 frequently	0	4	3
	7 occasionally	0	7	0
	7 rarely	5	2	0
glp	6 frequently	3	1	2
	7 occasionally	4	3	0
	8 rarely	6	2	0
gdt	0 frequently	0	0	0
	20 occasionally	4	15	1
	1 rarely	0	0	1