vicomtech
visual interaction
communication
technologies

From dynamic classifier selection to dynamic ensemble selection

Nikunj C. Oza, Kagan Tumer

Eider Sánchez

graphicsmedia.net ik4 research alliance

Contenidos

Objective of the article

- Introduce classifier ensembles
  - Definitions
    - Classifier ensembles
    - Bias/Variance tradeoff
    - Bayesian interpretation
  - Summarize leading ensemble methods
    - Simple averaging
    - Weighted averaging
    - Stacking
    - Bagging
    - Boosting
    - Order statistics
- Show real-word applications, in 4 different domains:
  - Remote sensing
  - Person recognition
  - One vs. all recognition
  - Medicine

Classifier ensembles

- Classification task:
  - Requires the construction of a statistical model that represents a mapping from input data to the appropriate outputs.
  - Model: intended to approximate the true mapping from the inputs to the outputs
  - Purpose: generate predictions of outputs for new, previously unseen inputs.

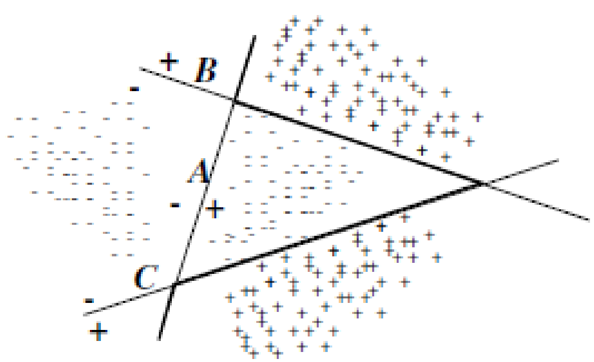
- Single classifier to make predictions for new examples.
  - BUT: many decisions affect the performance of that classifier.
  - Option A: selecting the best available classifier
    - BUT: distribution over new examples that the classifier may encounter during operation may vary
    - BUT: many classifiers are generally tried before a single classifier is selected. Therefore, valuable information discarded by ignoring the performance of all the other classifiers.
  - Option B: Classifier ensembles

Classifier ensembles

Classifier ensembles (combiners or committees)

- **Aggregations of several classifiers whose individual predictions are combined in some manner (e.g., averaging or voting) to form a final prediction.**

- Use all the available classifier information
  - Generally provide better and/or more robust solutions in most applications



Example:
An ensemble of linear classifiers (boldface line).
Each line A, B, and C is a linear classifier.

Classifier ensemble methods

- Simple averaging
- Weighted averaging
- Stacking
- Bagging
- Boosting
- Order statistics

Classifier ensemble methods - Simple averaging

If M classifiers $(h_i^m(x),\ m \in \{1, 2, \ldots, M\})$ are available, the class C_i output of the averaging combiner is:

$$h_i^{\text{ave}}(x) = \frac{1}{N} \sum_{m=1}^{M} h_i^m(x) \qquad (1)$$

- Benefits
  - Reduces the variance of the estimate of the output class posteriors
  - Simple: widely applied to real-world problems
  - Effective ensemble method, particularly in large complex data
- Problems
  - Reduces model error

$$E_{\text{model}}^{\text{ensemble}} = \frac{1 + \rho(M-1)}{M} E_{\text{model}}$$

ρ : Average correlation among the errors of the different classifiers

Classifier ensemble methods - Weighted averaging

- Different classifier weight

$$h_i^{\text{ave}}(x) = \frac{1}{M} \sum_{m=1}^{M} w_m h_i^m(x)$$

- Added degrees of freedom → better solutions
- But in practice
  - failed to provide improvement to justify its added complexity
  - When there is limited training data with which the weights can be properly estimated

Classifier ensemble methods - Stacking

- Actively seeks to improve the performance of the ensemble by correcting the errors
- Stacked generalization addresses the issue of classifier bias with respect to a training set, and aims at learning and using these biases to improve classification
- The main concept is to use a new classifier to correct the errors of a previous classifier

Classifier ensemble methods - Bagging

- Bootstrapped Aggregating (Bagging)
  - Combines voting with a method for generating the classifiers that provide the votes
  - Allow each base classifier to be trained with a different random subset of the patterns with the goal of bringing about diversity in the base classifiers.
- improve upon their base models more if the base model learning algorithms are unstable (ej. Decision trees)
  - differences in their training sets tend to induce significant differences in the models

Classifier ensemble methods - Bagging

- Bootstrapped Aggregating (Bagging)

Bagging(T, M)

For each $m = 1, 2, \ldots, M$,

$T_m = Sample\_With\_Replacement(T, |T|)$

$h_m = L_b(T_m)$

Return $h_{fin}(x) = \text{argmax}_{y \in Y} \sum_{m=1}^{M} I(h_m(x) = y)$

Sample_With_Replacement(T, N)

$S = \emptyset$

For $i = 1, 2, \ldots, N$,

$r = random\_integer(1, N)$

Add $T[r]$ to S.

Return S.

Classifier ensemble methods - Boosting

- AdaBoost algoritm
  - Generates a sequence of base models with different weight distributions over the training set

AdaBoost$(\{(x_1, y_1), \ldots, (x_N, y_N)\}, L_b, M)$
Initialize $D_1(n) = 1/N$ for all $n \in \{1, 2, \ldots, N\}$.
For $m = 1, 2, \ldots, M$:
$h_m = L_b(\{(x_1, y_1), \ldots, (x_N, y_N)\}, D_m)$.
Calculate the error of $h_m : \epsilon_m = \sum_{n:h_m(x_n) \neq y_n} D_m(n)$.
If $\epsilon_m \geq 1/2$ then,
set $M = m - 1$ and abort this loop.
Update distribution D_m:

$$D_{m+1}(n) = D_m(n) \times \begin{cases} \frac{1}{2(1-\epsilon_m)} & \text{if } h_m(x_n) = y_n \\ \frac{1}{2\epsilon_m} & \text{otherwise} \end{cases}$$

Output the final model:
$h_{fin}(x) = \text{argmax}_{y \in Y} \sum_{m:h_m(x)=y} log\frac{1-\epsilon_m}{\epsilon_m}$.

Classifier ensemble methods – Order statistics

- Order statistics combiners that selectively pick a classifier on a per sample basis
- Model error $$E_{\text{model}}^{\text{ensemble}} = \alpha E_{\text{model}}$$
  - Alpha is a factor that depends on the number of classifiers M and the order statistic chosen and the error model

Error reduction factors α, for the min, max and med combiners (Gaussian Error Model)

M	OS combiners	
	min/max	med
1	1.000	1.000
2	0.682	0.532
3	0.560	0.449
4	0.492	0.305
5	0.448	0.287
10	0.344	0.139
15	0.301	0.102
20	0.276	0.074

Real-world applications

- Remote sensing
- Person recognition
- One vs. all recognition
- Medicine

Real-world applications – Remote sensing

- Classification algorithms needs
  - large number of inputs
    - patterns collected repeatedly for large spaces
  - large number of features
    - data is collected across hundreds of bands
  - large number of outputs
    - classes cover many types of terrain (forest, agricultural area, water) and man-made objects (houses, streets)
  - missing or corrupted data
    - different bands or satellites may fail to collect data at certain times
  - poorly labeled (or unlabeled) data
    - data needs to be post-processed and assigned to classes

Real-world applications – Remote sensing

- Example applications
  - Random forests and mountainous terrain
  - Majority voting for agricultural land
  - Hierarchical classification of wetlands
  - Information fusion for Urban areas

Real-world applications – Person recognition

- Person recognition is the problem of verifying the identity of a person using characteristics of that person, typically for security applications
  - Iris recognition
  - fingerprint recognition
  - face recognition
  - behavior recognition
    - such as speech and handwriting
    - recognizing characteristics of a person, as opposed to depending upon specific knowledge that the person may have (such as usernames and passwords for computer account access)
- Problems
  - Involve multiple types of features
  - Difficulty in collecting good data
  - Different misclassification costs
    - Example, denying system access to a legitimate user vs. allowing access to an illegitimate user

Real-world applications – Person recognition

- Example applications
  - Unobtrusive person identification
  - Face recognition
  - Multi-modal person recognition
  - User-specific speech recognition

Real-world applications – One vs. all recognition

- Different types
  - Anomaly detection
    - problem of detecting unusual patterns
    - i.e. what does not fit into the set of identified patterns
  - Target recognition
    - finding what fits into an identified pattern
  - Intrusion detection
    - solved both ways:
    - A. target recognition: look for one of a set of known types of attacks
    - B. anomaly detection : look for anomalies in the usage patterns

vicomtech
visual interaction communication technologies

Real-world applications – One vs. all recognition

- Example applications
  - Modular intrusion detection
  - Hierarchical intrusion detection
  - Intrusion detection in mobile ad-hoc networks

Real-world applications – Medicine

- Different applications:
  - analyzing X-ray images, human genome analysis, and examining sets of medical test data to look for anomalies.
  - Root of all these problems: assessing the health of human beings
- Characteristics
  - limited training and test examples
    - i.e., few training examples due to the nature of problem and privacy concerns
  - imbalanced datasets
    - ie., very few anomalies or examples of patients with a disease
  - too many attributes
    - i.e., often many more than the number of training and test examples
  - different misclassification costs
    - i.e., false negatives significantly worse than false positives.

Real-world applications – Medicine

- Example applications
  - Pharmaceutical molecule classification
  - MRI classification
  - ECG classification

Conclusions

- Each ensemble method has different properties that make it better suited to particular types of classifiers and applications

- New applications, domains with complex and rich data
- Research areas:
  - Ensemble methods oriented at handling large amounts of diverse data
  - Clustering algorithms
  - Distributed classifier ensembles using active/agent-based methods

Gracias