

Genetically Designed Multiple-Kernels for Improving the SVM Performance

Laura Diosan, Mihai Oltean,
Department of Computer Science,
Babes-Bolyai University, Cluj-Napoca, Romania
lauras,moltean@cs.ubbcluj.ro

Alexandrina Rogozan and J. P. Pecuchet
LITIS, Institut National des Sciences
Appliquées, Rouen, France
arogozan,pecuchet@insa-rouen.fr

ABSTRACT

Classical kernel-based classifiers only use a single kernel, but the real-world applications have emphasized the need to consider a combination of kernels - also known as a multiple kernel - in order to boost the performance. Our purpose is to automatically find the mathematical expression of a multiple kernel by evolutionary means. In order to achieve this purpose we propose a hybrid model that combines a Genetic Programming (GP) algorithm and a kernel-based Support Vector Machine (SVM) classifier. Each GP chromosome is a tree encoding the mathematical expression of a multiple kernel. Numerical experiments show that the SVM embedding the evolved multiple kernel performs better than the standard kernels for the considered classification problems.

Categories and Subject Descriptors

I.2.6 [Learning]: Concept learning, Parameter learning

General Terms

Algorithms

Keywords

SVM, Genetic Programming, Kernel

1. INTRODUCTION

Support Vector Machines (SVMs) [4] are an example of intensively explored classifiers. These methods represent the data by means of a kernel function, which defines similarities between pairs of data [3]. One reason for the success of kernel-based methods is that the kernel function takes relationships that are implicit in the data and makes them explicit, the result being that the detection of patterns takes place more easily.

The kernel formalism allows different kernels to be combined. Basic algebra operations such as addition, multiplication and exponentiation preserve the key properties for a kernel (the positive semi-definiteness and the symmetry) and thus allow a simple, but powerful algebra of kernels to exist [3]. For instance, given a set of kernels $\{K_1, K_2, K_3\}$, we can form a linear combination of kernels or, even better, we can consider complex combinations of kernels: $K_1 + K_2 * K_3$ or $c * K_2 + K_3 * exp(K_1)$, where c is a constant.

Copyright is held by the author/owner(s).
GECCO'07, July 7–11, 2007, London, England, United Kingdom.
ACM 978-1-59593-697-4/07/0007.

Our hybrid approach combines a GP algorithm and an SVM algorithm. Each GP chromosome encodes the expression of a multiple kernel used by the SVM algorithm. The quality of a GP chromosome is given by the classification accuracy obtained through running the SVM algorithm (which involves the multiple kernel encoded in the current GP tree) in order to solve a particular classification problem.

A simple way of combining more kernels is very similar to the method used for evolving mathematical expressions (as in the case of regression problem): the kernels play the part of regression variables while regression operations are replaced with those performed between kernels. Despite the fact that any mathematical operator can be used in the regression problem, in our case we must restrict the functions involved in the MK expression. Why? Because not all the operations performed between kernels provide another valid kernel. Only several mathematical functions preserve the key properties of the kernel functions (the positiveness and the symmetry of the Gram matrix), such as *addition*, *multiplication* and *exponentiation* [3]. Therefore, the function set will be: $FS = \{+, \times, \exp\}$ The terminal set contains several classical kernels (such as linear kernel, polynomial kernel, radial basis function (RBF) kernel) and some ephemeral random constants [2]: $TS = KTS \cup \{c_1, c_2, \dots, c_m\}$, where KTS represents the terminal set of classical kernels. Note that a kernel-GP tree must contain at least one kernel in its leaves.

The quality of a GP chromosome is given by the accuracy rate computed by the SVM (that uses as kernel the eMK encoded into the current GP individual) on the test data set. For evaluating the quality of an eMK, we also have to look if that kernel is a valid one (with other words if the eMK satisfies the Mercer conditions [1]). It is possible to use an evaluation method that penalizes the “invalid” eMKs, but we do not need it because the initialization process and the genetic operators guarantee the correctness (from a kernel’s point of view) of a GP chromosome.

2. REFERENCES

- [1] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995.
- [2] J. R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, USA, 1992.
- [3] B. Schoelkopf and A. J. Smola. *Learning with Kernels*. The MIT Press, Cambridge, MA, 2002.
- [4] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer Verlag, New York, 1995.