Nuclear Instruments and Methods in Physics Research A 652 (2011) 29–32

Contents lists available at ScienceDirect

Nuclear Instruments and Methods in Physics Research A

journal homepage: www.elsevier.com/locate/nima

Global parameter optimization for maximizing radioisotope detection probabilities at fixed false alarm rates

David Portnoy*, Robert Feuerbach, Jennifer Heimberg

Johns Hopkins University Applied Physics Laboratory, 11100 Johns Hopkins Road, Laurel, MD 20723, USA

ARTICLE INFO

Available online 7 February 2011

Keywords:
Radiological and nuclear detection
Global optimization
Receiver operating characteristic
Homeland security

ABSTRACT

Today there is a tremendous amount of interest in systems that can detect radiological or nuclear threats. Many of these systems operate in extremely high throughput situations where delays caused by false alarms can have a significant negative impact. Thus, calculating the tradeoff between detection rates and false alarm rates is critical for their successful operation. Receiver operating characteristic (ROC) curves have long been used to depict this tradeoff. The methodology was first developed in the field of signal detection. In recent years it has been used increasingly in machine learning and data mining applications. It follows that this methodology could be applied to radiological/nuclear threat detection systems. However many of these systems do not fit into the classic principles of statistical detection theory because they tend to lack tractable likelihood functions and have many parameters, which, in general, do not have a onetoone correspondence with the detection classes. This work proposes a strategy to overcome these problems by empirically finding parameter values that maximize the probability of detection for a selected number of probabilities of false alarm. To find these parameter values a statistical global optimization technique that seeks to estimate portions of a ROC curve is proposed. The optimization combines elements of simulated annealing with elements of genetic algorithms. Genetic algorithms were chosen because they can reduce the risk of getting stuck in local minima. However classic genetic algorithms operate on arrays of Booleans values or bit strings, so simulated annealing is employed to perform mutation in the genetic algorithm. The presented initial results were generated using an isotope identification algorithm developed at Johns Hopkins University Applied Physics Laboratory. The algorithm has 12 parameters: 4 realvalued and 8 Boolean. A simulated dataset was used for the optimization study; the ''threat'' set of spectra contained 540 SNM and industrial signatures, and the ''benign'' set of spectra contained 240 NORM and medical signatures. As compared to a random parameter search, the statistical optimization was able to able to find parameters that yield significantly higher probabilities of detection for all probabilities of false alarm from 0 to 0.1 (and equal to for probabilities of false alarm greater than 0.1), in a relatively small number of iterations. The number of iterations used, 1000, is also many fewer than would be required for a reasonable systematic search of the parameter space.

1. Introduction

Today there is a tremendous amount of interest in systems that can detect radiological or nuclear threats. Many of these systems operate in extremely high throughput situations where delays can have significant negative impact. Thus, calculating the trade-off between detection rates and false alarm rates is critical for their successful operation. Receiver operating characteristic (ROC) curves have long been used to depict this tradeoff. The methodology was first developed in the field of signal detection [1]. In recent years it has been used increasingly in machine learning and data mining applications [2]. It follows that they could be applied to radiological/nuclear threat detections systems. However, typical isotope identification algorithms used by many radiological/nuclear threat detections systems do not neatly fit into the principles of statistical detection theory. They tend to lack tractable likelihood functions and have many parameters. Empirical ROC curves can be used to assess the performance of systems without tractable likelihood functions [3], but often these systems are further complicated by the fact that they tend to have many parameters, which, in general, do not have a one-to-one correspondence with the classes that are to be identified. In these cases the best strategy seem to be, finding those parameter values that maximize the probability of detection (P_d) for a selected number of probabilities of false alarm (P_{fa}). This work proposes a strategy to overcome these problems by

* Corresponding author.
E-mail address: david.portnoy@jhuapl.edu (D. Portnoy).

0168-9002/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.nima.2011.01.149

empirically finding parameter values that maximize the probability of detection for a selected number of probabilities of false alarm.

While ROC curves can be used to compare and evaluate the performance of classifiers, the main thrust of this work is to generate graphs that can be used for parameter selection. Assuming that when P_{fa} decreases the best P_d associated with that P_{fa} also decreases, a set of parameters for several P_{fa} values are desired such that the number of expected false alarms can be set according to various operating conditions. Thus, the objective of this work is to define a methodology that finds settings for the parameters of isotope detection systems, which yield the best detection probability for several predefined false alarm probabilities.

2. Global parameter optimization

Given enough time and resources, an estimation of all (P_d, P_{fa}) pairs for an algorithm's entire parameter space at a reasonable granularity may be generated. These estimates can then be displayed on ROC scatter-plot. In general, each P_{fa} value will have several corresponding P_d values. For this work we are interested only in the highest P_d and its associated parameter values, for a given P_{fa}. It is important to note that interpolation between points on this graph is liable to result in incorrect conclusions because it cannot be assumed that an interpolation between parameter values will generate the interpolated (P_d, P_{fa}) pair.

For algorithms with more than a handful of parameters performing this exhaustive search of the parameter space becomes virtually impossible. Take for example an algorithm with 10 parameters. If we plan to examine a relatively modest 5 values for each parameter, we have 5^{10} combinations of parameters for which P_d and P_{fa} values must be estimated. Now if each P_d and P_{fa} value requires a conservative 500 samples to compute, and each sample requires 1 s to process, the time to generate this very granular estimate of the (P_d, P_{fa}) pairs would take $(5^{10})(5\,0\,0)(1) \approx 4.9 \times 10^9$ CPU-seconds, or about 154.8 CPU-years. Even if one were to go through this exercise it is questionable whether such a granular estimate would even be useful. Thus we are left with two options: a guided approach where domain experts and algorithm designers make their best guesses as to the sub-regions of the parameter space to explore, or cast the problem into a global optimization task and use machine-learning techniques to perform an optimization. While exploring the parameter space using domain experts and algorithm designers is useful, it can lead to the exclusion of productive portions of the parameter space. The work described here uses a machine-learning method applied to the optimization problem with an aim towards reducing the time required to perform the search while at the same time reducing the bias that may be introduced by domain experts and developers.

Since only the maximum P_d for some predefined set of P_{fa} values (P_{fa} targets) is desired, the problem can be cast into to one of global optimization. For this we require an objective function and an algorithm to perform the optimization. The following objective function is proposed:

$$f(P_d,P_{fat},P_{fae}) = (1-m)(1-P_d) + m\left|\frac{P_{fae}-P_{fat}}{P_{fat}}\right|$$

where

$$m(P_{fat},P_{fae}) = 1-\exp\left(\frac{-(P_{fae}-P_{fat})^2}{2\sigma^2}\right)$$

Here P_{fat} is the probability of false alarm target, P_{fae} is the is the probability of false alarm estimated from the output of the identification algorithm run on the test data, and P_d is the probability of detection estimated from the output of the identification algorithm run on the test data. Because the optimization is attempting to find both the maximum P_d and the target P_{fa}, the function adjusts the weights of these depending on how far away the estimated P_{fa} is from the target P_{fa}. Thus, when there is a large difference between the estimated P_{fa} and the target P_{fa} the function is dominated by the term representing the difference between these two values. As the estimated P_{fa} gets closer to the target P_{fa}, the function becomes dominated by the P_d term. The mixing function m that is used to control the transition from P_{fa} to P_d is an unnormalized Gaussian, which has a single parameter σ. Just as the standard deviation in a Gaussian controls the spread of the distribution, the parameter σ in the mixing function controls how close the maximized P_d needs to be to the target P_{fa}. Smaller σ values will force the maximized P_d values closer to the P_{fa} targets.

While any stochastic optimization technique could be used with the aforementioned objective function, the optimization technique chosen for this work combines elements of simulated annealing [4] with elements of genetic algorithms [5,6]. Genetic algorithms were chosen because they can reduce the risk of getting stuck in local minima. However, classic genetic algorithms operate on arrays of Booleans or bit strings, so simulated annealing is employed to perform mutation in the genetic algorithm.

In the genetic algorithm vernacular, each target P_{fa} is assigned a population of chromosomes. Each chromosome is a set of parameter values; these parameters are the ones to be optimized. A population of chromosomes consists of the most "fit" parameter values as determined by the objective function described above. During one iteration of the optimization, a chromosome is randomly selected from one of the target P_{fa} populations. This chromosome is mutated and with some probability crossed-over with another chromosome. Simulated annealing is used to perform the mutation. New values (or mutated values) are selected from a normal distribution defined by the current value of the parameter, the parameter value range, and the current variance of the distribution. The variance is lowered as the "temperature" of the system decreases. This effectively reduces the size of the mutations as the system "cools" and allows "fit" chromosomes to converge on a solution. After mutation is performed, the newly mutated chromosome is crossed-over with another chromosome taken from one of the target P_{fa} populations with some probability (this probability also is reduced as the system cools). Crossover involves randomly choosing some portion of one chromosome and swapping it with some portion of another chromosome. This results in two new chromosomes that need to be evaluated for fitness. The fitness of the new chromosomes is evaluated for each target P_{fa} using the objective function. A chromosome that has a better fitness than any member of a target P_{fa} population is added to that population, and the least fit chromosome is removed from that population. Every time a new chromosome is added to a population the temperature of the system is reduced, cooling it.

The optimization algorithm has several parameters dealing with mutation and crossover probabilities, control of the system temperature, and σ of the objective function. The values selected were determined using a fake detection algorithm, which consisted of two functions that took 6 values and returned P_d and P_{fa} estimates. This allowed the optimization to be run extremely fast and the optimization algorithm parameters to be set using subjective evaluations. The optimization algorithm parameters used to generate the results shown in this paper are listed below. While they do work well for two isotope identification algorithms, the one described here and a PVT based portal algorithm (results currently unpublished), it may be overly optimistic to assume that they will work well for all detection systems. A more rigorous determination of these optimization algorithm parameters is left to future work. The population size for each target P_{fa} was set to 10; the starting σ for mutation was 6 and was lowered to 0.02 in 50 temperature reduction increments; the