Editorial Manager(tm) for International Congress Series
Manuscript Draft

Manuscript Number:

Title: A Methodology for Training and Validating a CAD System and Potential Pitfalls

Article Type: Full Length Article (FLA)

Section/Category:

Keywords: classification; leave-one-patient-out; cad validation

Corresponding Author: Mr. Mehmet Murat Dundar Siemens MEdical Solutions Inc, USA

First Author: Mehmet Murat Dundar, PhD

Order of Authors: Mehmet Murat Dundar, PhD; Glenn Fung, PhD; Luca Bogoni, PhD; M Macari, MD; A Megibow, MD; Bharat Rao, PhD

Abstract:

A Methodology for Training and Validating a CAD System and Potential Pitfalls

M. Dundar[1*], G. Fung[1], L. Bogoni[1], M. Macari[2], A. Megibow[2], B. Rao[1]

[1] Computer-Aided Diagnosis & Therapy, Siemens Medical Solutions
[2] NYU Medical Center

Abstract. In this study we first discuss potential pitfalls involved in training a classifier for a CAD system and then propose a methodology for a successful validation of a CAD system. Our approach tries to achieve a balance between performing well on the training data while generalizing well on new cases. We performed several experiments to justify each step of the proposed methodology. As our experimental results suggest, one can safely consider leave-one-patient-out in tuning the classifier and selecting the relevant features as a performance measure. However, the final classifier should always be evaluated on an independent test set.

Keywords: classification; leave-one-patient out; cad validation

1. Introduction

A typical process for a Computer-Aided Detection (CAD) system consists of: (1) automatically identify candidates in an image, (2) extract features for each candidate, (3) labels candidates as positive or negative (4) display positive candidates to the radiologist for final diagnosis [1,2,3]. The labeling (or classification) is performed by a classifier that has been trained off-line from a training dataset (a database of images where candidates have been labeled by an expert), and then frozen for use in the CAD system.

The critical requirement of a CAD system, and thus of its classifier, is its ability to generalize well. Namely, it should correctly label new datasets. Because we can build a very large number of different classifiers from the same training data (using hundreds of classification algorithms, each with adjustable parameters), the choice of the classifier is critical. Our methodology for training classifiers tries to achieve a balance between performing well on the training data while generalizing well on new cases. Our approach is summarized below:

A. Set aside a randomly chosen test dataset for final evaluation.
B. Use Leave-One-Patient-Out (LOPO) cross-validation as the principal evaluation metric.
C. Use the least number of features in the final classifier.
D. Assess that chosen classifier has good LOPO performance on the training data, and similarly good performance on the unseen test dataset.

* Corresponding author. *E-mail address*: murat.dundar@siemens.com

We present experimental results to justify each of the above steps. Additionally, we show that although cross-validation is a good scheme to prevent overfitting, it is not perfect. (We can build classifiers with low LOPO error that do generalize poorly.) The only true test of generalization is the classifier performance on totally unseen data.

2. Data Description

The database of high-resolution CT images used in these studies was obtained at NYU Medical Center. Prone and Supine views were acquired for 105 patients for CT-Colonography. 61/105 patients had positive findings with 71 total polyps. Sensitivity and specificity were established with respect to CTC by comparison to results from concurrent fiber-optic colonoscopy. CT data was acquired using a Volume Zoom CT scanner (Siemens Medical Systems, Forchheim, GDR), 4x1mm slice detector, 120kV, 0.5 sec. gantry rotation and effective 50mAs. Effective coverage speed varied between 12 and 14 mm /sec. CT images were reconstructed as 1.25 mm thick sections with a 1.0 mm reconstruction interval.

The proprietary algorithms (patents pending) composing the CAD system process include: data pre-processing, candidate generation, feature extraction, and classification.

The generated candidates and their extracted features (moments of tissue intensity, volumetric and surface shape and texture characteristics) are labeled as potential polyps by a classifier. In the actual workflow, potential polyps would be presented to the physician.

2.1 Training and Test Data

Training and Test Data: The 105 patients were randomly partitioned into two groups: training (n=64) and test (n=41). The test group was sequestered and only used to evaluate the performance of the final classifier. The 125 volumes in the 64 training patients (40 polyps) generated 3565 candidates; 3 patients had only 1 view. The 82 volumes in the 41 test patients (31 polyps) generated 2410 candidates. Many polyps were visible in both views. Only polyps of size >= 3mm were considered – the non-polyp candidates included air, stool, colon folds, and potentially other colonic structures.

3. Classifier Performance Measures

The two classifier performance metrics used are the sensitivity (a polyp is counted as "found" if it is detected in at least one of the volumes from the same patient) and the false positives/volume (false positives are all candidates in a volume that are not polyps). However, these metrics can be computed under different assumptions, which can dramatically change their values, as discussed below.

3.1 Test-Set Performance

This is the performance of a classifier as measured on the test dataset. Classifiers should not be run on the test dataset until all training is completed.

3.2 Leave-One-Patient-Out (LOPO) Cross Validation Performance (training dataset only)

In this scheme, we leave-out both the supine and prone views of one patient from the training data. The classifier is trained using the volumes from the remaining 63 (i.e., 64-1) patients, and tested on both volumes of the "left-out" patient. This process is repeated 64 times, leaving out each of the 64 patients in the training dataset, and the testing errors are averaged to compute LOPO performance. (Note that each LOPO run creates 64 different classifiers.) The entire LOPO process is usually repeated using different classification algorithms; the algorithm with the best LOPO is rerun on all 64 patients to create the final classifier.

3.3 Other Cross-Validation Metrics for training datasets:

Other performance metrics include (1) k-fold cross-validation (divide candidates randomly into k groups ("folds"): train classifiers on (k-1) folds, test on the k^{th} fold, and repeat k times), (2) Leave One Polyp Out cross-validation (create "N" folds, with one polyp in each fold, and do N-fold cross-validation), and (3) Leave One Volume Out cross-validation (like LOPO, except one volume is left out on each iteration).

LOPO is superior because it simulates actual use, wherein the CAD system processes both volumes for a new patient. For instance, with any of the above methods, if a polyp is visible in both views, the corresponding candidates could be assigned to different folds; thus a classifier may be trained and tested on the same polyp (albeit in different views).

4. Experimental Results

We discuss 4 sets of experiments, which in their ensemble verify our methodology approach - steps A through D presented in the first section.

4.1 Experiment 1: Feature Selection

Feature selection is a key component of classification. Each additional feature increases the discriminative power of a classifier, but also increases the risk of overfitting the training data. It is critical, therefore, to use as few features as necessary. We use the "wrapper" method for feature selection [4]: the classifier is run iteratively on the training data using different feature sets – during each iteration, one or more features are added, until cross-validation error converges. Therefore, with "wrapper" feature selection, the classifier itself determines the relevant features. Our final classifier contains 14 features, selected from an initial set of 66.

The following experiment demonstrates that a classifier built using all 66 features has inferior LOPO and test performance than a classifier built with the selected 14 features:

- Using all (66) features: LOPO on 64 training patients: 5.4 false positives/vol with 88% overall sensitivity. Test error: 5.9 fp/vol with 87% sensitivity – the classifier was trained on all 64 patients (with 66 features) and tested on the 41 test set patients.

- Using selected (14) features: LOPO on 64 training patients: 4.1 false positives/vol with 98% overall sensitivity. Test error: 3.9 fp/vol with 94% sensitivity.

4.2 Experiment 2: Re-substitution and LOPO errors

The recent introduction of the powerful kernel concept into classifiers, allows us to model a variety of distributions, but also increases the risk of overfitting. By using a high-capacity classifier (e.g. Kernel Fisher's Discriminant [5] with a radial basis Gaussian kernel) with small kernel width, we can "achieve" excellent re-substitution performance (100% sensitivity and 0.7 fp/vol). However, the same classifier generalizes very poorly, achieving just 26% sensitivity and 0.9 fp/vol on the test set. This is very similar to the LOPO performance on the training data (20% sensitivity, 0.8 fp/vol). The consistency between LOPO and test results clearly shows that LOPO is a better estimate of generalization than the re-substitution performance.

4.3 Experiment 3: Dangers of patient selection

When a system performs badly on some data, there is a strong temptation to exclude that data from the test set – and often it is easy to retrospectively justify the reason for eliminating a given dataset (such as, poor distension, poor image quality). Patients should be selected based on objective previously-defined inclusion criteria (image quality, clinical relevance), prior to running any experiments.

Recall that the 105 original patients were split into train (64) and test (41) randomly. Let us assume that we only had data from 64 patients, and instead of randomly dividing them into train and test sets, we manually "selected" patients in our test set after some preliminary classification runs. By selecting 40 specific patients into the test set and training a classifier on the 24 remaining patients, we achieve excellent performance on the "unseen" test set: 0.7 fp/vol and 100% sensitivity on 18 polyps! However, the LOPO on the 24 patients is just 60% sensitivity and 4.1 fp/vol, and this model would have very poor generalization. This extreme scenario, illustrates how by removing patients from a "test set" (either adding them to the train set or just ignoring them) we can seemingly obtain excellent performance on "test" data.

4.4 Experiment 4: The Need for a Test Set

LOPO is an excellent performance measure – for instance, we can conclude that poor LOPO guarantees poor generalization, and good LOPO performance usually means good generalization. However, good LOPO does not guarantee good generalization.

Recall the results in Experiment 1 (using 14/66 features). We achieved a re-substitution performance of 4.2 fp/vol with 98% sensitivity, LOPO of 4.1 fp/vol with 98% sensitivity, and test performance of 3.9 fp/vol with 95% sensitivity.

Using a larger initial feature set, and selecting 15 features after multiple runs, we achieve a similar LOPO performance of 5 fp/vol with 100% sensitivity. However, test performance is significantly inferior: 6.6 fp/vol with 84% sensitivity.

What has happened? Essentially, we have managed to overfit in cross-validation space. This was not an issue 10 years ago, when a single cross-validation run took days. With advances in computer technology, each LOPO run now takes seconds. The ability to iteratively "wrap" feature selection around a LOPO performance evaluation, allows us to evaluate many algorithms-feature set combinations, some of which, by chance, may have superior LOPO. Therefore, exceptional cross-validation performance, particularly on small datasets with many features, should be viewed with some degree of skepticism, until backed up by similar performance on completely unseen test data.

5. Conclusions

In this paper we addressed some of the key issues related to the validation of a CAD system. As our experimental results suggest, one can safely consider LOPO analysis in tuning the classifier and selecting the relevant features as a performance measure. However, the final classifier should always be evaluated on an independent test set. Satisfying LOPO results indicates good generalization only when supported with an acceptable test performance – the best test of generalization is when the LOPO and test performance are both good and similar. This is the case with our CAD system.

A final note of warning: a common mistake is that of tuning the classifier by continuously observing the classifier performance on the test data until a desirable performance is achieved. When the classifier is tuned according to its performance on the test data, then the test results lose all their credibility since the classifier no longer simulates real-world clinical settings. More importantly, such classifier loses its ability to generalize on new data, which is, as stated in the introduction, one of the critical characteristics of a CAD system.

References

[1] Jerebko A.K., Malley J.D., Franaszek M., Summers R.M.. "Multinetwork classification scheme for detection of colonic polyps in CT colonography data sets", Acad Radiol 2003; 10:154–160.[2] Nappi. J., Yoshida Y., "Automated Detection of Polyps with CT Colonography", Acad Radiol 2002; 9:386–397.

[3] Gokturk SB, Tomasi C, Acar B, Beaulieu CF, Paik DS, Jeffrey RB, + et al.. "A statistical 3-D pattern processing method for computer-aided detection of polyps in CT colonography", IEEE Trans Med Imaging - Dec 2001 (Vol. 20, Issue 12).

[4] G. H. John, R. Kohavi, K. Pfleger, "Irrelevant Features and the Subset Selection Problem", International Conference on Machine Learning, 1994.

[5] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, and K. R. Muller, "Fisher Discriminant Analysis with Kernels", *Neural Networks for Signal Processing IX IEEE*, 1999, pp. 41-48.