Building ICU Artifact Detection Models With More Data in Less Time

Christine L. Tsien, MD, PhD[a], Isaac S. Kohane, MD, PhD[b], Neil McIntosh, MB, DSc[c]
[a]BWH/MGH Harvard Affiliated Emergency Medicine Residency, Boston MA. [b]Children's Hospital Informatics Program, Boston MA. [c]Neonatal Unit, Royal Infirmary, Edinburgh.

As many as 86% of intensive care unit (ICU) alarms are false. Multiple signal integration of temporal monitor data by decision tree induction may improve artifact detection. We explore the effect of data granularity on model-building by comparing models made from 1-second versus 1-minute data. Models developed from 1-minute data remained effective when tested on 1-second data. Model development using 1-minute data means that more hours of ICU monitoring (including more artifacts) can be processed in less time. Compression of temporal data by arithmetic mean, therefore, can be an effective method for decreasing knowledge discovery processing time without compromising learning.

INTRODUCTION

False alarm rates in the intensive care unit (ICU) have been reported to be as high as 86%.[1-3] This can lead to compromised patient care.[4-7] Efforts to decrease false alarms therefore have much potential for effecting improvement in the ICU. Various methods for improving monitoring have been proposed (reviewed elsewhere[8]), but none have seen widespread practical application. We have previously found multiple signal integration of temporal bedside monitor data by decision tree induction to be one technique for detecting artifacts in neonatal ICU data signals.[9] That study, however, was limited because the data available for model development were of 1-minute granularity (one value per minute of monitored time), while in the actual ICU setting, monitored signals are available at a frequency of one value per second (1-second granularity). Moreover, artifacts in the ICU tend to be fleeting, on the order of seconds. A model built from 1-minute data would presumably miss these short-lived artifacts, and for that reason, would not be expected to perform well practically. In this study, we explore the effect of data granularity on model-building by developing new models using 1-second granularity data, and then comparing these with 1-minute granularity models. We also evaluate the performance of 1-minute models run on data of 1-second granularity to simulate performance in the ICU environment.

METHODS

Two different sets of data were used for the experiments: Set A consisted of approximately 200 hours of data values occurring at a frequency of one value per minute (1-minute granularity), while Set B consisted of approximately 74 hours of data values occurring at a frequency of one value per second (1-second granularity). Both sets were collected in 1996 from bedside monitors in the neonatal ICU (NICU) at Simpson Memorial Maternity Pavilion in Edinburgh, Scotland. Four physiological signals were present in each data set: electrocardiogram (ECG) heart rate (hr), measured in beats per minute; mean blood pressure (bp) from an indwelling arterial line, measured in millimeters of mercury; partial pressure of carbon dioxide (co2), collected transcutaneously and measured in kilopascals; and partial pressure of oxygen (o2), also collected transcutaneously and measured in kilopascals. Set A was derived from more than 100 different patients (approximately two hours of data from each patient), while Set B was derived from two different patients (approximately 24 hours from one patient and 50 hours from another patient). Raw data values were available from bedside monitors at a frequency of one value per second. One-second granularity data therefore reflects all values coming from the monitors. To collect 1-minute granularity data, for each of the four signals an arithmetic mean was calculated from the 60 raw values. Only these mean values were then recorded for that minute of bedside monitoring.

Occurrences of artifacts in each of the data streams were visually located and annotated retrospectively by an experienced clinician (N.M.) working constantly with the data collection system in the NICU. Annotated sections of the data streams manifested themselves as raw values affiliated with asterisks, one asterisk per raw value marked as artifact. A sample of the text data for annotated mean blood pressure values of 1-minute granularity is shown in Figure 1.

Derivation of feature attributes from the temporal data streams consisted of calculating (for each of the four signals) eight quantities thought to be potentially

1067—5027/01/$5.00

clinically useful for ICU event detection. These included moving mean ('avg'), median ('med'), maximum value ('high'), minimum value ('low'), range ('range'), standard deviation ('std_dev'), linear regression slope ('slope'), and absolute value of the linear regression slope ('abs_slope'). These eight quantities were calculated for each successively overlapping set of raw values. The number of raw values from which to derive feature attributes had been chosen arbitrarily to be three, five, and ten values in the study that developed 1-minute decision tree models,[9] corresponding to time intervals of three, five, and ten minutes, respectively.

63 62 59 56 55 67 74 54 54 53 55 55 56 57 57 58 60
* * * * * *

Figure 1. Sample annotated mean blood pressure values.

For this study, two different experiments were performed. In Experiment 1, the numbers of raw values with which to calculate feature attributes were chosen to be 180, 300, and 600, such that with 1-second granularity data, the identical quantities (in terms of time interval length) are calculated. In this way, previously developed (from Set A data) 1-minute artifact detection models can be directly compared to new artifact detection models developed from the 1-second data (Set B). The eight derived features, calculated for each of three time intervals and for each of the four signals, resulted in multi-signal feature vectors of size 96. These feature vectors comprised the inputs to machine learning programs; the output of these learning programs is a model that can classify previously unseen feature vectors as artifact or not. In Experiment 2, the numbers of raw values with which to calculate feature attributes were chosen to be three, five, and ten values to be consistent with the 1-minute models in terms of the *number* of values in each interval used to derive features. This corresponds to using time intervals of three, five, and ten seconds. The resulting 96-dimensional feature vectors were run through the 1-minute models.

Class labels were assigned to each feature vector of 96 values to facilitate supervised learning by decision tree induction[10] in Experiment 1, and testing of the built models in Experiments 1 and 2. Labels were assigned based upon the number of asterisks present (zero or one asterisk associated with each raw value) in the smallest time interval used in a given feature vector. The number of asterisks present in that time interval divided by the number of raw values present in the same interval gives the 'artifact average' value. Artifact average values are thus between 0 and 1, inclusive. For the blood pressure, carbon dioxide, and oxygen signals, all feature vectors were given a class label. Labels of 'artifact' were given to feature vectors with artifact average greater than 0.5, while labels of 'non-artifact' were given to feature vectors with artifact average less than or equal to 0.5. For the heart rate signal, the artifact class was given to only those feature vectors with an artifact average of 1, while the non-artifact class was given to those feature vectors with an artifact average of 0. Feature vectors with fractional artifact averages were not used to derive heart rate artifact models. These labeling methods were used in order to be consistent with the labeling used in the comparison study.[9]

Preprocessed data of 1-second granularity for Experiment 1 were split into training (70%), evaluation (9%), and test (21%) sets. Preprocessed data of 1-minute granularity had been similarly split. Each training set was used as input to c4.5,[10] a decision tree induction system. Varying performance of candidate models on the evaluation set helped to determine which candidate model would be chosen as each signal's final model. After experimentation, final models were implemented in the C language to facilitate testing of each model on reserved test data. Preprocessed 1-second data for Experiment 2 were also split into training, evaluation, and test sets; only the test set was used. One-minute models were run on both 1-minute and 1-second test data, while 1-second models were run on 1-second test data.

Performance metrics used for comparing different models include sensitivity, specificity, and area under the receiver operating characteristic (ROC) curve.[11] Sensitivity measures number of correct model-labeled artifact cases out of total number of actual artifact cases. Specificity measures number of correct model-labeled non-artifact cases out of total number of actual non-artifact cases. ROC curves were determined by first assigning to each tree leaf the probability of being an artifact for a set of cases that percolates to that point. These probabilities were based upon the ratio of artifacts to total cases in the training data that fell into each leaf. The threshold for considering a case to be artifact or non-artifact was then set at each leaf probability value. The resulting *(sensitivity, 1-specificity)* pairs were plotted to obtain the ROC curve. The area under each ROC curve was calculated by trapezoidal method.

RESULTS

Data were preprocessed as described in the Methods. Table 1 shows the breakdown of data cases by set and class label for each artifact signal type used in Experiment 1 (feature attributes made of identical time intervals). Table 2 shows the composition of the test sets by class, one for each artifact signal type, used in Experiment 2 (feature attributes made of identical numbers of values).

Figures 2 through 5 show decision trees developed from 1-second data. Class labels are represented by '1' (artifact) and '0' (non-artifact). Parentheses after a class label indicate the number of training cases that arrived at that node, shown where applicable as total number of training cases that arrived at that node followed by number of training cases incorrectly classified at that node. Attribute names are a concatenation of abbreviated signal name, abbreviated derived feature name, and number of values over which the derived feature was calculated.

Table 1. Breakdown of data cases by class label for each signal in Expt. 1 (180, 300, and 600 values in intervals used for derived feature calculation).

Signal	Class label	Training set	Evaluation set	Test set
bp	Non-artifact	156,769	20,027	47,287
	Artifact	441	50	132
	Total	157,210	20,077	47,419
co2	Non-artifact	153,083	19,557	46,178
	Artifact	4,127	520	1,241
	Total	157,210	20,077	47,419
o2	Non-artifact	151,646	19,362	45,723
	Artifact	5,564	715	1,696
	Total	157,210	20,077	47,419
hr	Non-artifact	153,862	19,656	46,419
	Artifact	474	76	139
	Total	154,336	19,732	46,558

Table 2. Breakdown of data cases in test sets by class label for each signal in Expt. 2 (3, 5, and 10 values in intervals used to derive features).

Signal	Class label	Test set
Blood pressure	Non-artifact	44,176
	Artifact	220
	Total	44,396
Carbon dioxide	Non-artifact	42,936
	Artifact	1,461
	Total	44,397
Oxygen	Non-artifact	42,545
	Artifact	1,852
	Total	44,397
Heart rate	Non-artifact	43,936
	Artifact	393
	Total	44,329

```
bp_med180 <= 0 : 1 (441.0)
bp_med180  > 0 : 0 (156769.0)
```

Figure 2. Blood pressure artifact detection decision tree model built from 1-second data.

```
co2_med180 <= 0.2 : 1 (4053.0/3.0)
co2_med180 > 0.2 :
| o2_low180 > 16 : 1 (30.0/3.0)
| o2_low180 <= 16 :
| | co2_avg180 > 0.4 : 0 (153040.0/9.0)
| | co2_avg180 <= 0.4 :
| | | hr_range180 <= 51 : 1 (40.0)
| | | hr_range180 > 51 : 0 (47.0/1.0)
```

Figure 3. Carbon dioxide artifact detection decision tree model built from 1-second data.

```
o2_med180 <= 0.5 : 1 (4725.0)
o2_med180 > 0.5 :
| o2_avg180 <= 15.7 :
| | o2_med180 <= 15.9 : 0 (150362.0/57.0)
| | o2_med180 > 15.9 :
| | | o2_std_dev180 <= 8.67 : 0 (638.0/61.0)
| | | o2_std_dev180 > 8.67 :
| | | | hr_avg600 <= 161.8 : 0 (100.0/6.0)
| | | | hr_avg600 > 161.8 :
| | | | | co2_std_dev300 <= 2.2 : 1 (190.0/5.0)
| | | | | co2_std_dev300 > 2.2 : 0 (80.0/17.0)
| o2_avg180 > 15.7 :
| | o2_med180 <= 20 : 0 (601.0)
| | o2_med180 > 20 : 1 (514.0/1.0)
```

Figure 4. Oxygen artifact detection decision tree model built from 1-second data.

```
hr_high180 <= 0 : 1 (474.0)
hr_high180 > 0 : 0 (153862.0)
```

Figure 5. Heart rate artifact detection decision tree model built from 1-second data.

Final decision trees developed from 1-minute data are reproduced in Figures 6 through 9 for convenience in comparing models. Table 3 shows results for each model run on test set data of its same granularity and for 1-minute models run on the two different types of preprocessed 1-second test sets.

DISCUSSION

Previous work showed that multi-signal detection of ICU artifacts by decision trees built from 1-minute data do well (ROC areas ranging from 89.41 to 99.93%).[9] In the current study, we found that decision trees built from 1-second data also perform effectively, even more so in fact (ROC areas ranging from 99.40 to 100.00%). Admittedly, 1-second data by nature contain more information than 1-minute

data. Surprisingly, however, models built from 1-minute data performed extremely well on test sets derived from 1-second data. Recall that our a priori assumption was that a model built from 1-minute data would not perform well when run on 1-second data because it would miss the short-lived artifacts found in the 1-second data; we have found that this is clearly not the case. ROC areas for Experiment 1 ranged from 96.30 to 100.00%, while ROC areas for Experiment 2 ranged from 99.70 to 100.00%. For three of the models (bp, co2, hr), 1-minute models run on 1-second data in both Experiments 1 and 2 performed better than the same 1-minute models run on 1-minute data. The 1-minute oxygen model run on 1-second data in Experiment 1 (ROC area 98.81%) performed well but not better than the 1-minute model run on 1-minute data (ROC area 99.93%). The results of running the 1-minute model on 1-second data in Experiment 2 (ROC area 99.92%) were equal to the results of the 1-minute model run on 1-minute data (ROC area 99.93%).

```
bp_med3 <= 4 : 1 (114.0/3.0)
bp_med3 > 4 :
| bp_range3 <= 7 : 0 (10959.0/72.5)
| bp_range3 > 7 :
| | bp_med10 > 46 : 0 (126.0/23.7)
| | bp_med10 <= 46 :
| | | bp_std_dev3 <= 5.51 : 0 (78.0/28.5)
| | | bp_std_dev3 > 5.51 :
| | | | co2_low10 <= 5.3 : 1 (46.0/10.1)
| | | | co2_low10 > 5.3 :
| | | | | hr_high5 <= 157 : 0 (27.0/12.8)
| | | | | hr_high5 > 157 : 1 (21.0/8.2)
```

Figure 6. Blood pressure artifact detection decision tree model built from 1-minute data.

```
co2_med5 <= 0.7 : 1 (207.0)
co2_med5 > 0.7 :
| o2_high3 > 14 : 1 (166.0/34.0)
| o2_high3 <= 14 :
| | co2_range3 <= 0.6 : 0 (10686.0/102.0)
| | co2_range3 > 0.6 :
| | | co2_low5 <= 4.5 : 1 (117.0/22.0)
| | | co2_low5 > 4.5 :
| | | | co2_slope3 <= 0.5 : 0 (150.0/26.0)
| | | | co2_slope3 > 0.5 : 1 (45.0/17.0)
```

Figure 7. Carbon dioxide artifact detection decision tree model built from 1-minute data.

These findings can be exploited: in situations in which an event develops slowly over several minutes, e.g., in some cases of pneumothorax, pneumothorax detection models could be developed with 1-minute data and then run on 1-second data processed using the same *time intervals*, as done in Experiment 1. On the other hand, in situations in which an event is fleeting, such as a short-lived false alarm lasting only seconds, models for false alarm detection could be developed with 1-minute data and then run on 1-second data processed using the same *numbers of values*, as done in Experiment 2. We found our a priori assumption—that 1-minute models would perform poorly on 1-second data—to be incorrect.

```
o2_med3 > 20 : 1 (90.0/1.6)
o2_med3 <= 20 :
| o2_low3 > 0 : 0 (11026.0/1.6)
| o2_low3 <= 0 :
| | o2_med5 <= 0 : 0 (154.0/5.4)
| | o2_med5 > 0 :
| | | o2_med3 <= 0.5 : 1 (79.0/1.6)
| | | o2_med3 > 0.5 : 0 (22.0/6.3)
```

Figure 8. Oxygen artifact detection decision tree model built from 1-minute data.

```
hr_low3 <= 113 :
| hr_range5 > 78 : 1 (180.0/3.0)
| hr_range5 <= 78 :
| | hr_low10 <= 30 : 1 (60.0/1.0)
| | hr_low10 > 30 :
| | | hr_med5 <= 121 : 0 (145.0/21.0)
| | | hr_med5 > 121 :
| | | | o2_low10 > 6 : 1 (41.0)
| | | | o2_low10 <= 6 :
| | | | | hr_range3 <= 38 : 0 (30.0/10.0)
| | | | | hr_range3 > 38 : 1 (37.0/8.0)
hr_low3 > 113 :
| hr_std_dev3 <= 8.14 : 0 (10565.0/66.0)
| hr_std_dev3 > 8.14 :
| | hr_range3 > 36 : 1 (41.0/6.0)
| | hr_range3 <= 36 :
| | | hr_low5 > 129 : 0 (160.0/28.0)
| | | hr_low5 <= 129 :
| | | | o2_low10 <= 4 : 0 (35.0/6.0)
| | | | o2_low10 > 4 :
| | | | | bp_abs_slope10 <= 0.21 : 1 (38.0/5.0)
| | | | | bp_abs_slope10 > 0.21 : 0 (37.0/15.0)
```

Figure 9. Heart rate artifact detection decision tree model built from 1-minute data.

A comparison of the decision tree models themselves is also interesting. For blood pressure artifact detection, both models found the median of three minutes of blood pressure raw values ('bp_med3' in 1-minute model and 'bp_med180' in 1-second model) to be a useful first predictor of artifact status. For detection of carbon dioxide artifacts, both models also found median value a useful first predictor, though the 1-minute model calculated median over 5-minute time intervals ('co2_med5'), while the 1-second model calculated it over 3-minute time intervals ('co2_med180'). In the oxygen artifact detection models, not only was the median over three

minutes present in each model, but both models used two identical threshold values for labeling a feature vector as artifact ('o2_med3 > 20' and 'o2_med3 <= 0.5' in the 1-minute model, 'o2_med180 > 20' and 'o2_med180 <= 0.5' in the 1-second model). The presence of identical attributes and thresholds is further reassurance that effective model development does not depend on use of one specific granularity of data. Usefulness of the median value in this domain is consistent with findings by Makivirta.[12]

Table 3. ROC curve areas for each model run on the test set of its own granularity, and for the 1-min models run on two different types of 1-sec test sets (Expt. 1 used 180, 300, and 600 values for feature derivation; Expt. 2 used 3, 5, and 10 values).

Signal	1-min model run on 1-min test set	1-sec model run on 1-sec test set	1-min model run on 1-sec test set (Expt 1)	1-min model run on 1-sec test set (Expt 2)
bp	89.41%	100.00%	100.00%	100.00%
co2	93.29%	99.70%	99.25%	99.70%
o2	99.93%	99.40%	98.81%	99.92%
hr	92.83%	100.00%	96.30%	99.95%

Our results indicate that while artifact detection models developed from 1-second data are effective when run on 1-second data, so too are models developed from 1-minute data effective when tested on 1-second data. This is a very important finding since developing models with 1-minute data has a tremendous advantage: during model development, more hours of ICU monitor data can be processed in less time. This is useful not only in general, but especially for data intensive domains such as the ICU. Because of the relative scarcity of artifacts, scattered sparsely amongst all the 'normal' values, voluminous amounts of physiological data streams need to be examined to ensure robust model development. The 1-minute models required processing of approximately 48,000 raw data values (200 hours multiplied by 60 minutes per hour multiplied by 4 data signals), while the 1-second models required processing of approximately 1,065,600 raw data values (74 hours multiplied by 3600 seconds per hour multiplied by 4 data signals). Thus, developing models from 1-minute data required roughly two orders of magnitude fewer calculations to process more than 2.5 times the number of monitor-hours. Moreover, these 1-minute models still performed well 'in the clinical setting' scenario, i.e., on 1-second monitor data. Data compression of temporal data by arithmetic mean, therefore, can be an effective method for decreasing knowledge discovery processing time without compromising learning. Future studies should focus on validating these techniques in other domains.

Acknowledgments

The authors would like to thank Peter Badger for assistance with data collection, Jon Doyle for support, and Peter Szolovits for advice. This work was supported in part by the AAUW Educational Foundation and DARPA contract F30602-99-1-0509.

References

[1]E.M.J. Koski, A. Makivirta, T. Sukuvaara, et al., Frequency and reliability of alarms in the monitoring of cardiac post-operative patients, Int J Clin Monit Comput 7 (1990) 129-33.

[2]S.T. Lawless, Crying wolf: false alarms in a pediatric intensive care unit, Crit Care Med 22 (1994) 981-5.

[3]C.L. Tsien, J.C. Fackler, Poor prognosis for existing monitors in the intensive care unit, Crit Care Med 25 (1997) 614-9.

[4]J.W.R. McIntyre, L.M. Stanford, Ergonomics and anaesthesia: auditory alarm signals in the operating room, in: R. Droh, W. Erdmann, R. Spintge, eds., Anaesthesia: Innovation in Management (Springer-Verlag, New York, 1985) 81-6.

[5]C. Meredith and J. Edworthy, Are there too many alarms in the intensive care unit? An overview of the problems, J Adv Nurs 21 (1995) 15-20.

[6]A. Meyer-Falcke, R. Rack, F. Eichwede, *et al.*, How noisy are anaesthesia and intensive care medicine? Quantification of the patients' stress, European Journal of Anaesthesiology 11 (1994) 407-11.

[7]C.A. Sara and H.J. Wark, Disconnection: an appraisal, Anaesth Intensive Care 14 (1986) 448-452.

[8]C.L. Tsien, TrendFinder: automated detection of alarmable trends. Lab for Computer Science TR 809, Massachusetts Institute of Technology, July, 2000.

[9]C.L. Tsien, I.S. Kohane, N.McIntosh, Multiple signal integration by decision tree induction to detect artifacts in the neonatal intensive care unit, Artificial Intelligence in Medicine 19 (2000) 189-202.

[10]J.R. Quinlan, C4.5 Programs for machine learning (Morgan Kaufmann Publishers, San Mateo, 1993).

[11]J.A. Hanley and B.J. McNeil, The meaning and use of the area under a receiver operating characteristic (ROC) curve, Radiology 143 (1982) 29-36.

[12]A. Makivirta, E. Koski, A. Kari, T. Sukuvaara, The median filter as a preprocessor for a patient monitor limit alarm system in intensive care, Comput Methods Programs Biomed 34 (1991) 139-44.