LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-TR-700884

Optimizing the Performance of Radionuclide Identification Software in the Hunt for Nuclear Security Threats

K. A. Fotion

August 18, 2016

Disclaimer

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Optimizing the Performance of Radionuclide Identification Software in the Hunt for Nuclear Security Threats

Katherine Fotion

Hosting Site: Lawrence Livermore National Laboratory

Mentor: Simon Labov

Abstract. The Radionuclide Analysis Kit (RNAK), my team's most recent nuclide identification software, is entering the testing phase. A question arises: will removing rare nuclides from the software's library improve its overall performance? An affirmative response indicates fundamental errors in the software's framework, while a negative response confirms the effectiveness of the software's key machine learning algorithms. After thorough testing, I found that the performance of RNAK cannot be improved with the library choice effect, thus verifying the effectiveness of RNAK's algorithms—multiple linear regression, Bayesian network using the Viterbi algorithm, and branch and bound search.

1. Internship Project

1.1. Introduction to the team

My first day working at Lawrence Livermore National Laboratory in the Global Security division began with a broad introduction to the purpose and impact of the team I would be working on for the next nine weeks. Within my first few hours, I sat in on a meeting for a project called ERNIE, or Enhanced Radiological Nuclear Inspection and Evaluation. ERNIE is a semi-truck radiation inspection apparatus currently stationed at the Tacoma port.

Figure 1. A semi-truck driving through ERNIE after loading cargo from a ship in the Tacoma port [2].

Ultimately, ERNIE's purpose is not to simply alarm upon encountering radiation, like many common technologies today, but rather to give an actual identification of the radioactive nuclide(s) found. The reason this feature is crucial is due to the fact that radiation is abundant in the environment, particularly in shipped goods. Items such as bananas, cat litter, and medical supplies can emit astonishing levels of radiation, particularly in shipping-grade quantities. These harmless materials have signatures, however, that are easily identifiable by nuclide type. Providing nuclide identification will allow officers to distinguish between a textbook banana shipment and a potential nuclear bomb threat, despite the fact that the intensity of the two signals may be identical.

As of now, ERNIE is merely collecting data in silence to help improve the current software; but eventually, the device will become an active staple of the Washington port, helping officers decrease the number of false positives without the concern of increasing the amount of false negatives. Though ERNIE does not rely on the same software that I have spent my internship dealing with, it does, in fact, share some of the same code base and was a suitable introduction into the reason my team must develop reliable, efficient and effective software.

1.2. Project overview

The Radionuclide Analysis Kit (RNAK) is pivotal software that my team has recently developed. Similar to ERNIE, the objective of RNAK is to correctly identify the various nuclides present in a given signal. The software is currently under test and constantly undergoing improvements. One of the tests my team planned to run, but temporarily shelved, was the evaluation of the *library choice effect*. My arrival at LLNL allowed the team to obtain these particular results significantly sooner than expected.

RNAK references a library for nuclide recognition. The library is essentially a list that defines all the nuclides that the software knows. The library is extensive, including even the extremely rare nuclides that will likely never be encountered in the real world. Due to the revolutionary software in the backbone of RNAK, this exhaustive list should not slow down or harm overall performance. A process called *downselect* has been designed to expertly sort through the possibilities and identify the correct nuclides in record time with satisfactory precision and recall.

The library choice effect will test the effectiveness of downselect by removing sets of rare nuclides from RNAK's library and measuring the software's performance. A result that indicates any improvement in the overall performance of RNAK with missing libraries implies an issue with downselect due to the fact that downselect was designed to counteract the burden of a large but thorough library. The alternative result will verify the efficacy of downselect as it applies to the overall performance of RNAK.

The following graphic demonstrates the process required by the assignment:

Figure 2. The library choice effect test divided into 3 steps.

As seen above, the first step was to generate a set of test spectra of the same relative difficulty to RNAK, a task that was much more involved than expected.

1.3. Generating a fitting test set

No test is operational until the perfect test set has been created. Though generating a test set is typically one of the final steps in my experience, it was suggested that I begin with this undertaking to become more familiar with RNAK and radiation spectra in general.

Considering that I was unacquainted with radiation detection coming into this internship, I gained countless takeaways simply from this portion of the project. One of the most crucial was a basic understanding of spectra. On the right is a sample Cs137 spectrum, one of the more common nuclides. The x-axis consists of bins representing energy channels, while the y-axis is the number of counts (i.e. the number of gamma rays) measured by a detector on a logarithmic scale. The peak that clearly protrudes above the background signal at the 220 energy level (corresponding to 660 keV) is the signature of Cs137. Many other nuclides have more than one peak, making Cs137 a simple and useful example to study.

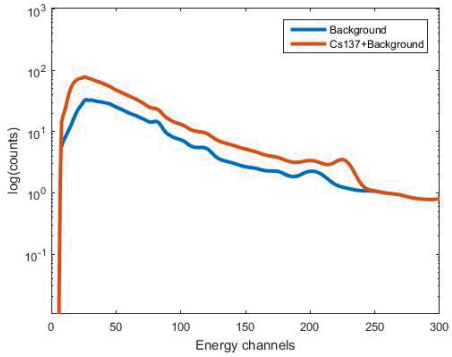

Figure 3. Sample Cs137 spectrum.

The spectrum seen above, however, is an easy problem for RNAK to solve. Cs137 is clearly identifiable and when in the above form is not an interesting test spectrum for the purpose of the library choice effect. A fitting test set must consist exclusively of difficult, but not impossible, spectra to get a better idea of how RNAK performs on suboptimal data.

1.3.1. Defining difficulty

I was introduced to the concept of signal-to-noise ratio (SNR) and weighted SNR when confronting the manipulation of spectral difficulty. The construction of difficult problems was necessary to build an ideal test set. I began by using the following formula for weighted SNR as a measure of difficulty, where a high weighted SNR indicates an easier problem.

$$SNR_{weighted} = \sqrt{\sum_i \frac{{S_i}^2}{B_i}} \tag{1}$$

Note that i in equation 1 iterates over the energy channels. Originally, the goal was to implement a binary search to find the weighted SNR of a particular nuclide that would guarantee a recall close to 0.75, thus isolating "difficult" but not too challenging of problems. This technique soon proved useless due to a weak correlation between RNAK recall and weighted SNR. Since the weighted SNR is a sum over all energy channels, it does not favor the peak regions. As a result, decreasing the weighted SNR did not necessarily decrease the recall of RNAK. A different approach was needed to properly create difficult spectra.

The next attempt to manipulate difficulty was through a fixed peak count value. The same binary search was implemented to find the conditions for a near 0.75 recall. This time, a stronger correlation was observed between fixed peak count and RNAK recall, but other problems presented themselves for nuclides more complicated than Cs137.

The final attempt proved successful when implementing a binary search on weighted SNR at peaks only. Below is the same Cs137 sample from figure 3 on the left and the adjusted, Poisson drawn spectrum on the right. For clarity, figure 5 is an example of how adjusting the weighted SNR based on the peak value can make a spectrum *easier* for RNAK to identify, although in the algorithm itself the goal is to make the spectra more challenging.

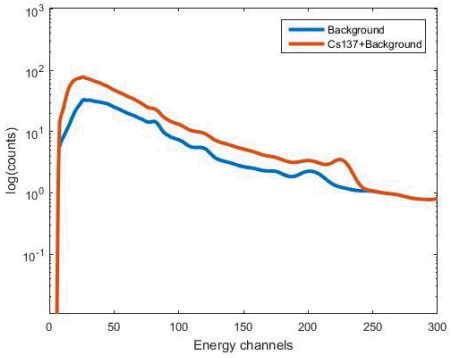

Figure 4. Original, non-Poisson Cs137 spectrum with background.

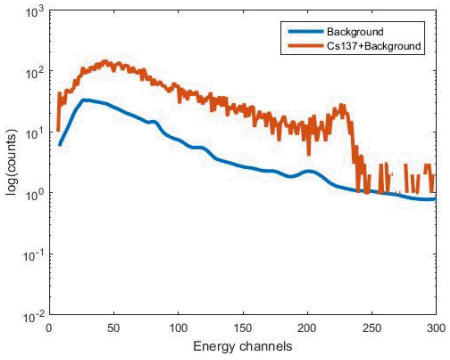

Figure 5. Randomized Cs137 spectrum with fixed weighted SNR at peak.

An optimal peak weighted SNR value was found for each nuclide and a massive test set of 100 samples for each of the 82 nuclides was generated. Such large processing power was possible due to Livermore Computing, an invaluable tool offered at the lab.

1.4. Creating a testing environment

Once the algorithms to generate a fitting test set were developed, it was time to develop the framework to run the test. All scripts were written in Matlab, relying heavily on preexisting and added features embedded in RNAK's Java-based software. The flowchart below is a breakdown of the testing process:

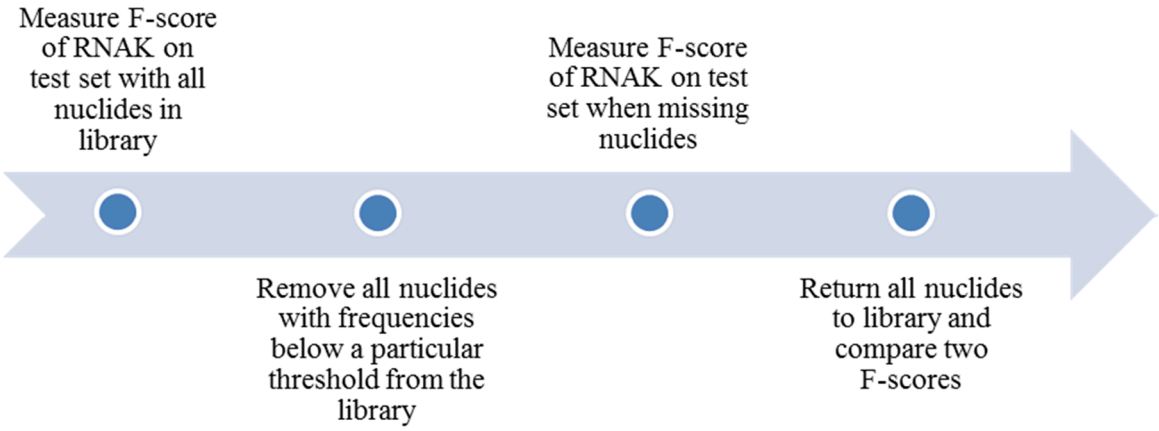

Figure 6. The series of steps involved in testing.

The library that is mentioned is an xml file that RNAK references when evaluating each spectrum. Removing nuclides from the library, therefore, involves tapping into this file and commenting out the unnecessary nuclides.

Choosing which nuclides to remove was determined by their frequency, which was calculated by values found in a spreadsheet compiled by the CBP Laboratory Scientific Services Teleforensics Center (LSS-TC). These weights were then written to a separate xml file and referenced by the testing framework when determining which nuclides to strike from the library at a given test.

1.4.1. Measuring performance

The ultimate goal of this test was to determine whether or not removing rare nuclides from the software's library will improve the performance of RNAK. *Performance*, however, is a subjective term that needed to be defined in a more tangible way. Performance could mean run time, recall, precision, accuracy, or a variety of other measurements. A definitive decision had to be made before moving forward.

After thorough background research, I found that when dealing with this particular type of software, namely a multi-labelled machine learning classification algorithm [4], the weighted F-score is the most appropriate measurement of performance, utilizing both precision and recall values.

$$F - score = (1 + \beta^2)\frac{precision \cdot recall}{\beta^2 precision + recall}, \textit{ where} \quad (2)$$

$$precision = \frac{true\ positives}{true\ positives + false\ positives} \quad and \quad recall = \frac{true\ positives}{true\ positives + false\ negatives} \quad (3, 4)$$

As seen in equation 2, there is a parameter called β that must be assigned. This selection of value can significantly impact the results of the test as it determines whether to place more emphasis on the precision measurement or the recall measurement.

In an application such as ERNIE, the semi-truck radiation detection apparatus, one would have to weigh whether or not it is more important to decrease the amount of false positives at the expense of potentially increasing the number of false negatives. If one's only goal is to decrease the amount of times an operator has to search a truck that is actually clear, the group may want to place more emphasis on precision and select a β value of 2. On the other hand, if someone's priority is to make sure that absolutely no potentially harmful substances get past the machine, then more emphasis should be placed on recall and a β value of 0.5 would be more appropriate. If both measurements are to be considered of equal importance, a β value of 1 should be chosen and the F-score is effectively the harmonic mean of the two. Since the specific application of RNAK is undetermined at this point, we ultimately used the latter in order to avoid favoring any particular measurement.

The difference between a *weighted* F-score and the F-score seen in equation 2 above, made up of equations 3 and 4, is simply the fact that the precision is calculated differently. A weighted precision takes into account the weights associated with each test nuclide when calculating the true positives and false positives. Both the formula for recall and the selection of β remains the same. The purpose of using a weighted F-score is that it considers the likelihood of encountering each nuclide. The recognition, or lack thereof, of an extremely rare nuclide should not affect the overall score of the algorithm as severely as imprecise predictions of a common nuclide. Using a weighted F-score adds an element of practical use to the measurement.

1.5. Results

After running the test on the generated test set, I evaluated whether or not the F-score when removing nuclides was higher than the F-score with the original library. I discovered that with one particular removal of nuclides, namely removing all nuclides with a frequency < 0.000081, the precision of RNAK spiked larger than previous measures (but not higher than the original value). In addition, the recall had already decreased to such an extent that the overall weighted F-score still fell below the original value. With each removal of nuclides, moving from those with the smallest frequency to those with larger frequencies, the weighted F-score consistently decreased. Both the precision and recall gradually decreased with each removal, experiencing occasional spikes. None of these spikes, however, were significant enough to improve RNAK's performance. Below is a table of 2 particular removals: removing the group of nuclides with the smallest frequency and the nuclide removal that results in the highest spike in precision. The F-scores when $\beta = 0.5$, $\beta = 1$, and $\beta = 2$ are shown to demonstrate the significance of β selection.

Table 1. Results of testing ~100 samples of each 82 nuclides.

	$\beta = 0.5$	$\beta = 1$	$\beta = 2$
Original F-score	0.5719	0.6251	0.6893
F-score after removing rarest group of nuclides	0.5499	0.5882	0.6321
F-score after removing nuclides with freq < 0.000081	0.5403	0.5496	0.5593

It is clear that the F-score worsens in these two cases in comparison to the original. This trend followed for all nuclide removals, proving that strategically choosing sub-libraries cannot improve the overall performance of RNAK.

1.6. Discussion

As expected, the results verify the software's ability to successfully sort through the massive library and eliminate clearly improbable nuclides immediately, naturally narrowing down the possibilities to a more attainable set. This process, called downselect, allows the software to handle the large library without a

negative impact on performance. The test showed that synthetically altering the library cannot deliver better results and, thus, the large library does not impair the performance of the software.
There was a point when I believed the results pointed to the contrary. When weighting the precision measurement, I originally was using purely the values from the LSS data. For example, according to the data, there is a 0.068416779 probability of encountering Cs137 in the environment. I was using this form of number as the test nuclide weights. The problem, however, arose when testing certain nuclides with fewer than 100 samples. This was only the case for a handful of nuclides, but still made a significant impact. Eventually, I discovered I was supposed to multiply each probability of occurrence by the number of samples for that test nuclide, resulting in 6.8416779 for Cs137. This difference in weight value significantly increased the precision measurements and caused minor spikes to no longer have a significant advantage over the constantly decreasing recall. I believe the team sighed in relief when discovering that their algorithm was not flawed.

1.7. Future work

Although this test did not incite any changes to RNAK's software, a number of other tests still need to be initiated to verify the effectiveness of RNAK's algorithms. Members of the team will conduct these tests over the next several months before gaining the confidence to deploy the software in a device similar to ERNIE. The test I designed has been converted from Matlab to Java in order to be compatible with Livermore Computing's Aztec system in case there is need for a retest or adjustment to the test.

2. Impact of Internship on My Career

This internship has been a fascinating portal into the world of research. I attended several lectures series about very *real* topics, including the emergency response plans in place in case an anthrax aerosol was ever released in a crowded city, the steps being taken to assist TSA in maintaining safe airports and flights, and the first ever simulated beating heart on Livermore Computing's Sequoia machine. The depth and breadth of the lab's impact is infinite and it was truly special to feel a part of such noble goals.

The actual skills I gained were also endless. Before arriving at the laboratory, I barely even knew what caused radiation, let alone how to adjust radiation spectra, how to operate a parallel computing machine, or how to test a machine learning algorithm. I went on to be one of the winners at the student poster

symposium and had a successful summer filled with programing research. Although I had taken a course on neural networks, I had never experienced the bridge between nets and machine learning. I can now enter my first graduate school course titled "Pattern Analysis and Machine Intelligence" with more confidence, knowing that I have seen those types of algorithms in action. I have since then applied and interviewed for two machine learning positions for the fall semester, feeling poised and collected instead of confused and flustered when encountering technical questions. This internship has been the perfect transition from undergraduate to graduate level knowledge.

Although this may be out of reach for the Department of Homeland Security, an interesting research area that I believe students would significantly benefit from, both due to its content and its multidisciplinary collaboration, is machine learning for biomedical applications. The future holds so many positive advancements for medical technology, and I believe the lab and DHS would both benefit immensely from attempting to delve into this up-and-coming field.

3. Acknowledgments

I would like to extend a special thanks to all those who assisted me along the way, including Karl Nelson, Brandon Seilhan, Yiming Yao, Patrick Beck and my research mentor, Simon Labov.

4. References

[1] Enghauser, M. (2016). Algorithm improvement program: nuclide identification algorithm scoring criteria and scoring application. *Sandia National Laboratories and DNDO*.

[2] ERNIE [Photograph found in Tacoma]. (n.d.). In SPIA. Retrieved July 26, 2016, from http://cits.uga.edu/uploads/1540compass/1540images//RPM1.jpg

[3] Knoll, G. F. (1979). Radiation detection and measurement. New York: Wiley.

[4] Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management, 45*(4), 427-437. doi:10.1016/j.ipm.2009.03.002