CNN Algorithms for Standoff Detection of Trace Explosives

Eric Yao

Winston Churchill High School

Maryland, U.S.A

ericyao0815@gmail.com

Chris Kendziora

Code 6365

US Naval Research Laboratory

Washington, DC, 20375 U.S.A

chris.kendziora@nrl.navy.mil

Abstract

As machine learning and artificial intelligence technologies continue to mature, these powerful computational tools have the potential to advance other areas of active research, including material sciences. Classifier algorithms able to detect trace explosives and dangerous substances through the use of computers - without human involvement - are of high relevance to the Navy. This paper proposes a convolutional neural network (CNN) based method to classify 55 different chemical substances (called analytes) on surfaces by active infrared (IR) backscatter imaging spectroscopy (IBIS) technology. We first analyzed the data by plotting the diffuse reflectance signature as a function of wavelength, with the intention to gain insight on how the analytes are affected by certain factors. We then created a convolutional neural network deep learning algorithm able to classify these analytes with a high accuracy when given their features. Through our research, we demonstrated that machine learning algorithms can still perform very well despite factors such as gaussian noise, very little analyte mass, and disruptions caused by the substrate (background material). This is partly due to the nature of CNNs, but numerous other procedures that enhance model performance were also used.

Keywords and Phrases: Convolutional neural network, Deep learning, Standoff explosives detection

1. Introduction

Machine learning is becoming increasingly prevalent in many fields to aid us on our quest for scientific discovery and technological advancement. The remote detection of trace explosives and other dangerous chemical substances at standoff distances is a vital topic of ongoing research and algorithm development. The objective of the present work is to detect and identify potential threats, including explosives and drugs of abuse. Because the target surfaces may be disrupted by contamination transferred by fingerprints, trace standoff

detection may be important to identify those associated with bomb making, narcotic synthesis such as fentanyl, or transportation of illegal drugs, and in forensic applications. OurIBIS technical approach implements active illumination for capturing the diffuse IR reflectance [1]-[6] signatures of particles deposited onto surfaces.

Our goal is to create and test a machine learning algorithm that could perform multi-class classification on the 55-analyte dataset. These analytes are represented in simulated diffuse reflectance spectra, each of which consists of 1,701 features and has varying amounts of mass and noise. The spectra are meant to test the capabilities and limits of machine learning. More specifically, this simulated dataset (consisting of 49,500 instances) is built with the intention to explore machine learning algorithms, in terms of behavior when fed noisy data, identification of analytes with tiny amounts of mass, and response to interference caused by the substrate. The construction of this dataset and the capturing of the reflectance spectra are based on the IBIS technology [7].

2. Proposed Method

Our multiclass classification algorithm lies at the intersection of machine learning and material sciences. In order to gain a deeper understanding of the problem and plan the construction of the algorithm, it is imperative to simulate the diffuse reflectance spectra of particles on substrates.

3. Simulation of Infrared Diffuse Reflectance Spectra

The classification algorithm learns to distinguish between analytes by associating certain features with certain analytes, and thus it is important to understand the differing features. Using the Plotly library, we generate the following two plots. The reflectance signature of dibenzoyl peroxide when it is deposited as particles on glass (Fig. 1) is compared to the reflectance signature of heroin deposited on glass (Fig. 2). The goal of these visualizations is to gain insight on these distinguishing features such that they can be accentuated and weighted in the model.

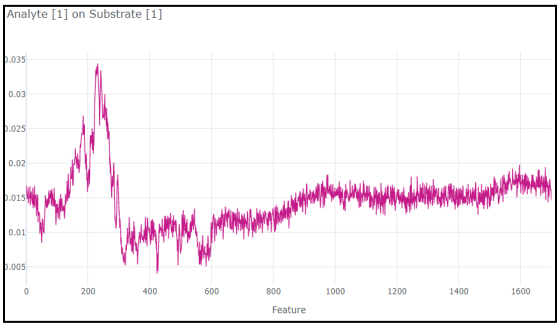

Figure 1: Analyte 1 (dibenzoyl peroxide) on substrate 1 (soda-lime glass), 0.1% noise

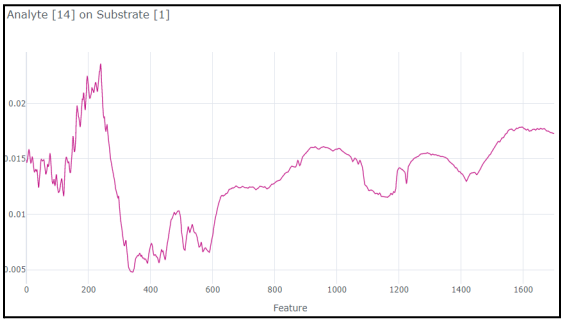

Figure 2: Analyte 14 (heroin) on substrate 1 (soda-lime glass), no noise

It becomes evident after analyzing the two plots that the large peak around feature 250 is caused by the substrate spectra, as there is much more background material than the substance itself, which explains the high values. This is thus the "background" that must be handled by the model. However, the sharp dip around feature 200 present in Fig. 1 but not in Fig. 2 suggests that feature 200 and those around it are distinguishing features of analyte 1. Similarly, between features 1000 and 1400 there is a series of winding curves present in Fig. 2 but not in Fig. 1, suggesting that those are distinguishing features of analyte 14. By choosing the correct machine learning model and tuning its hyperparameters accordingly, the algorithm will be able to pick up on these features and classify the analytes accurately.

Lastly, it can be observed that the plot in Fig. 1 is more erratic due to the added noise. It is therefore important to choose the correct hyperparameters for the model such that it does not mistake the noise for relevant features. The model should be precise enough to capture fine details without being confused by noise, and complex enough to capture the larger patterns such as the distinguishing features of analyte 14.

4. Convolutional Neural Network Algorithm

To implement the ideas discussed previously, we created a 1D convolution model that receives one instance of the data as input, in the shape (1701,1,1). During each convolution, filters of varying sizes are slid across the data in order to capture features of different sizes.

Based on the exploratory data analysis, we chose filter sizes of 10, 100, and 300. Through hyperparameter testing on validation data, we found that a filter size of 10 was optimal for capturing small features. If it were any smaller, it would pick up noise as features. The 100 and 300 sized filters were chosen to capture broader patterns and the general shape, or derivative information, of the data. We chose 3 convolutional layers, so that each layer may build off of the patterns found in the previous layer and recognize more complex patterns. Lastly, since there are more combinations of patterns to recognize as they get more complex, the number of filters increases in each layer accordingly (64 in layer 1, 128 in layer 2, 256 in layer 3). After performing many experiments tweaking these hyperparameters, this combination achieved the highest performance.

After the feature extraction of the convolutional layers is complete, these distinguishing features must now be flattened and passed into a multi-layer perceptron (MLP) to be classified as a certain analyte. The MLP used in this project consisted of

three fully-connected (dense) layers, with 300, 150, and 55 neurons respectively. The number of neurons in the first fully-connected layer must match the number of input features, which worked out to be 300 after the convolution layers. The number of neurons in the last layer must match the number of labels in multi-class classification problems. The number of neurons in the hidden layer is generally set to a number in between the input and output layers.

One problem the CNN faced is that it predicted the same label for all 12,375 testing instances. This is due to the vanishing gradients problem, where any neuron with a negative value would just be set to 0 due to the nature of the ReLU activation function (saturates for values less than 0). This is solved by replacing it with the LeakyReLU activation function, which has a slight gradient for values less than 0 and thus does not saturate (Fig. 3).

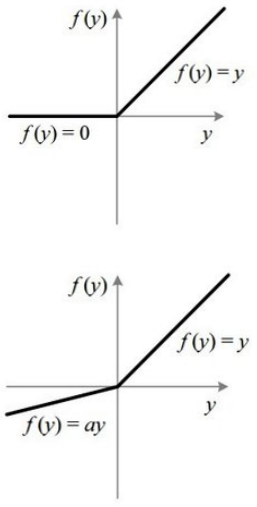

Figure 3: The ReLU activation function (top) is defined as f(x) = max(0,x) and saturates on the left side. The LeakyReLU activation function (bottom) is defined as f(x) = max(α*x,x) where the left side has a slope or "leak" with magnitude of α

In the final layer of the model is the softmax activation function, which converts its input neurons into a list of 55 probabilities all adding up to 1, representing the algorithm's confidence score for each class. When the model makes a prediction for a certain instance, it is set to take the highest probability from that list.

5. Experimental Results

5.1 Metrics

The primary metrics used to evaluate the performance of the CNN model were the F1-score and confusion matrix. After the model makes its final predictions on the test set, the F1-score is calculated for each of the 55 labels and averaged to produce a metric that takes into account both precision and recall. The maximum score is 1.0. In particular, let TP denote the number of true positives, FN denote the number of false negatives, and FP denote the number of false positives:

Recall: $R = \frac{TP}{TP+FN}$

Precision: $P = \frac{TP}{TP+FP}$

F1-score $= \frac{2 \cdot R \cdot P}{R+P}$

The second performance metric is a visual representation of precision and recall, which is the confusion matrix. The confusion matrix compares the model's predicted labels to the true labels, where a perfect model would have everything along the main diagonal Anything off-diagonal is a wrong prediction made by the model, as shown in Fig. 4.

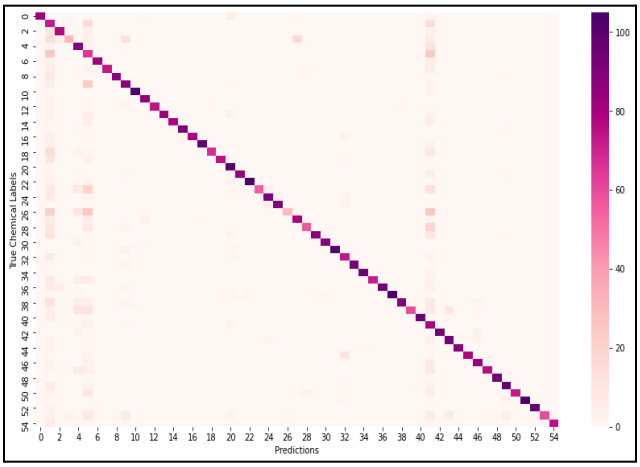

Figure 4: The confusion matrix comparing the CNN's predicted labels to the true chemical labels. The confusion matrix itself is a 55x55 array of numbers, and here it is represented as a heatmap with darker values representing higher accuracy.

It can be seen from the vertical stripes that the model tends to mislabel analytes as 1, 5, and 41. This could be because their reflectance signatures are similar to those of the substrate spectra, which, when the mass loading is low, dominates the combined spectra.

5.2 Optimization Strategy

Even after tweaking the hyperparameters in the CNN model, the model's F1-score is still far from optimized, at 74%. Thus, it is necessary to look into the model's learning process, which is governed by the stochastic gradient descent (SGD) optimizer. SGD is the mathematical process involved in backpropagation which makes the model learn. In short, it calculates the cost function of the model's predictions (how far off the predictions are), and then tries to minimize it by finding the global minimum through taking small steps down the steepest gradient.

SGD can be improved by tweaking the decay parameter. Our model has the SGD decay set to 0.0001, meaning that after each epoch (training iteration) the SGD takes smaller and smaller steps, thus reducing the chance that it will overshoot the global minimum.

Another optimization technique used in the CNN is batch normalization. As the millions of parameters in this deep learning network get updated after every epoch, the vanishing gradients problem may come back during training and drastically slow down the learning process, making it seem as though the model is not learning. Batch normalization solves this by standardizing the data after every iteration, as opposed to only scaling the data once before training.

Lastly, in each of the dense layers of the MLP, the He kernel initialization [8] is used to initialize the weights in a normal distribution with controlled variance, as

opposed to completely randomly. This is proven to improve model performance and speed up training.

The two confusion matrices shown in Fig. 5 compare the model's performance before and after the optimizers were added. Judging by the difference between the general spottiness of the confusion matrices, it becomes clear that these optimization techniques have worked. Indeed, the model performance improved to 84%.

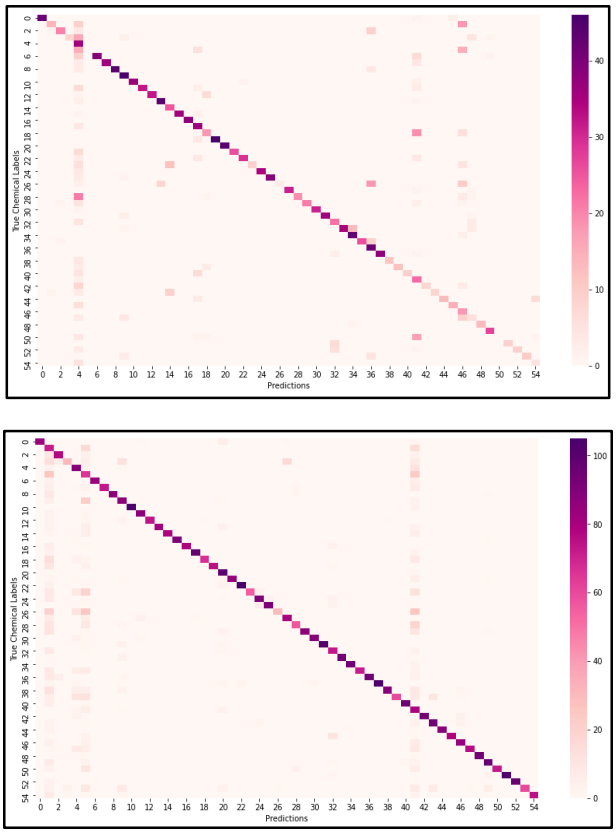

Figure 5: CNN's confusion matrix without any optimization techniques (left) and CNN's confusion matrix with SGD optimization and batch normalization (right)

6. Conclusion

To fully implement our IBIS technology for the standoff detection of trace chemicals on surfaces, we have developed a convolutional neural network that takes in a combination of analyte and substrate spectra as input, and returns the predicted label of the analyte as output. We demonstrated that machine learning algorithms can overcome many challenges posed by our large dataset (noisy data, small mass loadings), through the tweaking of hyperparameters (filter size, number of filters, number of convolutional layers). We also implemented several optimization techniques (SGD decay, batch normalization, kernel initialization) to further improve model performance.

In the future this work will bolster technological advancements in AI, allowing computers to outperform humans while keeping people safe in the process. Beyond the applications in material science and chemical detection, machine learning has other broader implications for the Navy. This includes autonomous warships capable of carrying out advanced warfare tactics, and unmanned submarines capable of conducting extensive undersea research.

ACKNOWLEDGEMENTS

The authors thank the Naval Research Laboratory Code 6365 for providing the large scale datasets. This work is supported in part by the Department of Defense (DoD) grant, #W911NF18104

REFERENCES

[1] Christopher J. Breshike, Christopher A. Kendziora, Robert Furstenberg, Tyler J. Huffman, Viet K. Nguyen, Norman Budack, Yohan Yoon, R. Andrew McGill. 2020. Hyperspectral imaging using active infrared backscatter spectroscopy for detection of trace explosives, Optical Engineering 59, 9, 092009-1. https://doi.org/10.1117/1.OE.59.9.092009

[2] Christopher A. Kendziora, Christopher J. Breshike, Yohan Yoon, Robert Furstenberg, Tyler J. Huffman and R. Andrew McGill, A system for rapid standoff detection of trace explosives by active infrared backscatter hyperspectral imaging, in Proc. SPIE 11392, Algorithms, Technologies, and Applications for Multispectral and Hyperspectral Imagery XXVI, 113920M, virtual, 2020. https://doi.org/10.1117/12.2558220

[3] Christopher J. Breshike, Christopher A. Kendziora, Robert Furstenberg, Yohan Yoon, T. J. Huffman, V.Nguyen and R. Andrew McGill, Rapid detection of infrared backscatter for standoff detection of trace explosives, in Proc. SPIE 11416, Chemical, Biological, Radiological, Nuclear, and Explosives (CBRNE) Sensing XXI, 114160W, Virtual, 2020. https://doi.org/10.1117/12.2558485

[4] Christopher J. Breshike, Christopher A. Kendziora, Yohan Yoon, Robert Furstenberg, Viet K. Nguyen and R. Andrew McGill, Infrared backscatter imaging spectroscopy for standoff detection of trace explosives, in Proc. SPIE 11010, Chemical, Biological, Radiological, Nuclear, and Explosives (CBRNE) Sensing XX, 1101004, Baltimore, 2019. doi: 10.1117/12.2518951.

[5] C. Breshike, Christopher A. Kendziora, Norman Budack, Yohan Yoon, Robert Furstenberg, Viet K. Nguyen and R. Andrew McGill, Active LWIR hyperspectral imaging and algorithms for rapid standoff trace chemical identification, in Proceedings of SPIE, vol. 10986, p. 109860K, 2019. https://doi.org/10.1117/12.2518720

[6] Christopher J. Breshike, Christopher A. Kendziora, Robert Furstenberg, Viet K. Nguyen, A. Kusterbeck and R. Andrew McGill, A system for rapid chemical identification based on infrared signatures, in Proceedings of SPIE, vol. 10639, p. 1063927, 2018. https://doi.org/10.1117/12.2304435

[7] Christopher A. Kendziora, Robert Furstenberg, Christopher J. Breshike, D. Finton, D. C. Kendziora, Tyler J. Huffman, R. Andrew McGill, Algorithms for identification of trace explosives by active infrared backscatter hyperspectral imaging, in: Proceedings of SPIE 11727, Algorithms, Technologies, and Applications for Multispectral and Hyperspectral Imaging, volume XXVII, 117270R, 2021. doi: 10.1117/12.2585982.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), volume XXVII, 117270R, 2021, pp. 1026-1034. doi: 10.1109/ICCV.2015.123.