LEADING EDGE

Summer 2021

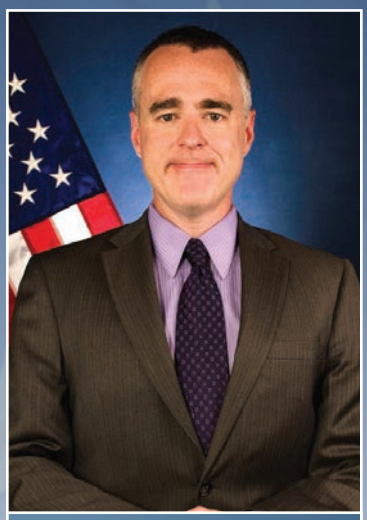
NAVAL SURFACE WARFARE CENTER, DAHLGREN DIVISION

NAVSEA WARFARE CENTERS DAHLGREN

NAVAL SURFACE WARFARE CENTER DAHLGREN DIVISION
DAHLGREN | DAM NECK

THE LEADER IN WARFARE SYSTEMS DEVELOPMENT & INTEGRATION

Capt. Casey Plew
Commanding Officer
Naval Surface Warfare Center
Dahlgren Division



Darren Barnes
Acting Technical Director
Naval Surface Warfare Center
Dahlgren Division

Table of Contents

The Rapid Rise of Neural Networks and Their Implication to Defense: A Cautionary Tale 04
By David A. Johannsen, Jeffrey L. Solka, and John T. Rigsby

Performance Numbers vs. Safety Numbers for Laser Hazard Evaluations 08
By Sheldon Zimmerman and Mary Zimmerman

Cybersecurity Analytics for Statisticians: A Case Study of Text Data 12
By David J. Marchette and Rakesh M. Verma

Naval Surface Warfare Center, Dahlgren Division (NSWCDD)
Capt. Casey Plew, *Commanding Officer*
Darren Barnes, *Acting Technical Director*
Alan Black, *Director, Corporate Communications*
Meghan Stoltzfus, *Branch Head, Congressional and Public Affairs*
Laura Driskell, *Layout Design & Graphic Artist*
David Johannsen, *Author*
David Marchette, *Author*
John Rigsby, *Author*
Jeffrey Solka, *Author*
Rakesh Verma, *Author*
Mary Zimmerman, *Author*
Sheldon Zimmerman, *Author*

The Leading Edge Magazine is a professional journal magazine of the Naval Surface Warfare Center Dahlgren Division (NSWCDD). This journal will draw upon the NSWCDD's rich legacy to instill a sense of pride and professionalism among community members and to enhance reader awareness of the relevance of naval surface warfare systems for our nation's defense. The opinions and assertions herein are the personal views of the authors and do not necessarily reflect the official views of the U.S. Government, the Department of Defense or the Department of the Navy.

Send feedback to:
Managing Editor, Leading Edge Magazine
Naval Surface Warfare Center Dahlgren Division
Corporate Communications Office, Code 103
6149 Welsh Road, Suite 213
Dahlgren, VA 22448
Email: NSWCDD.Info@navy.mil
Phone: 540-653-8152

Authorization
Leading Edge Magazine is published bi-annually from working capital funds by authority of the Naval Surface Warfare Center Dahlgren Division. Reproductions are encouraged with proper citation.

Introduction

Welcome back to the Leading Edge, the professional journal of Naval Surface Warfare Center Dahlgren Division. We are excited to announce the relaunch and redesign of this beloved publication! Leading Edge features in-depth articles and scholarly papers that provide a window into the groundbreaking work of our talented scientists, engineers, mathematicians and other experts here at Dahlgren. Going forward, expect biannual delivery every Spring and Fall.

This issue delves into the realm of neural networks, laser hazard evaluations and the evaluation of text data in cybersecurity. These technologies are poised to reshape the future of naval surface warfare. Members of our research and development community have contributed three insightful papers that explore what this means for our work at NSWCDD.

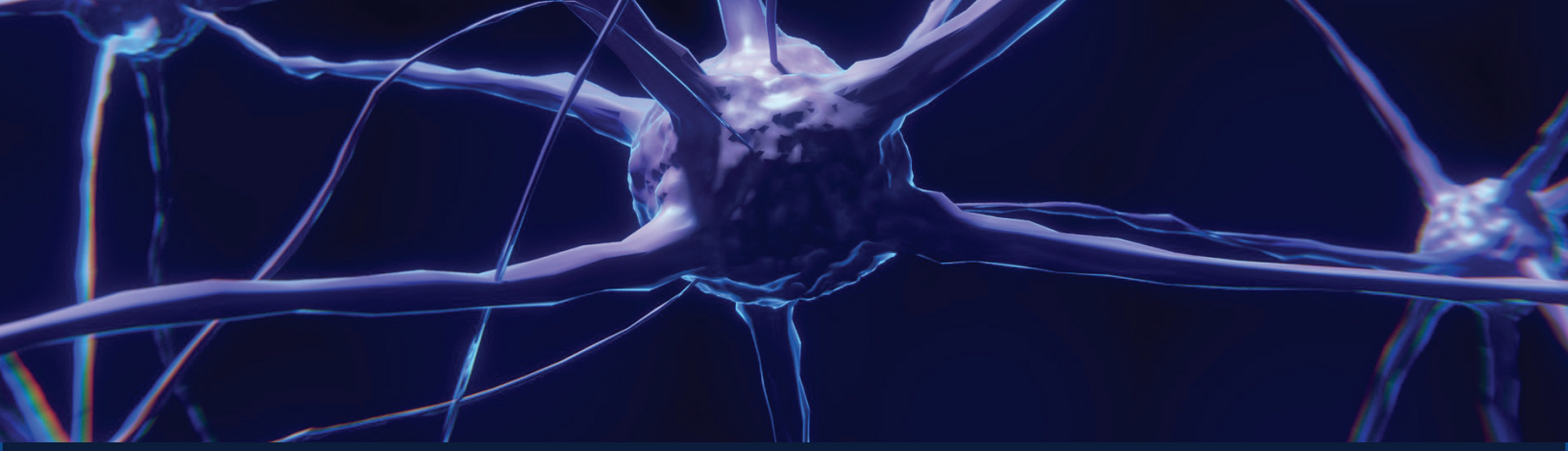
The first paper is a reality check. Authors David Johannsen, Jeffrey Solka, and John Rigsby share a cautionary tale about neural networks so that Navy leaders can be better informed about certain vulnerabilities and complications that this technology injects into our weapon systems. This paper reminds the reader of the dangers of overfitting, the black box nature of artificial neural systems, and their fragility while discussing some of the efforts at other organizations such as the Defense Advanced Research Projects Agency that are seeking to address these shortcomings.

In the second paper, Performance Numbers vs. Safety Numbers for Laser Hazard Evaluations, we learn how lasers are viewed by both laser manufacturers and laser safety professionals and what can be done to create a cohesive understanding and mutual betterment of laser safety calculations and measurements. This paper dives into the main laser assessment areas and how each calculation affects overall results.

The third paper, Cybersecurity Analytics for Statisticians: A Case Study of Text Data, discusses the technical defense developments for securing digital information against threats and attacks involving text data. The paper provides an overview of the areas vulnerable to these attacks and the statistical strategies and machine learning methods utilized to counter existing and future threats.

These thought-provoking articles are on the front lines of scientific research, development, test and evaluation at Naval Surface Warfare Center Dahlgren Division.

The Leader in Warfare Systems Development and Integration



The Rapid Rise of Neural Networks and Their Implication to Defense: A Cautionary Tale

By David A. Johannsen, Jeffrey L. Solka, and John T. Rigsby

Neural networks – a modern success story

Fueled by rapid increases in computer storage capacity and processing power (principally through the use of graphical processing units (GPUs)) and the widespread availability of powerful software for designing and implementing neural networks (such as Google's TensorFlow), the application of artificial neural networks to significant problems in the real-world has seen tremendous growth over the last several years. During this time, neural networks have demonstrated successes on problems as varied as automatic recognition of handwritten digits, automated image captioning and indexing, and have even beaten human masters in the game of Go. In fact, the field has reached the state of maturity that a person with only casual knowledge of computer programming can implement a neural network for whatever problem they might have at hand.

Given this climate of success, there is growing interest in fielding neural networks in Department of Defense (DoD) systems. In this brief note, we will discuss the nature of neural networks in a language which can be broadly understood, especially in the context of the unique environment within DoD, so that Navy leaders can be better informed about the strengths and limitations of this technology as it impinges ever more frequently on the DoD. We will first attempt to explain to non-specialists what an artificial neural network is. We will then discuss some of the inherent limitations of this class of machine learning tools and some of the ways that we and other members of the DoD are studying these limitations. Finally, we will discuss the consequences of these limitations.

What is a neural network?

Rather than giving a precise mathematical definition of a neural network, we will begin by giving a functionally oriented definition. Thus, we describe a neural network as a nonlinear function from the space of inputs to outputs. The particular function is chosen from a broad class of nonlinear functions through a process known as training. Often, in current practice, the choice of nonlinear function is underdetermined; that is, the function contains more parameters to be learned than the number of observations that one has at hand for training the algorithm.

For example, in the context of image captioning, the space of inputs is the collection of all possible digital images, and the output space is the collection of all meaningful captions. The neural network accepts a digital image as input, and produces a caption. Along the way, hidden from the end-user, the computer treats the image as a mathematical object, performs mathematical operations on it, and then produces a numerical output (which is often a vector of probabilities of membership in the various

classes). This vector of probabilities is then converted to a caption that is presented to the user.

We will now be a bit more precise in defining a neural network. The formulation of neural networks as a method of machine learning was motivated by analogy with the functioning of neurons in the human brain. Thus, neural networks consist of a set of nodes (neurons) with edges between them. Outputs of the nodes are multiplied by the weights associated with the edges and fed forward to the nodes in the next layer of the network. This information is adjusted by a threshold function associated with the node and then propagated through the neural network. In Figure 1 we present a 4 layered neural network with an input layer, two hidden layers, and an output layer. Following our discussions above, one might imagine that each node in the output layer provides the probability of the input belonging to one of three classes.

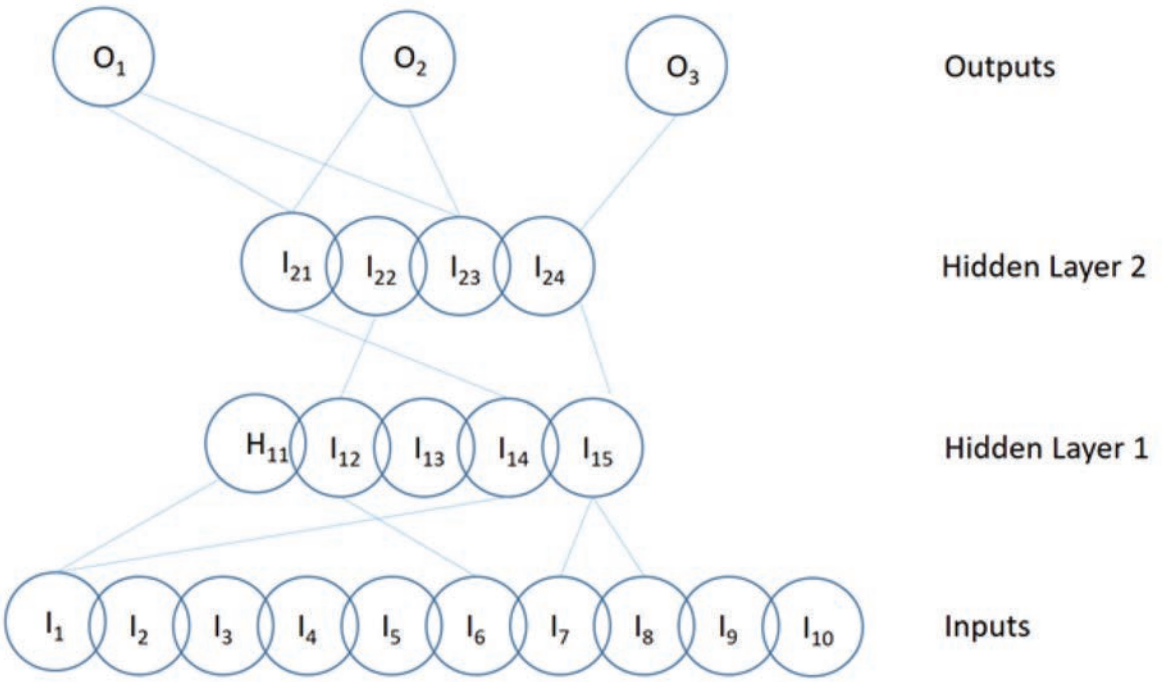


Figure 1. An example neural network with four layers.

A virtue is a vice

Statistical pattern recognition has historically involved a somewhat standard pipeline, see Figure 2.

The first three steps of this process (i.e., data collection, data processing/cleaning, and feature extraction) are often time consuming. If possible, a practicing statistician is well served to spend his time on the steps contained in the dotted box. The "extract features" and "dimension reduction" steps can be particularly daunting. The process of feature extraction and selection is usually the domain of subject matter experts (SMEs) and will often require significant time and experimentation to determine what features should be selected and have utility for the task at hand. Neural networks promise to revolutionize this pipeline by incorporating these two steps directly into the model building step without the need for SMEs. The virtue of artificial neural networks is that one can train a network to perform complex machine learning tasks such as interpolation, classification, regression, etc. without having to go through the process of feature generation and

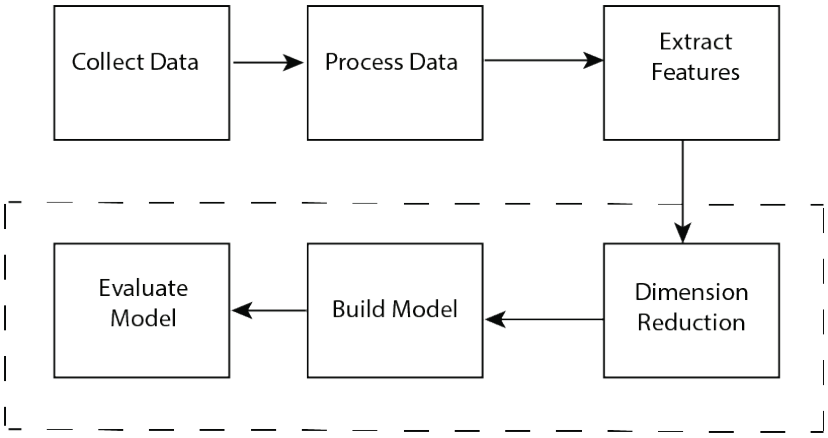


Figure 2. Pattern Recognition Pipeline.

dimensionality reduction. In our discussions below we will focus on the consequences of automating these two steps of the pattern recognition process.

Before getting too far along, we should note that in some settings it may be impossible to generate features specifically tailored for each of the possible classes. For example, in automatic image captioning, as the image may be of anything in the world, there are virtually a limitless number of different classes and therefore it is not possible to specify optimal features to be extracted for each of the classification tasks. Thus, it is indeed a benefit of neural networks that they free the scientist from the necessity of crafting specific features for each possible class. However, in many problem domains, neglecting to fully understand the processes that gave rise to the data (i.e., the training data) and then not actively participating in feature selection, yields a model of which we have no understanding. In the following paragraphs we will briefly describe some of the implications of this aspect of neural networks.

"Black box" nature

If one reads the scientific literature on neural networks, one will quickly see that they are often described as a "black box." What is meant by this? The complexity of a network trained to tackle non-trivial "real-world problems" is very high. That is, the internals of the network hide an incredible mathematical complexity. In fact, the complexity is so great that one cannot interpret how the input features provide a basis for the output. This is known in the statistics field as a non-attributable model. In the context of an application like image captioning, this is of some concern. For example, if the neural network errs and gives the label "dog" to an image of a "cat," the designer is troubled by not knowing exactly what features in the image caused the misclassification and therefore being unable to alter the algorithm in order to prevent future misclassifications. This lack of insight into the relation between input and output is much more troubling for DoD applications, where the consequences of misclassification are often much more serious. For example, if one is designing an autonomous vehicle, one would like to be able to predict how the vehicle will react to a given input from its sensors. With a neural network, it is generally impossible to know the output of the network prior to presenting the input to the system and observing the output.

Generalizability

Generalizability is the ability of the model to produce reasonable output when presented with an input that is different from the data used to train the network. The issue of generalizability is a central concern in DoD applications: how will a fielded system perform with subtle changes to the environment? As we do not know much about either the processes that gave rise to the training data

nor much about the "black box" nature of neural networks, it is impossible to predict the output. There are no guarantees of the behavior of the neural network, even when presented with inputs that do not vary substantially from the training data. In fact, there is currently no body of theory that governs the behavior of neural network outputs. The class of functions that a neural network model is chosen from has tremendous richness and great power to approximate highly nonlinear behavior. Though this expressive power is useful to learn from training, model richness is a potential problem when one wants to predict/classify a new observation. The figure below represents how a model (green line) created with far too much complexity suffers from overfitting and does not generalize to the simple linear model (black line) that generates the data points (black circles).

Studying the mitigation of these problems

Numerous organizations have begun programs to better understand these limitations. The Defense Advanced Research Projects Agency (DARPA) has started the Explainable AI (XAI) program which seeks to develop methods to better understand the decisions made by AI systems. DARPA has also begun the Lifelong Learning Machines (L2M) program. This program seeks to develop machine learning based systems that provide the capability to train themselves in the field in the face of new environmental or mission-based conditions.

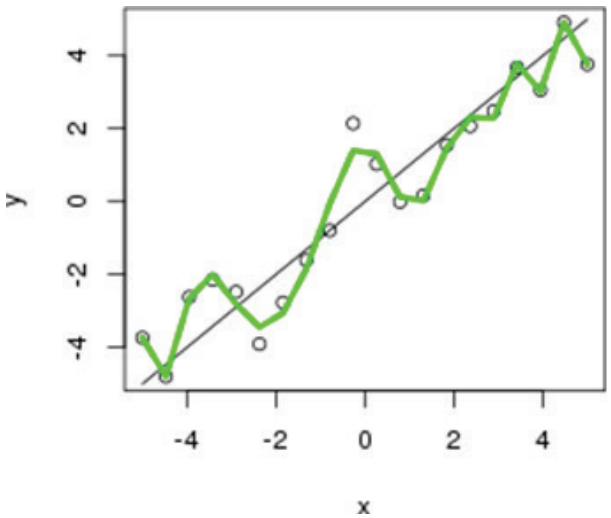
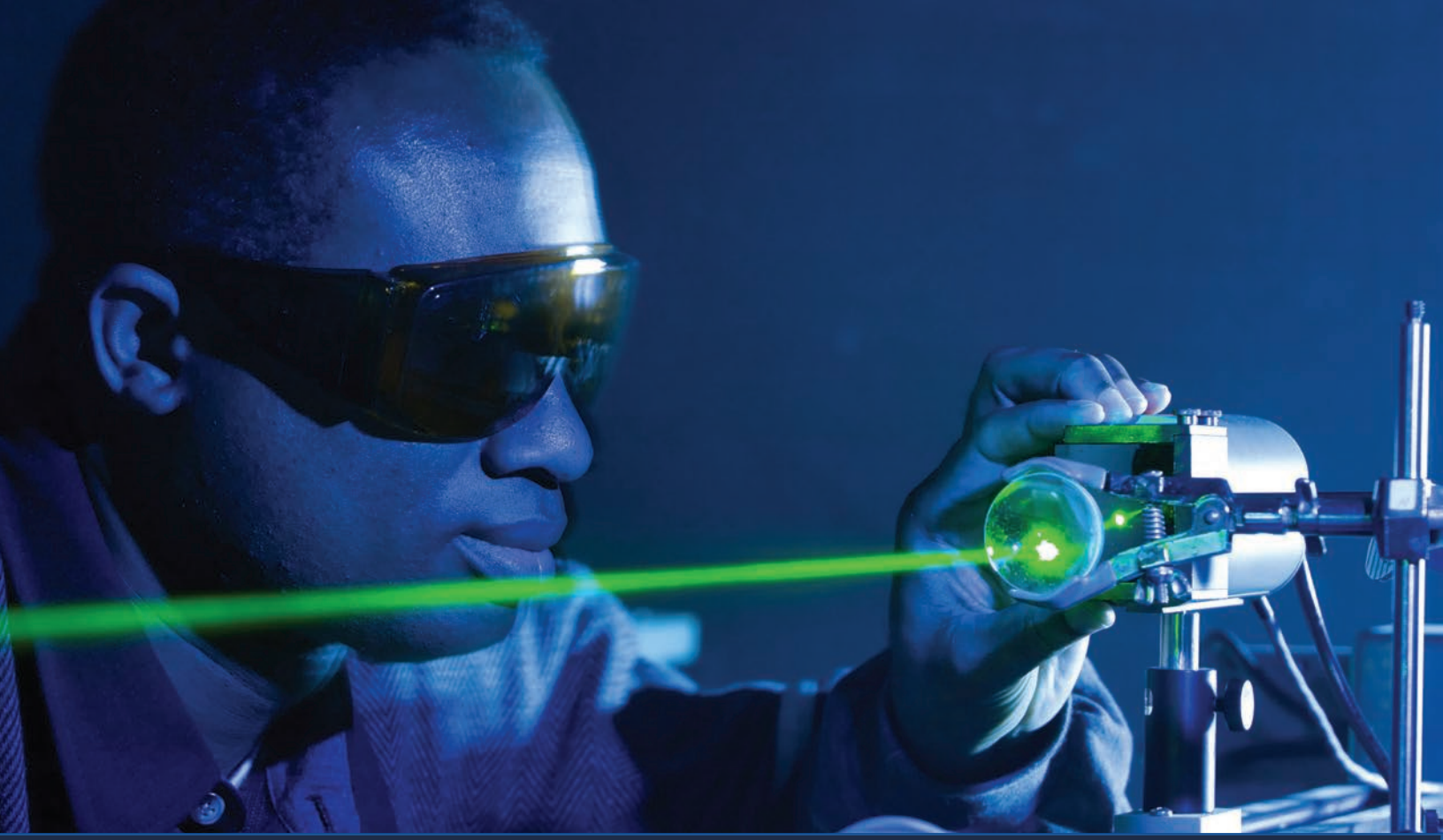


Figure 3. Simple linear model with white noise

Our own organization, the Naval Surface Warfare Center Dahlgren Division (NSWCDD), has also begun efforts to help better understand these shortcomings. Our ongoing effort, "Neural Networks for Manifold Discovery," seeks to apply advanced mathematical methodologies to better characterize the fragility of neural networks and other machine learning methodologies. Our new start effort, "Adversarial Learning for Robust AI," seeks to use recent research in "adversarial examples" to better understand how we can make neural network based systems more robust to environmental or enemy precipitated changes to operational environments. Both of these efforts were funded under the Naval Innovative Science and Engineering (NISE) program. The NISE program is designed to serve as a major innovation catalyst for the naval surface warfare centers.

Final comments

We hope that we have presented a fairly objective overview of artificial neural networks. We have tried to describe both the strengths of this class of machine learning algorithms, as well as illustrating some of their current limitations, especially in the unique environment of the DoD. We acknowledge the demonstrated successes of neural networks and believe that there are settings where the technology works very well; for example, developing AI for wargaming, planning, or training seems a very good use of the technology. In situations of complex environments where system performance errors have the potential for tremendous fiscal cost and potential to endanger lives, we need to be very cautious. We remain optimistic that programs at DARPA, NSWCDD, and other organizations can help better understand and ultimately mitigate these shortcomings. Until theory can catch up with practice, is a system whose outputs we can neither predict nor explain really all that desirable?



Performance Numbers vs. Safety Numbers for Laser Hazard Evaluations

By Sheldon Zimmerman and Mary Zimmerman

Abstract

Laser classification and hazard evaluation require certain laser beam parameters to perform the required measurements and calculations. When parameters are provided to a laser safety evaluator, they are often given from a performance perspective for a laser or laser system. While the performance numbers are not incorrect, they often are not what a laser safety evaluator really needs to do their measurements or calculations to perform a reasonable, realistic laser hazard evaluation.

It is hoped that this paper can help to bridge the gap between laser manufacturers and laser safety professionals by providing a better understanding of what a laser safety professional needs to know about a laser in order to better perform laser safety calculations and measurements.

The typical parameters required to perform a laser hazard evaluation for a continuous-wave (CW) laser are wavelength, power, beam size, and beam divergence. For a pulsed laser, the parameters needed include wavelength, energy per pulse, beam size, beam divergence, pulse repetition frequency, and pulse duration. Whether these parameters are provided from a performance perspective or a safety perspective can significantly change the results of a laser hazard evaluation, whether it is for the measurement portion (if measurements are done), or the calculations performed. What follows is a description of how each parameter can be looked at from each perspective and how the results can be affected because of those differences.

Wavelength

Depending on the wavelength of a laser beam, the precise wavelength may not be needed for either calculations or measurement, but there are times when it is crucial. Wavelength is particularly important for calculations near where the maximum permissible exposure (MPE) value is directly dependent on the wavelength, near MPE changed borders, and for measurements with a detector that has a strong wavelength dependence for its reading. For laser diodes in particular, the central wavelength in a wavelength range is often specified, but one of the extremes of that wavelength will determine the most conservative MPE. A measurement example could be where the manufacturer has specified a wavelength of 532 nm for a doubled Nd:YAG laser and they have not filtered out the fundamental 1064 nm radiation, but the evaluator did not know or think to use a filter to find out how much of the output is 532 nm and how much is 1064 nm. Without a filter, the portions of the beam that are at each wavelength cannot be measured precisely.

Beam Diameter/Size

It is somewhat uncommon for beam size or diameter to be very significant with regard to calculations of hazard distance, optical density, or even classification. However, with large beams the size can be of particular importance. The most often encountered difference here between the manufacturing and safety community is that, when a manufacturer specifies a Gaussian beam size or diameter, it is at a $1/e^2$ reference, where the safety community uses 1/e beam sizes to properly account for the maximum central beam irradiance or radiant exposure for the purposes of safety.

Beam Divergence

Beam divergence is usually provided as a maximum for performance, but the minimum beam divergence is needed for hazard calculations. If no minimum divergence is specified, the evaluator may have to use the diffraction-limited beam divergence, often resulting in overly conservative results. For example, for laser safety calculations, if someone is firing a 532 nm laser beam out of a 20 cm telescope, they may specify a divergence that is no more than 350 µrad, but the minimum divergence of the laser is not specified. The real minimum divergence may be 300 µrad, but the laser safety evaluator, not knowing the minimum divergence will likely assume a diffraction-limited divergence, about 17 µrad. If a value of 17 µrad is used instead of 300 µrad in the nominal ocular hazard distance (NOHD) calculations, the calculated NOHD will artificially be far longer than a much more realistic NOHD.

One measurement example could be where a manufacturer says that the divergence of their laser beam is no more than 1 mrad, and an evaluator has brought equipment to their facility to measure a divergence of approximately 0.5-0.75 mrad. However, upon arrival, the evaluator finds that the performance is significantly better, and the divergence is approximately 0.1-0.2 mrad, so the difficulty in measurement increases, and the equipment required to make a precise measurement may be different from what the evaluator brought. Certainly, this could lead to some difficulty, whether it is simply in taking extra time to figure out how to use the equipment the evaluator has on hand to measure the narrower divergence, perhaps with more space, using a different technique, having to have borrow equipment from the manufacturer, or having equipment shipped from the evaluator's facility.

Total Power

Total power is often specified as no less than some value, where the safety person needs to know the maximum power output of a laser device to perform calculations and measurements appropriately.

A calculation example could be where the output power of a CW visible laser beam is specified to be no less than 4 mW, while the maximum output of the laser is actually 10 mW, resulting in a Class 3B versus a Class 3R hazard classification. This certainly makes the likelihood of an error occurring, either in the laser class or in the actual quantification of

the hazards from the system. For measurements, if the power output is near the maximum for a detector response and the output is more than the specification, the measurement will be incorrect due to detector saturation or similar effect.

Total Energy per Pulse

Much like total power, total energy per pulse is often specified as no less than some value, where the safety person needs to know the maximum energy per pulse.

Similar to power, a pulsed calculation example could be where the pulse energy of visible laser beam is specified to be no less than 4 mJ, while the maximum output of the laser is actually 10 mJ, again increasing the likelihood of an error occurring, either in the laser class or in the actual quantification of the hazards from the system. For measurements, likewise, if the energy per pulse is near the maximum for a detector response and the output is more than the specification, the measurement will be incorrect due to detector saturation or similar effect.

Pulse Repetition Frequency

Pulse repetition frequency (PRF) is the laser output parameter that is most often matched from both a safety perspective and a performance perspective, usually because the highest PRF is the best for performance and the most conservative for safety. More often than not, the discrepancy is in the power given on the units, but even this is uncommon. For example, the output may be provided in kHz when a laser really is pulsing at MHz. For calculations this would result in an average power 1000 times less than the actual output, assuming the energy per pulse were correct as specified. When it comes to measurements, the detector chosen may not respond properly to a PRF that is significantly higher than reported.

Pulse Duration

Pulse duration is also usually matched from both a safety perspective and a performance perspective, usually because the shortest pulse duration is the best for performance and the most conservative for safety. Here also, the discrepancy is usually in the power given on the units, but it is uncommon. For example, the duration may be provided in ms when a laser really is pulsing at ns. This kind of error has a significant effect on the MPE calculation at the very least.

Concerning Measurements

When performing on-site measurements outside an evaluator's laboratory or facility, measurement difficulties are more likely to occur, simply because the evaluator does not have access to all of the equipment in his or her lab. Even when doing measurements in an evaluator's lab, there is the possibility of not having the required equipment for what turns out to be an unforeseen safety issue due to the specifications provided being with respect to performance.

Conclusion

Laser hazard evaluations performed based on performance numbers can be far too conservative or provide safety numbers that are inadequate to address a laser beam's true hazards. It is important that the numbers used in laser hazard evaluation be provided or found from a safety perspective to avoid erroneous safety numbers such as NOHD or required optical density for protection of users from hazardous laser radiation.

Cybersecurity Analytics for Statisticians: A Case Study of Text Data

By David J. Marchette and Rakesh M. Verma

Introduction

Cybersecurity, a huge area of research, is critical for protecting our computer networks and infras- tructure. Here we focus on one subfield of cybersecurity that lends itself naturally to statistical analysis: threat reports, security mechanisms and attacks, all of which involve some text data, e.g., attacks using emails/messages as vectors.

We take emails for granted now, but they have been around for less than 50 years, since the early 1970s. Initially they were limited to researchers in computer science, mostly in academia and government, and to small networks called local area networks. In the 1980s, a technical document called RFC[1] 821 was ratified, which described SMTP, the simple mail transfer protocol. However, the popularity of emails really skyrocketed when the Internet became widespread in the late 1990s. The Radicati Group, which researches email and social media usage and patterns, estimates that in 2018 more than 250 billion emails were sent.

With the easy access afforded by the Internet, a Pandora's box of problems was opened. Hackers started infiltrating weakly protected accounts and computer systems. Malware, software designed to damage computer systems or steal sensitive information, was perfected and deployed. Defenders were slow to catch up at first, but, over time, malware detectors (so called Anti-virus software) and intrusion detection systems were designed. Technical defenses of accounts and systems were also improved using software such as John the Ripper for cracking weak passwords proactively to ensure strong passwords were used, and the use of penetration testing, in which security experts attacked the network to discover (and fix) vulnerabilities. The arms race between the defenders and the attackers was on.

1 Request for Comments

Meanwhile in the email world, first came spam: emails containing advertisements for all kinds of products and services. Later, cybercriminals realized the true potential of email. Rather than breaching the technical defenses put up by defenders, it was much easier to create a fake website that mimicked a popular one, e.g., PayPal or eBay, and lure unwitting email recipients to the fake website with the goal of stealing the entered information. Similarly, it was much easier to spread malware by attaching it to an email and disguising it as an invoice or the agenda for the next meeting. Again, the unwitting human recipient would download it to the computer where it would infect the machine and silently steal information as long as it could beat the latest Anti-virus technology. Thus began the attack now known as phishing.

While estimating the losses due to phishing is difficult and the reported numbers are likely inaccurate, there is no question that vast amounts of money and time are lost through these attacks, and they can directly affect any one of us. Readers interested in these numbers can check out the reports published by the Anti-phishing Working Group, https://apwg.org/.[2]

Defenders now had to design spam and phishing email detectors and thereby hangs a tale. But before we can study the techniques behind these detectors, let us look at the structure of an email. An email consists of a header at the beginning of the email, which contains the information about the sender and recipient: much like the envelope of traditional letters. The header also contains a subject field, and routing information that is appended by the mail servers the email passes through, according to SMTP. The body of the email contains the actual message. Finally, emails can contain attachments – files that are sent along with the email.

While there are non-statistical methods to detect phishing - for example, ensuring that the link viewed by the user is actually the link that is accessed when it is clicked or analyzing the header information – clearly text analytics can be used to detect certain types of phishing and spam, which we discuss below. But, before doing that, it will be helpful to review text mining and natural language processing (NLP) briefly.

Text and Its Analysis

The overwhelming majority of data on the Internet is unstructured text. We generate more such data daily by writing emails, messages, memoranda, white papers, notices (e.g., from US-CERT), website updates, etc. As noted above, many cyberspace attacks start out with a well-crafted email/message. Collectively, such attacks via emails/messages are referred to as s*ocial engineering* attacks. This class includes business email compromise, job scams, and phishing/spearphishing attacks.

Humans are considered the weakest links in the cybersecurity chain. The popularity of these attacks stems from the fact that no technical defenses must be overcome, it is enough to just induce risky behaviors. Hence, they are likely to continue.

Another concern is deceptive content, e.g., "fake news," conspiracy theories, and the like. De- ceptive content is now proliferating on social networking sites and the Internet. It is insidiously undermining faith in public institutions and the news media, and destroying trust. A society, espe- cially one in which so many transactions are online, cannot function effectively when trust is lost. This is another fertile area for text mining and NLP approaches.

Statistical Analysis of Text.

A popular method for analyzing text data is the bag-of-words model. In this model, only the word frequencies are important; it's as if we jumble the words into a bag, forgetting word order, and simply count the number of times (or proportion of times) a given word appears in each document. While word order is important for understanding a document,

2 A 2020 report on cybercrime from NIST conducts an analysis of large, transparent studies. For 2016, it estimates staggering financial losses of between $160-770 billion in the US.

it turns out that these frequency histograms – also called term-document matrices – can be used to great effect in many document classification tasks. These features can be enhanced via the addition of word n-grams to encode phrases rather than single words. A word n-gram is an ordered sequence of *n* words. For example, in the phrase "my dog has fleas" there are three 2-grams (word bigrams) "my dog", "dog has" and "has fleas". In some applications, such as the analysis of passwords or the bytes in a malware executable, character n-grams are used instead of word n-grams.

The bag-of-word model loses grammar and syntax information, and for situations in which this loss may be critical, one can implement part-of-speech tagging, a subfield of NLP. This tags words or word phrases according to their parts of speech: nouns, verbs, adjectives, etc.

While the bag-of-words model does lose grammatical information, it can still provide consider- able information about the content. Topic models are a powerful class of statistical models that utilize term-document frequency matrices. The basic model is to view a document as a mixture of topics, and each topic as a mixture of words. This is a generative model, in which a document is generated by iteratively drawing a topic from the distribution of topics, then a word using that topic's distribution over words. See Blei et al. [2003], which describes the original Latent Dirichlet topic model.

Applying Text Mining and NLP to Cybersecurity

Besides spam, phishing and fake news, text mining and NLP techniques have also been used for password security, analyzing threat/security reports, identifying disgruntled employees, and attack generation. They have also been applied to malware detection, by treating the byte pattern of the malware as if tuples of bytes were words, and building models on n-grams of these "words."

Datasets & Techniques for Text-based Attacks

Before exploring statistical techniques, a few words on datasets are in order. Good, diverse and recent datasets are difficult to find because of privacy and reputational-risk concerns. Good, *labeled* ground-truth datasets are even harder, since much manual effort is required for labeling.

For password security there are several public datasets because of leaks and hacks. For example, about 10 million plaintext passwords, and SHA-1 hashes of hacked passwords are available.[3] For a study on disgruntled employees, researchers scraped a small dataset from Vault.com and Yahoo discussion groups. For attack attribution from threat intelligence reports, researchers collected publicly available reports, labeled them, and released the dataset.[4] It includes 249 labeled documents and over 20,000 unlabeled ones. For phishing, the IWSPA-AP and other datasets are available. This collection includes two datasets: emails with and without headers. They were collected from several sources and have undergone two rounds of cleaning and preprocessing to ensure high quality.

Statistical Techniques for Bad Passwords, Phishing Emails and Threats

We now discuss techniques for two security challenges: finding "bad" passwords and phishing email detection. The idea of detecting bad passwords comes from the observations that humans are very bad at producing random sequences, and they are even worse at remembering random sequences, so passwords tend to have patterns that the attacker can exploit. A dataset *D* of bad passwords was collected and a second-order Markov model, *M*, with 28 states was built. Our discussion follows Verma and Marchette [2019]. The states of *M* correspond to the 26 letters ignoring case distinctions, a state for space (SPC) and a state (OTH) for the remaining forty to fifty characters that include: digits, punctuation and special characters.

3 For example, see https://haveibeenpwned.com/Passwords

4 https://github.com/eyalmazuz/ThreatIntelligenceCorpus

For the transition probabilities, construct a frequency array f, , where $f[i,j,k]$ is the frequency of the character trigram ijk. For instance, for the password, passwd247, we get the overlapping trigrams pas, ass, ssw, swd, wdOTH, dOTHOTH and OTHOTHOTH. For each bigram ij, denote $f(i,j,\infty)$ as the total number of trigrams beginning with ij. Then $T[i,j,k] = f(i,j,k)/f(i,j,\infty)$, the maximum likelihood estimate of the transition probabilities. Good-Turing smoothing was used since many of the values were zeros.

Now the question of whether a given password is bad reduces to the likelihood that it is generated by M. The test used is a log-likelihood function. Let password $p = p_1 p_2 \ldots p_l$ then $llf(p) = \sum_{i=1}^{l-2} \ln(T[p_i, p_{i+1}, p_{i+2}])$

For the final test, calculate

$$BA(p) = \frac{\frac{llf(p)}{l-2} - \mu}{\sigma} \qquad (1)$$

where μ and σ are the mean and standard deviation of $\frac{llf(p)}{l-2}$.

Find the mean and standard deviation by computing $\frac{llf(p)}{l-2}$ for every password p in D.

Because of centering and normalization, BA(p) has a mean of zero and a standard deviation of one. The authors set a threshold of 2.6 standard deviations, about 99% of the area under the normal curve, and accept as good any password that has a value less than –2.6. Passwords close to the mean, zero, are viewed as being drawn from D and therefore unacceptable. Besides the normality assumption for *llf* values, the definition of a bad password depends on collecting a good dataset D. We turn to phishing next.

Anatomy of a Spearphishing Attack

Spearphishing is an attack in which the attacker tailors the email for a specific target. This specificity increases the probability of success. For example, an email came one morning to the first author purporting to be from the chair of his department. It just said, "are you available?" It had the name and position of the chair at the bottom. Nowadays, we are very used to reading our emails on cellphones that show very little email metadata. So, I responded. Then a reply came stating that: "I am in a meeting right now, so I can't call you. I want you to do something urgent for me."

Now the alarm bells started ringing. Upon checking the email header, I found the phisher had created a fake account with the department chair's name. This example illustrates the phishers are willing to go the extra mile for a higher chance of success. According to anecdotal reports, several faculty all over the US (and in other countries as well) have fallen for this specific attack already and lost considerable time and money in the process. Spearphishing attacks are also suspected to be the cause of the successful attack on the Ukraine power grid (Case [2016]), and there are other famous examples.

Phishing Email Detection

Next we analyze phishing emails from the "new" Nazario dataset,[5] https://monkey.org/~jose/ phishing/, corresponding to 3,388 phishing emails from 2007 to 2015. More recent email datasets, e.g., IWSPA-AP dataset[6] are available, but this one will serve just as well to illustrate the methods. We start with n-gram analysis, but instead of character n-grams for passwords and links, we consider word n-grams. Hypothesizing that phishing emails typically ask people to visit a website or download an attachment, we first check how many files have the 1-grams "click" and "download." The unigram"click" is in 1,125 files (of 3,388), close to a third, but "download" is in only 200 files. Of course, there may be some files containing both these words also. We next check for files containing the words "account," "html," "white"

5 To distinguish it from a 2004 dataset, which became popular in academic research on phishing.

6 http://ceur-ws.org/Vol-2124/

and "#ffffff" (hexadecimal for white).[7] From the table, we can see that Html is very popular with phishers as 4 out of 5 emails use it. Also, more than half of the emails have a reference to the word "account."

Readers might wonder why we chose the unigrams "white" and "#ffffff." The reason is that phishers might include in their emails some text to confuse spam/phishing detectors, which is obscured from the recipients by using the same color font for it as the typical background color.

Unigram	Number of Files	Percentage
click	1125	33.2
download	200	5.9
account	1950	57.7
html	2751l	81.2
white	1377	40.6
#ffffff	402	11.9

Table 1: Number of files and percentage containing each unigram (new Nazario dataset)

Next, we examine 3-grams (or trigrams). Since we want to leave some emails for testing our hypotheses, we just start with a toy (20 emails) random sample of phishing emails and give it to a concordance software. As seen in Table 1, there are some interesting patterns. These emails have headers as well, but we are focusing on the n-grams from the email bodies.

Trigram	Raw Frequency
your paypal account	13
all rights reserved	9
the link below	9
to your pay	9
in to your	8

Table 2: Top trigrams from 20 random phishing emails (new Nazario dataset)

This is for the phishing class. We also need to consider the legitimate class to determine viability of the n-gram approach. For this we check the IWSPA-AP 2.0 training dataset of legitimate emails with full headers. In 4082 emails we find that 160 files (3.9%) contain the unigram click and 206 files (5.0%) contain the unigram account. This gives us some hope that the n-gram approach may be worthwhile. But there is much work remaining.

This approach was studied in Verma and Hossain [2013]. They split the dataset into 70% for training and 30% for testing. The training dataset of phishing and legitimate emails was analyzed with a t-test to determine whether a feature's variance between two datasets is statistically significantly different. They used a two-tailed, two samples of unequal variances t-test since the phishing and legitimate datasets are of different sizes as well as variance.

After some experimentation, the researchers in Verma and Hossain [2013] considered frequencies of bigrams following the word "your." A bigram was chosen as a possible feature if its t-value exceeded the critical value for an α value of 0.01. As usual, a denotes the probability of a Type I error. Then weights were calculated for each selected bigram, b, as follows:

$$w(b) = \frac{(p_b - l_b)}{p_b} \qquad (2)$$

In Equation 2, p_b (respectively l_b) denotes the percentage of phishing emails (respectively legitimate emails) that contain b. Features that appeared in less than 5% of the emails or had weights less than 0 were discarded. Finally, a bigram b was selected, if $w(b) > \mu - \sigma$, where μ is the mean bigram weight and σ the standard deviation. The resulting set of bigrams is called PROPERTY, since it denotes bigrams referring to the user's property that has been

7 Strictly speaking, we should not examine the entire dataset. We should reserve a portion for testing and we should not use it for our learning or hypothesis making. However, these counts are just for illustration.

affected (e.g., "credit card"). A similar analysis was conducted for all the words that appear in sentences containing a hyperlink or any of the words: url, link, or website. This analysis leads to the set of words called ACTION. The two sets lead to the Action-Detector sub-classifier:

For each email, label it phishing if it has:

1. The word "your" followed by a bigram belonging to PROPERTY (e.g. "your paypal ac- count"), and
2. A word from ACTION in a sentence containing a hyperlink or any word from {"url", "link", "website"} (e.g. "click the link").

A Nonsense-detector is also designed to detect link-containing emails whose subjects do not match up with the body of the email, i.e., no word from the subject, after removing stopwords,[8] appears in the body. The Action-detector and the Nonsense-detector were composed sequentially.

The two detectors have several versions for robustness (recall that wily hackers are always trying to defeat these methods), the first versions of both detectors just use pattern matching. The second uses part-of-speech tagged features: bigrams that do not contain a noun or a named entity in the set PROPERTY, words that are not verbs in the set ACTION are discarded, and the Nonsense- detector only works on nouns, verbs, adjectives and adverbs in the subject of the email, and for subject-body similarity only nouns are used. The third adds sense tags to part-of-speech tagged features using SenseLearner Mihalcea and Faruque [2004], and the last extends the noun features using WordNet's[9] synonymy and the direct hyponyms of these synonyms. A few more changes are made in this last version, for more details see Verma and Hossain [2013].

We call these approaches feature engineering with statistical analysis/learning. Much early phishing detection work was in this direction. A recent survey of features and methods for phishing URL, website and email detection, and of phishing susceptibility studies, is given in Das et al. [2020]. A benchmarking evaluation of features and methods is in Aassal et al. [2020]. Another possibility is to avoid feature engineering and input the text of the emails using word embeddings, e.g., GLoVe[10], FastText[11] or ELMo[12] into a deep learning model that uses a lot of training data to automatically learn the features.

Threat Analysis

Finally, we consider threat analysis. The National Vulnerability Database[13] (NVD) contains reports of vulnerabilities covering two decades. A temporal topic model analysis was performed by Williams et al. [2020], showing the dynamic nature of the pattern of vulnerabili- ties, both over the applications and operating systems, and the types of vulnerabilities discovered. By analyzing these patterns, one can better understand the threat environment, which can help with planning and mitigation efforts. A similar study on attacks would be extremely valuable for understanding how the attackers exploit the vulnerabilities, how quickly they react to detected vul- nerabilities (and how often they utilize these vulnerabilities before the community detects them) and how the threat environment changes in time.

8 These are frequently occurring words such as conjunctions and prepositions.

9 https://wordnet.princeton.edu/

10 https://nlp.stanford.edu/projects/glove/

11 https://github.com/facebookresearch/

12 https://allennlp.org/elmo

13 https://nvd.nist.gov

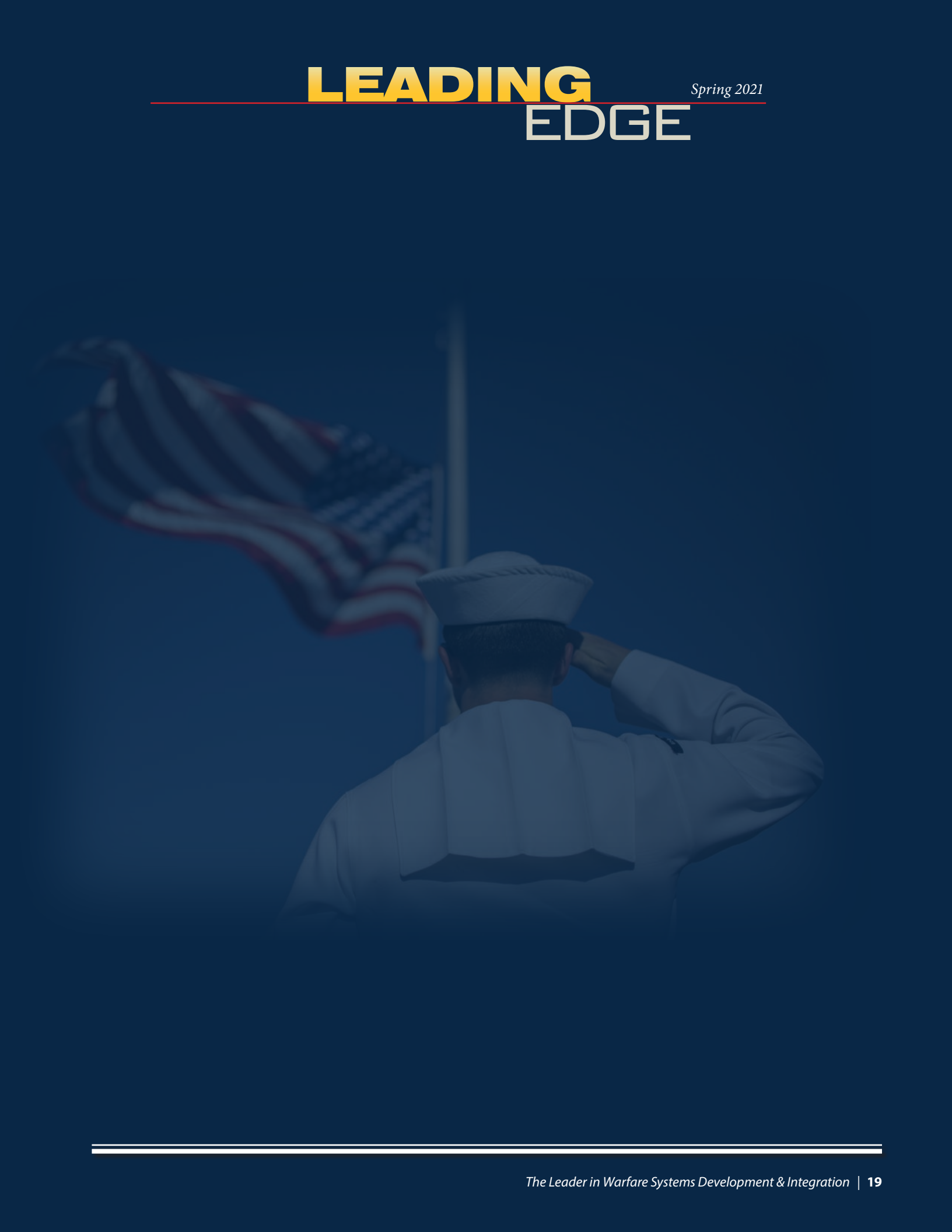
Discussion and Next Steps

Clearly, there are many opportunities for statisticians in cybersecurity. We have illustrated just text analysis for passwords, phishing and vulnerability reports. Others include: network analysis, the detection of malware, intrusion detection, and analyzing the behavior of users to detect insider threats. Since attackers are constantly trying to defeat defensive filters, an interesting direction for future research is adversarial machine learning Lee and Verma [2020]. It includes attack generation and techniques for building robust models. Readers interested in the state-of-the-art in phishing can look up the surveys cited above. For more information on text mining and natural language processing as well as other statistical and machine learning methods applied to cybersecurity, see Verma and Marchette [2019]. For a ranked list of free NLP tools see the URL below.[14]

References

1. Ayman El Aassal, Shahryar Baki, Avisha Das, and Rakesh M. Verma. An in-depth benchmarking and evaluation of phishing detection research for security needs. IEEE Access, 8:22170–22192, 2020.
2. David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet Allocation. JMLR, 3(Jan): 993–1022, 2003.
3. Defense Use Case. Analysis of the cyber attack on the Ukrainian power grid. Electricity Information Sharing and Analysis Center (E-ISAC), 388, 2016.
4. Avisha Das, Shahryar Baki, Ayman El Aassal, Rakesh M. Verma, and Arthur Dunbar. SoK: A comprehensive reexamination of phishing research from the security perspective. IEEE Commun. Surv. Tutorials, 22(1):671–708, 2020.
5. Daniel Lee and Rakesh Verma. Adversarial machine learning for text. In Proceedings Sixth Inter- national Workshop on Security and Privacy Analytics, page 33–34, New York, NY, USA, 2020. ACM.
6. Rada Mihalcea and Ehsanul Faruque. Senselearner: Minimally supervised word sense disambigua- tion for all words in open text. In Proceedings Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text, SENSEVAL@ACL 2004, Barcelona, Spain, 2004.
7. Rakesh M. Verma and Nabil Hossain. Semantic feature selection for text with application to phishing email detection. In Hyang-Sook Lee and Dong-Guk Han, editors, Information Security and Cryptology, 16th International Conference, Seoul, Korea, Revised Selected Papers, volume 8565 of LNCS, pages 455–468. Springer, 2013.
8. Rakesh M Verma and David J Marchette. Cybersecurity Analytics. CRC Press, 2019.
9. Mark A Williams, Roberto Camacho Barranco, Sheikh Motahar Naim, Sumi Dey, M Shahriar Hossain, and Monika Akbar. A vulnerability analysis and prediction framework. Computers & Security, 92:101751, 2020.

14 https://www.predictiveanalyticstoday.com/top-free-software-for-text-analysis-text-mining-text-analytics/



DAHLGREN

The Leader in Warfare Systems Development & Integration




NAVSEA
WARFARE CENTERS
DAHLGREN

NSWCDD Corporate Communications
540.653.8152 | NSWCDD.Info@navy.mil
http://navsea.navy.mil/Home/Warfare-Centers/NSWC-Dahlgren