2020 NDIA GROUND VEHICLE SYSTEMS ENGINEERING AND TECHNOLOGY SYMPOSIUM
VEA TECHNICAL SESSION
AUGUST 11-13, 2020 - NOVI, MICHIGAN

Latent Dirichlet Allocation (LDA) for Anomaly Detection in Ground Vehicle Network Traffic

Adam Thornton[1], Brandon Mieners[1], Donald Poole[1], Mark Russell[2]

[1]Southwest Research Institute, San Antonio, TX
[2]U.S. Army Combat Capabilities Development Command (CCDC) Ground Vehicle Systems Center (GVSC) VEA

ABSTRACT

Latent Dirichlet Allocation (LDA) and Variational Inference are applied in near real-time to detect anomalies in ground vehicle network traffic for VICTORY enabled networks. The technical approach, that utilizes the Natural Language Processing (NLP) technique to detect potential malicious attacks and network configuration issues, is described and the results of a proof of concept implementation are provided.

Citation: A. Thornton, B. Meiners, D. Poole, M. Russell, "Latent Dirichlet Allocation (LDA) for Anomaly Detection in Ground Vehicle Network Traffic", In *Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium* (GVSETS), NDIA, Novi, MI, Aug. 11-13, 2019.

1. INTRODUCTION

The detection of anomalies in network traffic can help network operators secure and control their networks. Additionally, detection of anomalies that do not necessarily pose a security threat are also of interest and serve as the basis for revealing network misconfigurations and failures. Many existing anomaly and intrusion detection systems are rule and pattern based [1] [2] and require extensive domain knowledge to develop the rules and pattern matching. To that end, Latent Dirichlet Allocation (LDA) was employed, which is a Natural Language Processing (NLP) concept. NLP is a branch of Artificial Intelligence (AI) that encompasses the interaction between computers and human (natural) language. Using LDA the research has demonstrated that anomalies in network traffic can be detected and potentially described in real-time [1] and utilize data-driven statistical models that can adapt and learn from captured data [2]. This would provide network operators greater insight into the operational health and condition of networks so more informed decisions could be made, helping to mitigate or completely avert failures. This approach can be applied across many domains for U.S. Army ground vehicles, including anomaly detection in Modular Open System Approach (MOSA) based networks, condition-based maintenance from automotive Controller Area Network (CAN) bus data, and sensor abnormality identification and failure detection for Command, Control, Computers, Communications, Cyber, Intelligence, Surveillance and Reconnaissance (C5ISR)/Electronic Warfare (EW), and autonomous sensing systems.

To give greater sense of the capability and potential of using LDA for network traffic anomaly detection, the first discussion will involve how

LDA was applied to model network traffic. Then, the application of developing and deploying LDA to a simulated Vehicular Integration for Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance (C4ISR) Interoperability (VICTORY) Data Bus (VDB) and its potential expandability into different applications will be presented in the following sections:

- Experimental Use Case: This concept was implemented and evaluated on the Defense Advanced Research Projects Agency (DARPA) Building Resource Adaptive Software Systems (BRASS) project and was applied against a simulated VDB. The simulated VDB contained 4 simulated services (Position, Orientation, Direction of Travel and Camera Gimbal), where the data rates of each service where randomly selected and the packet contents were statically assigned (all values were within VICTORY specification). A review of the design will be presented, including how training data was gathered, how the data was processed and composed into the model, and how it was applied against a live network to detect anomalies with VICTORY services and provide the results of the approach when it was applied against a live VDB.
- Extensible Use Case: Discuss a hypothetical application of the technique to provide vehicle health monitoring towards condition-based maintenance from a VICTORY Automotive Service or J1939 CAN-Bus data.

2. METHODOLOGY: TOPIC MODELING FOR NETWORK TRAFFIC ANOMALY DETECTION

In the realm of network security, traditional systems for detecting threats, such as SNORT [3] and Bro [4], are rules-based network security frameworks that require sufficient domain expertise to implement and do not adapt well to new threats. Thus, being ill-suited for deployment in modern networks that are constantly adapting and evolving. While research has been performed in the domain of network fault detection and misconfiguration [5] [6], very little has resulted in deployment. With the advent and rise of machine learning, techniques such as LDA can help combat these adaption issues. LDA, a machine learning technique traditionally used within NLP to model topics in documents, can be applied to model network traffic. It was hypothesized that if LDA can model the structures and relations between text documents and words, it can model the similar structures and relationships between network traffic and the Internet Protocol (IP)/User Datagram Packet (UDP) packet header elements that make up the individual network packets. This relationship can be described through the concept of a bag-of-words model, which is the input to an LDA model. Within the field of NLP, where the concern is finding relationships between text (such as a sentence or a document) and the words within the text, the text is represented as a "bag" of its words, without regard to grammar or word ordering. Just keeping the count or frequency of the occurrence of words in the text. This concept can be extended to model the relationship between network traffic and UDP packet header information where each time slice of traffic is treated as the "text" and the count or frequency of the various UDP header fields and values are viewed as words.

3. WHAT IS LDA?

Before delving into the applications of LDA it would help to provide a high-level overview of the idea behind LDA. It is a form of unsupervised machine learning, a branch of machine learning that looks for undetermined patterns in data without existing class labels and very little human supervision, used within topic modeling to ascertain topics within a set of documents. LDA views these documents as a bag-of-words. It does not care about the order of the words in the document. Thus, the overarching theme of LDA is that every document can be described by a distribution of topics and every topic can be

described by a distribution of words. To arrive at this idea, imagine a large corpus of 1000 documents, where each document can be reasonably described from a set of 1000 words. Connections can be made between documents with a subset of common words, as depicted in Figure 1. However, it is difficult to understand what category or topic each document belongs with only a single level of abstraction.

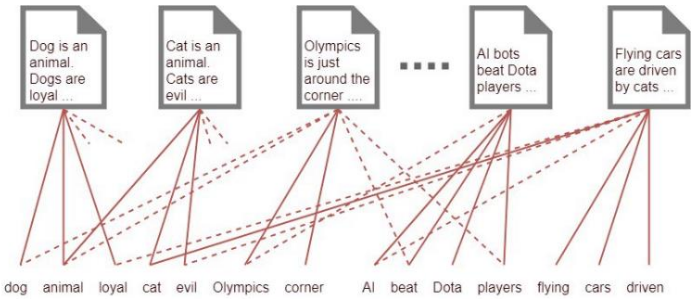

Figure 1: Document to words relationship (Bag of Words) [7]

To rectify this problem, an extra level of abstraction can be added by introducing a latent (hidden) layer within the lexicon of machine learning. This latent layer is said to be hidden because only the words and documents can be observed, while the layers and topics are not directly observable. With the observations of the words and documents, in addition to the inclusion of the hidden layer, connections between documents and topics and between topics and words can be mapped. In all cases, the number of topics/themes comprising the hidden layer is known. In the case of Figure 2, there are three topics/themes. One thing to note is that the topics of "Animals," "Sports," and "Tech" are not what LDA actually produces. Because LDA is learning a distribution of words to model topics, it would, for example, produce an output such as [0.2 * AI, 0.05 * beat, 0.05 * Dota, 0.2 * players, 0.2 * flying, 0.2 * cars, 0.1 * driven] for the topic "Tech" (note the distribution adds up to 1).

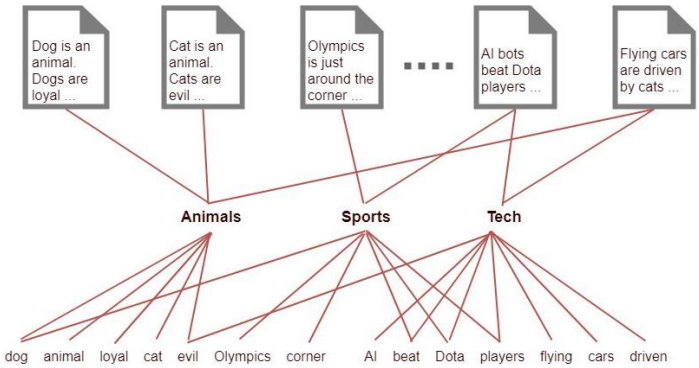

Figure 2: Latent (hidden) layer providing topic abstraction [7]

3.1. LDA: Mathematical Intuition

Given the 10,000 ft high-level view of LDA, a bit of math can be introduced to provide some formalism. Although LDA was described as going from a document, to a bag-of-words, then to topics, LDA operates in somewhat of a reverse-engineered fashion. LDA assumes that a document was generated by picking a set of topics, and then for each topic it picks a set of words. Figure 3 provides a graphical model depicting this reverse-engineered generative modeling process of LDA.

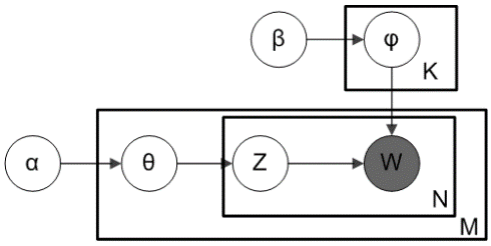

Figure 3: Graphical model of LDA showing Dirichlet distributed topic words [8] [9]

The boxes are plates that represent repeated entities. So, the outer plate represents the documents and the inner plate represents the repeated words in each document. The hyperparameters that control the LDA process are:

M – The number documents
N – Number of words in a given document
K – The number of topics
α – The parameter of the Dirichlet prior on the per-document topic distributions

β – The parameter of the Dirichlet prior on the per-topic word distribution
θ_i – The topic distribution for document i
φ_k – The word distribution for topic k
z_ij – The topic for the j-th word in document i
w_ij – The specific word

You will notice that the W, the words w_ij, is grayed out. This is because the words are the only observable variable in the process. All the other variables are latent. When thinking of the topic and word distributions θ and φ, respectively, it is helpful to recall the previous bag-of-words discussion. The entity θ_i is a matrix where the rows are documents and the columns are the word counts. So, θ, is a Dirichlet distribution (that will be discussed later) that covers all of the documents M. In a similar fashion, the entity φ_k is a matrix where the rows are topics and the columns are words counts. Thus, φ is the set of word distributions for all topics, K, which is also modeled by a Dirichlet distribution.

With this model, LDA can generate documents. Since a single document has N words, LDA will generate N topics that are to be filled (or distributed) with words. Those words will be filled by the generation K words conditioned on each topic.

3.2. What is a Dirichlet Distribution?

In describing LDA, parameters α and β were mentioned, which control the Dirichlet priors, but haven't discussed what a Dirichlet distribution is. A Dirichlet distribution is a multivariate generalization of the Beta distribution. In Figure 4, the plots show how a change in α changes the distribution

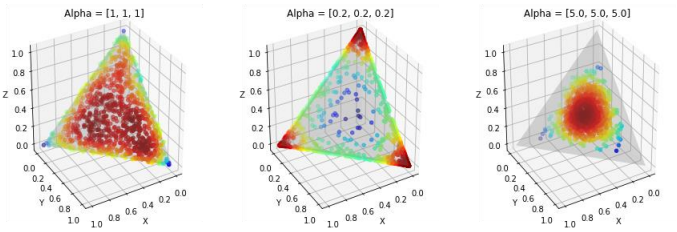

Figure 4: Dirichlet Distributions for different values of alpha [9]

The distribution moves to the middle with larger α values and moves to the corners with smaller α values.

4. VARIATIONAL INFERENCE

Finding the joint posterior probability of θ (topic distribution per document), z (the topics in each document) and β (the distributions of words for each topic) is intractable. Variational Bayes provides an approximation via known probability distribution that closely matches the true posterior. Ultimately, this becomes an optimization problem where the free variational parameters, γ, ϕ and λ, that θ, z, and β that must be found were approximated to minimize the Kullback-Leibler (KL) divergence between variational distribution and the true posterior.

5. EXPERIMENTAL USE CASE: GROUND VEHICLE NETWORK

Using LDA on networks to detect anomalies that are indicative of not just malicious attacks but also indicators of network misuse and misconfiguration has been done other researchers [1] and [10], for example. This technique could be applied not to just detect typical network anomalies but also anomalies with the producers of data on ground vehicle networks.

5.1. Apples to Oranges; Documents to Networks

Before delving into the technical approach, we'll briefly provide clarification of the terminology differences between the realm of NLP, which includes LDA, and TCP/IP networks. Table 1 provides a breakdown of this analogy:

Document Terminology	Network Equivalent
Total # of Documents	Total # of network snapshots
Document	Network snapshot
Words in Document	Unique features in traffic

Table 1: Analogy Between Text and Network Traffic

The "words" that are used are the extracted features in Table 2:

Packet Layer	Extracted Feature (words)
Ethernet Layer	MAC source MAC destination
IP Layer	Total Len, Type of Service, Frag. Flags, TTL, IP source, IP dest.
UDP Layer	Port source Port destination checksum

Table 2: Network Packet Features

5.2. Technical approach

As a part of the DARPA BRASS program, it was hypothesized that this technique could be utilized to detect performance and configurations issues with a VICTORY network, a VDB. Specifically, it was desired to detect VICTORY services that were publishing incomplete data, not publishing data at the configured rate, or were unexpectedly publishing or not publishing data. These anomalies could be indicative of malicious activity or problems with the VICTORY services and data providers. Although this technique was evaluated against a VICTORY network, the same technique could be applied to detect anomalies with any sort of structured or standardized data network such as (Controller Area Network) CAN, MIL-STD-1553, etc.

The high-level overview of the technical approach involves:

- Stand up a VDB that will generate training data for the LDA model by utilizing VICTORY service simulators based on the VDB configuration defined in a VICTORY Configuration Language (VCL). These simple simulators will generate VDB traffic in a more regular and predicable manner than actual VICTORY- enabled devices because they are not reliant on external sources for data. They generate random data at their configured rate. An alternative approach to obtaining training could utilize a network capture from a real VDB where the data has been scrubbed and confirmed to be normal and ideal or manually writing network captures to represent the ideal network. These approaches could be applied for any other types of networks or buses, but the approach will depend on the tools that are available for simulating or generating sample data for the domain.
- Run the simulated training data capture VDB for a length of time and save the capture using pcap, a network traffic capture tool.
- Generate the Traffic Vectors, or bag-of-words models, from the capture.
- Feed the Traffic Vectors into the LDA to generate a model of the VDB.
- Configure real VICTORY-enabled devices running on a live VDB using a VCL file. This instructs the VICTORY services as to what VICTORY data to provide and at what rate. The live VDB is also captured via pcap. The live VDB is where anomalies are injected for this experiment.
- Generate Traffic Vectors from the live capture.
- Feed the live VDB Traffic Vectors along with the model of the training network into the Variational Inference algorithm.
- The output of this is a log likelihood estimate that the last snapshot (one second) of live VDB capture contained an anomaly.
- The snapshot and likelihood estimate are fed into a simple binary classifier (based on the likelihood threshold range found during training) to identify the type of anomaly.

Figure 5 summarizes the technical approach.

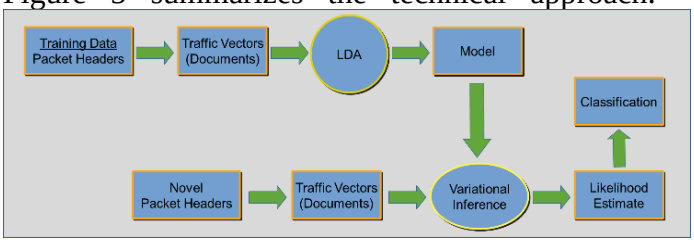

Figure 5: Technical Approach of VDB Anomaly Detection

5.3. Results

As demonstrated on the DARPA BRASS effort, the classifier was able to indicate the presence of the following conditions almost immediately after the triggering event for the following anomalies:

- VICTORY services that were not publishing data at their configured rate. The acceptable rate variability is determined by the variability in publish rates in the training data.
- VICTORY services with missing or extraneous data. The sensitivity to variability in data size depends on how much variance exists in the training data. The first training collection was a short capture. The rollover of a counter variable in the live data from tens place value to hundreds place values caused the sample to get classified as containing anomaly. This emphasizes the importance building a test model that is representative of the normal, real-world operating conditions of the network.
- VICTORY services that were supposed to be publishing but were not publishing
- VICTORY services that were not supposed to be publishing but were publishing

The goal of the effort for DARPA BRASS was to develop a proof of concept not to measure performance. Therefore, performance metrics are merely subjective as observed by the developers. The testing infrastructure and source code were not modified and instrumented in such a way to objectively gauge the performance.

There are upper and lower bounds for classifying the likelihood estimate as an anomaly. Tightening or relaxing those bounds will result in less or more allowance, respectively, in the variation of observed features before a sample is classified as an containing an anomaly. This allows tuning of the classification sensitivity.

The first test of the implementation did result in false positives that resulted from using a capture of the training network of only a few minutes. When running with the live network, a sequence counter in the data messages rolled over in place values which resulted in a change in the payload size which, in turn, was identified as anomalous. The correction for that was to run a longer capture on the training network. A capture of several hours of training data did not result in this false positive detection. Sequential tests with increasing capture times to ascertain how long the training capture needs to be to not have this false positive was not performed. No false negatives occurred.

With these results, the current implementation can be used to determine when a vehicle network is not performing and behaving as planned. It would be able to detect unexpected traffic which could be indicative of an improperly configured or behaving device or a malicious attack. It can indicate whether a device is not generating data or is generating more or less data than planned which likewise could be a problem with the configuration of the device or a deeper problem with the device. A production version of this approach could save in a database or log the questionable sample and likelihood estimate for later evaluation. One point to note is that at this point the implementation only indicates the presence of an anomaly in a sample. Additional development on the simple binary classifier stage of the approach, which currently only outputs whether the snapshot likely contained or didn't contain an anomaly, is necessary for the algorithm to provide recommendations on the cause(s) of the anomaly(s).

6. EXTENSIBLE USE CASE

We believe this technical approach can be adapted and applied to the data contained in data packets,

not just the headers of data packets. For instance, when applied to data from a VICTORY Automotive service or a CAN J1939 bus, this updated approach should be able to detect anomalous readings and behaviors for individual sensors and data points but also anomalous patterns across multiple sensors and data points. Extending the approach presented in this paper from not just detecting anomalies in network packets but also detecting anomalies in the data payload in a network packet supports Condition Based Maintenance (CBM) as the anomaly detection system warns the operators and maintainers of the vehicle that conditions are present which might be indicative of engine, transmission, or other hardware failure. This warning could be displayed on a driver display much like a check engine light with more informative sub-screens. Additionally, all anomalies would be logged so vehicle maintainers and engineers can recover and analyze the logs once the vehicle returns to the depot and make a determination if maintenance is required.

For the demonstration effort, the algorithm ran on a virtual machine with 4 processor cores and 4 GB (gigabyte) of memory and there were no performance issues with detecting anomalies on the small number of features being analyzed. There, however, is potential risk involving the performance of the approach when applied against a very large set of features. The demonstration effort only modeled a handful of features from network packet headers while the number of features for an automotive system is orders of magnitude larger and the number of features across all systems on a vehicle is even larger. This will require scalability testing. A few remedies could include careful down-selection of features to monitor or adding a larger dedicated processing unit to the vehicle for running the algorithm.

While this leads to very close to real-time failure detection, applying this technical approach to live data only means the failure being detected will most likely be imminent. This approach could be extended even further to observe historical data captures, perhaps over the entire lifespan of the vehicle and a culmination of captures from all vehicles for a given platform. This would allow analysis and failure prediction and provide maintenance recommendations with ample time before catastrophic failure, thus leading to predictive maintenance.

7. SUMMARY

Even though LDA is typically associated with NLP and processing documents, the technique was adapted to the realm of information contained in network packet headers. Through the application of LDA and Variation Inference, missing and extraneous VICTORY services, services publishing at a rate different than configured the configured rate, and data messages containing more or less data then expected were detected on a VDB via the model-based machine learning technique rather than laborious rule-based approaches. An extensible use-case of the approach has potential to provide Condition Based Maintenance warnings and recommendations, thus reducing the risk of breakdowns of deployed vehicles.

8. REFERENCES

[1] B. Newton, "Anomaly Detection in Network Traffic Traces Using Latent Dirichlet Allocation," ~, 2012.

[2] C. Cramer and L. Carin, "Bayesian topic models for describing computer network behaviors,"2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, 2011, pp. 1888-1891, doi: 10.1109/ICASSP.2011.5946875.

[3] M. Roesch, "Snort - Lightweight Intrusion Detection for Networks," in Proceedings of LISA '99: 13th Systems Administration Conference, Seattle, WA, 1999.

[4]V. Paxson, "Bro: A System for Detecting Network Intruders in real-time," in Proceedings of the 7th USENIX Security Symposium, San Antonio, TX, 1998.

[5]F. Le, S. Lee, T. Wong, H. S. Kim and D. Newcomb, "Detecting Network-Wide and Router-Specific Misconfigurations Through Data Mining," IEEE/ACM Transactions on Networking, vol. 17, no. 1, pp. 66-79, February 2009

[6] S. Lee, T. Wong and H. S. Kim, "Improving dependability of network configuration through policy classification," in IEEE International Conference on Dependable Systems and Networks With FTCS and DCC (DSN), Anchorage, AK, USA, 2008.

[7] T. Ganegedara, "Light On Math Machine Learning: Intuitive Guide to Latent Dirichlet Allocation," Medium: Towards Data Science, 23 August 2018. [Online]. Available: https://towardsdatascience.com/light-on-math-machine-learning-intuitive-guide-to-latent-dirichlet-allocation-437c81220158. [Accessed 5 December 2019].

[8] D. M. Blei, A. Y. Ng and M. I. Jordan, "Latent Dirichlet Allocation," Journal of Machine Learning, pp. 993-1022, 2003.

[9] W. Contributors, "Latent Dirichlet Allocation -- Wikipedia, The Free Encyclopedia," 2004. [Online]. Available: https://en.wikipedia.org/wiki/Latent_Dirichlet_allocation. [Accessed 6 December 2019].

[10] X. Cao, B. Chen, H. Li and Y. Fu, "Packet Header Anomaly Detection using Bayesian Topic Models," Cryptology ePrint Archive, pp. 1-12, 17 January 2016.