Streaming Estimation of Information-Theoretic Metrics for Anomaly Detection

Sergey Bratus, Joshua Brody, David Kotz, and Anna Shubina

Institute for Security Technology Studies, Department of Computer Science, Dartmouth College

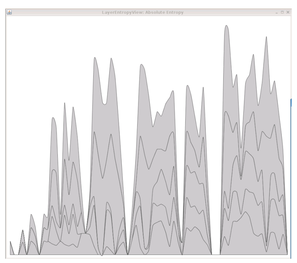
Motivation

Information-Theoretic Statistics applied to monitoring of network traffic can be useful in detecting *changes in its character.*

- Frequencies *of frames of given type/subtype*
  - *ex: too many Deauth or Deassoc frames: classic DoS flood.*
  - *require constant memory.*
- Distributions *of header field values*
  - *changes in* entropy *of a particular field value suggest anomaly. [4,5]*
  - *require memory linear in number of distinct values.*
- Joint distributions *of header field values*
  - *track co-occurence of feature values. Changes in predictability of feature pairs indicate anomaly.*
  - *require memory quadratic in number of distinct values.*

Memory requirements become prohibitively expensive for sophisticated measurements. However, recent advances in streaming estimation algorithms give hope that such computations can be made practical.

Single Feature Entropies

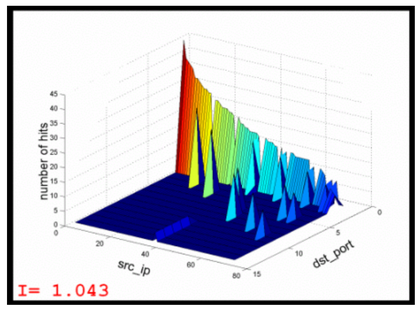


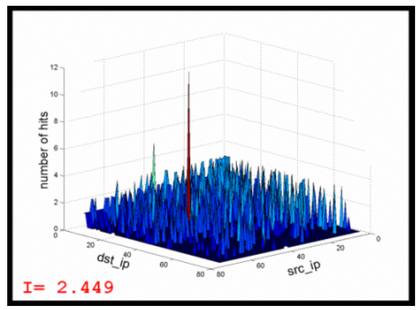
Viewing each header field as a stream of discrete values, we can measure the *entropy* of this stream. This entropy serves as a one-number summary of its observed distribution. Monitoring changes in entropy over time might reveal traffic anomalies.

Calculating entropy requires memory proportional to the number of distinct values for the field. For streams with a large number of values, this cost is prohibitive, so efficient algorithms that accurately estimate entropy are required.

Joint Distributions and Mutual Information

Measuring joint distributions and the mutual information between field values allows us to monitor trends in co-occurence of feature values. If $I(A;B)$ is high, then A is a good predictor of B and vice versa. Changes in the predictability of one variable given another indicate anomaly.





Estimating Entropy and Mutual Information

Recent algorithms by Chakrabarti et al. [2], Lall et al. [6], and Bhuvanagiri and Ganguly [1] provide efficient ways of estimating Shannon entropy $H(A)$. Mutual information and conditional entropy can be efficiently estimated with an additive error, using the formulae $I(A;B) = H(A) + H(B) - H(A,B)$ and $H(B|A) = H(A,B) - H(A)$.

The **Lall et al.** algorithm: updates depend on previous history, but some value counts precise.

Preprocessing:

– Select a sequence of random locations in stream.

Online:

– Maintain counter for each location j.

– Counter tracks how many times a_j appears after location j.

Postprocessing:

– Process above data to get estimate of entropy.

Reduce error by running several copies in parallel.

The **Bhuvanagiri/Ganguly** algorithm: parallelizable, but no value counts precise.

Preprocessing:

– Generate several random hash functions for hash table.

– Generate hierarchy of random substreams.

Online:

– Use hash table to maintain frequency counts.

– Multiple hash tables reduce expected error.

– Random substreams enable better accuracy.

Postprocessing:

– Use estimates of frequency of each item to calculate estimate of entropy.

Chow-Liu Classifiers

Existing classifier implementations have been forced to make the strongest possible independence assumptions between features. If these assumptions are relaxed, the accuracy of the classifier can be increased, at the expense of additional memory and computation power. A reasonable trade-off between accuracy and computational complexity is to restrict the classifier to second-order dependencies, where each feature depends on at most one other feature.

Chow-Liu trees [3] provide an efficient method of determining nearly optimal second-order dependencies by comparing mutual information of feature pairs. We propose to build a second-order classifier by dynamically building Chow-Liu trees from estimated data.

References

1. Lakshminath Bhuvanagiri and Sumit Ganguly. Estimating Entropy over Data Streams. ESA 2006.
2. Amit Chakrabarti, Graham Cormode, and Andrew McGregor. A near-optimal algorithm for computing the entropy of a stream. SODA 2007.
3. C. K. Chow and C. N. Liu. Approximating Discrete Probability Distributions with Dependence Trees. Transactions on Information Theory 1968.
4. Guofei Gu, Prahlad Fogla, David Dagon, Wenke Lee, and Boris Skoric. Towards an information-theoretic framework for analyzing intrusion detection systems. ESORICS '06.
5. Anukool Lakhina, Mark Crovella, and Christophe Diot. Mining anomalies using traffic algorithms for estimating entropy of network traffic. SIGCOMM 2006.
6. Ashwin Lall, Vyas Sekar, Mitsunori Ogihara, Jun Xu, and Hui Zhang. Data streaming algorithms for estimating entropy of network traffic. SIGMETRICS 2006.

Conclusion

Measuring information-theoretic statistics of traffic has proved useful in detecting network anomalies; however, the more complex the statistic, the more memory is required. Recent advances in streaming algorithms enable us to *estimate* statistics, giving us accurate approximations statistics while using a reasonable amount of memory.