Characterizing Cyber Attacks through Variable Length Markov Models

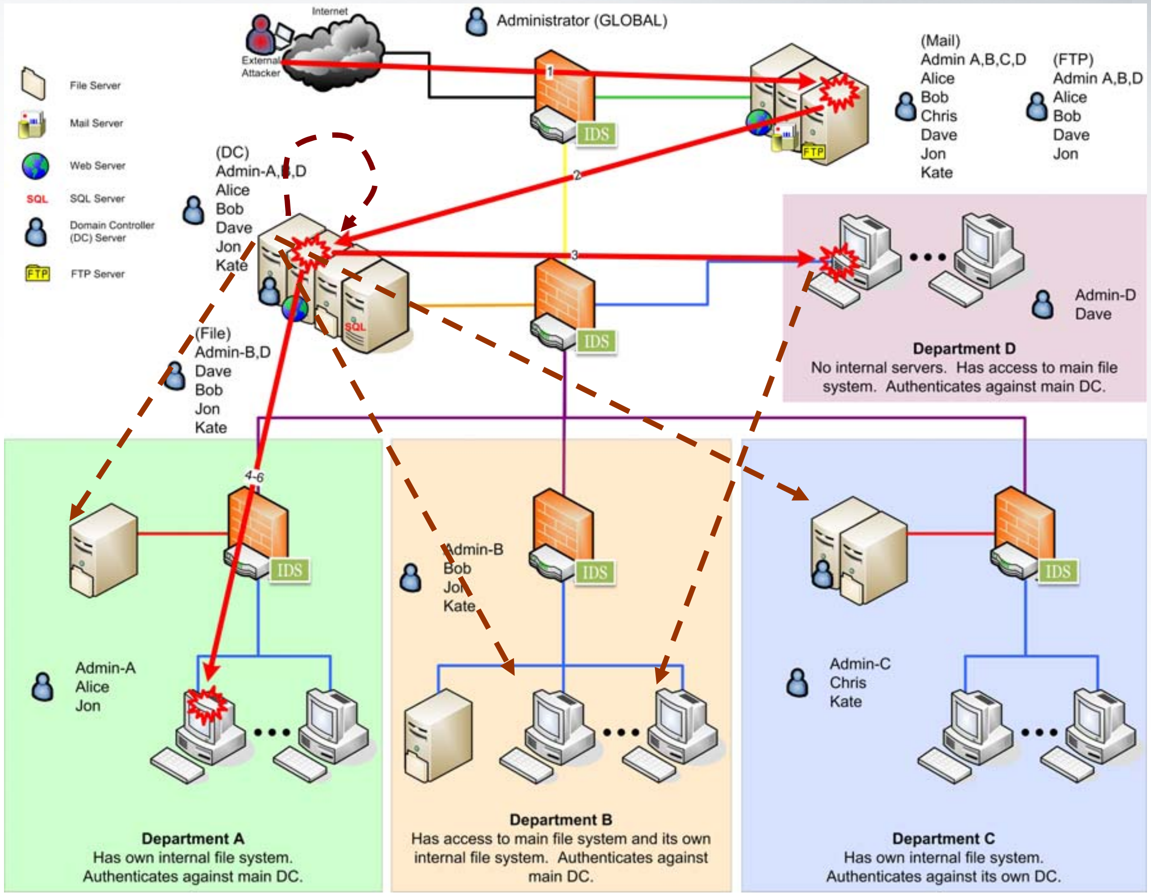
Dr. Shanchieh Jay Yang
and Daniel Fava

Department of Computer Engineering
Rochester Institute of Technology

National Center on Multi-source Information Fusion (NCMIF)
under technical direction of AFRL, Rome NY

Problem Statement

- Goal:
  Projecting next actions of multistage terrorist cyber attacks
- Objects:
  sequences of exploits
- Environment:
  cyber space
- Observables:
  - Intrusion Detection System (IDS) Alerts
  - Attack tracks (attack graphs) containing correlated alerts



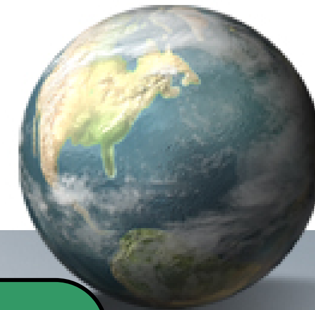
Why is it challenging?

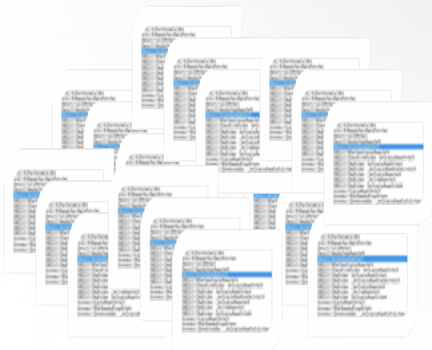
- Comparing to traditional attacks…

Missile attacks	Cyber attacks
Missile trajectory governed by laws of physics	Attack maneuvers in cyber space is governed by ???
Intention is to destroy	Intention can be for fun, to steal, to impair operations…
New missile technologies invented over years	New vulnerabilities and attack methods invented weekly or daily
Higher cost and harder to execute attacks -> fewer attacks	Low entry cost and cyber space is open -> more and often attacks

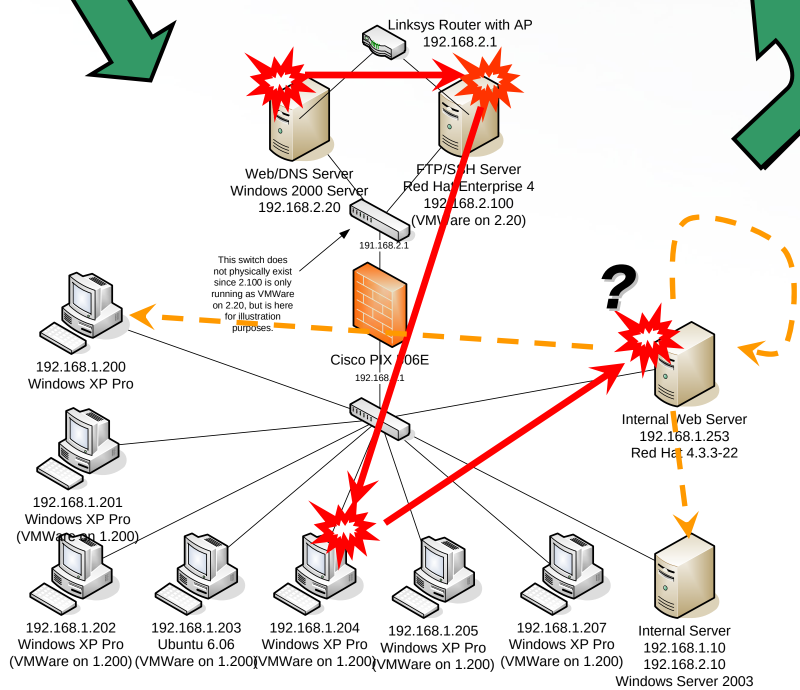
- So what do we do?

Approach: Terrain vs. Behavior





Behavior Models:
Behavior extraction using
Variable Length Markov Model,
Hidden Markov Model



Virtual Cyber Terrain:
Contextual reasoning
based on logical
connectivity, service,
vulnerability, impact

Behavior Analysis – How?

- Expert developed behavior model
  - E.g., guidance template, Bayesian Network
  - Diverse SME opinions (knowledge elicitation?)
  - Costly to maintain and update
- Attack tracks → time-stamp ordered sequences of symbols
- Context-based model
  - Adaptive Bayesian Network [Qin,Lee'04], Data Mining [Li etal.'07]
  - 0^{th}, 1^{st}, 2^{nd}, 3^{rd} order Markov Model
  - Variable-length Markov Model (VLMM)
  - Universal Predictor [Jacquet etal '02]
  - *Q*: What should be the context?
- State-based model
  - Hidden Markov Model (feasible?)

Translating Alerts

- <Alert>
  - <Description>ICMP PING NMAP</Description>
  - <Dest_IP>100.20.0.0</Dest_IP>
  - <Category>Recon_Scanning</Category>
- </Alert>
- <Alert>
  - <Description>SCAN SOCKS Proxy attempt</Description>
  - <Dest_IP>100.10.0.1</Dest_IP>
  - <Category>Recon_Scanning</Category>
- </Alert>
- <Alert>
  - <Description>WEB-IIS nsiislog.dll access</Description>
  - <Dest_IP>100.20.0.0</Dest_IP>
  - <Category>Intrusion_Root</Category>
- </Alert>

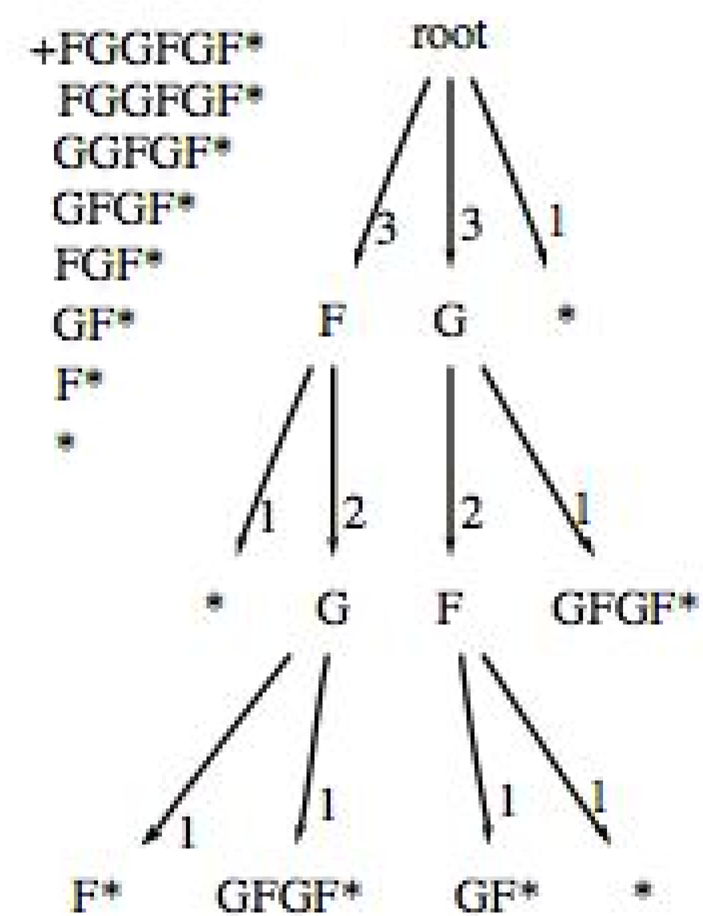
Category & target IP (Ω_t): AaB

Description (Ω_d): ABC

Category (Ω_c): AAB

Suffix Tree and Prediction

- +FGGFGF*
  - +: start of attack track
  - F: WEB-IIS nsiislog.dll access
  - G: WEB-MISC Invalid HTTP Version String
  - *: end of attack track
- What follows +GF?
  - -1th order: P=1/3
  - 0th order: P{G}=P{F}=3/7, P{*}=1/7
  - 1st order:
    P{G|F} = 2/3, P{*|F} = 1/3
  - 2nd order:
    P{G|GF} = 1/2, P{*|GF} = 1/2
  - VLMM – blending the estimates



Suffix tree from historical data

- Historical attack sequences builds suffix tree
- Suffix tree embeds patterns exhibited in finite-contexts
- Each unfolding attack sequence matches part of suffix tree for prediction

VLMM for prediction

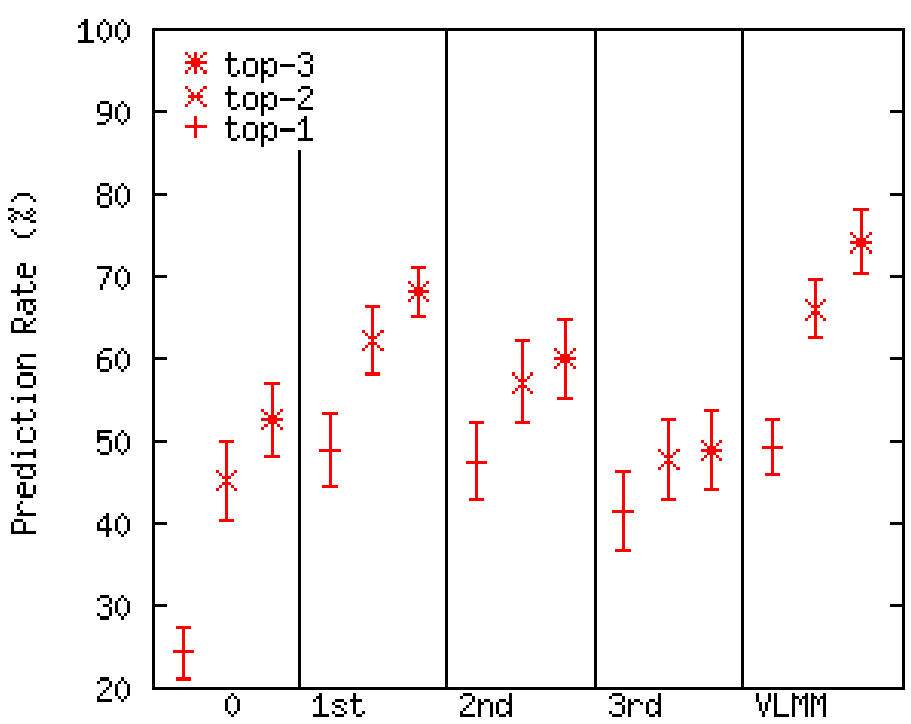
- Predict next action (x_{n+1}) given:
  - an unfolding sequence of attack: $s = \{x_1, x_2, \ldots, x_n\}$
  - a data-set containing representative attack tracks
- Example: FFGF?
- Procedure:
  - Create suffix tree from representative attack sequences
  - From suffix tree, find:
    - FFGF: $P_4\{X_5|X_1 = F, X_2 = F, X_3 = G, X_4 = F\}$,
    - FGF: $P_3\{X_5|X_2 = F, X_3 = G, X_4 = F\}$,
    - GF: $P_2\{X_5|X_3 = G, X_4 = F\}$,
    - F: $P_1\{X_5|X_4 = F\}$,
    - : $P_0\{X_5\}$, (frequency count)
    - : $P_{-1}\{X_5\}$, (1/alphabet size)
  - Blend P_m, $P_{m-1} \ldots P_{-1}$
    - $P(X) = \Sigma_{o=-1 \ldots m}\, w_o \cdot P_o$
    - $w_m = 1 - e_m$, $w_n = (1-e_n)\Pi_{i=n+1 \ldots m} e_i$
    - e_i: escape probability for context of length i

Experiment Setup

- Ground truth data generated via scripted attacks on a VMWare network
- A total of 1,113 attack sequences composed of 4,723 alerts after $\Delta t=1$ filtering [Valuer'04]
- 10 independent runs with random 50-50 splits of training vs. test data
- Alphabet choices:
  - Specific attack method (Ω_d)
  - Category of attack method (Ω_c)
  - Category + target IP (Ω_t)
- Top-*k* prediction rate (*k*=1, 2, 3):
  - % of correct prediction falls in the top-*k* choices

0 to 3rd Order and VLMM (Ω_d)

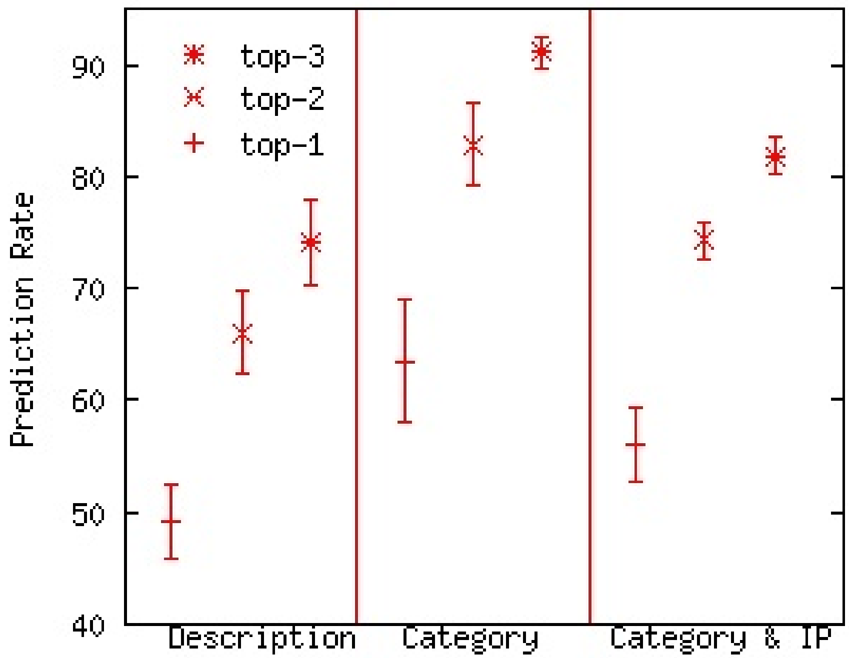
- Dominance of 1st order prediction
- VLMM combines n-order and offers better predictions
- Top 3 actions:
  - ICMP PING NMAP (43%), WEB-MISC Invalid HTTP Version String (22.4%), (http inspect) BARE BYTE UNICODE ENCODING (9.0%)
  - ICMP PING NMAP followed by ICMP PING NMAP 87.7% of the time
- Predicts better for repeating actions? Blending with longer context helps for predicting transitions?



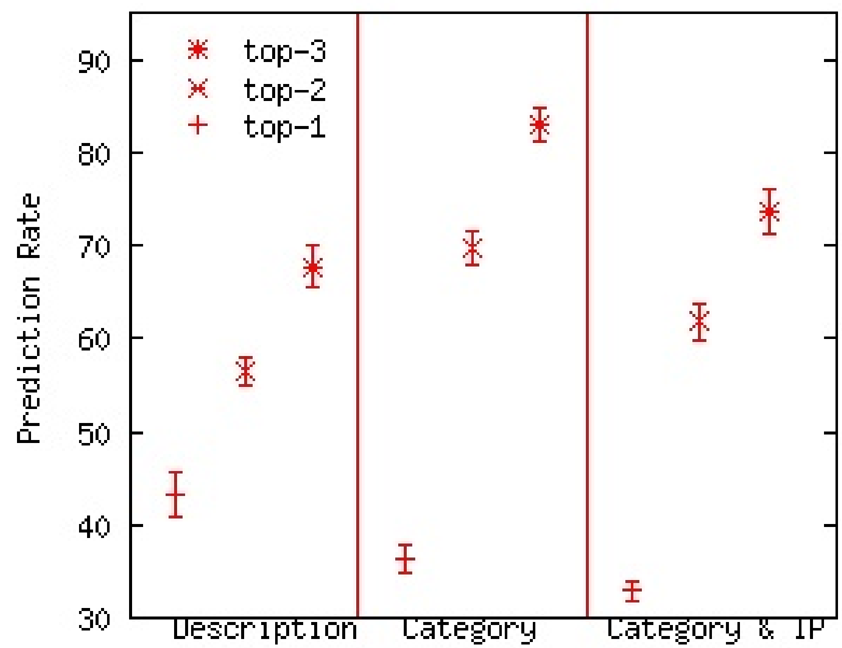
Prediction rate for transitions

- Predicting transitions will be better off by training with data sets with no repetition
- Predicting attack category is easier and more reaonable than predicting specific attack method

Trained with no repetition



Trained with repetition

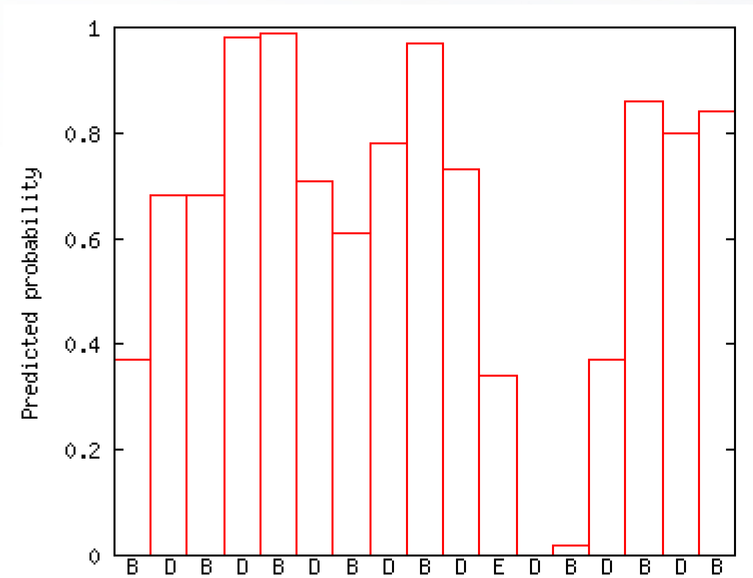


Some Observations

- Many repetitive attack actions
  - One attack action results in multiple alerts
  - No need to use an algorithm to predict repeating actions (exploit methods)
  - removing repetitive actions allows
    - Better capturing of transitions of attack actions
    - Smaller model size and faster algorithm execution
- More occurring actions predicted better …
- Except …
  - Signature in attack sequence

    `WEB-MISC bad HTTP/1.1 request, Potentially worm attack' always followed by `MISC OpenSSL Worm traffic'
  - Overshadowing

    High frequency actions are overshadowed by even higher freq. ones



Entropy of Predictions?

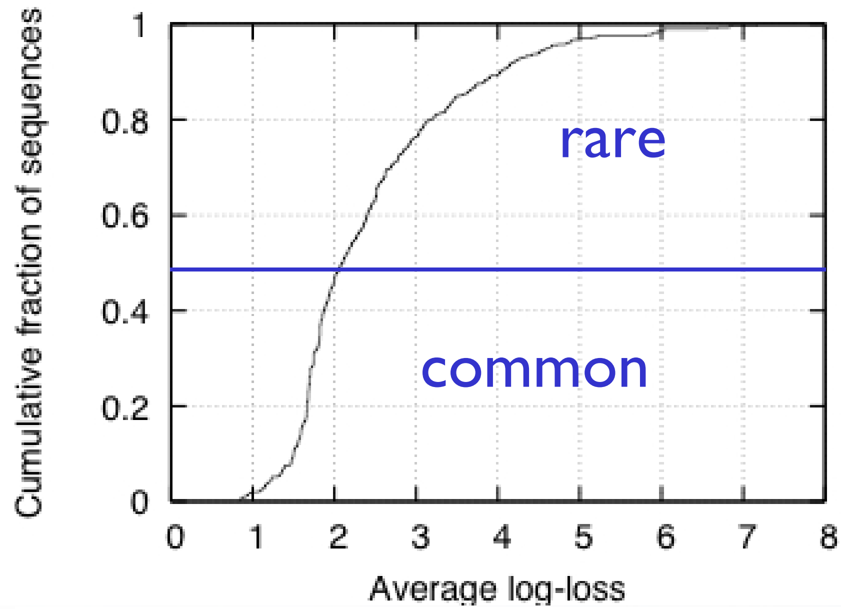
- Intuition:
  Uncertainty/variability → higher entropy for mis-predicted actions

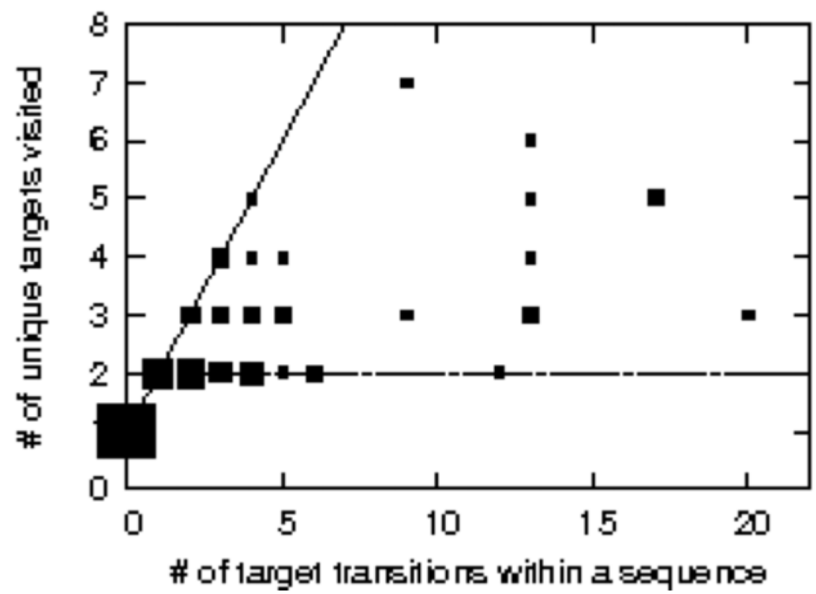
	Repetition	Category	Category & IP	Description
Correct	No	0.62 ± 0.48	0.91 ± 0.60	1.07 ± 0.69
Mis-pred.	No	0.93 ± 0.63	1.41 ± .81	1.35 ± 0.71
Correct	Yes	0.52 ± 0.51	0.58 ± 0.57	0.58 ± 0.68
Mis-pred.	Yes	0.88 ± 0.63	1.04 ± 0.75	1.23 ± 0.92

- Higher entropy for
  - Mis-predicted, finer granularity of Ω, and no-repetition set,
- Large standard deviation – entropy is not that indicative?!

Classification?

- Can we categorize cyber attack types (with no ground truth)?
- Average Log-loss:
  - Rarity of attack sequence
  - Threshold=2.0 (Ω_c, no repetition)
    - O.83 vs. 0.69 prediction rates
- # target trans vs. # targets visited:
  - Agility of attack
  - Most targets suffered 2 scans
  - Most popular targets: 1,735 and 814 out of a total of 4,723
  - Are more agile attacks harder to predict?





Conclusion

- A new theoretical and real-world problem
  - Finite sequences (and can be short)
  - Diverse and changing behavior (in terms of exploitation methods & transitions)
  - Noisy (intentional & unintentional)
- Context-based (VLMM) prediction:
  - Combine longer with shorter contexts helps
  - Training with no-repetition helps to extract attack transition behavior
  - Suffix tree embed diverse behavior and potential for real-time implementation
- Future work
  - Complex objects instead of simple symbols?
  - Classification for better prediction?
  - Prediction of rare and high-impact events?