Spam Detection in Voice-over-IP Calls through Semi-Supervised Clustering

Yu-Sung Wu, **Saurabh Bagchi**
Purdue University, USA

Navjot Singh
Avaya Labs, USA

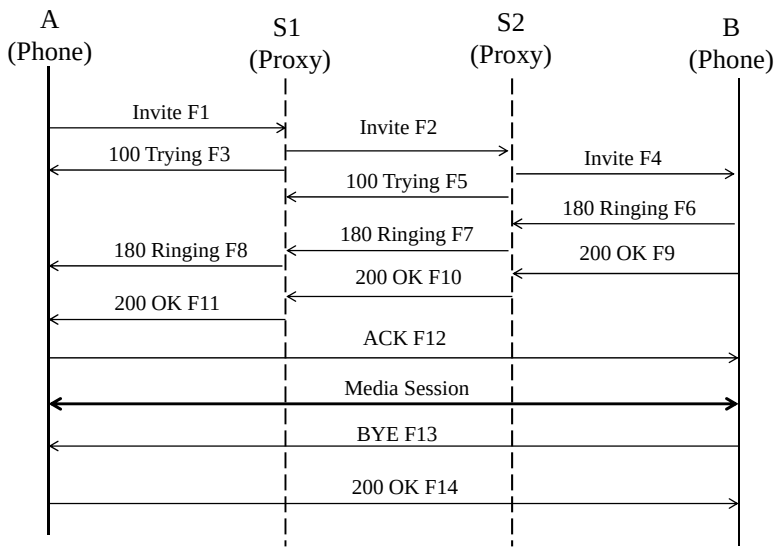
Ratsameetip Wita
Chulalongkorn University, Thailand



Voice-over-IP (VoIP) Overview

- Session Initiation Protocol (SIP) or H.323 for signaling
- Real-time Transport Protocol (RTP) for media
- Media flow happens after a successful call setup, which is achieved through signaling
- Real-time Transport Protocol (RTCP) for feedback
- Other supporting protocols: DNS, DHCP, ICMP

Sample Call Flow in VoIP



Outline

1. VoIP Overview
2. **Challenges in VoIP Spam Detection**
3. System Architecture
4. Semi-supervised Clustering
5. Efficient Clustering for Spam Detection: e-MPCK-Means, p-MPCK-Means
6. Call Trace and Experiments
7. Conclusions

Spam Calls in VoIP Systems

- SPam over Internet Telephony (SPIT)
- Unsolicited and unwanted phone calls from (malicious) parties
  - Telemarketing calls
  - Harassing calls
  - Survey / polling calls
- Why is this a growing phenomenon?
  - VoIP calls are cheap to make
  - SPIT is very easy to automate
- Comparison with e-mail spam:
  - Motives and impacts are analogous
  - But, more disruptively, a VoIP spam intrudes in real-time

PURDUE UNIVERSITY

Challenges for Dealing with VoIP Spam

- A spam call in many ways appears like a normal (non-SPIT) call
  - Both follow the same protocols (SIP, H.323, RTP, RTCP)
  - No malformed packets
  - No exploitation of protocol vulnerabilities
  - Existing NIDS systems (Snort, SCIDIVE[1],…) do not apply
- VoIP is a real-time system
  - Before you pick up the call, can you tell if it's going to be a spam call?

[1] Y-S. Wu, S. Bagchi, S. Garg, N. Singh, T. Tsai, "SCIDIVE: A Stateful and Cross Protocol Intrusion Detection Architecture for Voice-over-IP Environments," DSN 05, pp. 401-410.

PURDUE UNIVERSITY

Challenges for Dealing with VoIP Spam

- VoIP system is a dynamic environment
  - Call duration, call frequency, the words you say, … can all be changing from one deployment to another
  - Different persons have different perspectives on what constitute a spit call
    - Some might be interested in buying merchandise from telemarketers while they do dislike other harassing phone calls.
  - Therefore, fixed threshold-based rules for detection are not suitable for filtering spam calls

PURDUE UNIVERSITY

Contribution

- Identify features from a VoIP call for spam detection
- Clustering of VoIP calls to identify spam calls
- Use of user-feedback and semi-supervised clustering technique to differentiate between spam and legitimate calls
- Adapting the original MPCK-Means[2] algorithm into:
  - eMPCK-Means : A *O(N)* algorithm for clustering a batch of VoIP calls
  - pMPCK-Means : A real-time algorithm for detecting VoIP spam

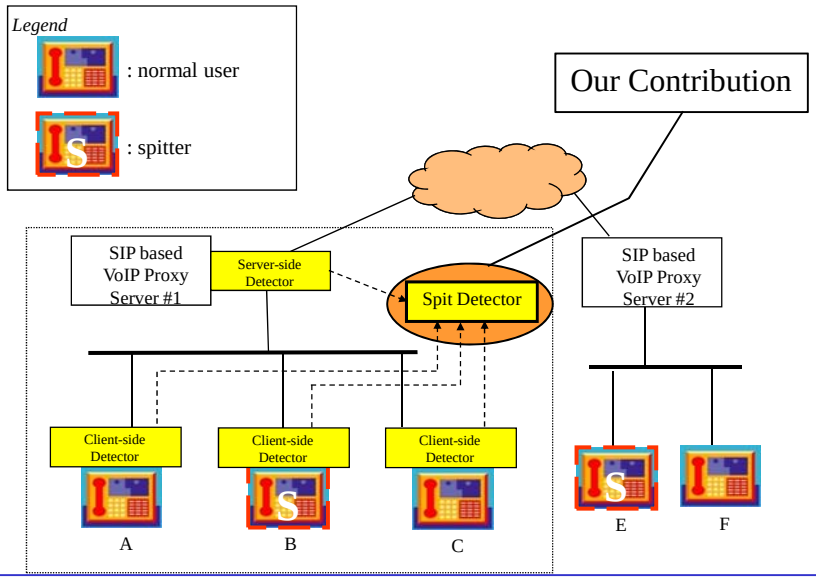
[2] M. Bilenko, S. Basu, and R. J. Mooney, "Integrating constraints and metric learning in semi-supervised clustering," in *ICML*, 2004, pp. 81-88.

PURDUE UNIVERSITY

Outline

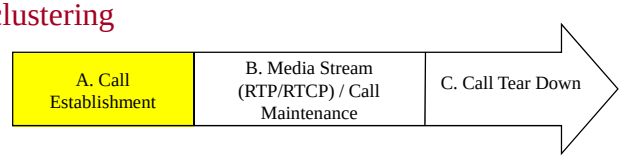
1. VoIP Overview
2. Challenges in VoIP Spam Detection
3. **System Architecture**
4. Semi-supervised Clustering
5. Efficient Clustering for Spam Detection: e-MPCK-Means, p-MPCK-Means
6. Call Trace and Experiments
7. Conclusions

System Architecture



VoIP Call Features

17 call features extracted from VoIP signaling and media traffic used here for clustering



1-2. From/To URI
3. Start time
4. Duration
5. # of SIP INVITE messages
6. # of SIP ACK messages
7-8. # of SIP BYE messages from caller/callee
9. Time since the last call from the originator of the current call
10-15. # of 1xx, 2xx, 3xx, 4xx, 5xx, and 6xx SIP response messages
16. Call frequency of the originator of the current call
17. Ratio of non-silence duration of the callee to the caller media streams

Outline

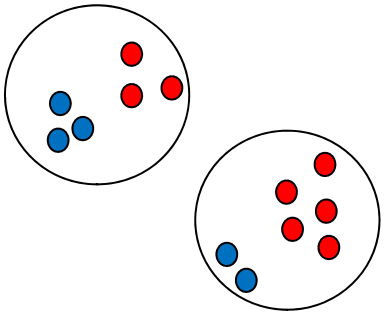
1. VoIP Overview
2. Challenges in VoIP Spam Detection
3. System Architecture
4. **Semi-supervised Clustering**
5. Efficient Clustering for Spam Detection: eMPCK-Means, pMPCK-Means
6. Call Trace and Experiments
7. Conclusions

Basic Clustering

- Objective: Cluster calls into legitimate and spam calls
- Classic K-Means clustering

$$\sum_{j=1}^{K} \sum_{x_i \in X_j} \left\| x_i - \mu_i \right\|^2 \quad \text{is minimized}$$

- Objective function puts weight on each feature evenly
- However, there may be only a few call features that can distinguish between the different clusters
- Putting equal weight on all the selected features can drown out the influence of these distinguishing features



Semi-supervised clustering

- MPCK-Means
  - Distance from centroids (reweighted by *A* matrix)
  - Cost from violating must-link constraints (*pairs of data points which should be put in the same cluster*)
  - Cost from violating cannot-link constraints (*pairs of data points which should be put in different clusters*)

How to Update *A* matrix

- The A matrix A_h for cluster *h* is acquired by solving $\frac{\partial \tau_{\text{mpckm}}}{\partial A_h} = 0$

- Covariance of data points in cluster *h*
- Cost from violating must-link constraints related to cluster *h*
- Cost from violating cannot-link constraints related to cluster *h*

$$A_h = |X_h| \left(\sum_{x_i \in X_h} (x_i - \mu_h)(x_i - \mu_h)^T \right.$$

$$+ \sum_{(x_i, x_j) \in M_h} \frac{1}{2} w_{ij} (x_i - x_j)(x_i - x_j)^T 1[l_i \neq l_j]$$

$$+ \sum_{(x_i, x_j) \in C_h} \left(\overline{w_{ij}} (x_h' - x_h'')(x_h' - x_h'')^T \right.$$

$$\left. \left. - (x_i - x_j)(x_i - x_j)^T 1[l_i = l_j] \right) \right)^{-1}$$

Outline

1. VoIP Overview
2. Challenges in VoIP Spam Detection
3. System Architecture
4. Semi-supervised Clustering
5. **Efficient Clustering for Spam Detection: e-MPCK-Means, p-MPCK-Means**
6. Call Trace and Experiments
7. Conclusions

Our Contribution: eMPCK-Means

- Batch mode of operation
- Improvement in runtime:
  - A *O(N)* approximation version of MPCK-Means
    - MPCK-Means is $O(N^3)$
  - *O(N)* complexity cluster initialization
    - Skip the pair-wise constraints => $O(N^2)$
    - Use the set of flagged spam calls, flagged legitimate calls, and the set of the rest of calls directly for cluster initialization
  - Efficient estimation of maximally separated points
    - Embed the estimation in the distance calculation
  - Use a constant number of constraints in cluster assignment step
    - Experiment results from [2] suggest that MPCK-Means can work reasonably well with only a few constraints

Our Contribution: eMPCK-Means

- Improvement in clustering quality:
  - Pre metrics update on the starting cluster(s)
    - Update *A* matrix once before entering the main-loop of MPCK-Means
    - Results in an initial *A* matrix which reflects the user feedback information better
    - In comparison, an identity matrix is used as the initial *A* matrix in MPCK-Means

pMPCK-Means

- For real-time spam detection: Hang up a suspect call even before media flow starts
- Only allowed to use features available at call establishment phase
  - From URI, To URI, Start time, and Time since the last call from the originator of the current call
- For most of the time, each new data point (an incoming call) only involves a cluster assignment operation
  - *O(1)* complexity
- Occasionally, eMPCK-Means is invoked to recondition the clustering
  - Re-compute the clusters, *A* matrix, etc.
  - Can be carried out in an asynchronous manner in the background

eMPCK-Means (multi-class)

- With MPCK-Means, eMPCK-Means, and pMPCK-Means, we create only two clusters:
  - Cluster of spam calls and cluster of legitimate calls
  - Because user feedback only provides a binary predicate on whether a call is spam / legitimate
- eMPCK-Means (multi-class)
  - Use of expert knowledge to differentiate different types of calls
  - Split each cluster (spam or legitimate) into three sub-clusters based on call types:
    - Calls going to voice mail box
    - Calls terminated by the user immediately after the call is established
    - The remaining types of calls

Outline

1. VoIP Overview
2. Challenges in VoIP Spam Detection
3. System Architecture
4. Semi-supervised Clustering
5. Efficient Clustering for Spam Detection: e-MPCK-Means, p-MPCK-Means
6. **Call Trace and Experiments**
7. Conclusions

PURDUE UNIVERSITY

Call Traces for Experiments

Name	Legitimate Call Length	Legitimate Call Inter-arrival time	Spam Call Length	Spam Call Inter-arrival time	Total # of Legitimate Calls	Total # of Spam Calls
v4	5	30	1	2	171	212
v5	5	10	1	10	338	45
v6	5	30	1	10	289	94
v7	5	30	5	10	302	81

Common characteristics for spam calls:

- There are 6 spitters in the system
- 10% chance of a call being hung up by the caller
- Non-silence period in media stream is dominated by the spitter

Common characteristics for legitimate calls:

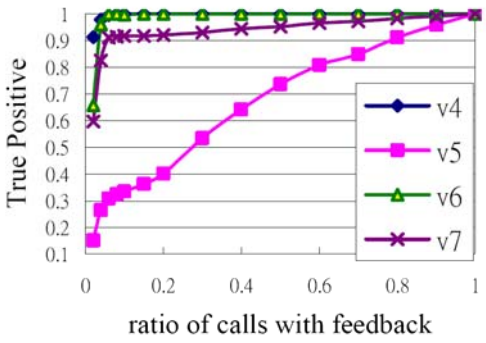
- There are 90 legitimate users in the system
- 60% chance of a call being hung up by the caller

PURDUE UNIVERSITY

Experiment: Effect of user feedback

eMPCK True Positive Rate across call traces

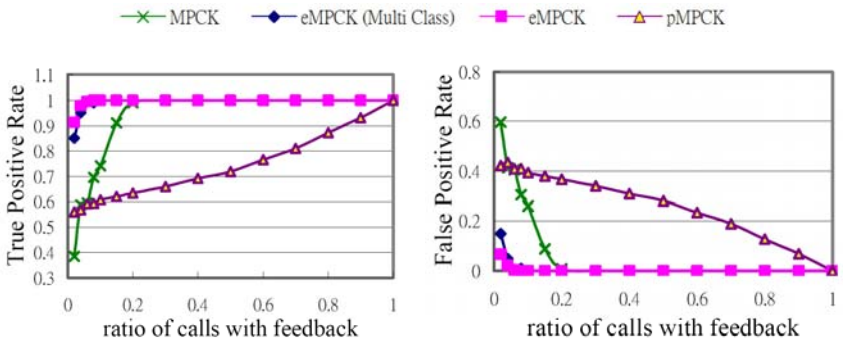
True Positive: (# of actual spam calls detected) / (# of detected calls)



v4 is the easiest, followed closely by v6, and then v7. v5 is the hardest.

Experiment: Effect of user feedback

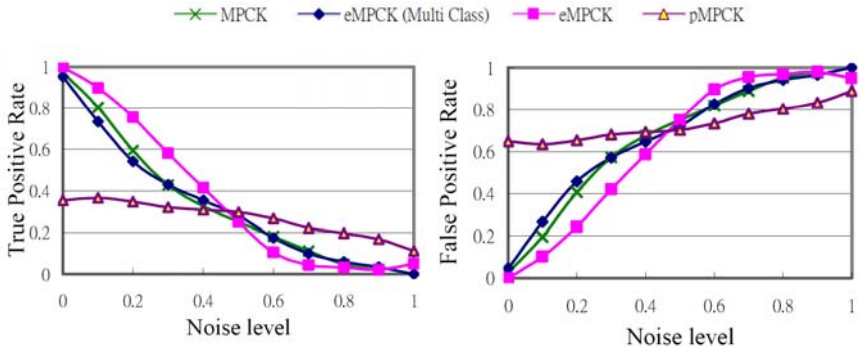
Comparing 4 algorithms (use call trace v4)



- Pre metric update boosting improves the performance in eMPCK
- A small amount of user feedback is enough to make the detection accurate enough

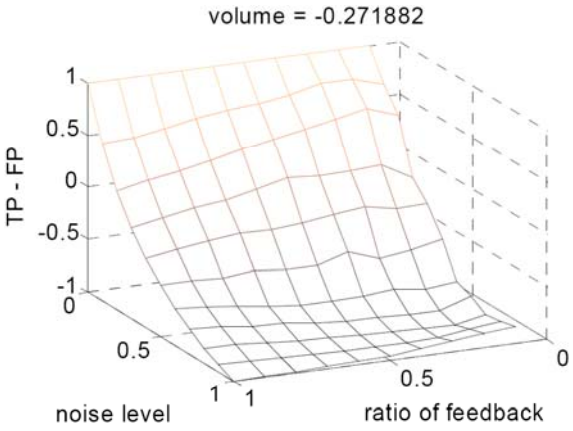
Experiment: Noise in user feedback

Call trace 6, user feedback fixed at 0.3



- pMPCK is not really usable
- The others work with low noise level
- The use of pre-metric update hurts the performance of eMPCK when noise level is past 0.5 (the majority of user feedback is inaccurate)

Experiment: Quality and quantity of user feedback



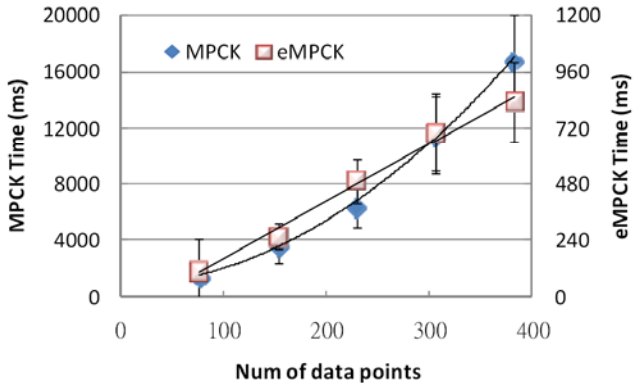
eMPCK (TP-FP) for call trace v6

$$Volume = \int_{n=0}^{1}\int_{f=0.1}^{1} (TP - FP) \cdot df \cdot dn$$

n: noise level, f: feedback ratio

TP-FP	v6	avg. across all traces
MPCK	-0.319	-0.314
eMPCK (Multi Class)	-0.330	-0.314
eMPCK	-0.272	-0.287
pMPCK	-0.371	-0.341

Experiment: Scalability



Call trace 7 is used for this experiment.

- eMPCK is at least 15X faster than MPCK
- eMPCK exhibits linear time complexity

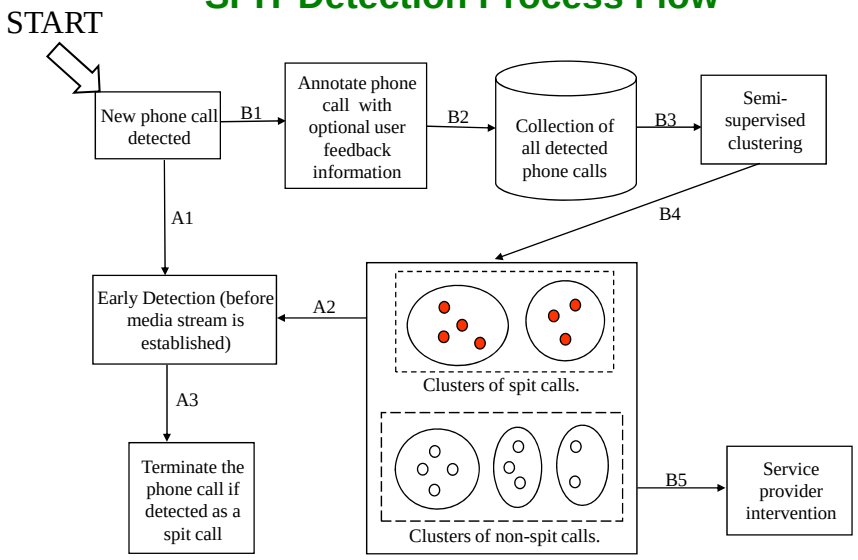
Outline

1. VoIP Overview
2. Challenges in VoIP Spam Detection
3. System Architecture
4. Semi-supervised Clustering
5. Efficient Clustering for Spam Detection: e-MPCK-Means, p-MPCK-Means
6. Call Trace and Experiments
7. **Conclusions**

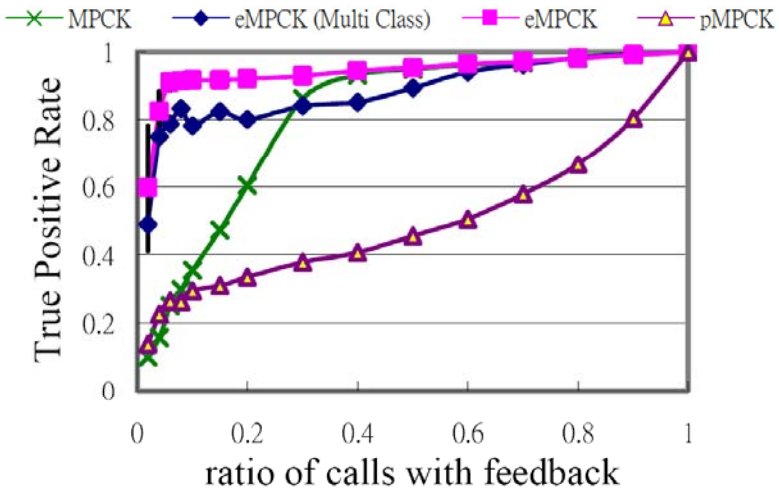
Conclusion

- Propose a solution to detect VoIP spam
- Our solution is built upon semi-supervised clustering
  - Able to adapt to different environments and needs
- Come up with scalable algorithm for batch detection of VoIP spam
  - Useful and practical for service provider
- Detect VoIP spam in real-time is hard
  - pMPCK-Means is barely usable due to the limited available features during call establishment
- Future Work
  - Better real-time detection
  - Sharing signatures of spam calls across ISPs

SPIT Detection Process Flow



Experiment / Effect of user feedback



Call trace v7 / True Positive rate