Data-driven models of reputation in cyber-security

Pierre Lison
Norwegian Computing Center (NR)

Second Workshop on Text Analytics for Cybersecurity and Online Safety (TA-COS)

11th International Conference on Language Resources and Evaluation (LREC 2018)

12/05/2018

Introduction

- Blacklists and whitelists (= **reputation lists**) often employed to filter network traffic
- Manually curated by security experts

Introduction

- Shortcomings of blacklists and whitelists:
  - Slow reaction time
  - Maintenance is difficult and time-consuming
  - Limited coverage
  - Static (can be circumvented through techniques such domain flux and fast flux networks)

Introduction

Can we use **machine learning** to automatically predict the reputation of end-point hosts?

1. Predictions in real-time, without human intervention
2. Less vulnerable to human errors and omissions
3. Full coverage of end-point hosts

Introduction

Can we use **machine learning** to automatically predict the reputation of end-point hosts?

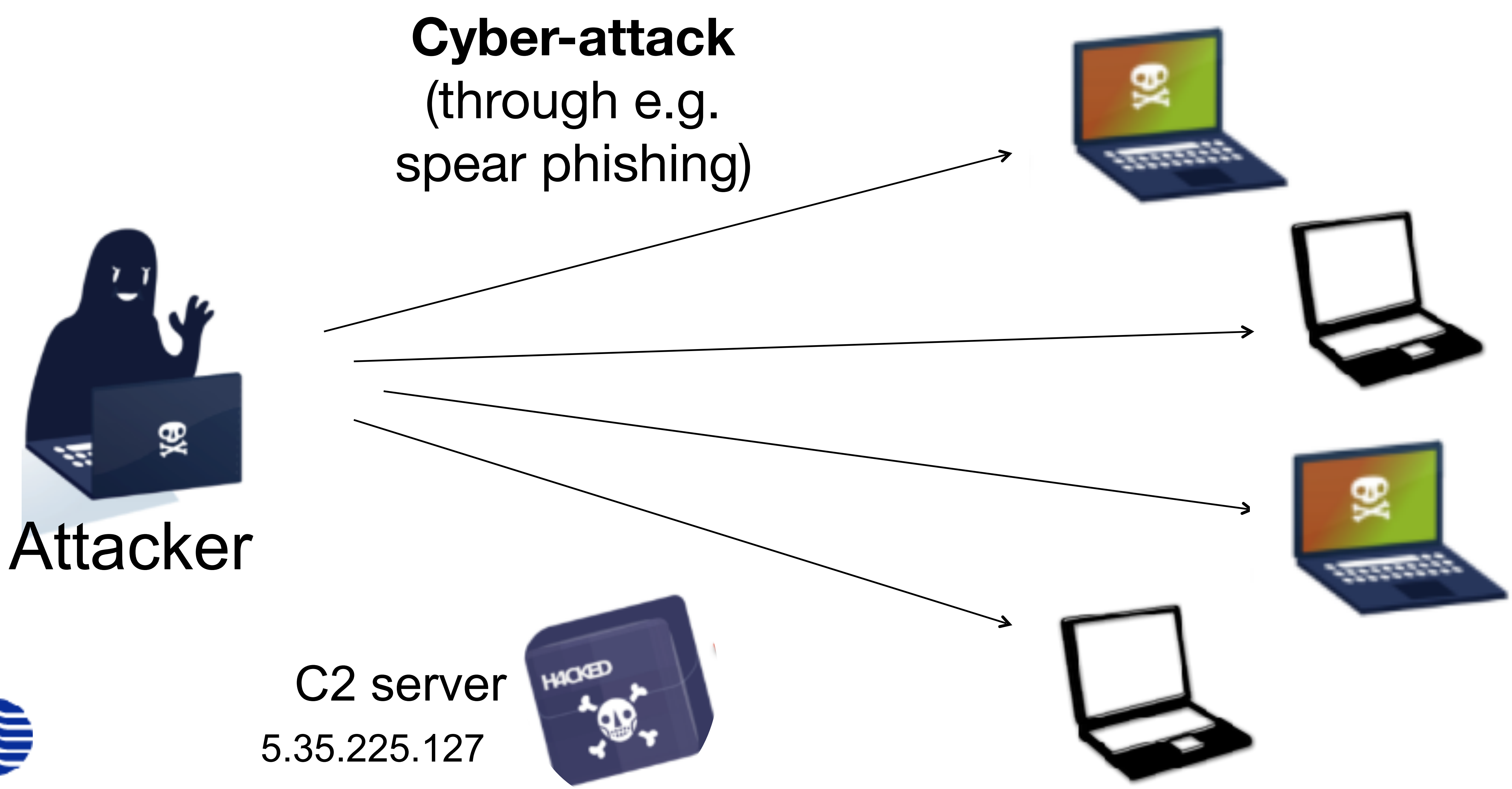
Detecting domain names generated by malware with RNNs

Predicting the reputation of domains and IP addresses from passive DNS data

Part 1: Detecting domain names generated by malware

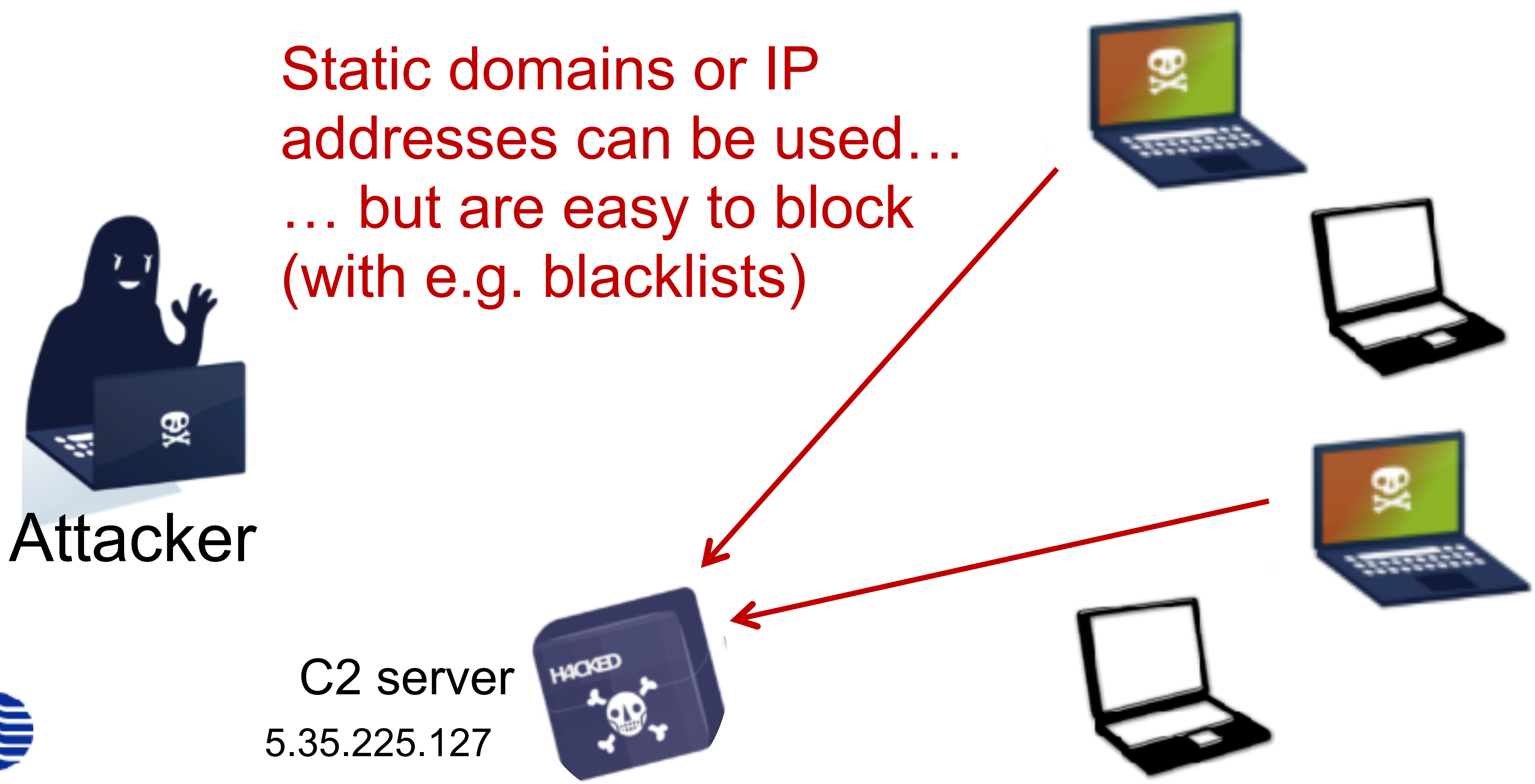
Domain-generating malware

- Most malware must connect compromised machines with a *command and control* (C2) server for their operations



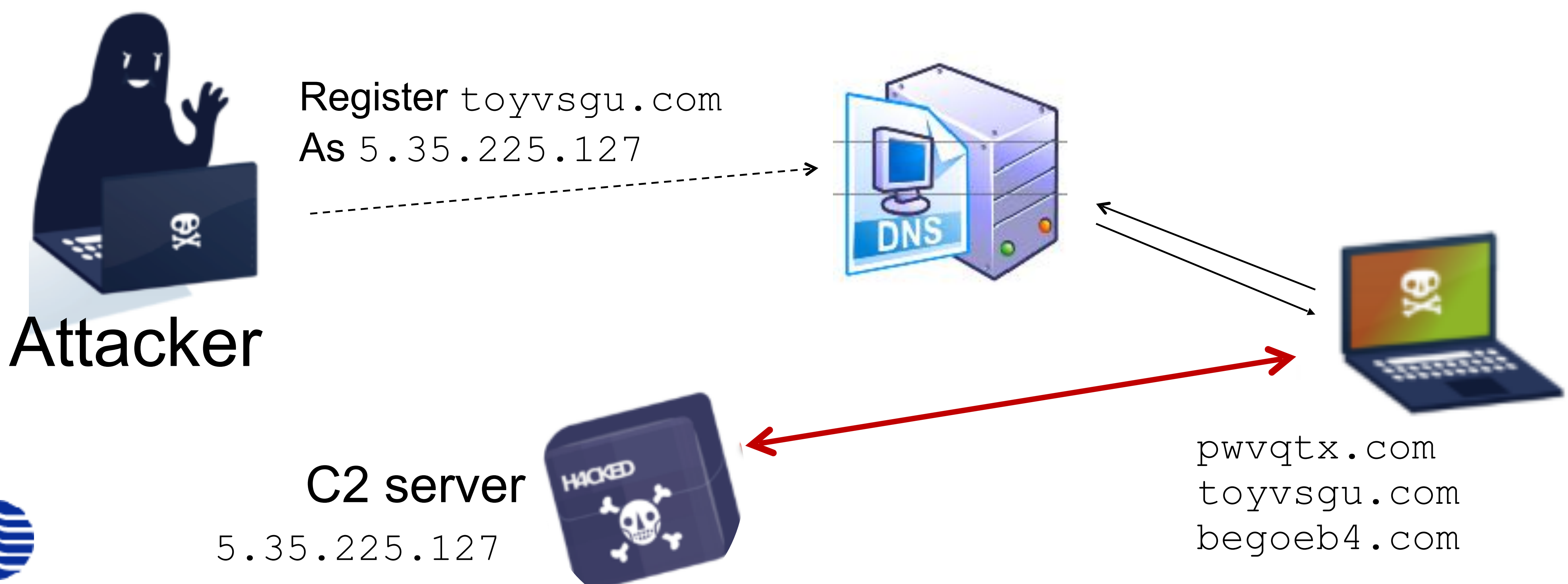
Domain-generating malware

- Most malware must connect compromised machines with a *command and control* (C2) server for their operations



Domain-generating malware

- With domain-generation algorithms (DGA), compromised machines will attempt to connect to a large number of pseudo-random domain names…
- The attacker can then simply register a few of these artificial domains to establish a rendez-vous point



Domain-generating algorithms (DGAs)

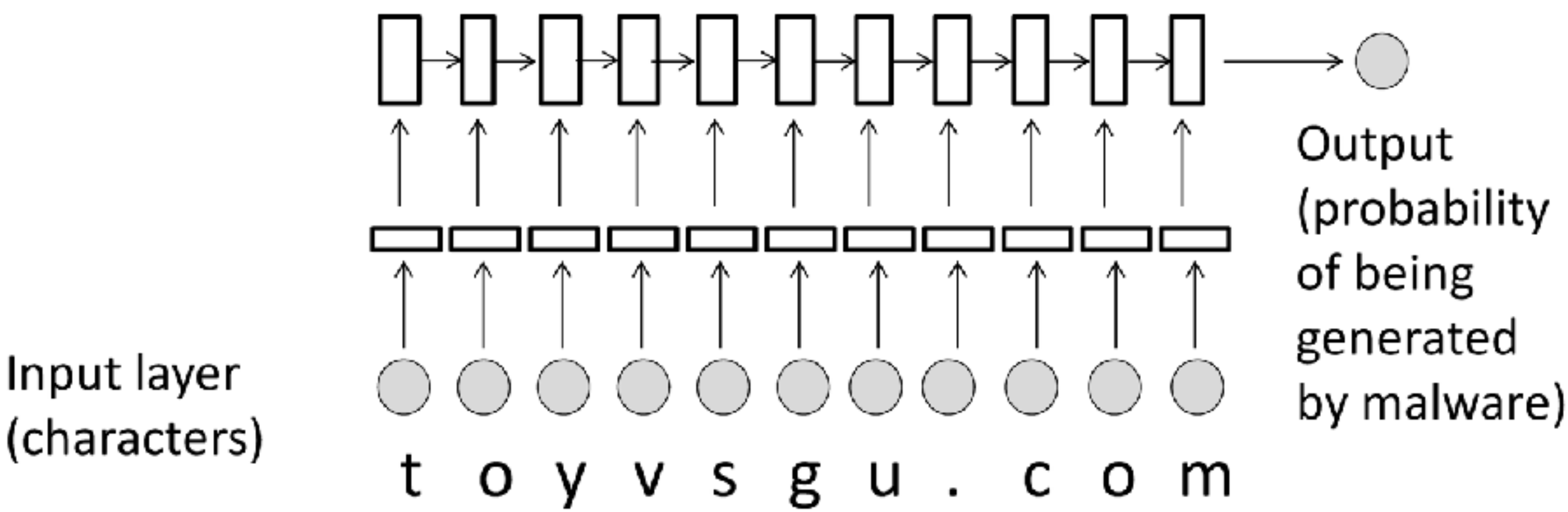
- DGAs increasingly popular as command-and-control (C2) rendez-vous mechanism in botnets
  - First observed in the Kraken botnet (2008)
- DGAs generate a large number of seemingly random domain names based on a *shared secret* (**seed**)
- Highly *asymmetric* situation:
  - Malicious actors only need to register a single domain to establish a C2 communication channel
  - But security professionals must control the full range of potential domains to contain the threat (*counter-measures*: pre-registering, blacklists, or sinkholes)

Taxonomy of DGAS

- **Time dependence:**
  - Are the seeds fixed or are they only valid for a specific period (by including a time source in their calculation?)
- **Determinism:**
  - Are the seeds computed through a deterministic procedure, or do they include unpredictable factors (weather forecasts, stock markets prices, etc.)
- **Generation scheme:**
  - How are the domains generated from the seeds? Popular techniques include alphanumeric combinations, hash-based techniques, wordlists and permutations.

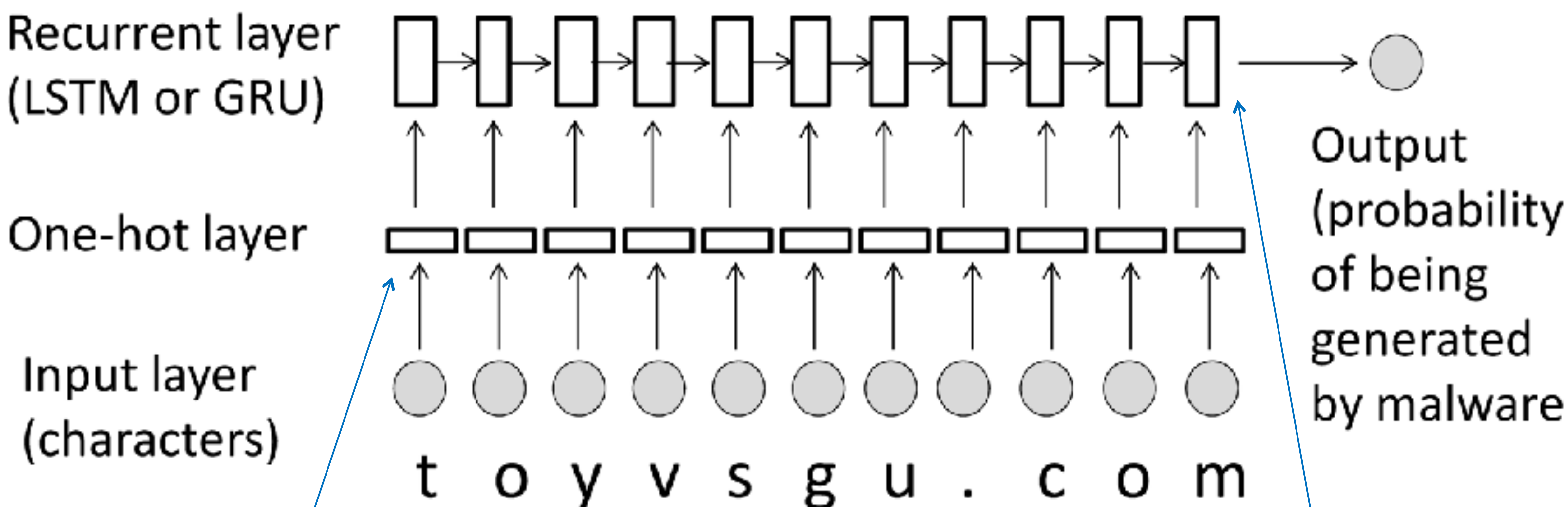
Detection of DGAs

- **Recurrent neural network** trained on a large dataset of benign & malicious domains
  - Ability to learn complex sequential patterns
- Purely data-driven – easy to apply and update



Architecture

Recurrent layer builds up a representation of the character sequence as a dense vector



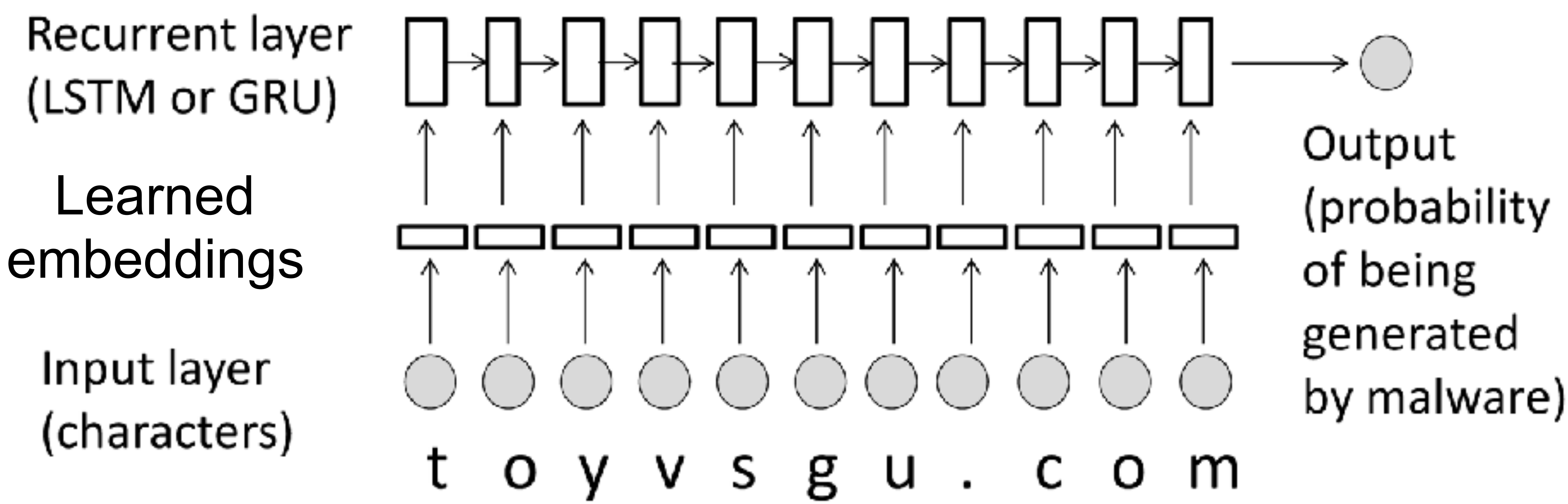
First layer encode each character as a "one-hot" vector

Domain name is fed to the neural network character by character

Final vector is used to predict whether the domain is DGA

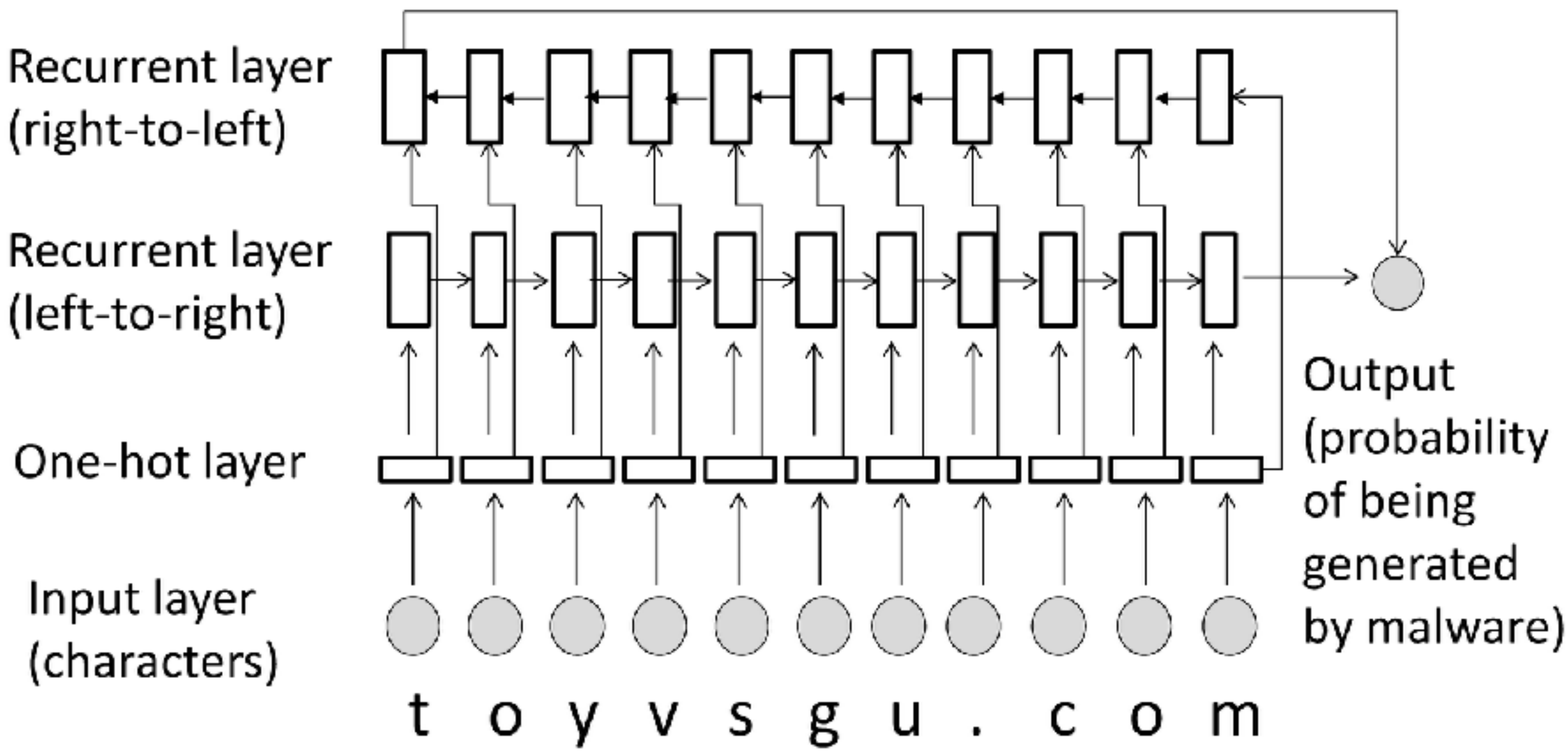
Extensions

- **Embeddings**
- Bidirectionality
- Hidden layer
- Multi-task learning



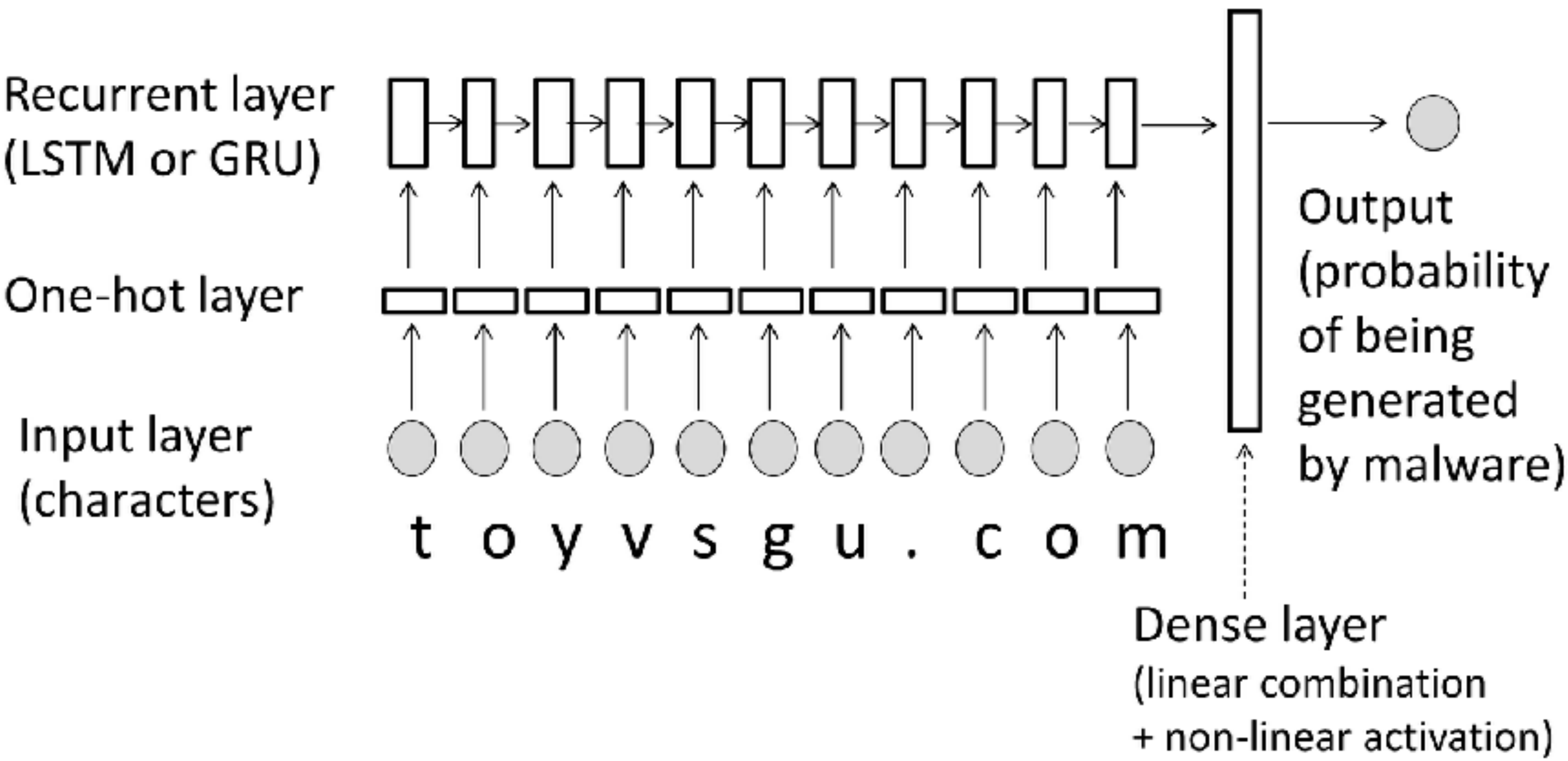
Extensions

- Embeddings
- **Bidirectionality**
- Hidden layer
- Multi-task learning



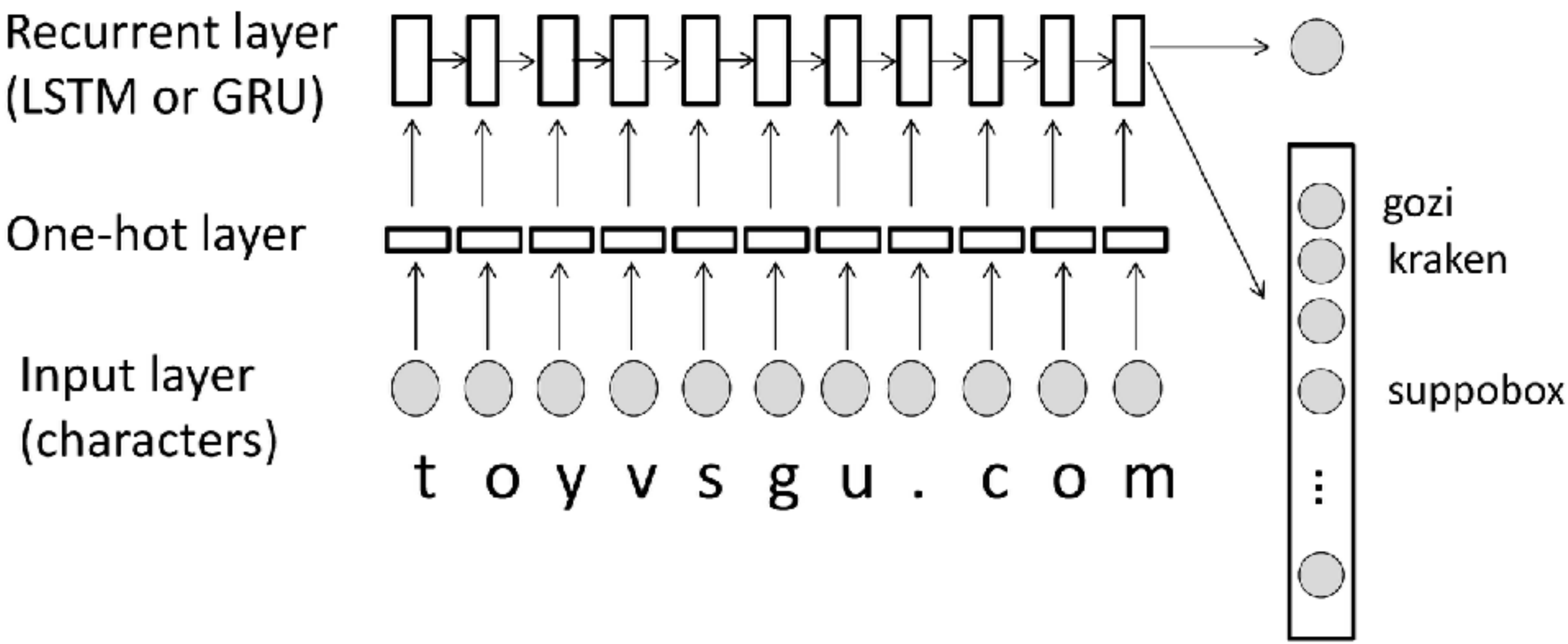
Extensions

- Embeddings
- Bidirectionality
- **Hidden layer**
- Multi-task learning



Extensions

- Embeddings
- Bidirectionality
- Hidden layer
- **Multi-task learning**



Data

- The parameters of the neural model must be estimated from training data
- **Negative examples** (benign domains):
  - Snapshots from the Alexa top 1 million domains
  - Total: over 4 million domains
- **Positive examples** (malware DGAs)
  - DGA lists from the DGArchive (63 types of malware)
  - Feeds from Bambenek Consulting
  - Domain generators for 11 DGAs
  - Total: 2.9 million domains

Data

Malware	Frequency
bamital	40 240
banjori	89 984
bedep	15 176
beebone	420
blackhole	732
bobax	19 288
conficker	400 000
corebot	50 240
cryptolocker	55 984
cryptowall	94
dircrypt	11 110
dnschanger	40 000
downloader	60
dyre	47 998
ekforward	1 460
emotet	40 576
feodo	192
fobber	2 600
gameover	80 000
gameover_p2p	41 000
gozi	105 631
hesperbot	370
locky	179 204
madmax	192
matsnu	12 714
modpack	52
murofet	53 260
murofet_w	40 000
necur	40 000
necurs	36 864
nymaim	186 653
oderoor	3 833
padcrypt	35 616
proslikefan	75 270
pushdo	176 770
pushdotid	6 000
pykspa	424 215
pykspa2	24 322
qadars	40 400
qakbot	90 000
ramdo	15 984
ramnit	90 000
ranbyu	40 000
ranbyus	12 720
rovnix	40 000
shifu	4 662
simda	38 421
sisron	5 936
suppobox	41 014
sutra	9 882
symmi	40 064
szribi	16 007
tempedreve	453
tinba	80 000
torpig	40 000
tsifiri	59
urlzone	34 536
vawtrak	1 050
virut	400 600
volatilecedar	1 494
xxhex	4400
Total	2 925 168

Evaluation

- 10-fold cross validation on the full dataset
- **Baseline**: logistic regression on character bigrams
  - Toyvsgu.com → (to, oy, yv, vs, sg, gu, u., .c, co, om)
- Metrics: accuracy, precision, recall, F_1 score

$$\text{precision} = \frac{\text{\# correctly classified malware domains}}{\text{\# domains classified as malware by model}}$$

$$\text{recall} = \frac{\text{\# correctly classified malware domains}}{\text{\# actual known malware domains}}$$

$$F_1 \text{ score} = 2\frac{p \times r}{p + r}$$ (harmonic mean of the two)

Model selection

- The use of embeddings, bidirectional layers, and additional hidden layers did not improve the performance
- Multi-task learning (i.e. simultaneously learning to detect DGAs and to classify them) yielded the same results as networks optimised for these two tasks separately
  - The two tasks can use a shared latent representation
- The recurrent layer used GRU units with dimension=512
- Model trained on GPU with a batch size of 256, two passes and RMSProp as optimisation algorithm

Results

Area Under the Curve (AUC) of the ROC curve (see next slide)

- Detection

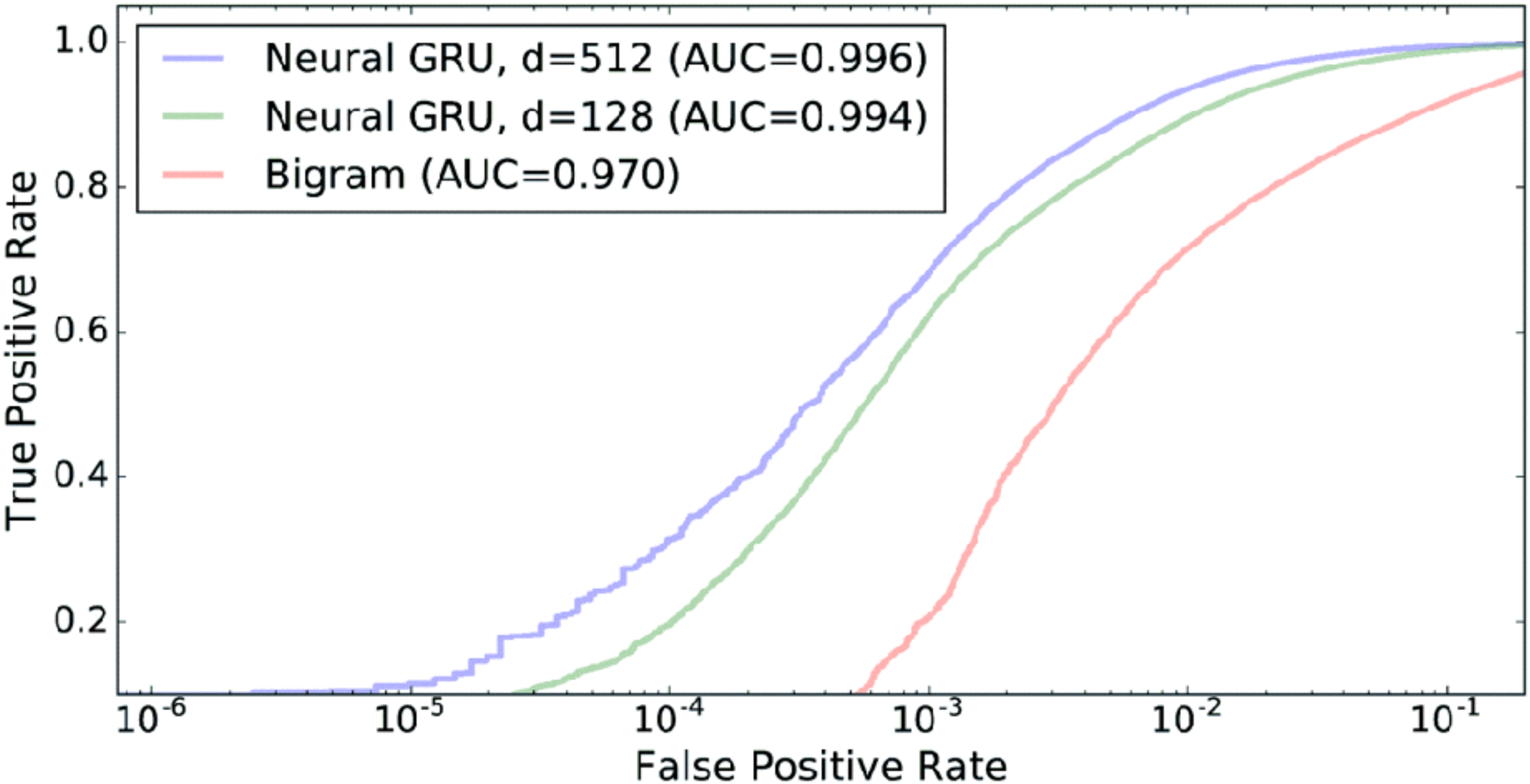
	Accuracy	Precision	Recall	F_1 score	ROC AUC
Bigram	0.915	0.927	0.882	0.904	0.970
Neural model	**0.973**	**0.972**	**0.970**	**0.971**	**0.996**

- Classification

	Accuracy	Precision		Recall		F_1 score	
		Micro	Macro	Micro	Macro	Micro	Macro
Bigram	0.800	0.787	0.564	0.800	0.513	0.787	0.522
Neural model	**0.892**	**0.891**	**0.713**	**0.892**	**0.653**	**0.887**	**0.660**

Micro: weighted averages over all classes
Macro: unweighted averages

ROC curve



Neural GRU, d=512 (AUC=0.996)
Neural GRU, d=128 (AUC=0.994)
Bigram (AUC=0.970)
True Positive Rate
1.0
0.8
0.6
0.4
0.2
False Positive Rate
10^{-6}
10^{-5}
10^{-4}
10^{-3}
10^{-2}
10^{-1}

Discussion

- Neural model is also able to detect dictionary-based DGAs such as `suppobox` (recall of 93%, compared to only 12% for baseline) when given enough training examples
- Some DGAs still remain difficult to detect, such as `matsnu` (not enough training data to learn underlying wordlists)
- See our paper for detailed results for each malware family

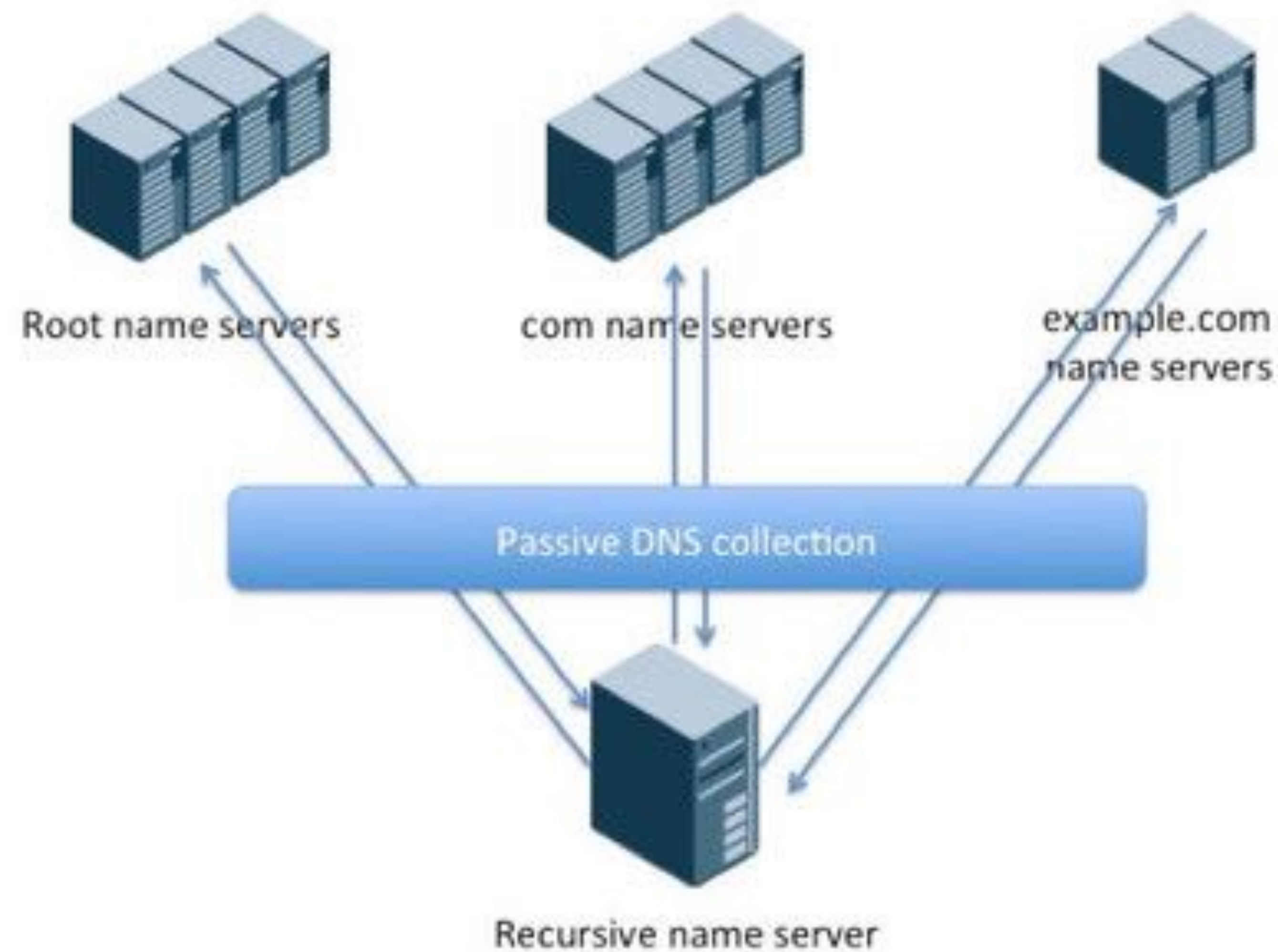
[Lison, P., & Mavroeidis, V. (2017). Automatic Detection of Malware-Generated Domains with Recurrent Neural Models. In *Proceedings of NISK 2017*.]

Part 1: Predicting the reputation of domains and IP addresses from passive DNS data

Passive DNS

- Can we automatically predict the reputation of domain names and IP addresses from DNS data?

- **Passive DNS data** is highly useful:
  - Inter-server DNS messages captured by sensors
  - Less privacy concerns (not tied to personal information)



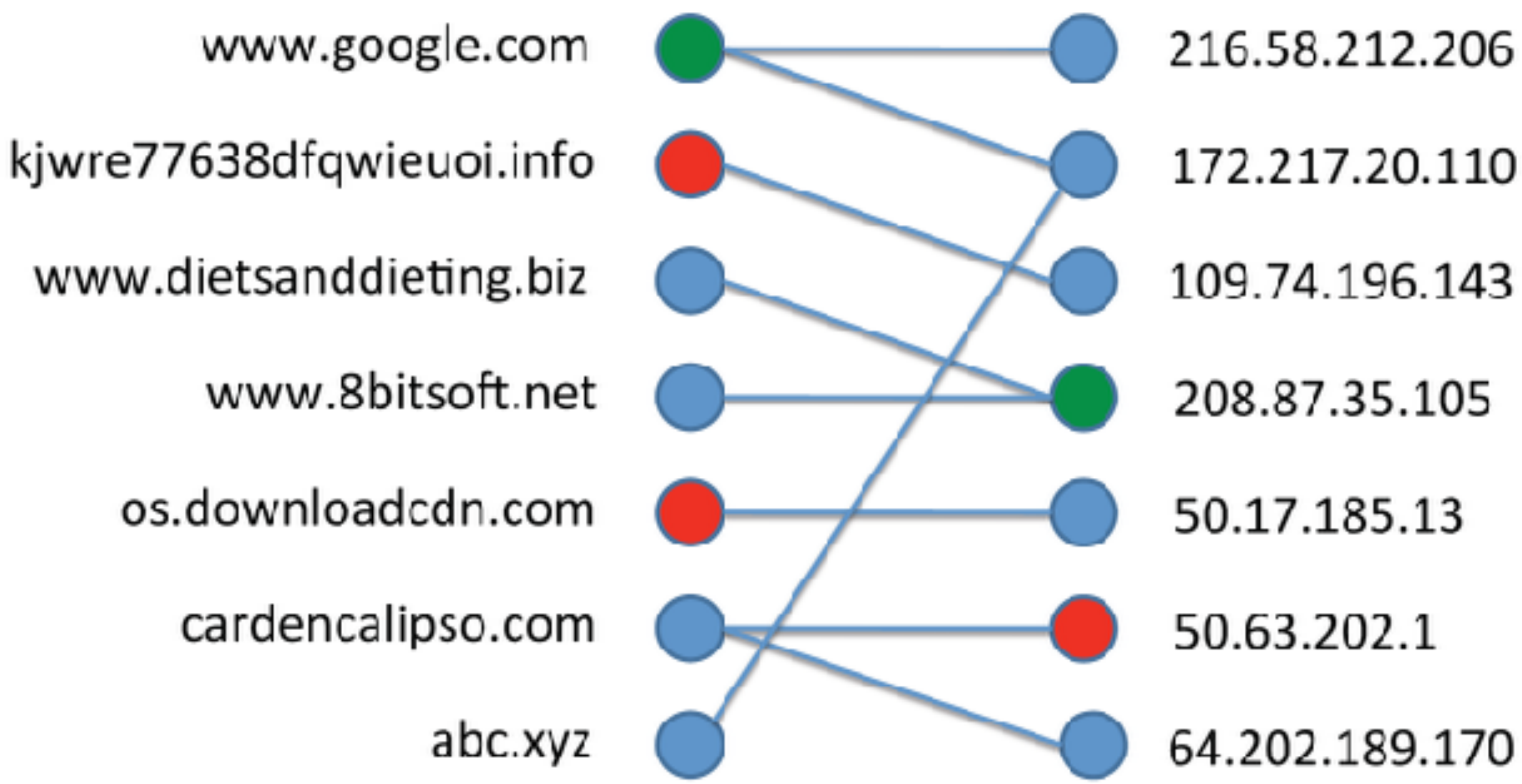
Passive DNS

- Collaboration with Mnemonic AS, a Norwegian cyber-security company [www.mnemonic.no]
- Dataset of *720 million aggregated DNS queries* collected over a period of four years
- Each entry is defined by:
  - A record type (A, CNAME, etc.)
  - A query and its answer,
  - A Time-to-Live (TTL) value
  - A number of occurrences
  - Timestamps for the first and last occurrence of the entire

84% A records,
11% CNAME
4% AAAA

Data

Labelled dataset of **720 million** records

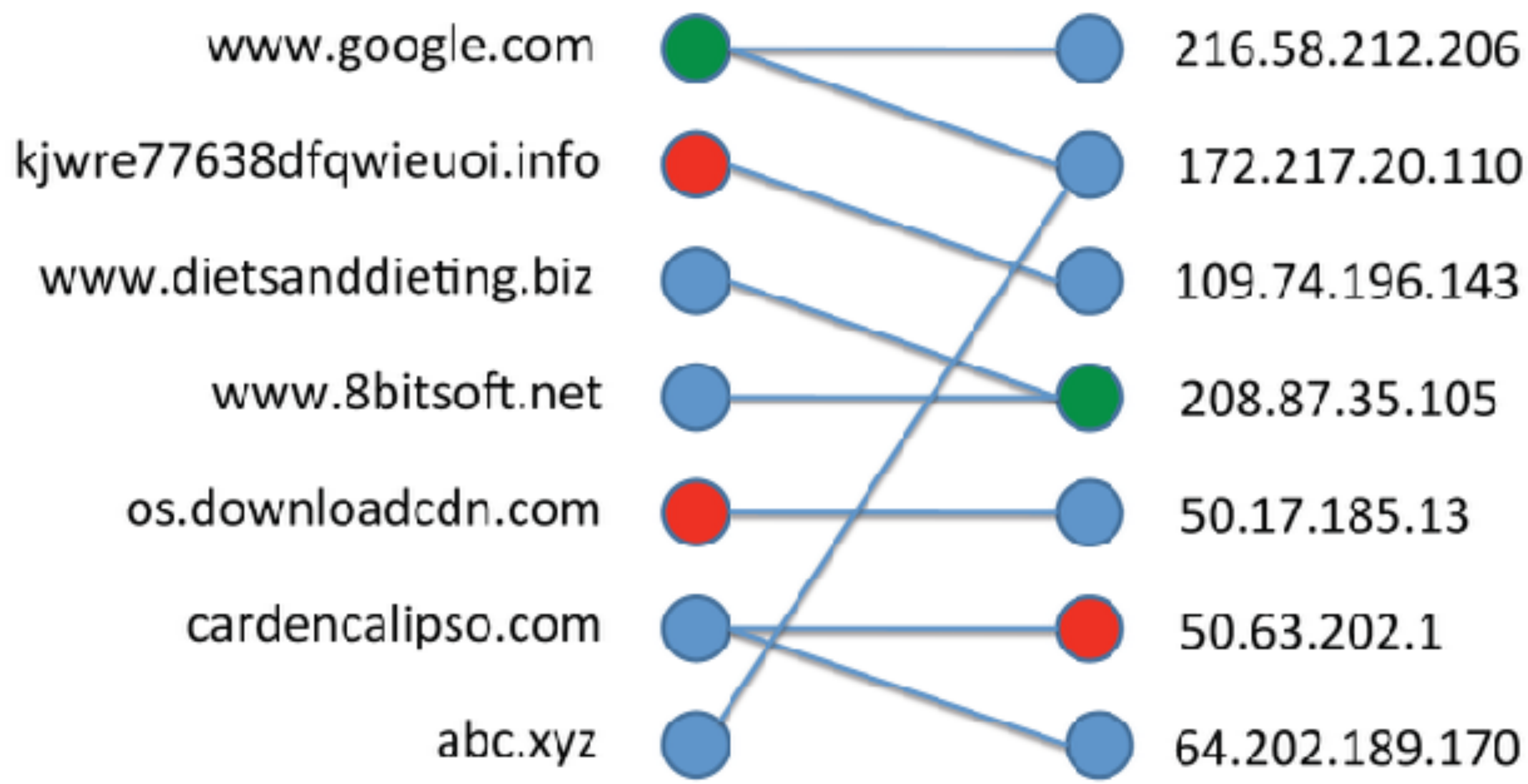


We enriched the passive DNS data with:

- ► Reputation labels from existing blacklists and whitelists
- ► IP location(geoname identifiers) and ISP data

Data

Labelled dataset of **720 million** records (**102 M** records labelled as benign, **8.2 M** records as malicious and **614 K** records as sinkhole)



- The reputations are associated with a *confidence level* (from the reputation source and description)
- Employed to derive reputations for DNS records (edges)

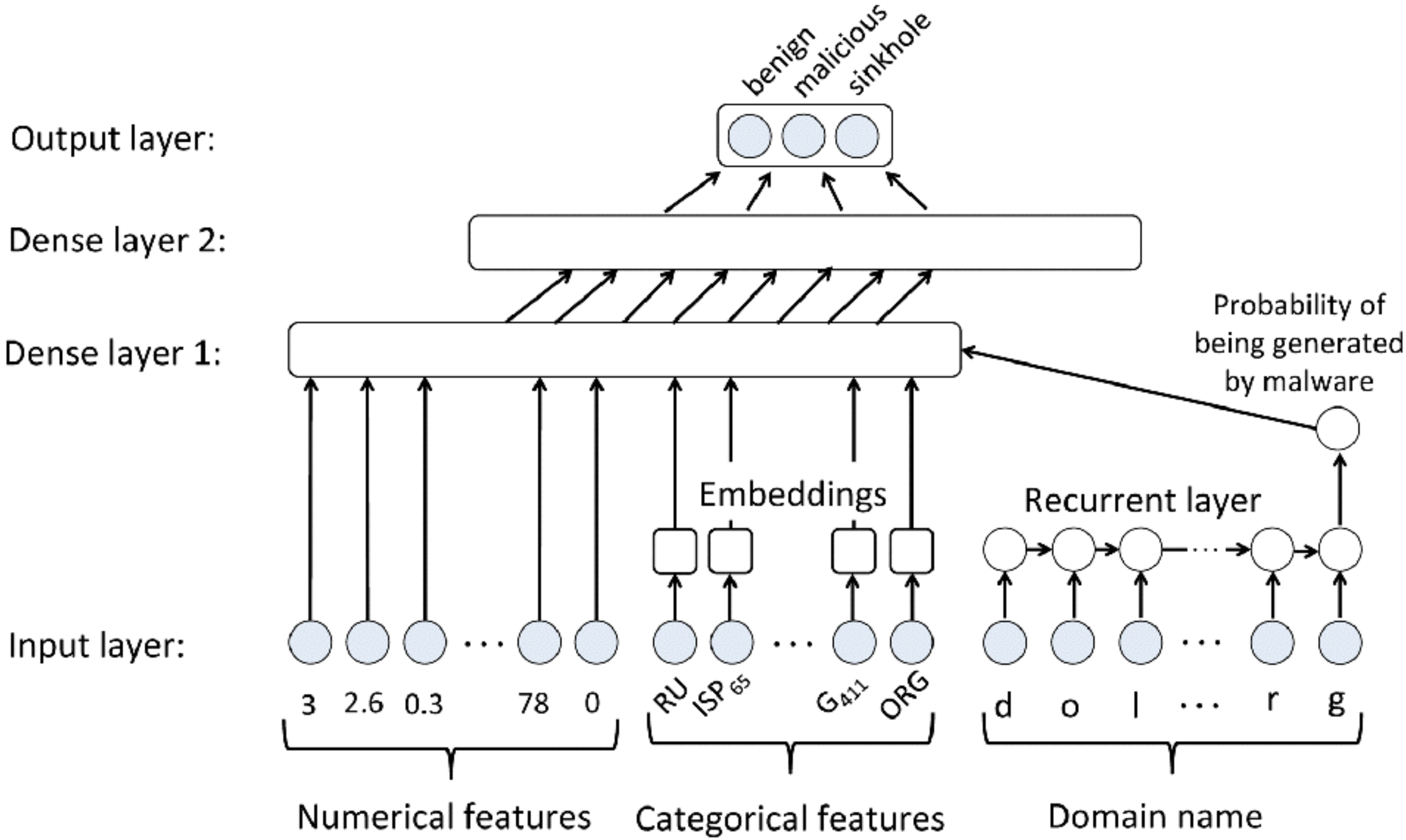
Graph inference

- **Local neighbourhood** is important for the reputation
- Traversal of bipartite graph to extract the number of neighbours and their reputations
- Experiments with adapted versions of PageRank

Features

- Numerical features derived from the records:
  - Lifespan, number of queries (for record, domain or IP), number of distinct countries or ISP, TTL values, etc.
- Categorical features:
  - ISP, geolocation, top-level domain, etc.
- Ranking features from Alexa
- Features extracted from neighbouring records
  - Number of records at distance 1 and of reputation X
- Sequence of characters from the domain

Neural model

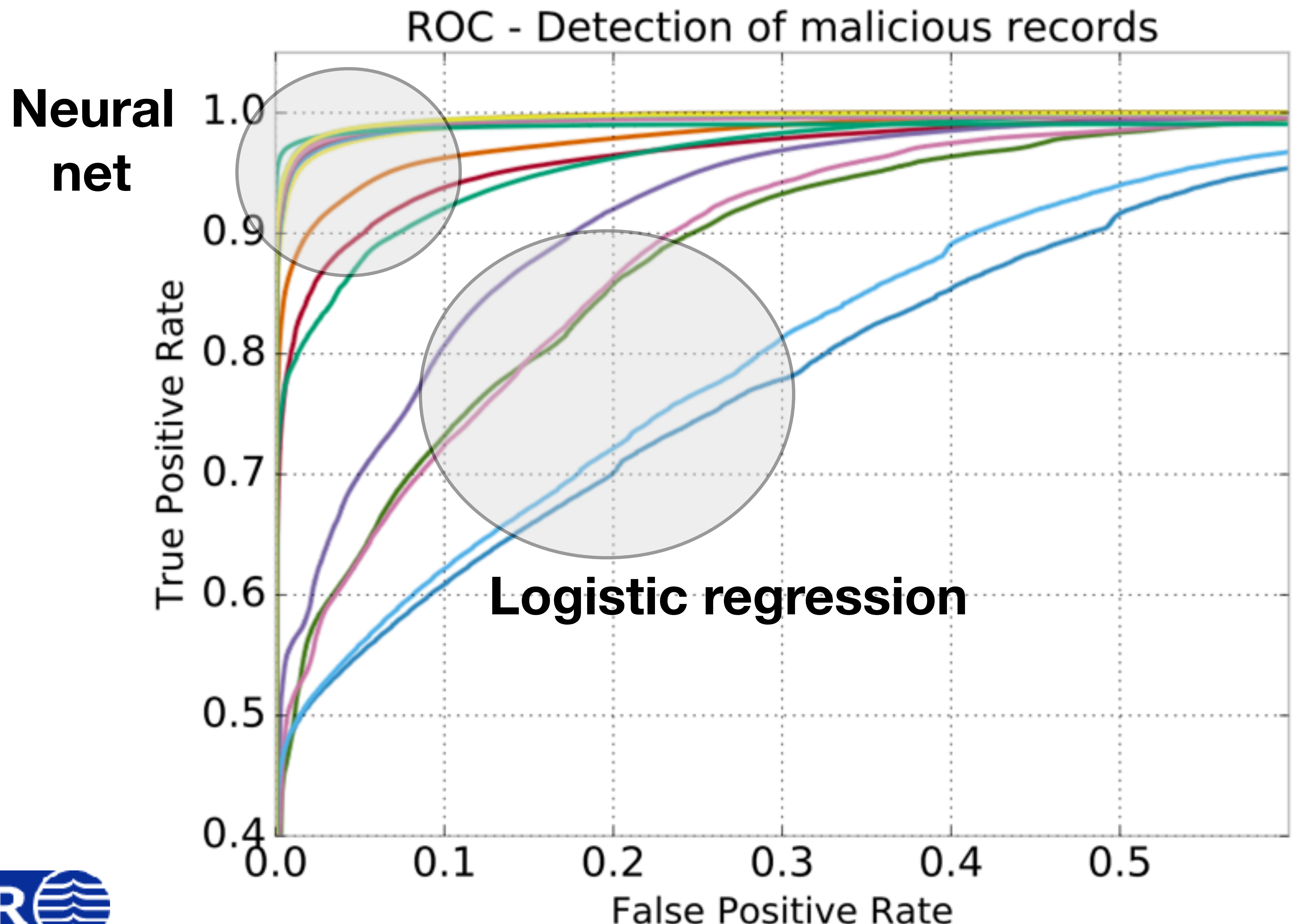


Results

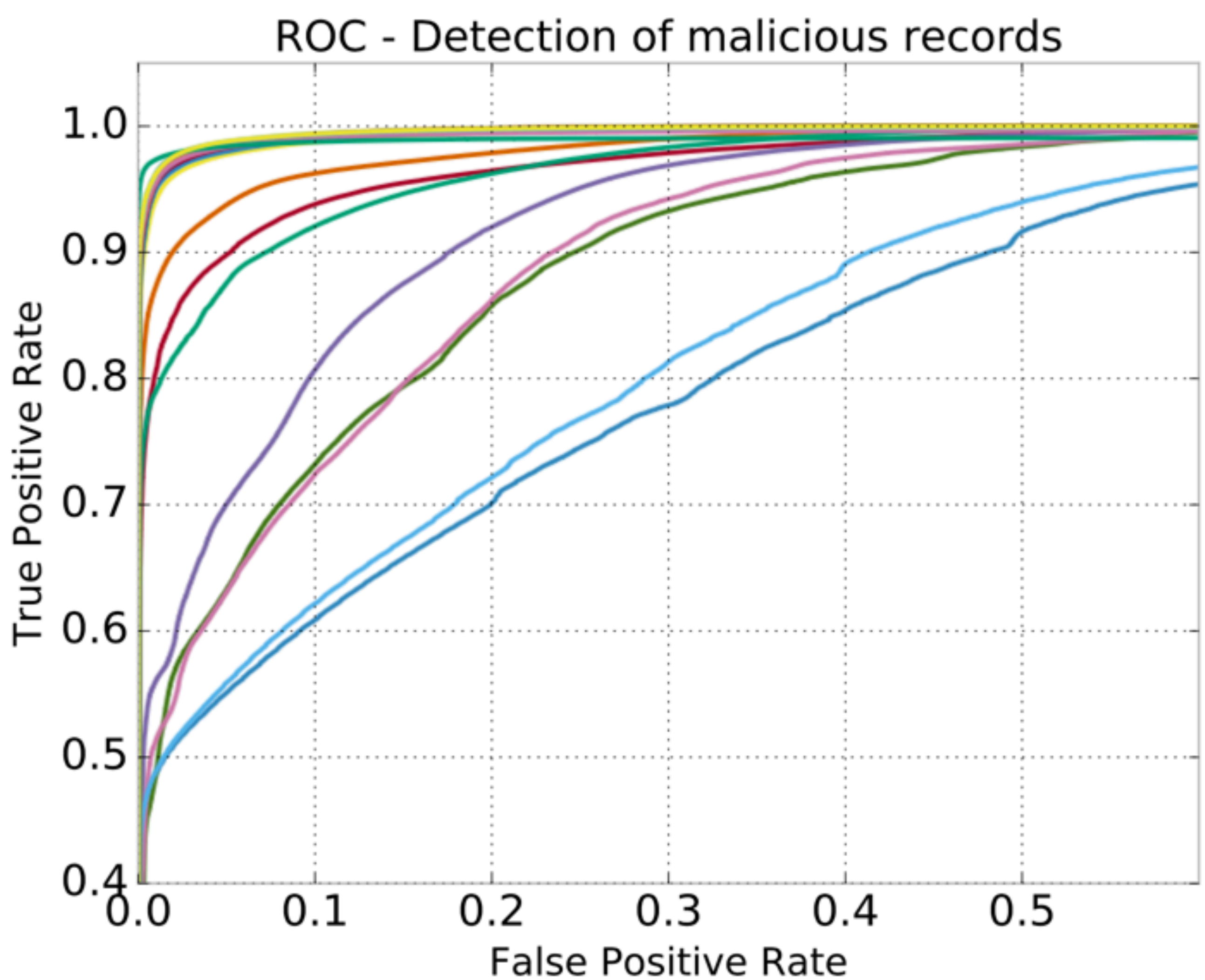
Model	Benign			Malicious			Sinkhole			Accuracy
	P	R	F_1	P	R	F_1	P	R	F_1	
nb_domain_queries < 10	0.98	0.44	0.61	0.10	0.87	0.19	0.0	0.0	0.0	0.54
Logistic regression	0.97	0.97	0.97	0.60	0.65	0.62	0.51	0.26	0.35	0.944
Neural net (with 1 hidden layer)	0.99	0.99	0.99	0.93	0.93	0.93	0.99	1.00	0.99	0.990
Neural net (with 2 hidden layers)	1.00	0.99	0.99	0.92	0.95	0.93	0.98	1.00	0.99	0.990

In this setting, the neural net is first trained on the labelled dataset and applied to predict the reputation of unlabelled records, which are then used to get better estimates of the "neighbour" features.
The model is then trained again on these new feature values.

ROC curve


ROC - Detection of malicious records
Neural net
Logistic regression
True Positive Rate
False Positive Rate
0.4
0.5
0.6
0.7
0.8
0.9
1.0
0.0
0.1
0.2
0.3
0.4
0.5

ROC curve



With the best performing model, we achieve a recall of:

- 0.74 for a false positive rate of 1:100K
- 0.86 for 1:10K
- 0.92 for 1:1000
- 0.97 for 1:100
- 0.99 for 1:10.

Future work

- Exploiting **semi-structured information sources**
  - Security reports, alerts on cyber-security websites, etc.
  - Knowledge discovery, information extraction necessary
- **Benefit**: go beyond simple reputation labels and understand *why* a host should or should not be trusted
- **Challenges**:
  - Lack of annotated text data for this domain
  - Inconsistent naming conventions for cyber-threats

Conclusion

- Neural networks can be successfully used to predict the **reputation** of end-point hosts
  - Detection of DGA from the domain names
  - Detection of malicious records from passive DNS

- Can be integrated in software tools for cyber-threat intelligence

- Current work:
  - Consolidate experimental results
  - Integration of unstructured data sources (i.e. textual data)