The Alan Turing Institute

UCL

KING'S College LONDON

DRMD: Deep Reinforcement Learning for Malware Detection under Concept Drift

Shae McFadden[1,2,3], Myles Foley[2], Mario D'Onghia[3], Chris Hicks[2], Vasilios Mavroudis[2], Nicola Paoletti[1], Fabio Pierazzi[3]

[1]King's College London, [2]The Alan Turing Institute, [3]University College London

AAAI-26 / IAAI-26 / EAAI-26

JANUARY 20–27, 2026 | SINGAPORE

Partially Funded By

National Cyber Security Centre

a part of GCHQ

EPSRC

Engineering and Physical Sciences Research Council

Android Malware Detection

Thousands of new apps per day

Limited capacity for manual review

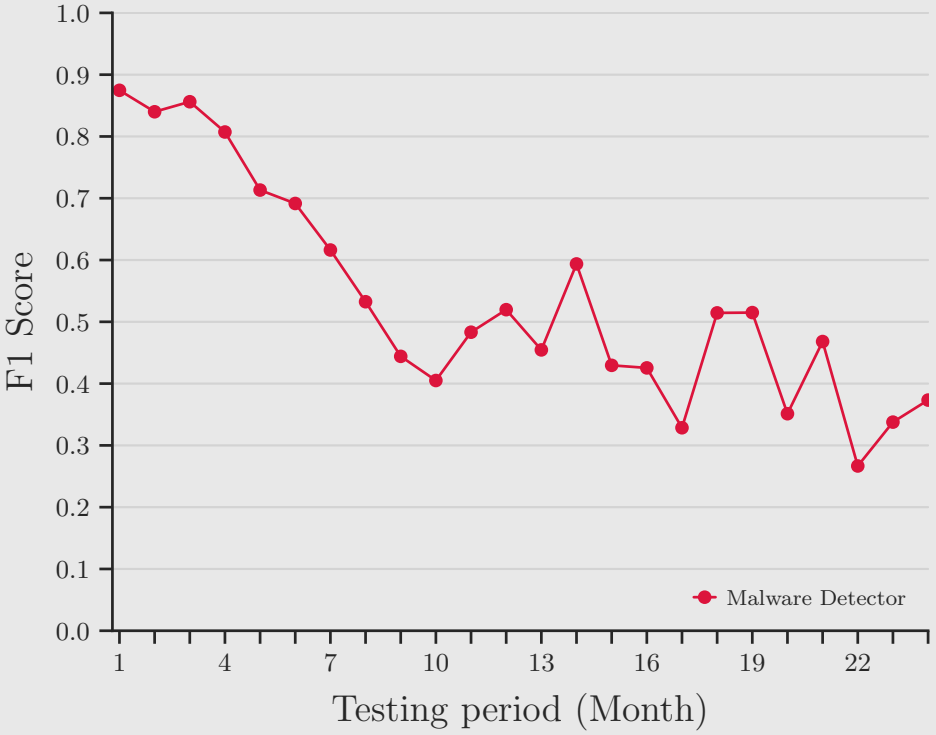
Concept Drift

ML Assumption: data is stationary over time

Reality: apps constantly evolve

Result: Performance degradation

Yesterday's training data becomes less relevant for today's threats.



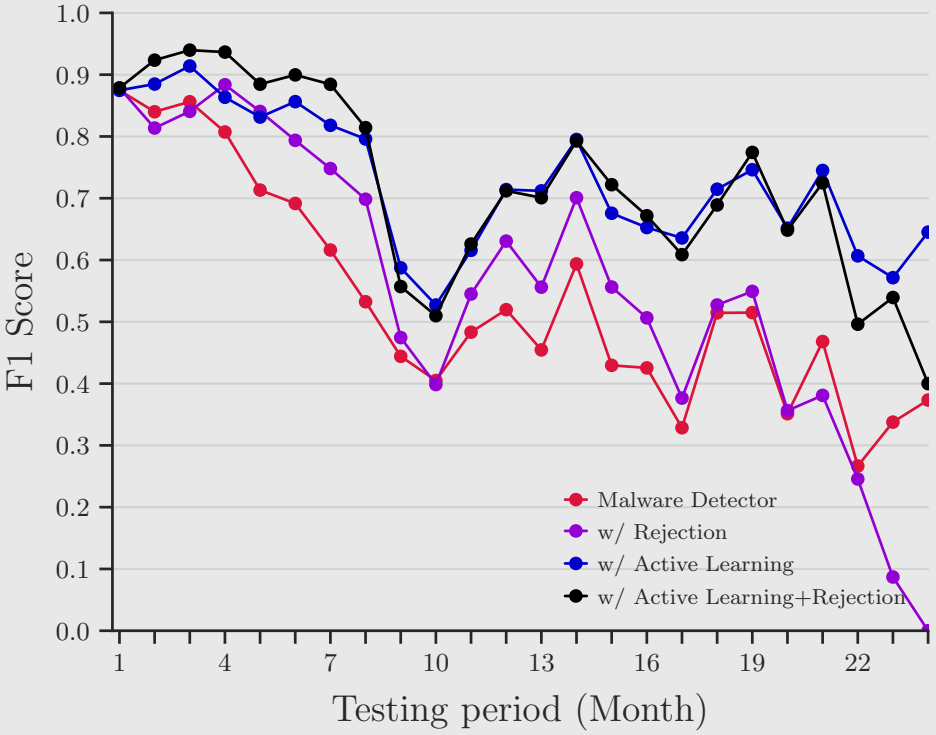
Concept Drift Mitigations

Rejection:
limiting the impact of drift

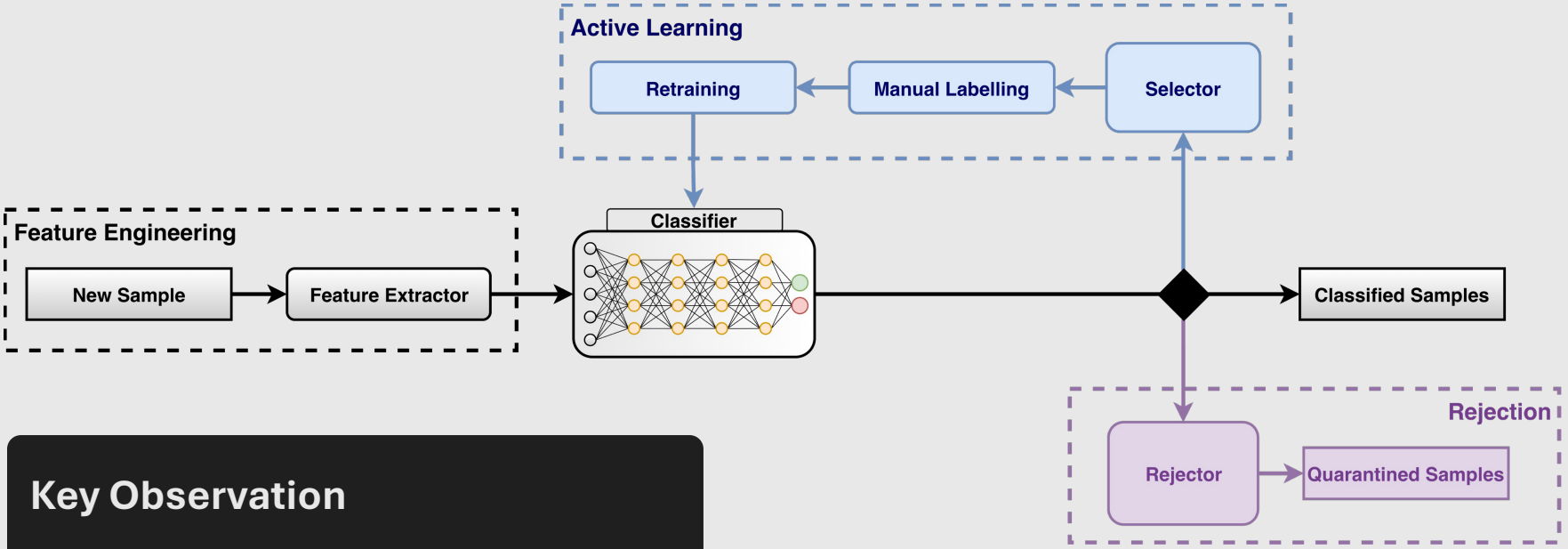
Selects samples at a high-risk of being misclassified to be quarantined.

Active Learning:
adapting the detector to drift

Selects an informative subset of new samples for manual labelling and retraining.



Malware Detection Pipeline



Key Observation

Existing approaches treat active learning, rejection, and detection independently

DRL-Based Malware Detection

Intuition

Treat malware detection as a **unified** decision-making problem

Not just "is this malware?" but also ”am I certain?"

Formulation (MD-MDP)

One-step MDP (Contextual Bandit)
Corrects spurious dependencies of prior work, ICMDP [Appl. Intell.’20]

Action Space

✓ **Classify as Goodware**

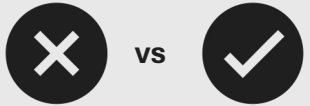
✗ **Classify as Malware**

? **Reject** → **Active Learning**

Rewards

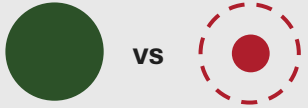
Accuracy

Provides the foundation
+1 correct, -1 incorrect



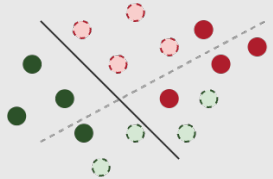
Class Imbalance

Upscales rewards for malware based on distribution (~10%)



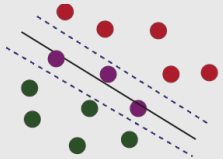
Temporal Robustness

Upscales rewards for samples based on temporal position

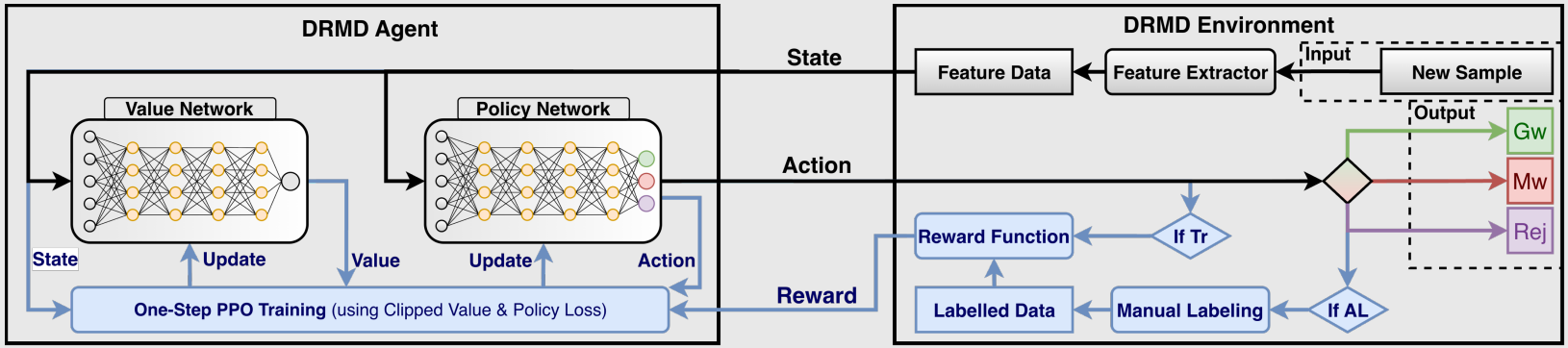


Rejection

Balances rewards for rejection relative to misclassification risk



DRMD Pipeline



Proximal Policy Optimization (PPO)

Learns policy from experience through clipped updates

Classification-Only Policy

DRMD in existing pipelines

Classification-Rejection Policy

Unified Malware Detection

Experimental Settings

Datasets

Transcendent (Tr)
2014-2018 | 259,230 apps | ~10% malware

Hypercube (Hc)
2021-2023 | 159,839 apps | ~10% malware

Feature Spaces

Drebin (D)
10,000-D sparse binary vector including: hardware and app components; requested and used permissions; filtered intents; restricted and used API calls; and network addresses

Ramda (R)
379-D binary vector including: permissions, intents, and sensitive APIs.

Evaluation

Time-aware
Train on first year, test on remaining years using monthly periods for active learning and rejection.

AUT Metric
Area Under Time (AUT) of the F_1 Score, measures performance stability over time under concept drift.

Baselines

Drebin (SVM)

DeepDrebin (MLP)

Ramda (VAE+MLP)

SL-DRMD (supervised)

Classification-Only Policy

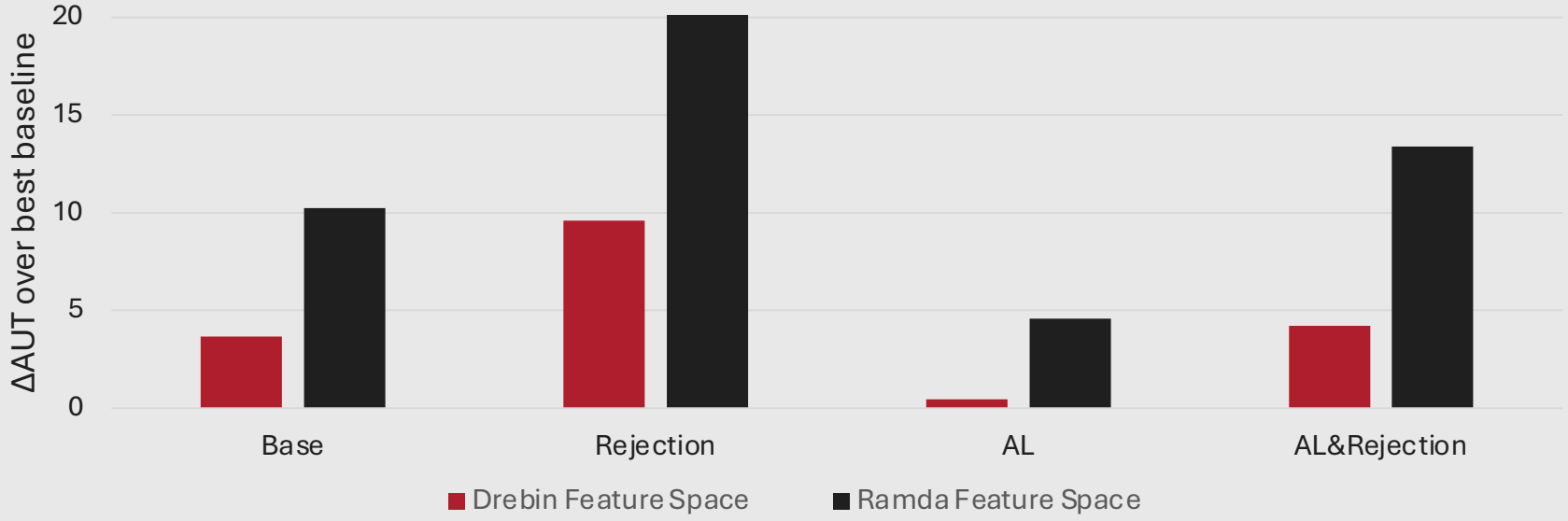
Classifier Comparison

Same AL and rejection budgets
DRMD outperforms Baselines

90%
settings

79%
statistically significant

+8.66
ΔAUT



Classification-Rejection Policy

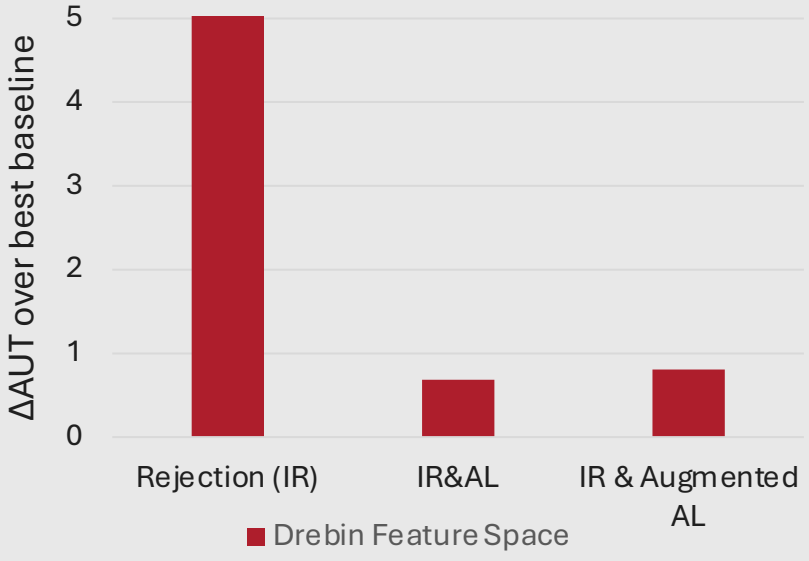
Pipeline Comparison

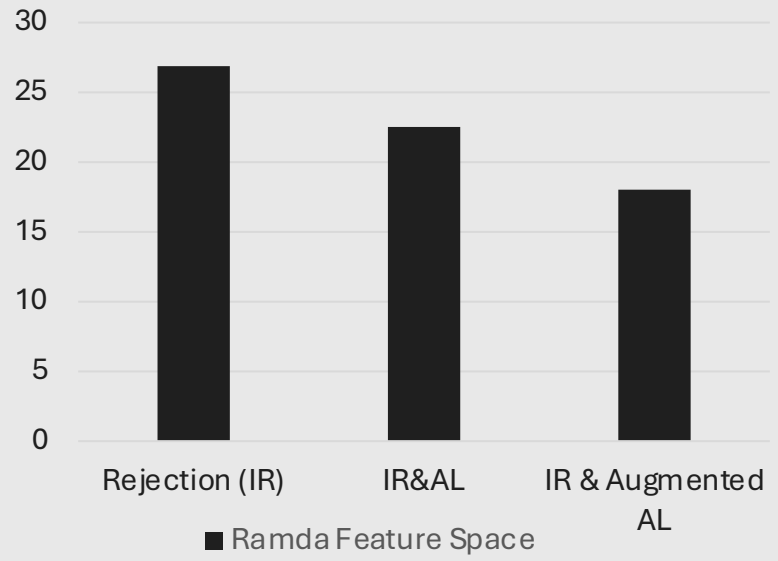
Same AL and rejection budgets
DRMD outperforms Baselines

81%
settings

68%
statistically significant

+10.90
ΔAUT





MD-MDP vs ICMDP

ICMDP (Prior Work)

Used in DQNimb [Appl. Intell.'20] & *SINNER* [Info.'24]

✗ Episodes span multiple samples

✗ State transitions can create correlations between independent samples

✗ Does not consider concept drift or mitigations

MD-MDP (Our Approach)

DRMD

✓ One-step MDP

✓ Each sample is an independent episode

✓ Drift-aware reward design that integrates rejection and active learning

Formulation Comparison

Same architectures using CO policy
MD-MDP outperforms ICMDP

97%
settings

45%
statistically significant

+1.94
ΔAUT

One-Step PPO vs DCBs

Deep Contextual Bandits (DCBs)

NeuralTS [ICLR'21] & *NeuralUCB* [ICML'20]

✓ Each sample is an independent

✗ Updates are performed over the history of all experiences

✗ Retains experiences from past samples

One-Step PPO

DRMD

✓ Each sample is an independent

✓ Updates are performed over new experiences

✓ Uses temporal sliding window of samples to generate new experiences

Approach Comparison

Same rewards using CO policy
One-Step PPO outperforms DCBs

100%
settings

100%
statistically significant

+9.77
ΔAUT

Key Takeaways

1. Adaptive Decision-Making, Not Just Classification
Learning what to predict and when to abstain in one policy

2. One-Step MDP Formulation
Treats samples independently to avoid correlation between samples

3. Concept Drift-Aware DRL
Reward structure captures spatial and temporal dynamics

4. Integration Matters
Integrated rejection and AL can act in real time and adapt as the agent does

+8.66 AUT classification-only policy
+10.90 AUT classification-rejection policy
Across 172 experimental settings

DRMD: Deep Reinforcement Learning for Malware Detection under Concept Drift

Shae McFadden[1,2,3], Myles Foley[2], Mario D'Onghia[3], Chris Hicks[2], Vasilios Mavroudis[2], Nicola Paoletti[1], Fabio Pierazzi[3]

[1]King's College London, [2]The Alan Turing Institute, [3]University College London

AAAI-26 / IAAI-26 / EAAI-26
JANUARY 20-27, 2026 | SINGAPORE

Partially Funded By
National Cyber Security Centre
a part of GCHQ
EPSRC
Engineering and Physical Sciences Research Council