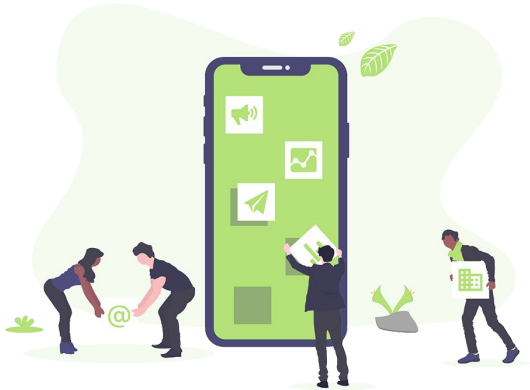
Repackaged Android App Detection

Columbia University
Data Science Capstone Project
Team 1



Presented by
Jianfeng Zhuang, Huazhang Liu, Chengyou Ju, Shaofeng Wu, Weitao Chen

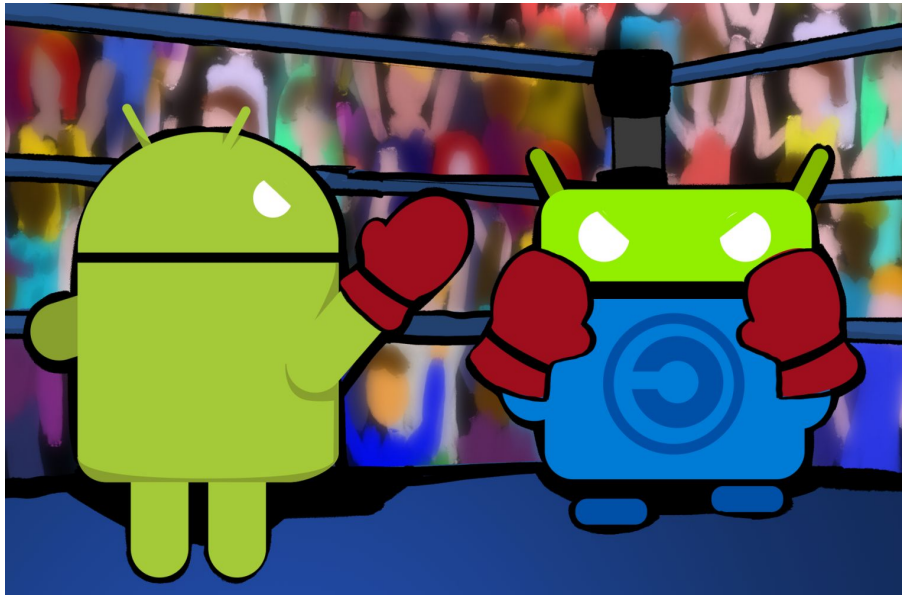
Faculty Mentor: Professor Gail Kaiser
PhD Mentor: Shirish Singh

COLUMBIA | ENGINEERING
The Fu Foundation School of Engineering and Applied Science

Overview

Harms of Android apps repackaging

- Deprive benefits
- Spread malwares
- Increase workload



How the Reverse Engineering Happened

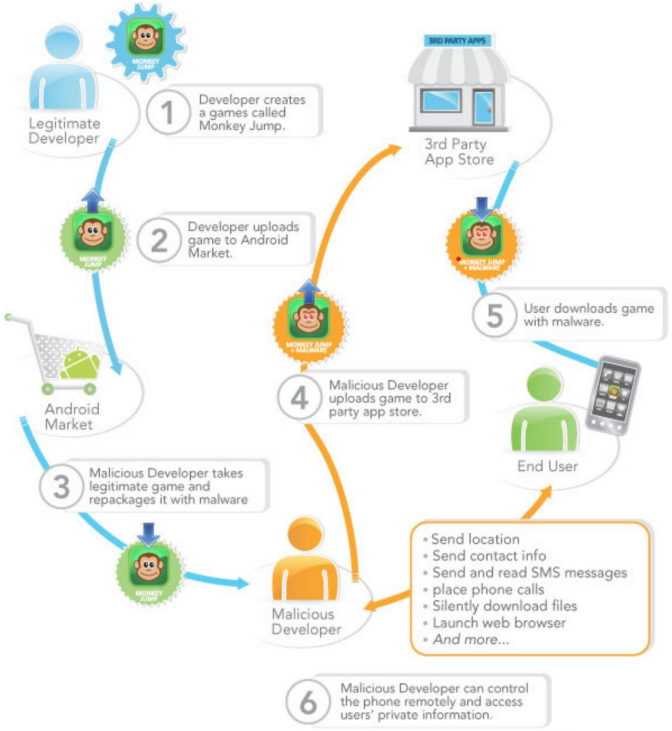


Fig Taken from Lookout Mobile Threat Report 2011, accessed Dec, 2020.

Dataset: RePack

- Collected From AndroZoo
- 2776 original apps
- 15,297 repackaged apps

Pipeline to Extract Java Code

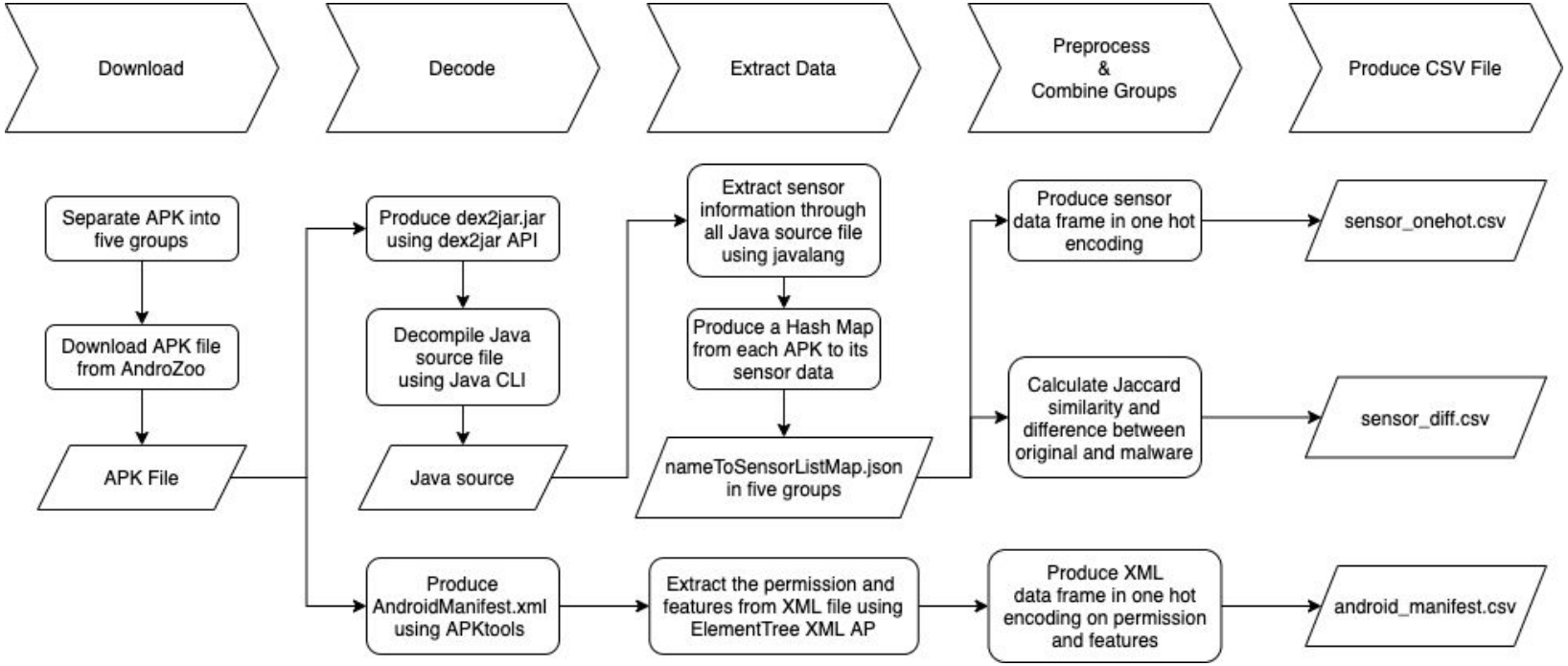
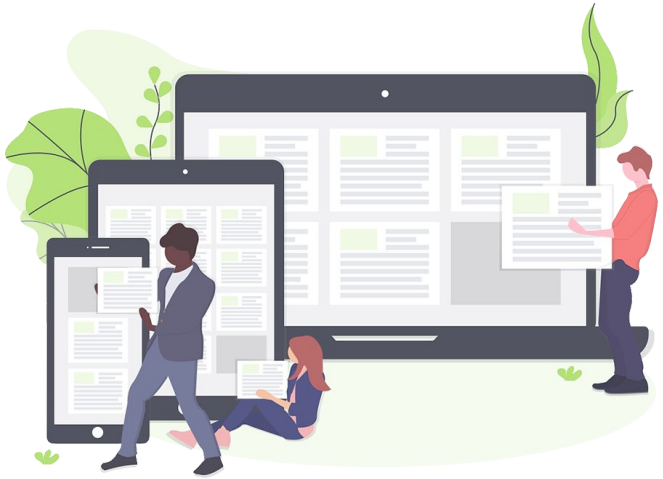


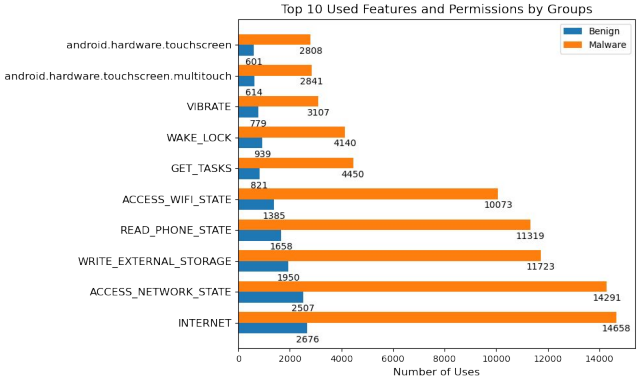
Fig. Java Code Extraction Pipeline

Exploratory Analysis and Observations

- 14827 pairs of repackaged and original apps extracted successfully
  - 7178 pairs have same sensors
  - 6789 pairs do not have any sensors
  - 804 repackaged apps have additional sensors than original apps
  - 61 original apps have additional sensors than repackaged apps
- 29 used sensors
- 32 used hardware and software features
- 114 used permissions
- Most used:
  - Sensor: Accelerometer
  - Feature: Touchscreen multitouch
  - Permission: Internet



Exploratory Analysis and Observations



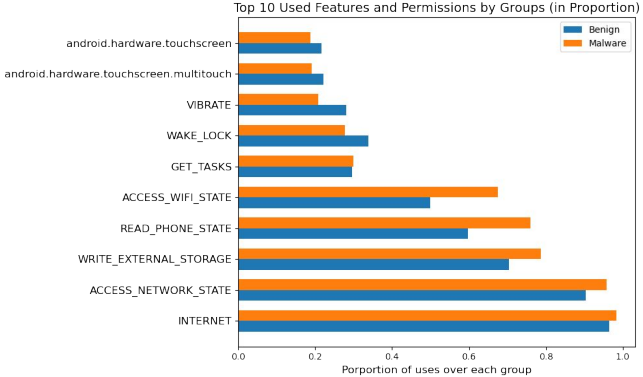
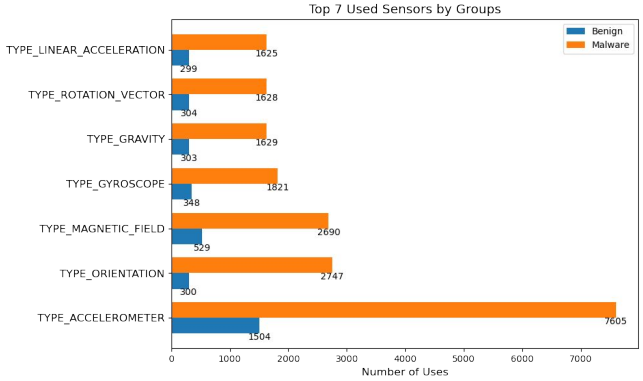


Fig. Top Used Features and Permissions



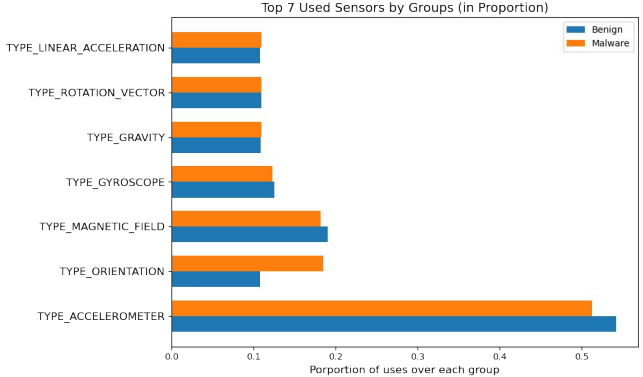


Fig. Top Used Sensors

Exploratory Analysis and Observations

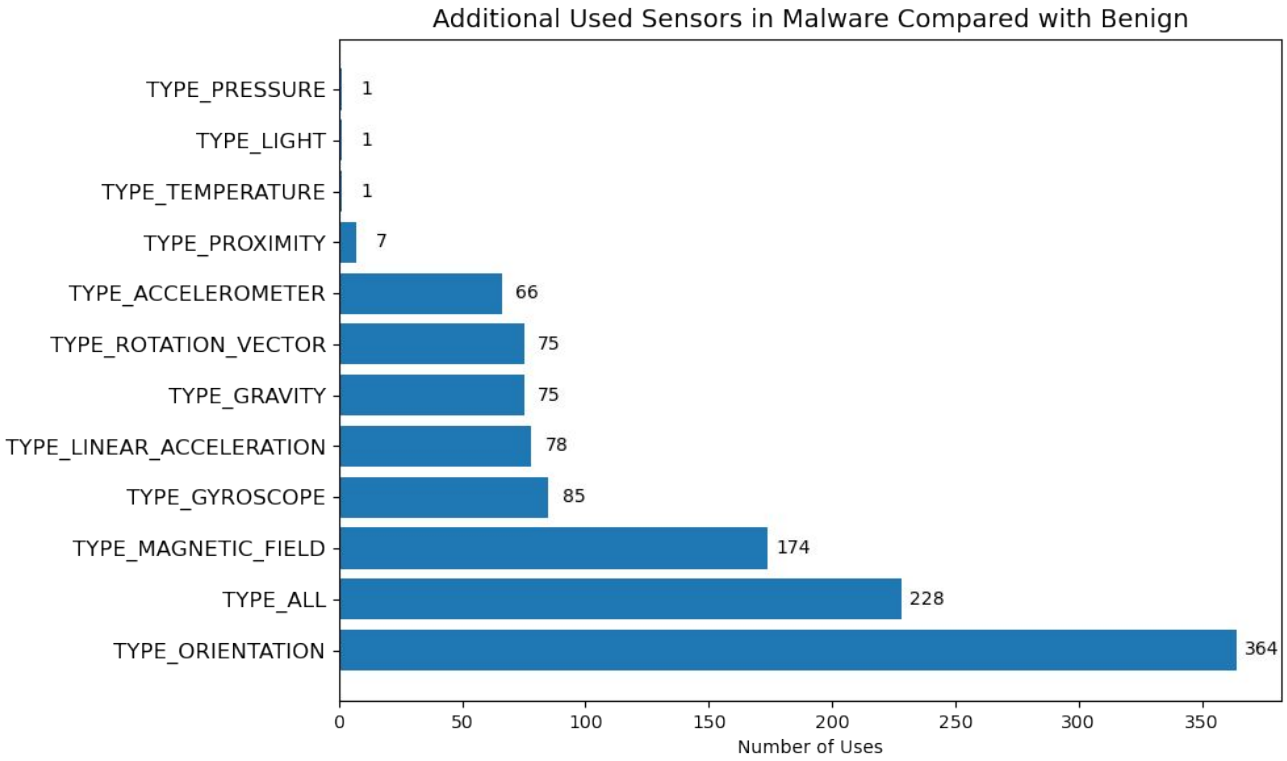


Fig. Additional Used Sensors in Repackaged than Original

Works with Imbalanced Dataset

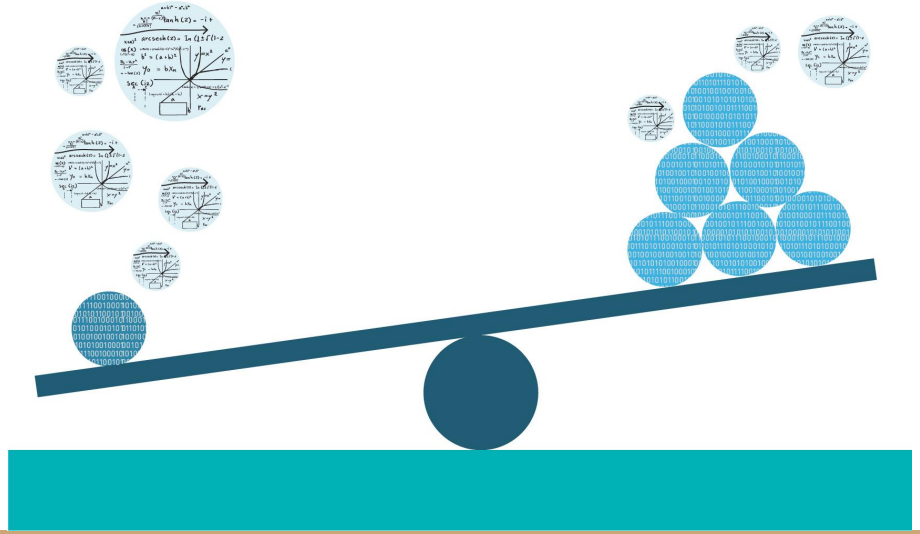
Original dataset contains 15297 pairs of repackaged-to-original apps

Solve the imbalance problem by:

- Down Sampling
- Random Oversampling
- Duplicating original apps
- SMOTE

Major Evaluation Metrics:

- Balanced accuracy scores



Modeling on Sensor, Feature, Permission Data

The models we used:

- Baseline Model (predict everything as malware)
- Logistic Regression
- Support Vector Machine (SVM)
- K-Nearest Neighbor (KNN)
- Random Forest
- XGBoost
- Multilayer Perceptron (MLP)

Two methods on oversampling:

- Oversample minor class
- SMOTE (Synthetic Minority Oversampling TEchnique)

Modeling on Sensor, Feature, Permission Data

Model results in balanced accuracy:

- Oversampling Model - Top 3 Models:
  - Random Forest: 0.70
  - Logistic Regression: 0.69
  - SVM: 0.69

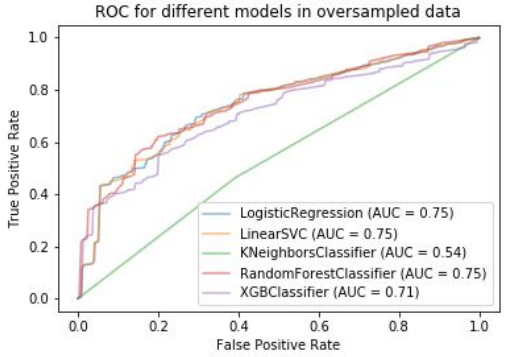


Fig. ROC Curve on Oversampling Model

- SMOTE Model - Top 3 Models:
  - Logistic Regression: 0.68
  - Random Forest: 0.68
  - XGBoosting: 0.67

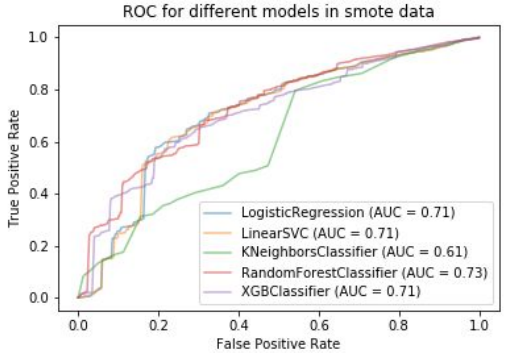


Fig. ROC Curve on SMOTE Model

Modeling on Sensor, Feature, Permission Data

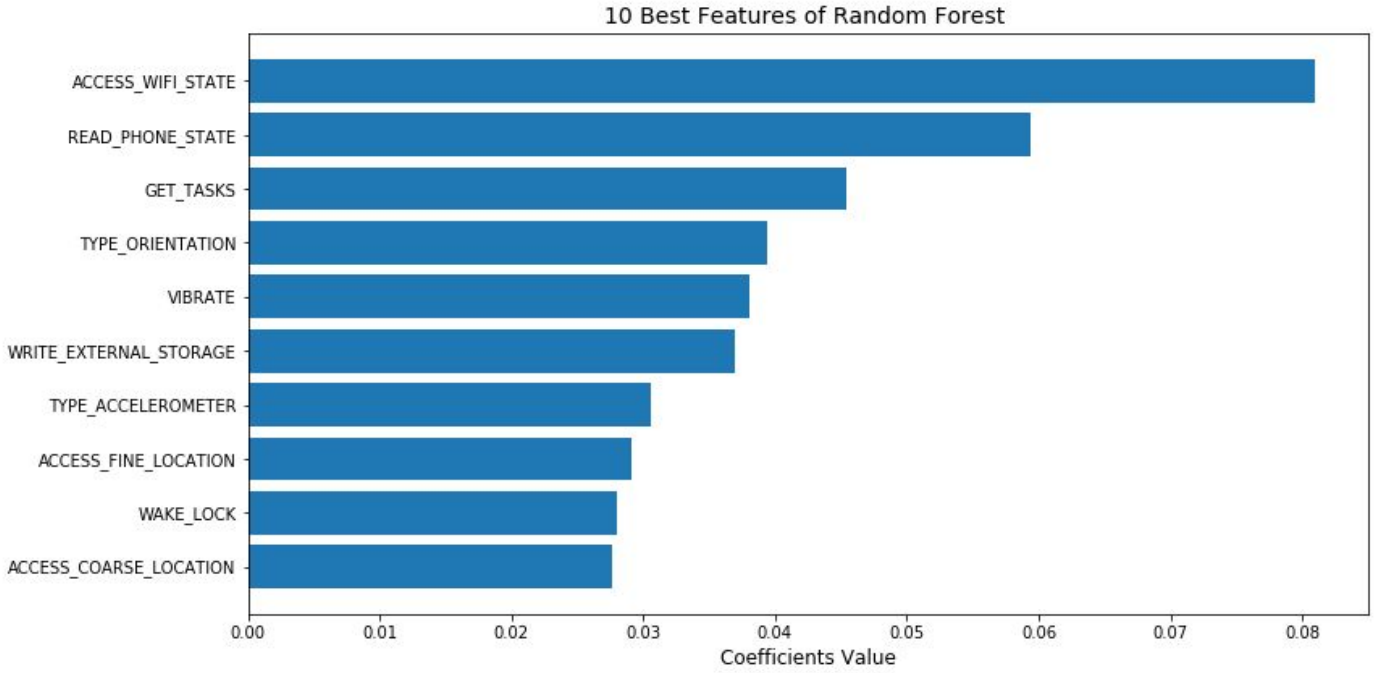
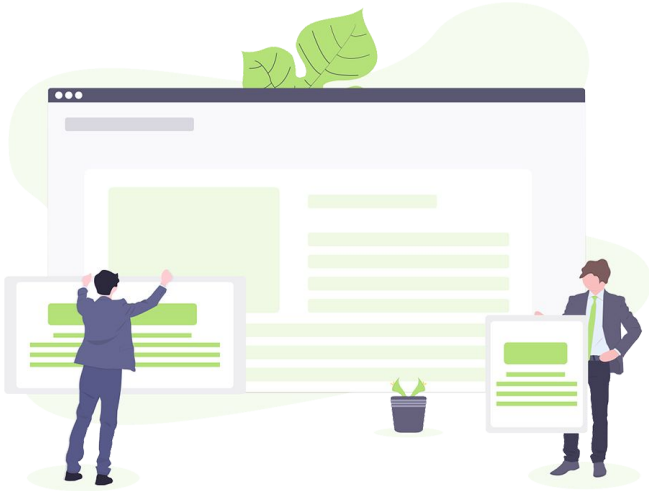


Fig. Feature Importance on Random Forest

Flow Data



I. Control Flow Graph (CFG)

- Static analysis and compiler application
- Tool used: Androguard
- Pipeline

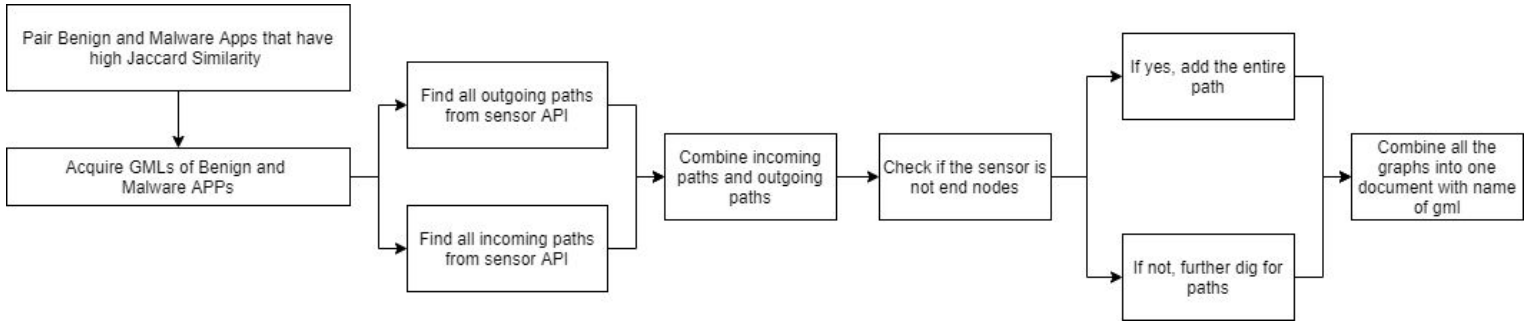


Fig. Control Flow Graph Pipeline

Flow Data

II. Data Flow Graph (DFG)

- Flow Droid: a generic, platform-independent data flow tracker and platform-specific extensions
- Pipeline

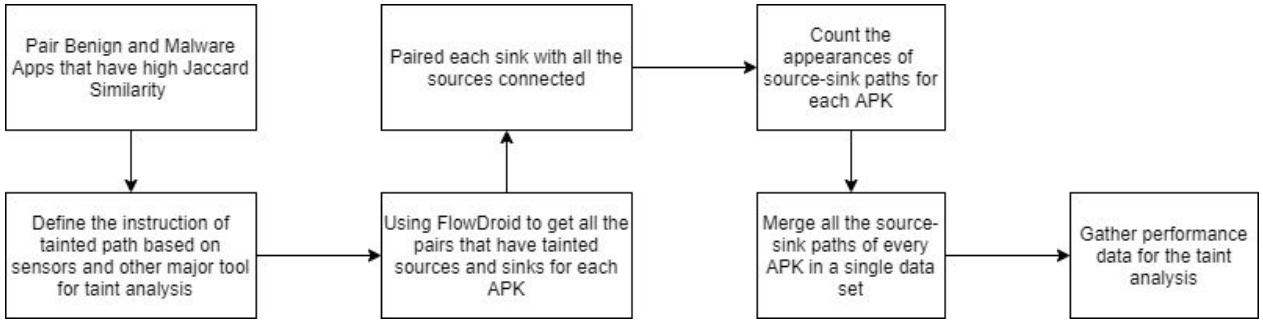


Fig. Data Flow Graph Pipeline

Flow Data

DFG - Taint Analysis

- Find untrustworthy sources and mark them as tainted
- Follow the “tags” to trace the flow of tainted objects

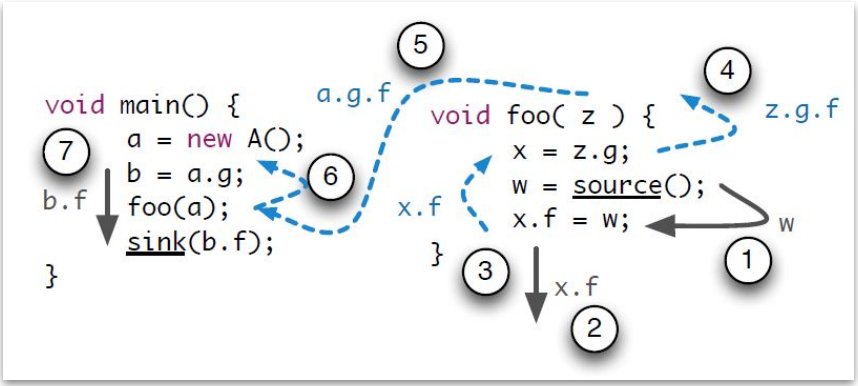


Fig. Taint Analysis

Modeling on Flow Data

- Mainly focus on 804 pairs
  - Repackaged apps have extra sensors
  - Analyze in CFG, DFG and Both
- Additional CFG Analysis on 7997 pairs:
  - Adding 7178 pairs with same sensors
- Strategies on Imbalance:
  - Random Oversampling
  - Duplicating originals by pairs
- Model Selection:
  - Logistic Regression
  - Support Vector Machine (SVM)
  - K-Nearest Neighbors (KNN)
  - Random Forest
  - Gradient Boosting
  - XGBoost
  - Multi-layer Perceptron (MLP)
- Evaluation Metrics:
  - Balanced Accuracy (Main)
  - ROC-AUC Score
  - F1-Score

Modeling on Flow Data

Model results in balanced accuracy:

- 804 pairs on CFG - Top 3 Models:
  - MLP: 0.96
  - Gradient Boosting: 0.94
  - Random Forest: 0.89

- 804 pairs on DFG - Top 3 Models:
  - SVM: 0.77
  - Logistic Regression: 0.74
  - XGBoost Classifier: 0.71

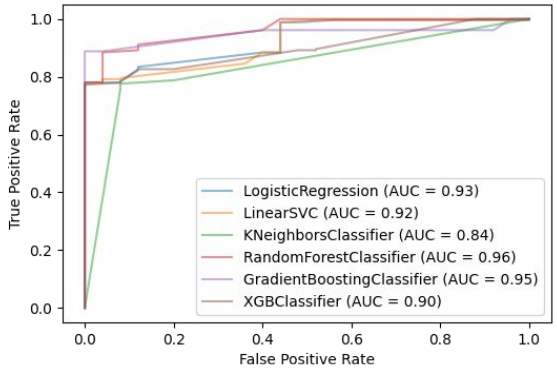


Fig. ROC Curve on CFG of 804 pairs

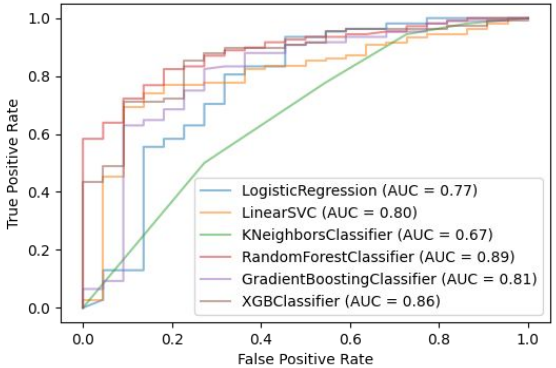


Fig. ROC Curve on DFG of 804 pairs

Modeling on Flow Data

Model results in balanced accuracy:

- 804 pairs on both CFG and DFG - Top 3 Models:
  - Logistic Regression: 0.96
  - SVM: 0.94
  - Gradient Boosting: 0.94

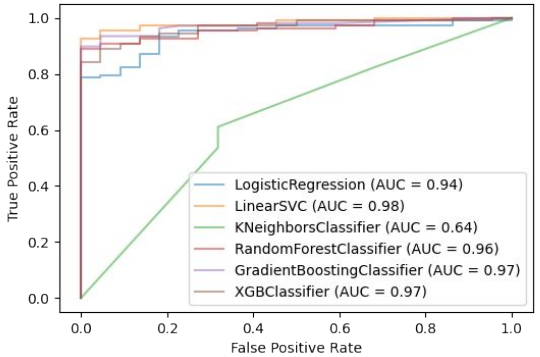


Fig. ROC Curve on all flow data of 804 pairs

- 7997 pairs on CFG - Top 3 Models:
  - Logistic Regression: 0.64
  - Random Forest: 0.62
  - XGBoost: 0.62

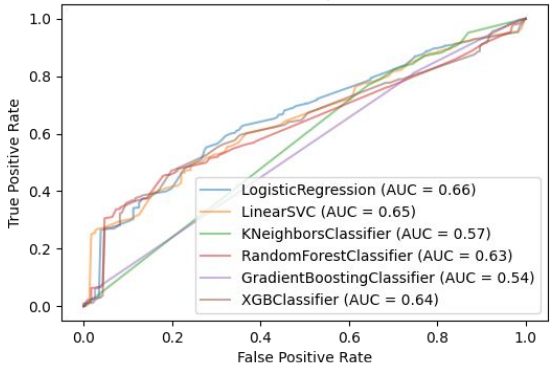


Fig. ROC Curve on CFG of 7997 pairs

Modeling on Flow Data

Feature importance in best Gradient Boosting model of 804 pairs:

- Top 3 Feature in CFG
  - Mobclix Browser Activity to Sensor Manager
  - Full Screen Activity to Sensor Manager
  - Unit Player to Sensor Manager
- Top 3 Feature in DFG
  - Call Graph Construction Time
  - Number of Sinks
  - Maximum of Memory Consumption
- Top 3 Feature in both CFG and DFG
  - Full Screen Activity to Sensor
  - Call Graph Construction Time
  - Full Screen Activity to Sensor Manager

Conclusion

Repackaged Apps thread Android ecosystem

Sensor based detection on repackaged apps helps

- Java code information based on sensor classifies some original-repackaged pairs
- Flow paths (CFG and DFG) through sensor also are able to classify pairs

Potential future works:

- Java-language-based detection
- Other methods to extract sensor information

Thank You For Listening

Links

Light Talk on Youtube: https://youtu.be/ECMqzvwGnes

Androzoo: https://androzoo.uni.lu

Javalang: https://github.com/c2nes/javalang

AndroGuard: https://androguard.readthedocs.io/en/latest/

FlowDroid: https://github.com/secure-software-engineering/FlowDroid

Appendix - Model Results

	accuracy	roc_auc_score	precision_score	recall_score	balanced_accuracy_score
RF	0.644388	0.752475	0.951397	0.624267	0.703111
LR	0.708347	0.745679	0.936959	0.713710	0.692694
SVM	0.711545	0.745366	0.936424	0.718109	0.692388
xbg	0.603454	0.710569	0.939716	0.582845	0.663603
mlp	0.597378	0.697650	0.921884	0.588343	0.623745
KNN	0.480652	0.535450	0.889831	0.461877	0.535450
Baseline	0.872402	0.500000	0.872402	1.000000	0.500000

Fig. Model Results on Sensor, Feature, and Permission Dataset using Oversampling

	accuracy	roc_auc_score	precision_score	recall_score	balanced_accuracy_score
LR	0.715382	0.712449	0.930647	0.728006	0.678539
SVM	0.710265	0.710404	0.930529	0.721774	0.676677
xbg	0.628718	0.712880	0.940912	0.612903	0.674873
RF	0.579149	0.732736	0.939601	0.553152	0.655022
mlp	0.563799	0.673253	0.916870	0.549853	0.604501
KNN	0.457307	0.614943	0.890826	0.430718	0.534908
Baseline	0.872402	0.500000	0.872402	1.000000	0.500000

Fig. Model Results on Sensor, Feature, and Permission Dataset using SMOTE

Appendix - Model Results

	Balanced Accuracy		ROC AUC Score		F1 Score	
	Random Oversampling	Duplicate Oversampling	Random Oversampling	Duplicate Oversampling	Random Oversampling	Duplicate Oversampling
Logistic Regression	0.89	0.89	0.92	0.93	0.88	0.88
Support Vector Classifier	0.89	0.89	0.92	0.92	0.87	0.87
KNN	0.84	0.85	0.88	0.84	0.86	0.87
Random Forest	0.89	0.89	0.97	0.96	0.87	0.87
Gradient Boosting Classifier	0.92	0.94	0.97	0.95	0.93	0.94
XGBoost Classifier	0.85	0.85	0.89	0.90	0.87	0.88
Multilayer Perceptron	0.96	0.95	0.99	0.99	0.98	0.97

Fig. Model Results on CFG of 804 pairs

	Balanced Accuracy		ROC AUC Score		F1 Score	
	Random Oversampling	Duplicate Oversampling	Random Oversampling	Duplicate Oversampling	Random Oversampling	Duplicate Oversampling
Logistic Regression	0.79	0.74	0.77	0.77	0.88	0.86
Support Vector Classifier	0.76	0.77	0.85	0.80	0.77	0.77
KNN	0.64	0.62	0.62	0.67	0.83	0.82
Random Forest	0.70	0.64	0.87	0.89	0.90	0.91
Gradient Boosting Classifier	0.62	0.67	0.79	0.81	0.85	0.91
XGBoost Classifier	0.67	0.71	0.86	0.86	0.90	0.91
Multilayer Perceptron	0.55	0.56	0.74	0.56	0.91	0.74

Fig. Model Results on DFG of 804 pairs

Appendix - Model Results

	Balanced Accuracy		ROC AUC Score		F1 Score	
	Random Oversampling	Duplicate Oversampling	Random Oversampling	Duplicate Oversampling	Random Oversampling	Duplicate Oversampling
Logistic Regression	0.86	0.96	0.94	0.98	0.93	0.96
Support Vector Classifier	0.94	0.93	0.98	0.96	0.96	0.96
KNN	0.65	0.63	0.64	0.66	0.73	0.86
Random Forest	0.94	0.92	0.96	0.97	0.93	0.95
Gradient Boosting Classifier	0.94	0.89	0.97	0.97	0.96	0.95
XGBoost Classifier	0.91	0.86	0.97	0.96	0.94	0.93
Multilayer Perceptron	0.78	0.61	0.91	0.78	0.72	0.92

Fig. Model Results on both CFG and DFG of 804 pairs

	Balanced Accuracy		ROC AUC Score		F1 Score	
	Random Oversampling	Duplicate Oversampling	Random Oversampling	Duplicate Oversampling	Random Oversampling	Duplicate Oversampling
Logistic Regression	0.61	0.63	0.61	0.66	0.62	0.70
Support Vector Classifier	0.49	0.61	0.46	0.65	0.34	0.69
KNN	0.55	0.53	0.57	0.57	0.83	0.85
Random Forest	0.50	0.62	0.57	0.63	0.21	0.47
Gradient Boosting Classifier	0.61	0.52	0.64	0.54	0.60	0.12
XGBoost Classifier	0.61	0.62	0.64	0.64	0.61	0.69
Multilayer Perceptron	0.51	0.53	0.51	0.60	0.31	0.73

Fig. Model Results on CFG of 7997 pairs