CSC3059 Malware Analysis Questions and Answers

Niall McLaughlin

1. Automatic Malware Detection

(a) The Table below shows the frequency of code-based properties of 1000 samples each of malware and normal Android application code respectively. From the table calculate the following probability values for the code property getNetworkOperator: P(Ri=1), P(Ri=0), P(C=M|Ri=1), P(C=M|Ri=0), P(C=B|Ri=1) and P(C=B|Ri=0) Show your working out.

Code Properties	Malware Frequency	Benign Frequency
getSubscriberID	742	42
getSimSerialNumber	455	35
DexClassLoader	152	16
createSubprocess	169	0
.jar (secondary payload)	252	87
KeySpec (code encryption)	254	99
getNetworkOperator	125	754
Chown	107	5

Table 1. Malware and Benign code-based property frequency.

What do we know?

Dataset

1000 Malware Samples
1000 Benign (Clean) Samples

Feature

getNetworkOperator

Malware Frequency 125 / 1000
Benign Frequency   754 / 1000

```
P(R_i = 0) =       = (2000 – (125 + 754)) / (1000 + 1000)
                   = (2000 – 879) / 2000
                   = 1121 / 2000
                   = 0.5605
```

P(R_i = 0)
Probability feature does not occur

```
P(R_i = 1) =       = (125 + 754) / (1000 + 1000)
                   = 879 / 2000
                   = 0.4395
```

P(R_i = 1)
Probability feature occurs

```
P(C=B|R_i=0) =     = (1000 – 754) / ((1000 – 754) + (1000 – 125))
                   = 246 / (246 + 875)
                   = 246 / 1121
                   = 0.2195
```

P(C=B|R_i=0)
Probability that sample is bengin, given that feature does not occur

```
P(C=B|R_i=1) =     = 754 / (125 + 754)
                   = 754 / 879
                   = 0.8578
```

P(C=B|R_i=1)
Probability that sample is benign, given that feature occurs

```
P(C=M|R_i=0) =     = (1000 – 125) / ((1000 – 754) + (1000 – 125))
                   = 875 / (246 + 875)
                   = 875 / 1121
                   = 0.7805
```

P(C=M|R_i=0)
Probability that sample is malware, given that feature does not occur

```
P(C=M|R_i=1) =     = 125 / (125 + 754)
                   = 125 / 879
                   = 0.1422
```

P(C=M|R_i=1)
Probability that sample is malware, given that feature occurs

(b) The mutual information (MI) value for getSubscriberID is 0.28
Using your answers from part (a), determine whether getNetworkOperator is a more, or less, discriminative feature by calculating its mutual information using the formula:

$$MI(R_i, C) = \sum_{r=0}^{1} \sum_{c \in \{mal, ben\}} P(R_i = r) P(C = c | R_i = r) \log_2 \left[\frac{P(C = c | R_i = r)}{P(C = c_j)} \right]$$

Show your working out.

- We have already worked out most of the terms in Question 1 (a).
- We still need P(C=B) and P(C=M)

$$P(C = B) = 1000 / (1000 + 1000)$$
$$P(C = B) = 0.5$$

$$P(C = M) = 1000 / (1000 + 1000)$$
$$P(C = M) = 0.5$$

$$MI(R_i,C)=P(R_i=0)\left\{P(C=B|R_i=0)\log_2\left[\frac{P(C=B|R_i=0)}{P(C=B)}\right]+P(C=M|R_i=0)\log_2\left[\frac{P(C=M|R_i=0)}{P(C=M)}\right]\right\}$$
$$+P(R_i=1)\left\{P(C=B|R_i=1)\log_2\left[\frac{P(C=B|R_i=1)}{P(C=B)}\right]+P(C=M|R_i=1)\log_2\left[\frac{P(C=M|R_i=1)}{P(C=M)}\right]\right\}$$

```
MI = 0.56 * (0.22 * log2(0.22/0.5) + 0.78 * log2(0.78/0.5)) +
     0.44 * (0.86 * log2(0.86/0.5) + 0.14 * log2(0.14/0.5))


MI = 0.56 * (0.22 * log2(0.44) + 0.78 * log2(1.56)) +
     0.44 * (0.86 * log2(1.72) + 0.14 * log2(0.28))


MI = 0.56 * (0.22 * -1.18 + 0.78 *  0.64) +
     0.44 * (0.86 *  0.78 + 0.14 * -1.84)


MI = 0.56 * (-0.2596 +  0.4992) +
     0.44 * ( 0.6708 + -0.2576)

MI = 0.316
```

- MI for getSubscriberID = 0.28
- MI for getNetworkOperator = 0.32

Therefore getNetworkOperator is more discriminative than getSubscriberID

(c) An unknown executable file is analysed and the following features are detected: **GetSubscriberID**, **DexClassLoader**, **keySpec**, **GetNetworkOperator** and **Chown**. Using the information in Table 1, calculate the probabilities that this executable file is malware or benign and hence state the final classification decision. Show your working out.

Feature Vector (R) = (1,0,1,0,0,1,1,1)

$(R_1=1|C=M) = 742/1000 = 0.742$

$(R_2=0|C=M) = 1-(455/1000) = 0.545$

$(R_3=1|C=M) = 152/1000 = 0.152$

$(R_4=0|C=M) = 1-(169/1000) = 0.831$

$(R_5=0|C=M) = 1-(252/1000) = 0.748$

$(R_6=1|C=M) = 254/1000 = 0.254$

$(R_7=1|C=M) = 125/1000 = 0.125$

$(R_8=1|C=M) = 107/1000 = 0.107$

$$P(C=M|R) = 0.742*0.545*0.152* 0.831*0.748*0.254* 0.125*0.107 = 0.0001298$$

$(R_1=1|C=B) = 42/1000 = 0.042$

$(R_2=0|C=B) = 1-(35/1000) = 0.965$

$(R_3=1|C=B) = 16/1000 = 0.016$

$(R_4=0|C=B) = 1-(0/1000) = 1$

$(R_5=0|C=B) = 1-(87/1000) = 0.913$

$(R_6=1|C=B) = 99/1000 = 0.099$

$(R_7=1|C=B) = 754/1000 = 0.754$

$(R_8=1|C=B) = 5/1000 = 0.005$

$$P(C=B|R) = 0.042*0.965*0.016* 1*0.913*0.099*0.754* 0.005 = 2.2e-7$$

Final decision – file is more likely to be Malware because probability of malware given features is higher than probability of benign given features i.e. P(M|R=r) is greater than P(B|R=r)

During training, a neural network is used to classify several malware files. Table 2 shows the network's prediction for each file and the correct label. For each malware file, calculate the value of the network's cost function. Show your working out.

Filename	Network's prediction	Correct Label
1.qiqhsj9bzn.apk	0.7	1
2.gmcl3ueS7m.apk	0.2	0

Table 2. Network's prediction and correct label for two malware samples.

The cost function is given by the equation

Cost(y,y*) = -y ln(y*) – (1-y) ln (1-y*)

Where the correct answer is y and the network’s prediction is y*

```
File 1
Correct label = 1
Prediction = 0.7
Cost(1,0.7) = -1*ln(0.7) – (1-1)*ln(1-0.7)
Cost(1,0.7) = -1*ln(0.7) – 0*ln(1-0.7)
Cost(1,0.7) = -1*ln(0.7)
Cost(1,0.7) = -1*-0.36
Cost = 0.36
```

```
File 2
Correct label = 0
Prediction = 0.2
Cost(0,0.2) = -0*ln(0.2) – (1-0)*ln(1-0.2)
Cost(0,0.2) =             – (1-0)*ln(1-0.2)
Cost(0,0.2) = –1*ln(0.8)
Cost(0,0.2) = –1*-0.22
Cost = 0.22
```