Detection of Unknown Malicious Code via Machine Learning

Robert Moskovitch (Jointly with Dima Stopel, Clint Feher, Nir Nissim and Yuval Elovici)

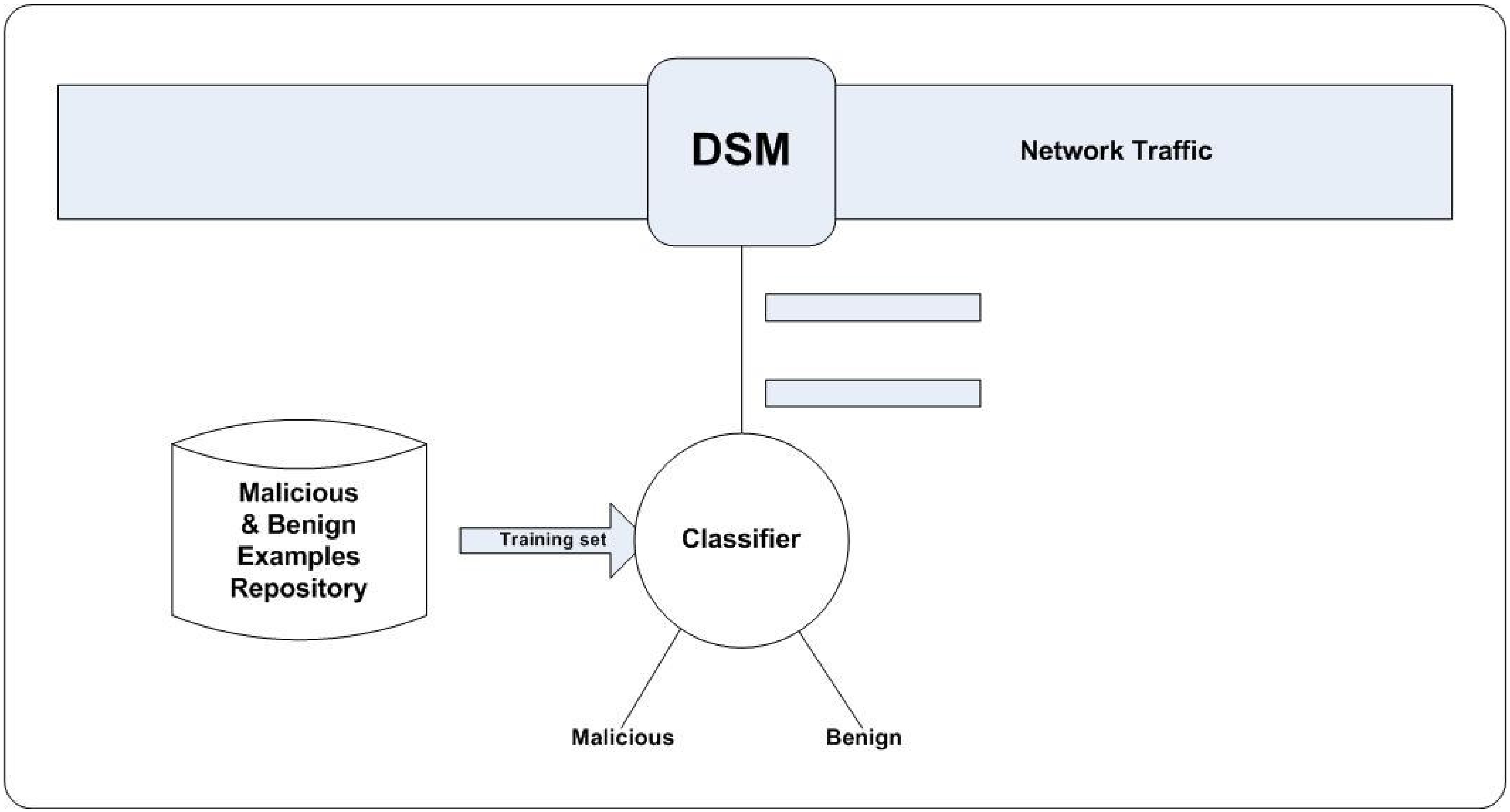
Deutsche Telekom Laboratories at Ben Gurion Univeristy, Beer Sheva, Israel.

OWASP, IDC, Herzeliya, September 14, 2008

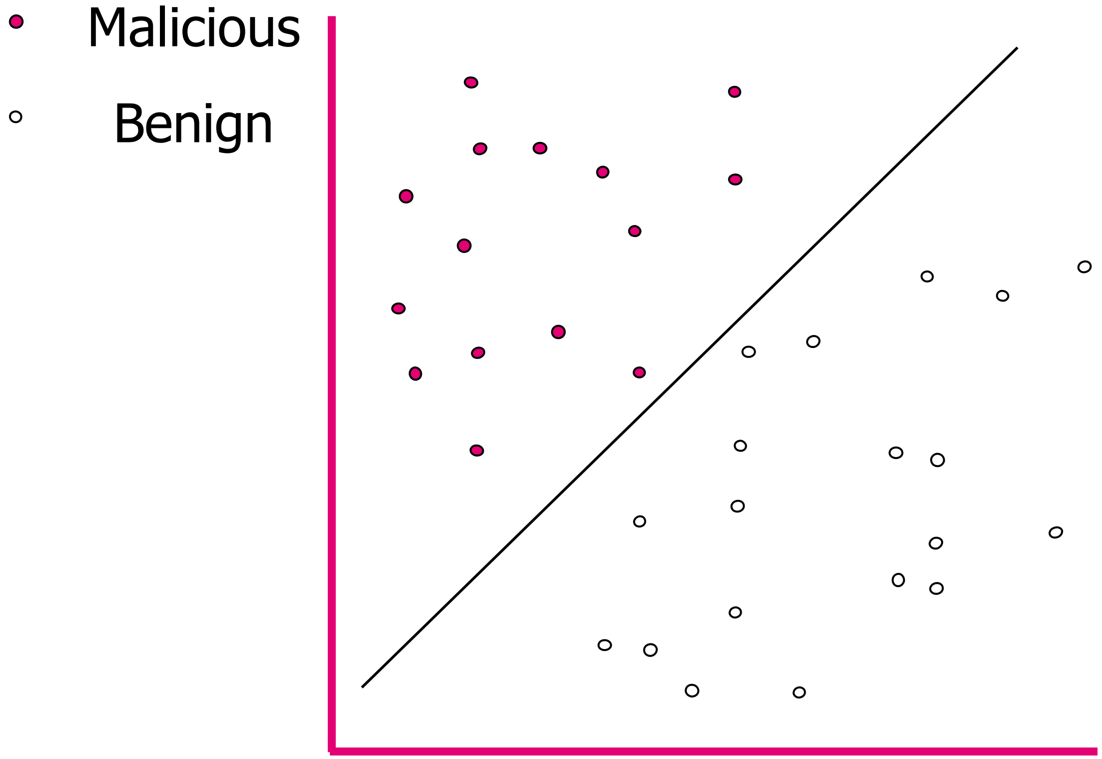
Motivation and Goals

- Currently most Anti Viruses perform a signature based detection.
- This method is very accurate, but is helpless against unknown malcodes.
- Some Anti Viruses use heuristics which extend their capabilities.
- Detecting unknown malicious code (which can't be detected by an anti-virus) is an important feature.
- Recent approaches suggest to employ machine learning for unknown malicious code detection [Abou-Assaleh et al, 04; Kolter, JMLR06].
- The GOAL:
  - Develop a generalized anti-virus
  - To use it for Malcode acquisition over network traffic

Detecting Malcodes in Network Traffic

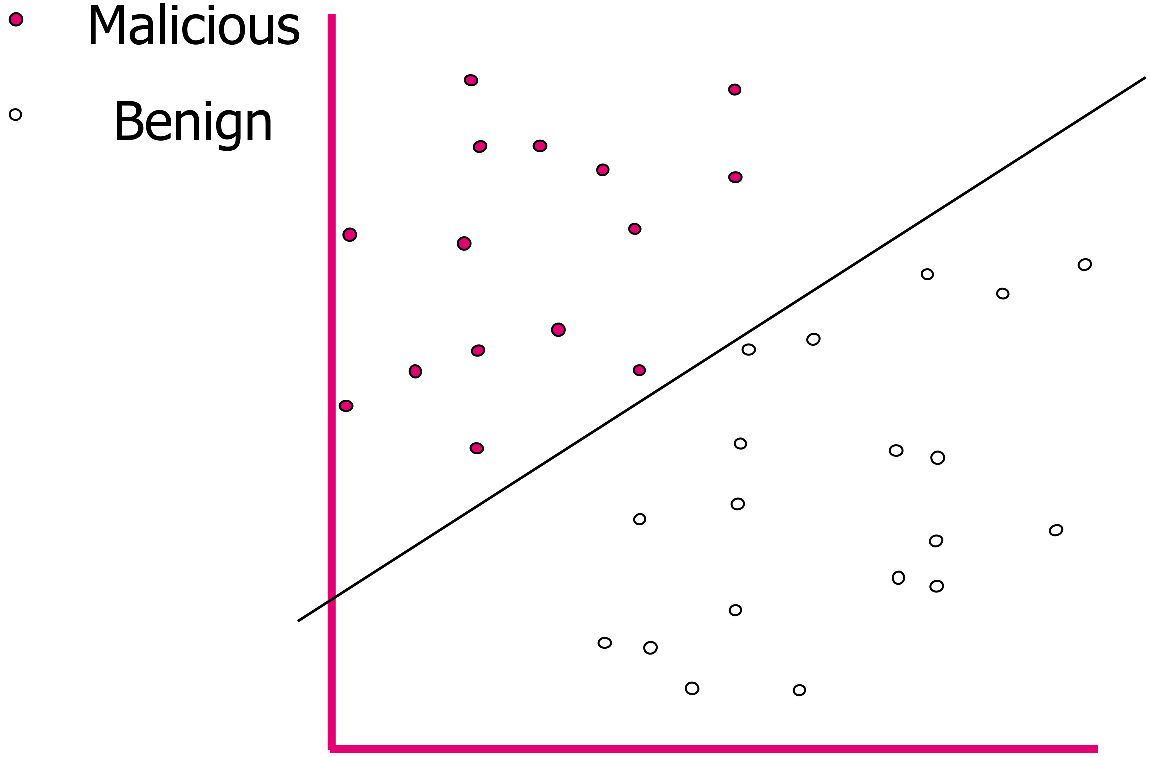


Linear Classifiers



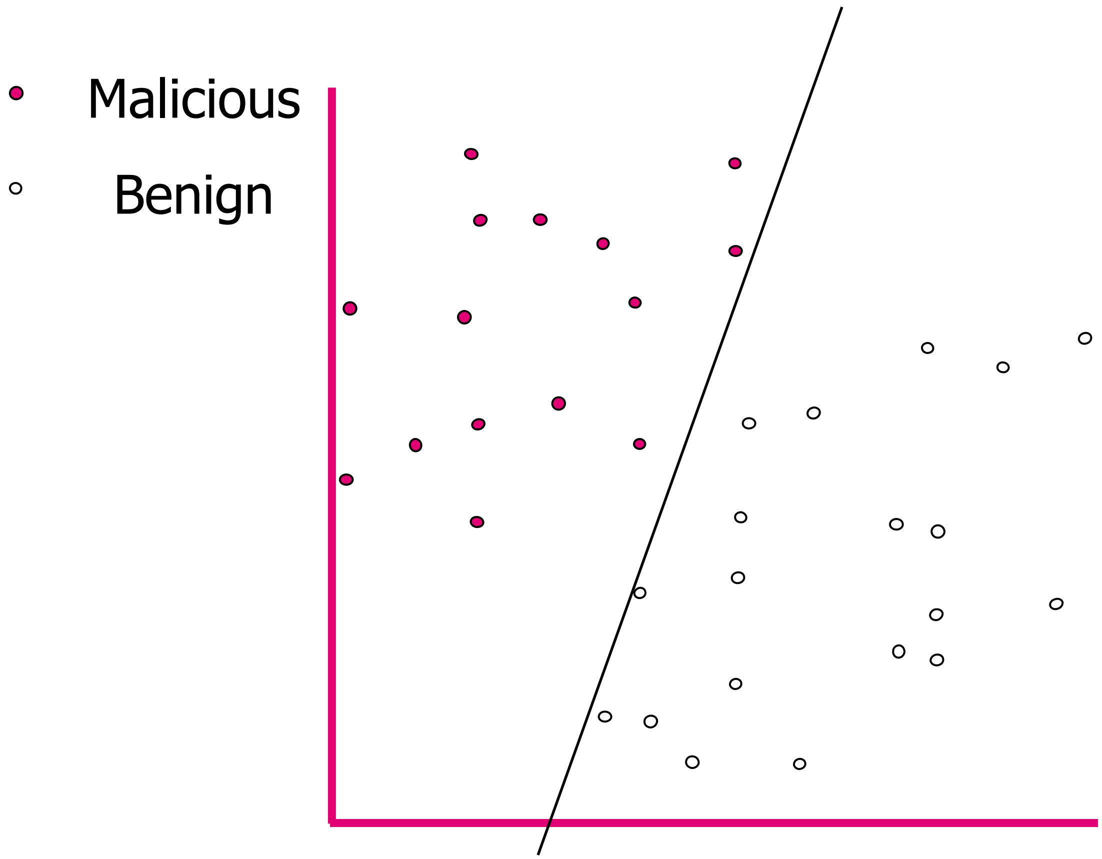
How would you classify this data?

Linear Classifiers



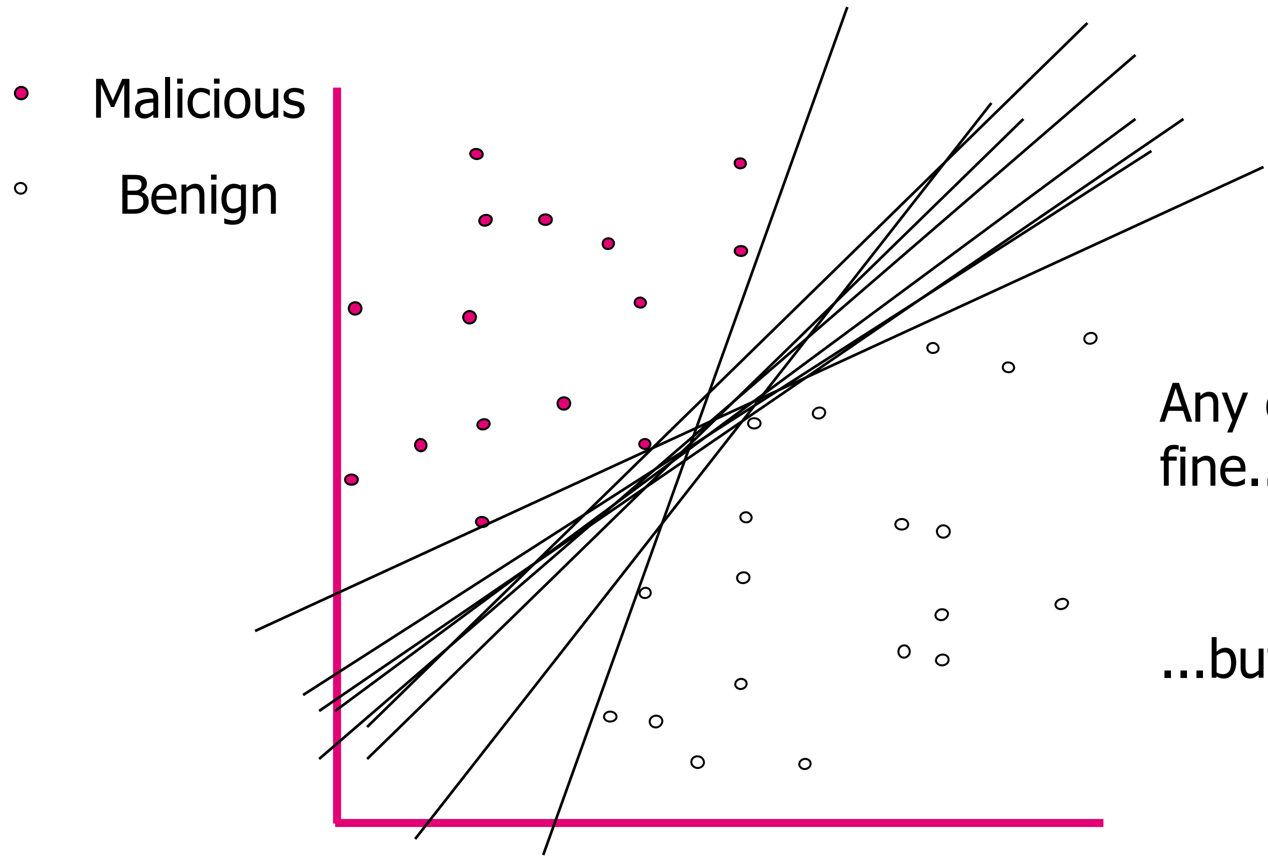
How would you classify this data?

Linear Classifiers



How would you classify this data?

Linear Classifiers



Any of these would be fine..

...but which is the best?

Concepts from Text Categorization

- Classifying text to categories.
- Common task is *spam email filtering* (Spam/Good).
- Given a set of labeled documents a *supervised learning* is applied, after which a *new* document can be classified accordingly.
- Commonly a *vocabulary* of *words* is *extracted* from the document collection. Each term is *stopped* and *stemmed* and represented by: $w_i = n_i / n_{max}$.
- Then a *tfidf* $= w_i * \log(N/n_i)$ *measure* is computed to represent the existence of the word in the entire collection.

t_1	t_2	t_3	t_4	t_5	*label*
0.4	0.5	0.2	…	0	Spam
0.5	0.7	0.1	…	0.5	Good
0.2	0.9	0.5	…	0.2	Spam
0.4	0.3	0.7	…	0.6	Good

$$Given:$$

$$Ci = \{w_1 t_1, w_2 t_2 \ldots w_N t_N\} \quad and \quad dj = \{v_1 t_1, v_2 t_2 \ldots v_N t_N\}$$

$$Sim(Ci, dj) = \frac{\vec{C}_i \times \vec{d}_j}{|\vec{C}_i| \times |\vec{d}_j|} = \frac{\sum_{n=1}^{N} (w_n \times v_n)}{\sqrt{\sum_{n=1}^{N} w_n^2 \times \sum_{n=1}^{N} v_n^2}}$$

$$where \quad 0 \le Sim(Ci, dj) \le 1$$

Analogous of Malcode Detection as Text Categorization

- Classifying *Malicious Code* can be analogous to *Text Categorization*.
- Texts ←→ Malicious Code & Benign (Files)
- Words ←→ Code expressions
- Then weighting functions, such as *tf* or *tfidf* can be used.

Extracting malcode features

- There are explicit fixed properties given within the *header* of the file.
- There are several common approaches:
  - N-Grams
  - PE header
  - Imported DLLs and DLLs' functions.
  - Disassembly terms
  - And more…
- Common *N-grams* method:
  - Sequence of characters
  - Collect statistics using a "sliding window" of length n.
  - Build profiles (signatures) of most frequent *n-grams*
- In order to reduce the amount of features, *feature selection* approaches can be applied.

How do n-grams work?

`Marley was dead: to begin with.  There is no doubt whatever about that.  …`

(from Christmas Carol by Charles Dickens)

n = 3

```
Mar    1
Arl    1
rle    1
ley    1
ey_    1
y_w    1
_wa    1
was    1
…
```

NGrams Extraction Example

```
00000000: 4D 5A 90 00 03 00 00 00|04 00 00 00 FF FF 00 00 | MZ█a╕aaa♪aaa██aa
00000010: B8 00 00 00 00 00 00 00|40 00 00 00 00 00 00 00 | ¸aaaaaaa@aaaaaaa
00000020: 00 00 00 00 00 00 00 00|00 00 00 00 00 00 00 00 | aaaaaaaaaaaaaaaa
00000030: 00 00 00 00 00 00 00 00|00 00 00 00 E0 00 00 00 | aaaaaaaaaaaaXaaa
00000040: 0E 1F BA 0E 00 B4 09 CD|21 B8 01 4C CD 21 54 68 | ┤º┤a´Ü,!¸ÌL,!Th
00000050: 69 73 20 70 72 6F 67 72|61 6D 20 63 61 6E 6E 6F | is program canno
00000060: 74 20 62 65 20 72 75 6E|20 69 6E 20 44 4F 53 20 | t be run in DOS
```

n = 3

4D5A90

5A9000

900003

000300

030000

000000

000004

000400

…

Dataset

- We acquired the malicious files from the VX Heaven website - 7688 malicious files.
- The benign set, including executable and DLL (Dynamic Linked Library) files, were gathered from machines running Windows XP operating system on our campus, containing 22,735 files.
- The Kaspersky anti-virus program was used to verify that these files were completely virus-free, or malicious.
- Creating Vocabularies (TF Vector)

N-Grams	Vocabulary Size
3-gram	16,777,216
4-gram	1,084,793,035
5-gram	1,575,804,954
6-gram	1,936,342,220

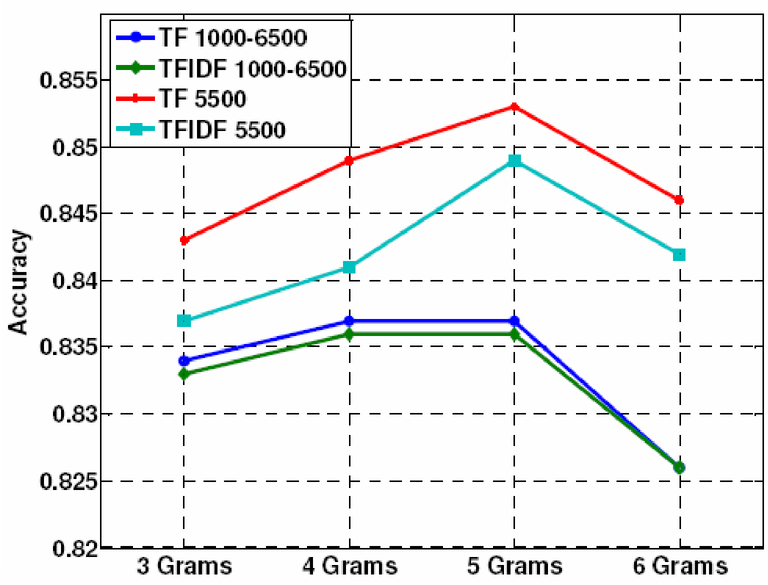
- Calculating TF and TFIDF For Each Document
- Preliminary Feature Selection was based on the DF measure:
  - Top 5500 terms
  - Top 1000 – 6500 terms

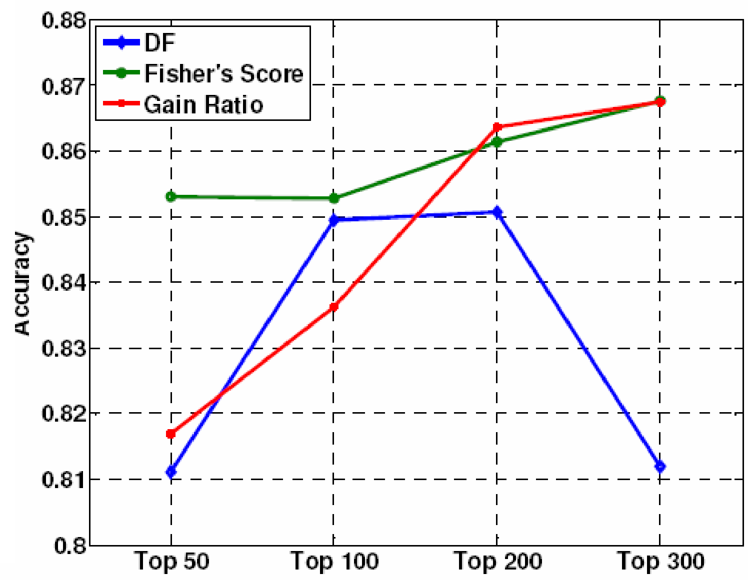
Experiment 1
determine best conditions

- A wide and comprehensive set of evaluation runs was designed:
  - All the combinations of the optional settings: N-grams, Feature Selection and Top Selection.
  - For each of the aspects
  - For all 8 classifiers.
- Training set ≠ Testing set (represents unknown files).

Global Feature Selection vs. n-grams

- Mean accuracies quite similar: Top 5500, TF, 5-gram.
- Top300 for GainRatio and FisherScore outperform.
- FisherScore in general was better.





Classifiers

- Under the best conditions presented above, the classifiers that achieved the highest accuracies, with lowest false positive rates, are:
  - Boosted Decision Tree
  - Decision Tree
  - Artificial Neural Network

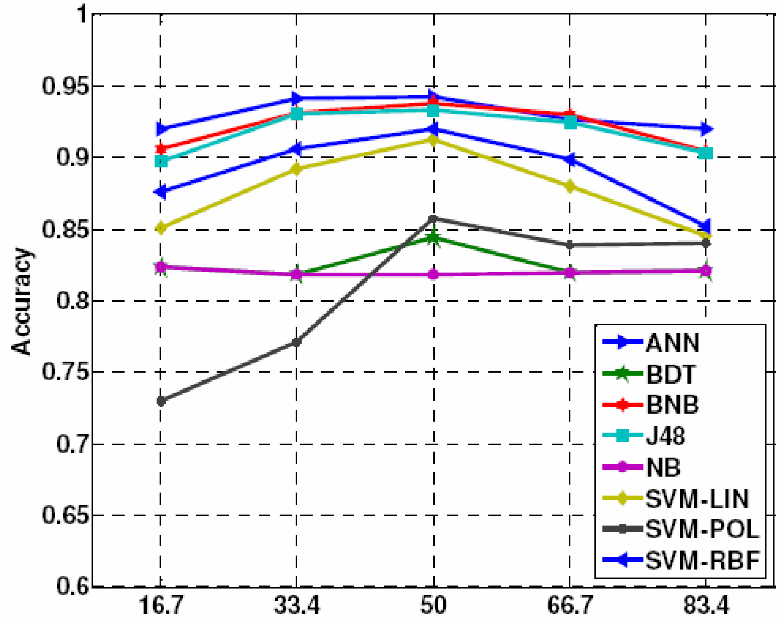
Classifier	Accuracy	FP	FN
ANN	**0.941**	0.033	0.134
DT	**0.943**	0.039	0.099
NB	0.697	0.382	0.069
BDT	**0.949**	0.040	0.110
BNB	0.697	0.382	0.069
SVM-lin	0.921	0.033	0.214
SVM-poly	0.852	0.014	0.544
SVM-rbf	0.939	0.029	0.154

Experiment 2.
Investigation of the imbalance problem

- Performed under the best conditions found at experiment 1.
- 5 levels of Malicious Files Percentage (MFP) in the training set (16.7, 33.4, 50, 66.7, 83.4).
- 17 levels of MFP for testing sets : (5, 7.5, 10, 12.5, 15, 20, 30, 40, 50, 60, 70, 80, 85, 87.5, 90, 92.5, 95).
- 85 (17*5) runs for each of the 8 classifiers.

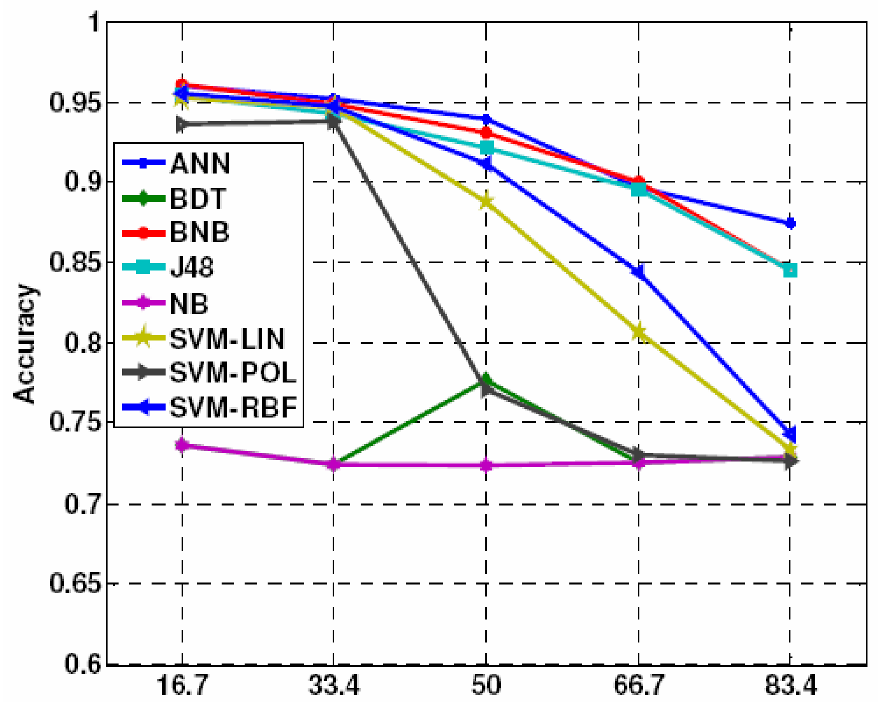
Training-Set Malcode Percentage

The most accurate & relatively stable classifiers across all the MFP's: ANN, BNB, DT

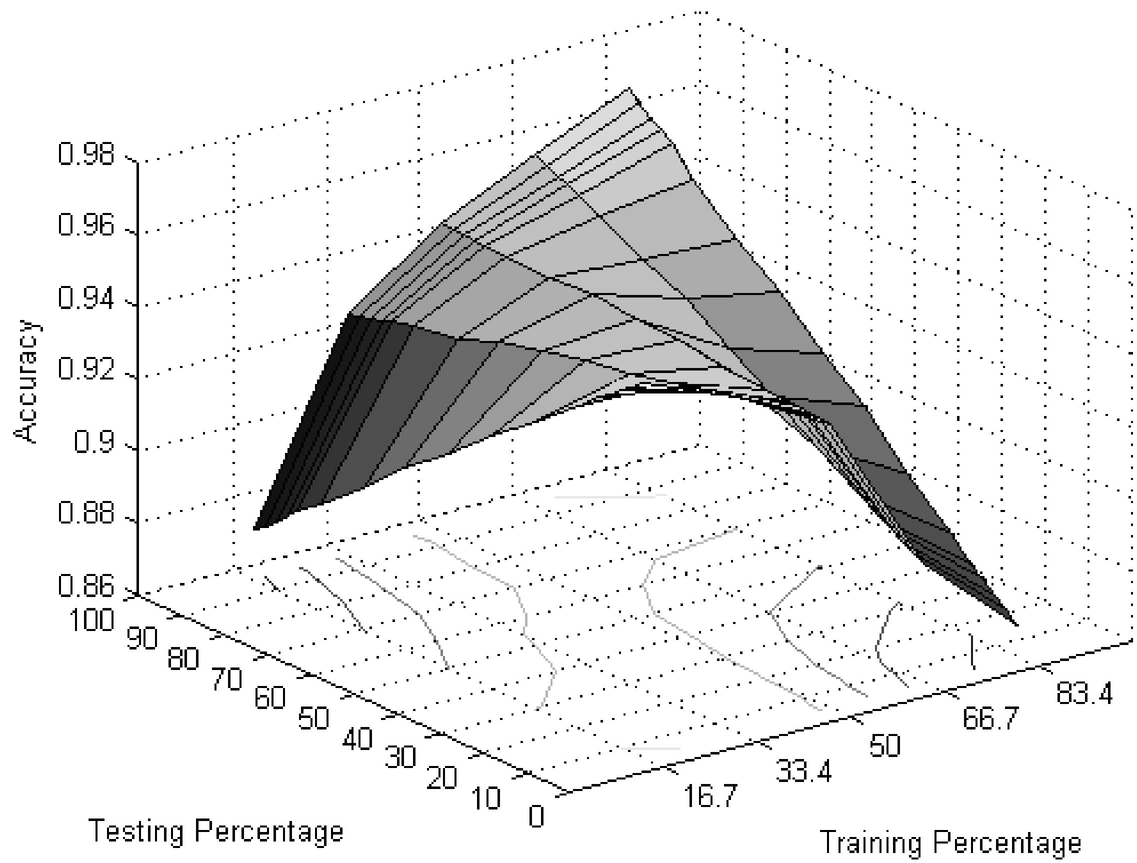


10% Malcode Percentage in the Test Set

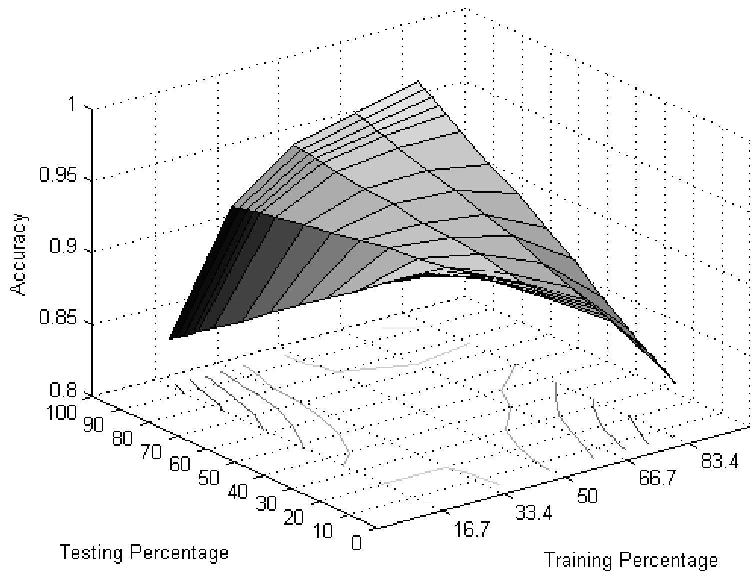
- Is The realistic scenario at most networks.
- Different MFP of training set, for fixed testing set's MFP (10%).
- The most accurate results achieved at the level of 16.7 for training set.



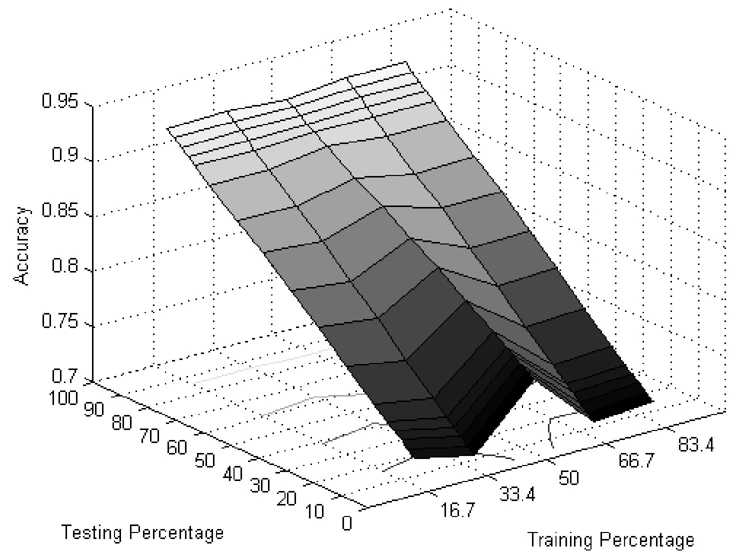
ANN - 3D results representation



DT – 3D results presentation

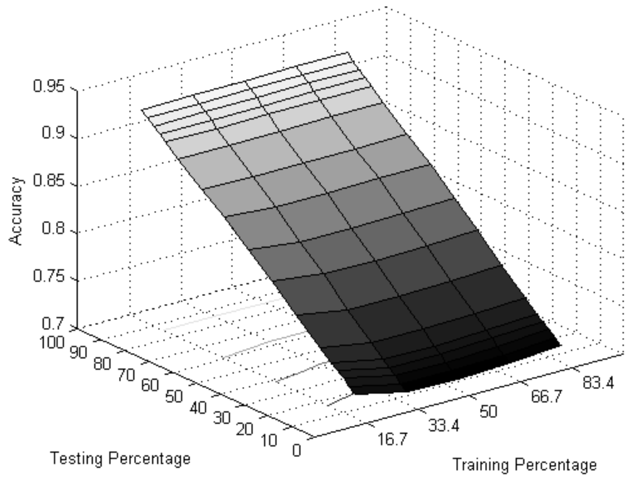


DT

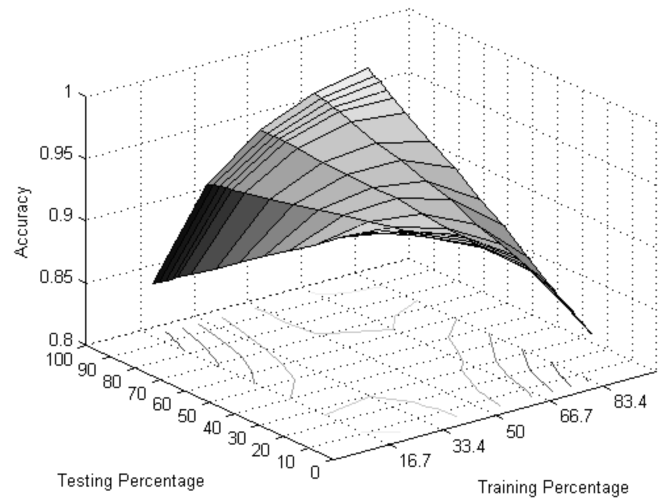


BDT

NB - 3D results presentation

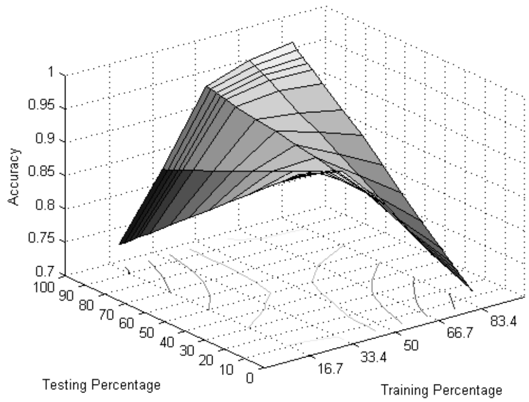


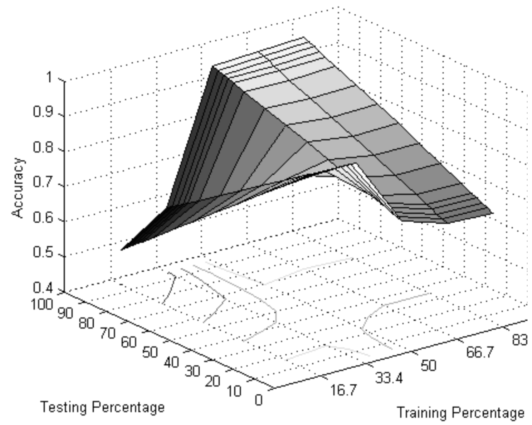
NB

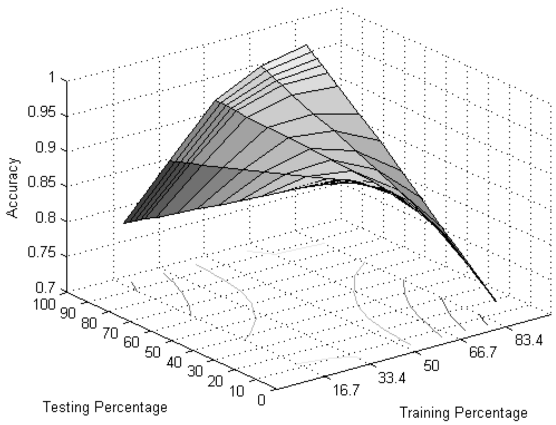


BNB

SVM – 3D results presentation







SVM-LIN

SVM-POL

SVM-RBF

- More details can be found in several publications on my website at http://medinfo.ise.bgu.ac.il/medLab/MembersHomePages/homePage_Robert.htm

- For any questions and any additional information do not hesitate to email me: robertmo-replaceby(at)-bgu.ac.il