ПОРІВНЯННЯ АЛГОРИТМІВ МАШИННОГО НАВЧАННЯ ДЛЯ ПОШУКУ ШКІДЛИВОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Підготувала: Мєтєлєва Любов, ФБ-12

Постановка проблеми

У високонавантажених систем є свої особливості, що вимагають відмінного від інших комп’ютерних систем підходу.

Для таких систем є недопустимим виділення значних ресурсів для запуску підозрілих файлів у “пісочниці”, потокової перевірки виконуваних файлів антивірусом чи для використання інших методів поведінкового аналізу.

Статистичний аналіз

+ вимагає менше ресурсів комп’ютера
- вимагає постійного оновлення
- може не враховувати суттеві ознаки

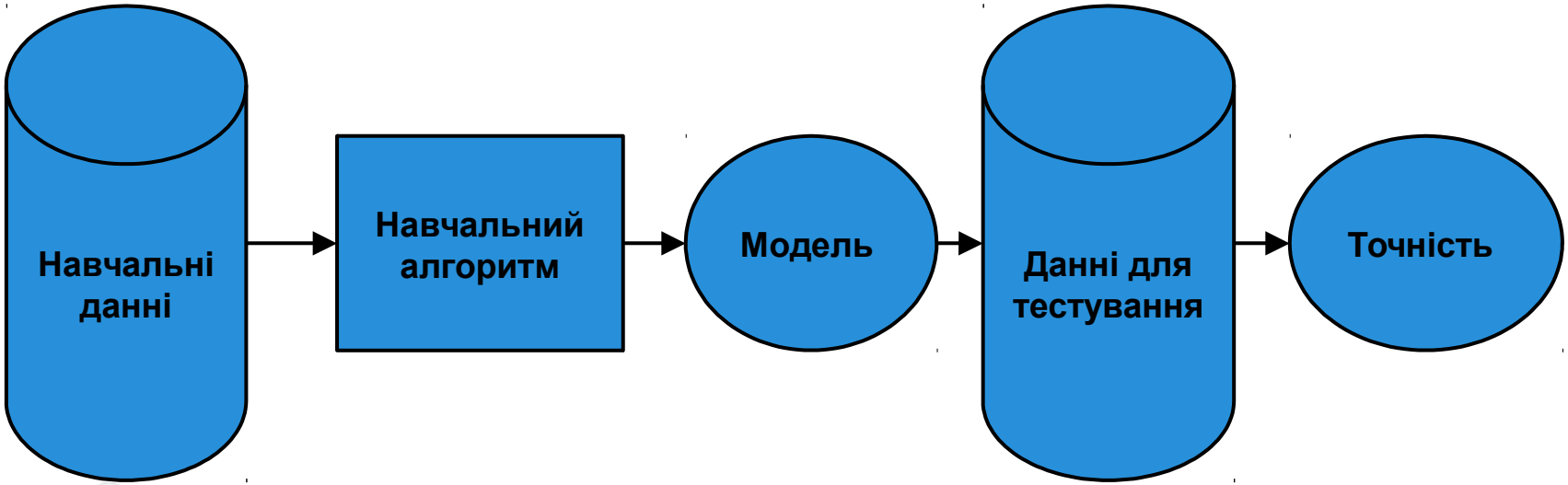
Поведінковий аналіз

- вимагає більше ресурсів комп’ютера
+ може застосовуватися без постійного оновлення
+ демонструє найкращі результати на практиці

Процес роботи

Навчання

Тестування



Навчальна вибірка

55 000 бінарних виконуваних файлів Win32 (EXE або DLL), поділені на три групи: "чисті", "інфіковані" та "невідомі".
На основі методу головних компонент (principal components analysis, PCA) в PE заголовках відокремлено 20 атрибутів:

- NumberOfSections
- SizeOfCode
- SizeOfInitializedData
- SizeOfUninitializedData
- AddressOfEntryPoint
- BaseOfCode
- BaseOfData
- ImageBase
- SizeOfImage
- SizeOfHeaders
- SizeOfStackReserve
- SizeOfStackCommit
- SizeOfHeapReserve
- SizeOfHeapCommit
- NumberOfRvaAndSizes
- ImageVersion
- IatRVA
- DebugSize
- ExportSize
- ResourceSize

Тестова вибірка

Розмір вибірки: 18 500 файлів
Для формування вибірок застосовувались загальнодоступні бази шкідливого програмного забезпечення:

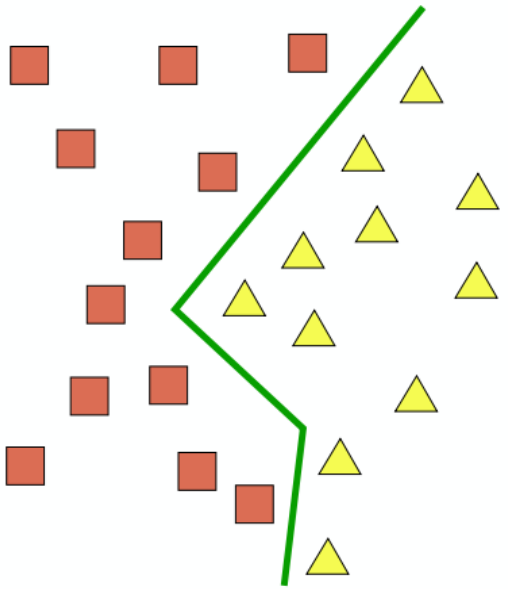
- VirusShare.com
- vxheaven.org/vl.php

У якості ’чистих’ файлів використовувались файли з дистрибутивів ОС Microsoft Windows 7 та 8 x86

Досліджені алгоритми машинного навчання

В роботі побудовано класифікатори на базі наступних:

- штучні нейронні мережі;
- дерева прийняття рішень (J48);
- список прийняття рішень (PART);
- метод опорних векторів (SVM);
- інші.

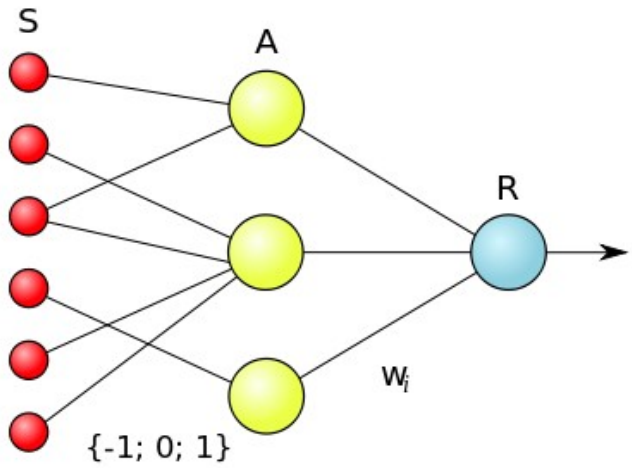


Штучні нейронні мережі

На основі багатошарового перцептрону.
Найкращий результат при 500 нейронах у приховному шарі

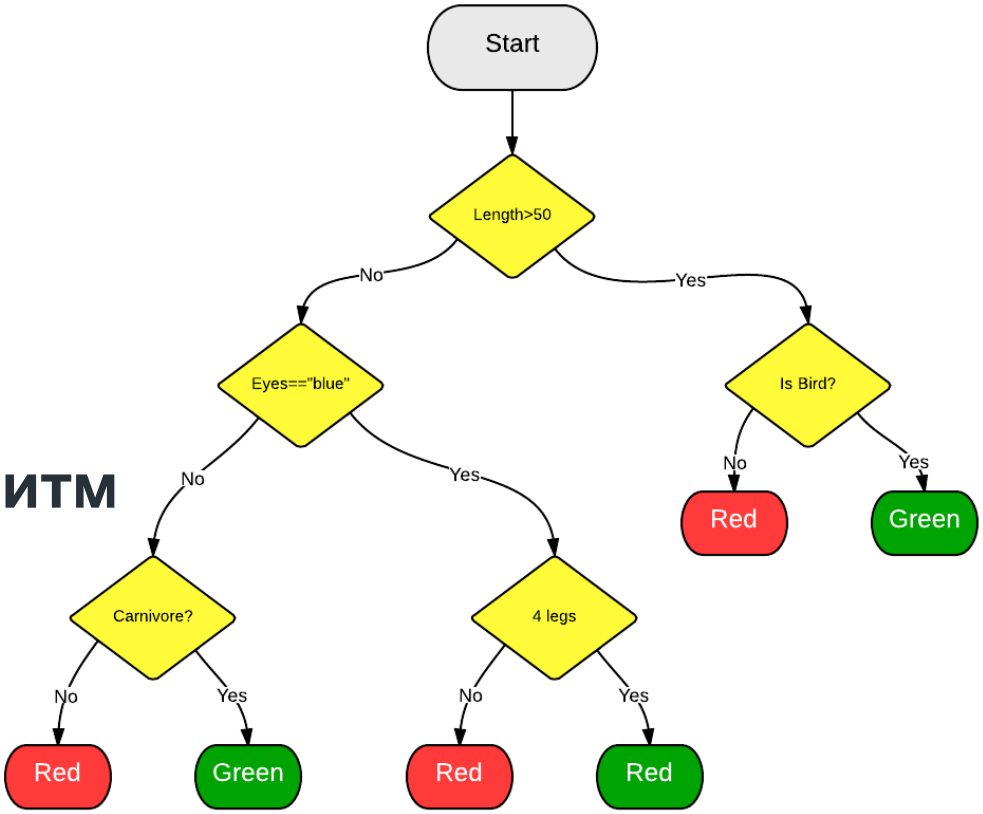
weka.classifiers.functions.MultilayerPerceptron -L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a

Точність: 50.8349 %



Дерева прийняття рішень

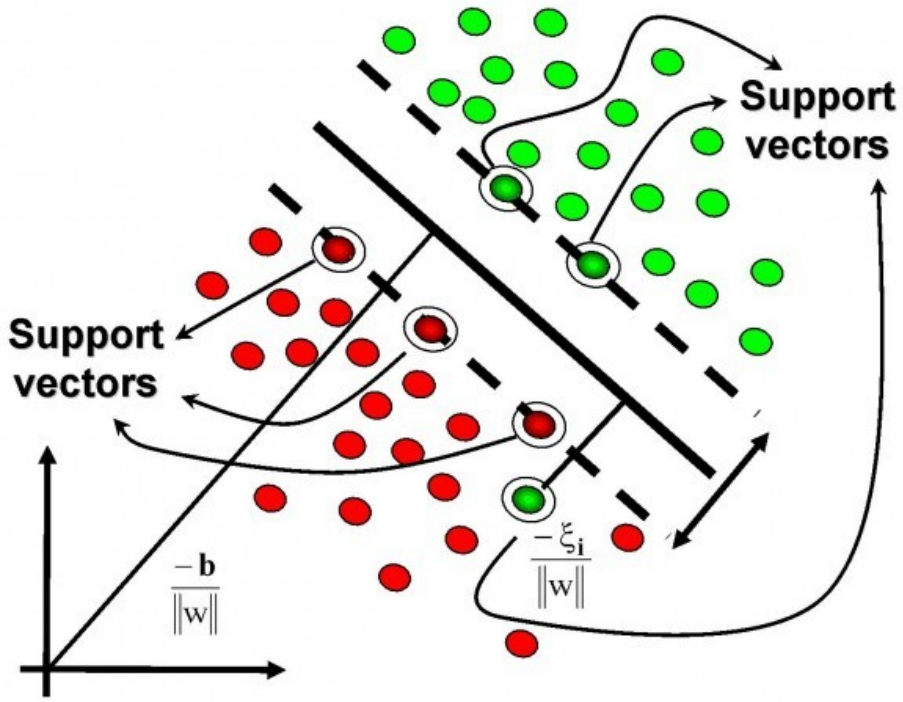
J48 алгоритм
Точність: 68.2839 %
Кількість листів : 172
Розмір дерева : 343



PART decision list алгоритм
Точність: 65.6568 %
Кількість правил : 99

Метод опорних векторів

SMO algorithm
(Sequential minimal optimization)
Кількість оцінок ядра: 28746503
Точність: 75.2026 %



Результати доследжння

Алгоритм	Вірно класифіковані		Невірно класифіковані	
Multilayer Perceptron	9346	50.8349 %	9039	49.1651 %
J48	12554	68.2839 %	5831	31.7161 %
SVM	13826	75.2026 %	4559	24.7974 %
PART	12071	65.6568 %	6314	34.3432 %

Підсумок

Для підвищення ефективності уже відомих алгоритмів можна використати метод бустингу.

В результаті досліджень, отримуємо статистичний метод класифікації програмного забезпечення для застосування в високонавантажених системах мережевої фільтрації.

Дякую за увагу!