ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

ΕΥΦΥΕΙΣ ΤΕΧΝΟΛΟΓΙΕΣ ΔΙΑΔΙΚΤΥΟΥ - WEBINTELLIGENCE

Προσδιορισμός προηγμένων κυβερνοεπιθέσεων χρησιμοποιώντας μηχανική μάθηση

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΙΩΑΚΕΙΜ ΣΤΑΥΡΟΥ

Επιβλέπων : Χρήστος Ηλιούδης
Καθηγητής ΔΙ.ΠΑ.Ε.

Θεσσαλονίκη, Σεπτέμβριος 2021

ΔΙΕΘΝΕΣ ΠΑΝΕΠΙΣΤΗΜΙΟ ΤΗΣ ΕΛΛΑΔΟΣ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ ΕΥΦΥΕΙΣ ΤΕΧΝΟΛΟΓΙΕΣ ΔΙΑΔΙΚΤΥΟΥ – WEB INTELLIGENCE

Προσδιορισμός προηγμένων κυβερνοεπιθέσεων χρησιμοποιώντας μηχανική μάθηση

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΙΩΑΚΕΙΜ ΣΤΑΥΡΟΥ

Επιβλέπων : Χρήστος Ηλιούδης
Καθηγητής ΔΙ.ΠΑ.Ε.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή στις Choose a date.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Όνομα Επώνυμο	Όνομα Επώνυμο	'Ονομα Επώνυμο
Choose an item. ΔΙ.ΠΑ.Ε.	Choose an item. ΔΙ.ΠΑ.Ε.	Choose an item. ΔΙ.ΠΑ.Ε.

Θεσσαλονίκη, Choose a date

(Υπογραφή)

.....

Click here to enter text.

Click here to enter text.

Περίληψη

Οι αλγόριθμοι Εξόρυξης Γνώσης και Μηχανικής Μάθησης εφαρμόζονται σε πληθώρα δραστηριοτήτων της ψηφιακής καθημερινότητας του ανθρώπου ώστε να προβλέψουν μελλοντικές καταστάσεις. Τα τελευταία χρόνια εφαρμόζονται με επιτυχία και για την αναγνώριση εισβολών και παραβιάσεων σε πληροφοριακά και επικοινωνιακά συστήματα. Η άμεση αναγνώριση τέτοιων παραβιάσεων είναι ζωτικής σημασίας για οποιονδήποτε οργανισμό και επιχείρηση. Η ανάγκη έχει προσελκύσει το ενδιαφέρον τόσο της επιστημονικής κοινότητας της εξόρυξης γνώσης και της μηχανικής μάθησης όσο και της βιομηχανίας της Πληροφορικής. Ως αποτέλεσμα, έχουν προταθεί διάφορες τεχνικές και μέθοδοι με στόχο την αποτελεσματική κατηγοριοποίηση σε τέτοιου είδους συστήματα και ως επακόλουθο έχουν αναπτυχθεί συστήματα ανίχνευσης εισβολών (Intrusion Detection Systems) με βάση αυτές τις τεχνικές και μεθόδους. Η παρούσα διπλωματική εργασία αφορά την εκτεταμένη ανασκόπηση της βιβλιογραφίας που αφορά τους αλγορίθμους και τις τεχνικές εξόρυξης γνώσης και μηχανικής μάθησης για την αποτελεσματική αναγνώριση εισβολών και κακόβουλων ενεργειών σε πληροφοριακά και επικοινωνιακά συστήματα καθώς και της επισκόπησης των αντίστοιχων συστημάτων ανίχνευσης εντοπισμού εισβολών.

Λέξεις Κλειδιά: Μηχανική Μάθηση, Συστήματα ανίχνευσης εισβολών, Ασφάλεια Πληροφοριακών Συστημάτων

Abstract

Nowadays, data mining and machine learning algorithms are applied to a variety of human activities in order to predict future situations. Moreover, they have been successfully applied for intrusion detection in the information and communication environments. The on-time recognition of such intrusions is vital for any organization. This need has attracted the interest of both the scientific community of datas mining and machine learning and the IT industry. As a result, various algorithms and techniques have been proposed in the bibliography in order to classify intrusions into such systems with high accuracy. As a consequence, many Intrusion Detection Systems have been developed based on these algorithms and techniques. This thesis concerns the extensive literature review on data mining and machine learning algorithms for intrusion and malicious actions detection in information and communication systems as well as the review of the intrusion detection systems.

Keywords: Data Mining, Machine Learning, Intrusion Detection Systems, IDS

Πίνακας περιεχομένων

1

Εισαγωγή

1.1 Κατηγοριοποίηση δεδομένων

Η εξόρυξη δεδομένων και η Μηχανική μάθηση είναι πεδία της επιστήμης των υπολογιστών που επικεντρώνονται στην ανάλυση των δεδομένων για την εξαγωγή συμπερασμάτων, έπειτα από λεπτομερή επεξεργασία τους. Οι τεχνικές της εφαρμόζονται σε πολλούς τομείς της καθημερινότητας, όπως η ιατρική για την πρόβλεψη και διάγνωση διαφόρων ασθενειών και παθήσεων, στην οικονομία για την ορθότερη λήψη αποφάσεων σχετικά με οικονομικά δεδομένα που ενισχύουν τις αγορές, την τηλεπικοινωνία για την βελτιστοποίηση της ποιότητας των υπηρεσιών, στην ασφάλεια πληροφοριακών και επικοινωνιακών συστημάτων κ.α. Πρόκειται, για επιστημονικά πεδία της επιστήμης των υπολογιστών τα οποία εξελίσσονται συνέχεια, καθώς ο όγκος των δεδομένων αυξάνεται καθημερινά με ταχείς ρυθμούς και η διαχείρισή τους ολοένα και γίνεται δυσκολότερη.

Η κατηγοριοποίηση δεδομένων ή εποπτευόμενη μάθηση όπως αλλιώς αναφέρεται στην ορολογία της μηχανικής μάθησης, αφορά την κατάταξη ενός στοιχείου (ή στιγμιότυπου) σε μια κατηγορία (ή κλάση) με βάση κάποια χαρακτηριστικά. Η συγκεκριμένη διαδικασία λαμβάνει χώρα σε πολλούς τομείς της καθημερινότητας όπως για παράδειγμα, στην οικονομία με το διαχωρισμό των πολιτών βάσει κάποιων χαρακτηριστικών, για να λάβουν επιδόματα ή στο να προσδιοριστεί αν ένα μήνυμα ηλεκτρονικού ταχυδρομείου είναι ανεπιθύμητο ή όχι. Η κατηγοριοποίηση αποτελεί σημαντικό κομμάτι αφου οι αλγόριθμοι κατηγοριοποίηση καταφέρνουν πολλές φορές να επιτύχουν υψηλή ακρίβεια κατά την κατηγοριοποίηση στην

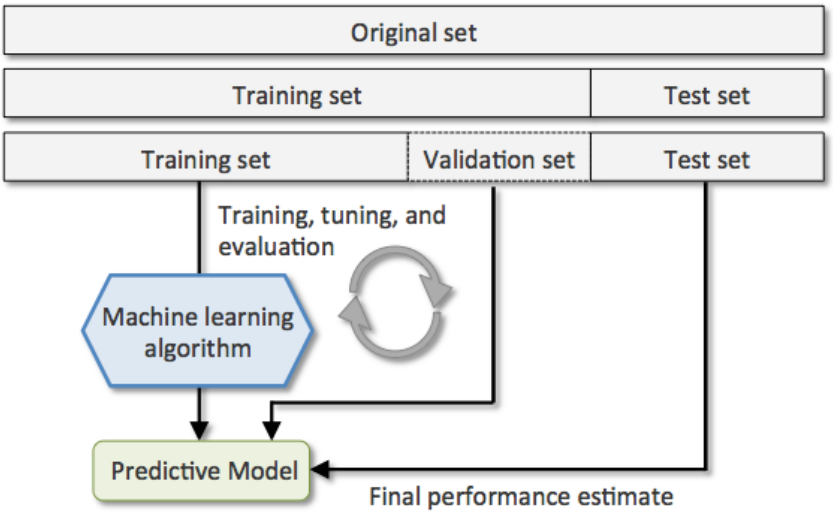
κατηγοριοποίηση στιγμιότυπων όπου η κλάση τους δεν είναι γνωστή. Τα τελευταία χρόνια, η κατηγοριοποίηση δεδομένων έχει βρει εφαρμογή στο πεδίο της ασφάλειας των πληροφοριακών και επικοινωνιακών συστημάτων με στόχο τον εντοπισμό κακόβουλων επιθέσεων. Η παρούσα διπλωματική εργασία ασχολείται με το αυτό το αντικείμενο.

Η διαδικασία της κατηγοριοποίησης αποτελείται από δύο στάδια. Στο πρώτο στάδιο, ο κατηγοριοποιητής τροφοδοτείται με ένα σύνολο δεδομένων εκπαίδευσης, όπου τα στιγμιότυπα έχουν γνωστή κλάση, και "εκπαιδεύεται" στο να αναγνωρίζει τις κλάσεις του συνόλου δεδομένων εκπαίδευσης βάσει των χαρακτηριστικών των στιγμιότυπων. Η διαδικασία αυτή ονομάζεται εκπαίδευση και το αποτέλεσμα τις είναι η κατασκευή ενός μοντέλου κατηγοριοποίησης που έχει ως στόχο την σωστή κατηγοριοποίηση νέων στιγμιότυπων όπου η κλάση τους δεν είναι γνωστή. Η διαδικασία της εκπαίδευσης μπορεί να παρομοιαστεί με τη διαδικασία εκμάθησης των γραμμάτων της αλφαβήτου από ένα νήπιο. Αρχικά, το νήπιο εκπαιδεύεται βλέποντας τα γράμματα και ακούγοντας τον γονέα του να του αναφέρει ποιο γράμμα είναι το καθένα. Μετά το στάδιο αυτό, ιδανικά, το νήπιο είναι σε θέση να αναγνωρίζει το κάθε γράμμα χωρίς να ακούει ποιο είναι. Το δεύτερο στάδιο της διαδικασίας της κατηγοριοποίησης αφορά την εφαρμογή του κατηγοριοποιητή. Στο στάδιο αυτό, το μοντέλο εφαρμόζεται κατηγοριοποιώντας στιγμιότυπα των οποίων οι κλάσεις δεν είναι γνωστές.

Οι αλγόριθμοι κατηγοριοποίησης (ή αλλιώς αλγόριθμοι μάθησης με επίβλεψη) όπως και η αλγόριθμοι συσταδοποίησης (ή αλλιώς αλγόριθμοι μάθησης χωρίς επίβλεψη) μπορούν να χρησιμοποιηθούν για την επίλυση διαφορετικών προβλημάτων αναγνώρισης προτύπων [1,2]. Ένα από αυτά αφορά τον εντοπισμό εισβολών σε υπολογιστικά συστήματα. Η κατηγοριοποίηση, όπως αναφέρθηκε πριν, βασίζεται στη χρήση των δεδομένων εκπαίδευσης για τη δημιουργία μιας συνάρτησης, στην οποία κάθε ένα από τα δεδομένα εκπαίδευσης περιέχει ένα ζεύγος διανυσμάτων εισόδου και εξόδου (δηλαδή την ετικέτα κλάσης). Ο σκοπός της μάθησης είναι να υπολογιστεί κατά προσέγγιση η απόσταση μεταξύ των παραδειγμάτων εισόδου - εξόδου προκειμένου να δημιουργηθεί ένας ταξινομητής (μοντέλο). Η δημιουργία του μοντέλου δίνει την δυνατότητα της κατηγοριοποίησης άγνωστων στιγμιότυπων σε ετικέτες μιας μαθημένης κλάσης.

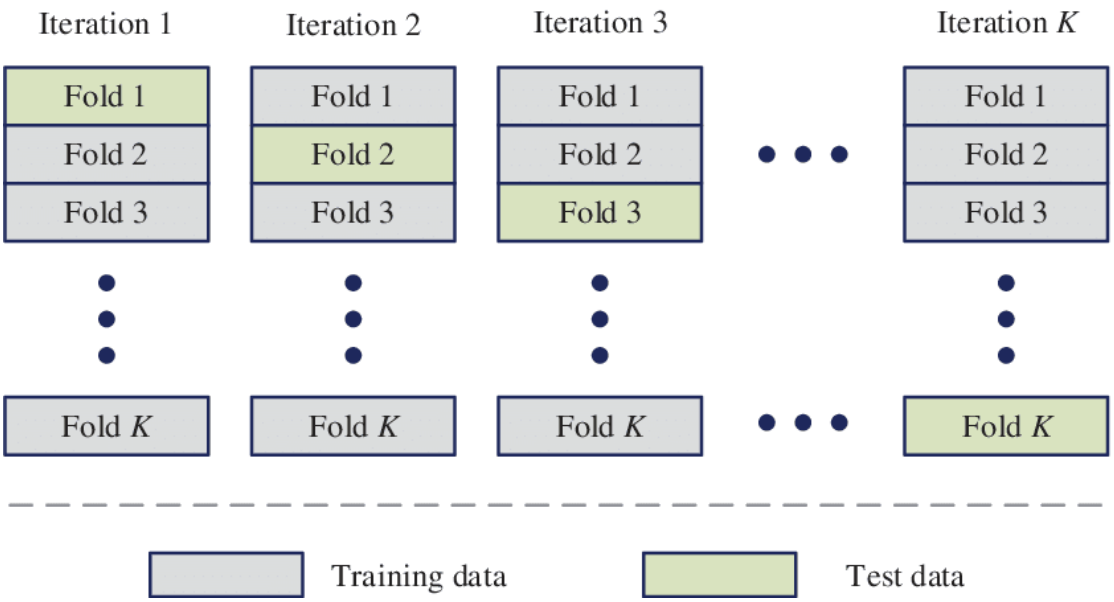
Ασφαλώς, πριν το στάδιο της εφαρμογής και μετά την κατασκευή του μοντέλου κατηγοριοποίησης, συνήθως χρησιμοποιείται μια τεχνική αποτίμησης της απόδοσης και της αποτελεσματικότητας του κατηγοριοποιητή. Μια από αυτές τις τεχνικές είναι το hold out βάσει του οποίου τα διαθέσιμα δεδομένα με γνωστές κλάσεις χωρίζονται σε δύο σύνολα, το ένα είναι το σύνολο δεδομένων εκπαίδευσης και άλλο είναι το σύνολο δεδομένων δοκιμής. Το μοντέλο κατασκευάζεται βάσει του συνόλου δεδομένων εκπαίδευσης και στη συνέχεια, η απόδοση και αποτελεσματικότητα του κατηγοριοποιητή δοκιμάζεται βάσει του συνόλου δοκιμής. Αν κατηγοριοποιητής έχει παραμέτρους, αυτές πρέπει με κάποιον τρόπο να οριστούν από τον

χρήστη. Σε αυτή την περίπτωση, το αρχικό σύνολο δεδομένων χωρίζεται σε τρία σύνολο. Το σύνολο δεδομένων εκπαίδευσης χρησιμοποιείται για την ανάπτυξη του μοντέλου, το σύνολο επικύρωσης (validation set) χρησιμοποιείται για τις ανάγκες καθορισμού των παραμέτρων (parameter tuning) του μοντέλου. Η διαδικασία καθορισμού του παραμέτρων συνήθως περιλαμβάνει την επαναληπτική εκτέλεση της κατασκευής του μοντέλου με διαφορετικές τιμές παραμέτρων και τελικά, επιλέγεται το πιο αποτελεσματικό μοντέλο. Στο τέλος, το σύνολο δοκιμής χρησιμοποιείται για τη εκτίμηση της απόδοσης και αποτελεσματικότητας του κατηγοριοποιητή. Η διαδικασία αυτή παρουσιάζεται στην παρακάτω εικόνα.



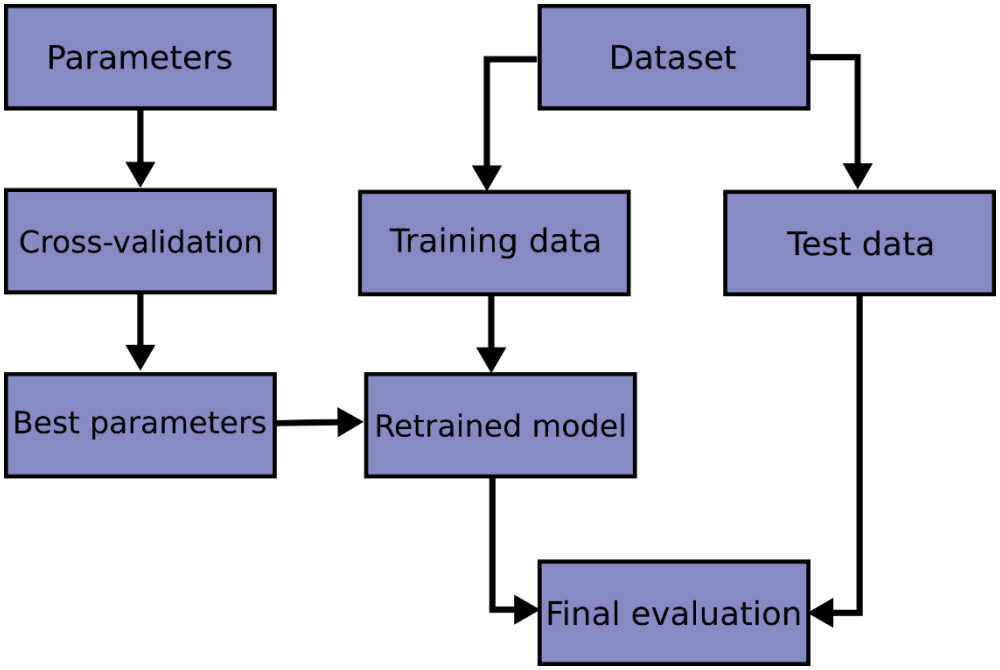
Σχήμα 1. Διαδικασία κατασκευής του μοντέλου κατηγοριοποίησης

Μια άλλη τεχνική επικύρωσης της απόδοσης και της αποτελεσματικότητας των κατηγοριοποιητών είναι η Διασταυρούμενη Επικύρωση κ τμημάτων (k-folds cross validation) Στη μέθοδο αυτή το σύνολο δεδομένων διαιρείται σε κ υποσύνολα. Κάθε υποσύνολο περιέχει διαφορετικά στιγμιότυπα. Η επιλογή των υποσυνόλων είναι τυχαία. Ένα από τα υποσύνολα χρησιμοποιείται ως σύνολο δοκιμής και τα υπόλοιπα κ-1 ενώνονται και δημιουργούν το σύνολο εκπαίδευσης. Το μοντέλο εκπαιδεύεται χρησιμοποιώντας το σύνολο εκπαίδευσης και δοκιμάζεται έναντι του συνόλου δοκιμής. Η διαδικασία επαναλαμβάνεται κ φορές, κάθε φορά χρησιμοποιώντας ένα διαφορετικό σύνολο ως σύνολο δοκιμής και τα υπόλοιπα εννέα ως σύνολο εκπαίδευσης. Στο τέλος υπολογίζεται η μέση επίδοση του μοντέλου. Η μέθοδος μπορεί να διαφοροποιηθεί ως προς το πλήθος των τμημάτων. Ονομάζεται μέθοδος επικύρωσης κ τμημάτων, όπου κ συμβολίζει τον αριθμό των υποσυνόλων και των επαναλήψεων. Οι τιμές που χρησιμοποιούνται συνήθως είναι το κ=10 και το κ=5.



Σχήμα 2. Διασταυρούμενη Επικύρωση κ τμημάτων

Σε αυτό το σημείο αξίζει να σημειωθεί ότι η Διασταυρούμενη Επικύρωση κ τμημάτων χρησιμοποιείται συχνά και για τον καθορισμό των παραμέτρων ενός κατηγοριοποιητή και συνδυάζεται με τη μέθοδο hold out που αναφέρθηκε προηγουμένως. Πιο συγκεκριμένα, αφού τα αρχικά δεδομένα χωριστούν σε σύνολο δεδομένων εκπαίδευσης και σύνολο δεδομένων δοκιμής, το σύνολο δεδομένων εκπαίδευσης χωρίζεται σε κ υποσύνολα κατάλληλα και η Διασταυρούμενη Επικύρωση κ τμημάτων εκτελείται επαναληπτικά για διαφορετικές τιμές παραμέτρων. Στο τέλος, επιλέγεται το πιο αποτελεσματικό μοντέλο (συνήθως αυτό που επιτυγχάνει την υψηλότερη ακρίβεια κατηγοριοποίησης) και τελικά το μοντέλο αυτό δοκιμάζεται κατηγοριοποιώντας τα στιγμιότυπα του συνόλου δεδομένων δοκιμής.



Σχήμα 3. Συνδυασμός Διασταυρούμενης Επικύρωσης κ τμημάτων και μεθόδου hold out

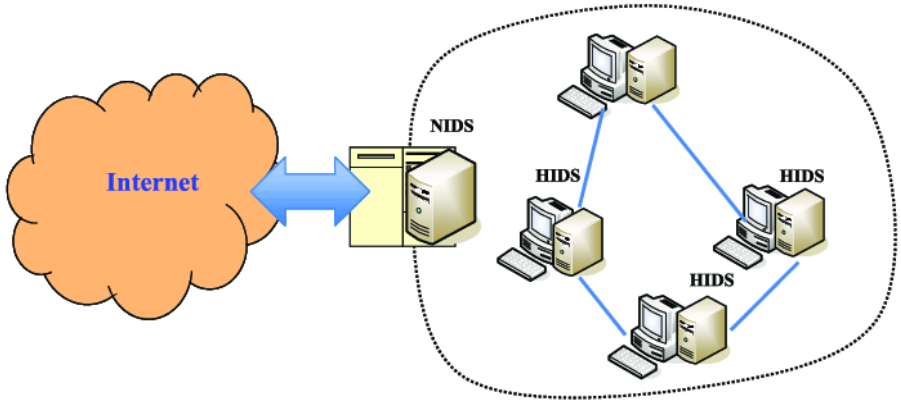
Μια κατηγορία αλγορίθμων κατηγοριοποίησης δεν κατασκευάζει μοντέλο. Στην πράξη, τα ίδια δεδομένα εκπαίδευσης παίζουν τον ρόλο του μοντέλου κατηγοριοποίησης. Αυτή η κατηγορία αλγορίθμων ονομάζονται «οκνηροί» (lazy) και κατηγοριοποιούν τα νέα στιγμιότυπα εξετάζοντας τα δεδομένα εκπαίδευσης την στιγμή της κατηγοριοποίησης. Χαρακτηριστικό παράδειγμα αυτής της κατηγορίας είναι ο κατηγοριοποίηση κ εγγύτερων γειτόνων. Ο συγκεκριμένος κατηγοριοποιητής θα περιγράφει αναλυτικά σε επόμενο κεφάλαιο της παρούσας διπλωματικής εργασίας. Αντίθετα, οι κατηγοριοποιητές που κατασκευάζουν μοντέλο κατηγοριοποίησης ονομάζονται «πρόθυμοι» (eager) κατηγοριοποιητές.

Συγκρίνοντας τους πρόθυμους και τους οκνηρούς κατηγοριοποιητές προκύπτουν κάποια χρήσιμα συμπεράσματα. Ένα από αυτά είναι πως οι πρόθυμοι κατηγοριοποιητές υπερισχύουν σε χρόνο και σε απαιτήσεις αποθηκευτικού χώρου, καθώς δημιουργούν μοντέλο κατηγοριοποίησης πριν την είσοδο νέου στιγμιότυπου και δεν χρειάζονται τα δεδομένα εκπαίδευσης για την κατηγοριοποίηση αυτού. Αντίθετα, οι οκνηροί κατηγοριοποιητές υπερισχύουν σε ευρύ φάσμα υποθέσεων που μπορούν να καλύψουν, γιατί έχουν πάντα διαθέσιμο ολόκληρο το σύνολο δεδομένων εκπαίδευσης. Επιπρόσθετα, δεν απαιτούν υπολογιστικό κόστος για την κατασκευή του μοντέλου. Ωστόσο, η κατηγοριοποίηση είναι υπολογιστικά συνήθως πιο δαπανηρή.

1.2 Συστήματα ανίχνευσης εισβολών

Ένα σύστημα ανίχνευσης εισβολών παρακολουθεί τα δεδομένα που διακινούνται σε ένα υπολογιστικό ή επικοινωνιακό σύστημα και αναλύει ενδείξεις εισβολών την στιγμή που συμβαίνουν ή έχουν συμβεί. Το σύστημα ανίχνευσης εισβολών καταγράφει συμβάντα και είναι σε θέση να διακόψει συνδέσεις και διεργασίες. Επιπρόσθετα, ειδοποιεί τους αντίστοιχους διαχειριστές των συστημάτων ώστε αυτοί, με τη σειρά τους να λάβουν τα κατάλληλα μέτρα προστασίας.

Τα συστήματα ανίχνευσης εισβολών υπολογιστικού συστήματος (Host based Intrusion Detection Systems - HIDS) δεν αφορούν το δίκτυο. Αφορούν αποκλειστικά το υπολογιστικό σύστημα. Τα συστήματα ανίχνευσης εισβολής δικτύου (Network Intrusion Detection Systems-NIDS) αφορούν τη διασφάλιση του δικτύου από επιθέσεις. Τα συστήματα ανίχνευσης εισβολής δικτύου εξετάζουν μπορεί να είναι είτε συστήματα πραγματικού χρόνου είτε συστήματα εκτός σύνδεσης. Τα πρώτα εξετάζουν τη δομή των δικτυακών πακέτων για να εντοπίσουν πιθανές εισβολές. Αν συμβεί αυτό παράγουν ειδοποιήσεις. Τα συστήματα εκτός σύνδεσης καταγράφουν τη ροή των πακέτων σε ένα δίκτυο και κατασκευάζουν χαρακτηριστικά που βασίζονται σε συνδέσεις. Έτσι, δημιουργούν ένα σύνολο δεδομένων. Τέτοια σύνολα δεδομένων χρησιμοποιούνται από αλγορίθμους εξόρυξης δεδομένων και μηχανικής μάθησης για την ανίχνευση επιθέσεων. Περισσότερες πληροφορίες για τα συστήματα ανίχνευσης εισβολών καταγράφονται σε επόμενο κεφάλαιο της παρούσας διπλωματικής εργασίας.



Σχήμα 4. Συστήματα ανίχνευσης εισβολών

1.2.1 Συνεισφορά

Τα τελευταία χρόνια, τόσο οι πρόθυμοι όσο οι οκνηροί κατηγοριοποιητές έχουν εφαρμοστεί με επιτυχία για την αναγνώριση κακόβουλων εισβολών και παραβιάσεων σε πληροφοριακά και επικοινωνιακά συστήματα. Η άμεση αναγνώριση τέτοιων παραβιάσεων είναι ζωτικής

σημασίας για οποιοδήποτε οργανισμού και επιχείρηση. Αυτή η ανάγκη έχει προσελκύσει το ενδιαφέρον της επιστημονικής κοινότητας της εξόρυξης γνώσης και της μηχανικής μάθησης. Ως αποτέλεσμα, έχουν προταθεί διάφορες τεχνικές και αλγόριθμοι που έχουν ως στόχο τη βελτίωση της κατηγοριοποίησης σε τέτοια συστήματα και ως επακόλουθο έχουν αναπτυχθεί συστήματα ανίχνευσης εισβολών (Intrusion Detection Systems) με βάση τους αλγορίθμους αυτούς. Το γεγονός αυτό αποτελεί το κίνητρο εκπόνησης της παρούσας διπλωματικής εργασίας.

Η συνεισφορά της παρούσας εργασίας είναι η εκτεταμένη ανασκόπηση της βιβλιογραφίας που αφορά τους αλγορίθμους και τις τεχνικές εξόρυξης γνώσης και μηχανικής μάθησης για την αποτελεσματική αναγνώριση εισβολών και κακόβουλων ενεργειών σε πληροφοριακά και επικοινωνιακά συστήματα καθώς και της επισκόπησης των αντίστοιχων συστημάτων ανίχνευσης εντοπισμού επιθέσεων.

1.3 Οργάνωση κειμένου

Στο κεφάλαιο 2 γίνεται μια ανασκόπηση στις επιθέσεις μέσω δικτύου και στα συστήματα ανίχνευσης εισβολών (IDS). Στο κεφάλαιο 3 γίνεται μια ανασκόπηση των αλγορίθμων μηχανικής μάθησης που μπορούν να βρουν εφαρμογή σε συστήματα IDS. Στο κεφάλαιο 4 παρουσιάζεται μια στατιστική ανάλυση που αφορά τη βιβλιογραφία, ενώ στο τέλος παρουσιάζονται τα συμπεράσματα.

2

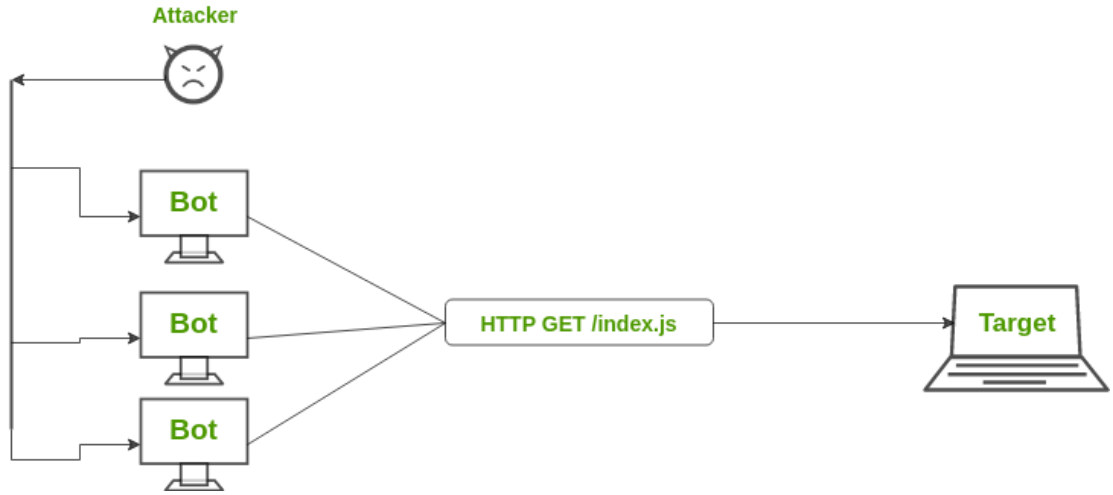
Επιθέσεις και Συστήματα Εντοπισμού Εισβολών

Οι παρακάτω ενότητες περιλαμβάνουν μια σύντομη επισκόπηση των επιθέσεων δικτύωσης, των ταξινομήσεων τους καθώς και διαφόρων στοιχείων που αφορούν τα συστήματα ανίχνευσης εισβολής.

2.1 Επιθέσεις δικτύωσης

Αυτή η ενότητα περιλαμβάνει τις τέσσερις βασικές κατηγορίες επιθέσεων δικτύου. Κάθε επίθεση σε ένα δίκτυο μπορεί άνετα να τοποθετηθεί σε μία από αυτές τις ομαδοποιήσεις [15].

Denial of Service (DoS): Μια επίθεση DoS είναι ένας τύπος επίθεσης κατά την οποία ο κακόβουλος χρήστης ή αλλιώς χάκερ καθιστά έναν υπολογιστή ή τους πόρους μνήμης αυτού πολύ απασχολημένους ή πολύ "γεμάτους" για να εξυπηρετήσει νόμιμα (legitimate) αιτήματα δικτύωσης και ως εκ τούτου αρνείται την πρόσβαση των χρηστών σε ένα μηχάνημα π.χ. τα apache, smurf, neptune, ping of death, back, mail bomb, UDP storm κ.α. είναι όλα επιθέσεις DoS. Με άλλα λόγια πρόκειται για επιθέσεις εναντίον ενός υπολογιστή, ή μιας υπηρεσίας, με σκοπό να καταστήσουν τον υπολογιστή ή την υπηρεσία ανίκανη να δεχτεί άλλες συνδέσεις και έτσι να μην μπορεί να εξυπηρετήσει άλλους χρήστες. Για παράδειγμα, πολλαπλά αιτήματα http μπορούν να προκαλέσουν την μη απόκριση ενός εξυπηρετητή ιστού.



Σχήμα 5. Επίθεση Denial of Service (DoS)

Αν και ο το συγκεκριμένο είδος επίθεσης αφορά κυρίως δικτυακές υπηρεσίες, δεν περιορίζεται μόνο σε αυτές αλλά αναφέρεται και σε άλλα πεδία όπως ο μία επίθεση καταναλώνει τους πόρους ενός μικροεπεξεργαστή και έτσι αυτός δεν μπορεί να εξυπηρετήσει άλλες διεργασίες.

Remote to User Attacks (R2L): Η επίθεση από απόσταση σε χρήστη είναι μια επίθεση κατά την οποία ένας χρήστης στέλνει πακέτα σε ένα μηχάνημα μέσω διαδικτύου, στο οποίο δεν έχει πρόσβαση για να εκθέσει τα τρωτά σημεία των μηχανών και να εκμεταλλευτεί προνόμια που ένας τοπικός χρήστης θα είχε στον υπολογιστή π.χ. xlock, guest, xnsnoop, phf, sendmail dictionary κ.λπ.

User to Root Attacks (U2R): Αυτές οι επιθέσεις είναι εκμεταλλεύσεις κατά τις οποίες ο κακόβουλος χρήστης (χάκερ) ξεκινά στο σύστημα με έναν κανονικό λογαριασμό χρήστη και επιχειρεί να καταχραστεί ευπάθειες στο σύστημα προκειμένου να αποκτήσει προνόμια υπερχρήστη π.χ. perl, xterm.

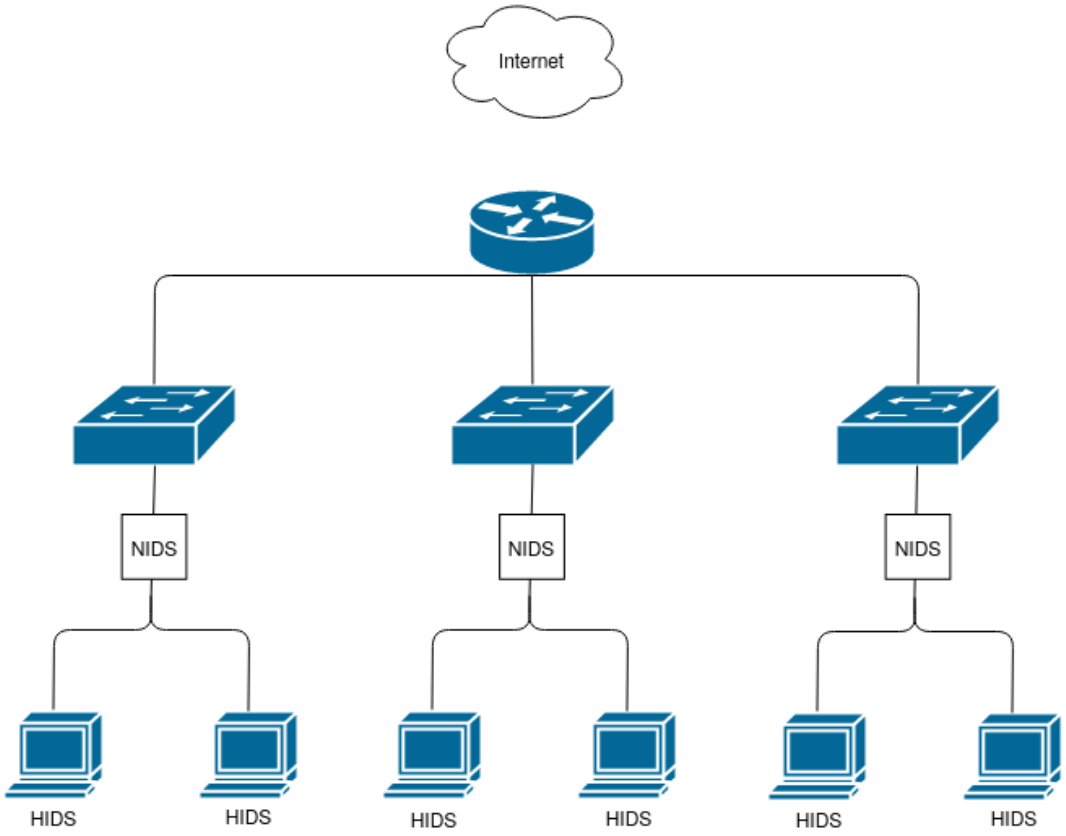
Probing (Ερευνα): Το probing είναι μια επίθεση κατά την οποία ο επιτιθέμενος, με την βοήθεια συνήθως κατάλληλου λογισμικού, σαρώνει ένα μηχάνημα ή μια συσκευή δικτύωσης προκειμένου να προσδιορίσει αδυναμίες ή τρωτά σημεία που μπορεί αργότερα να αξιοποιηθούν για να θέσουν σε κίνδυνο το σύστημα. Αυτή η τεχνική χρησιμοποιείται συνήθως στην εξόρυξη δεδομένων π.χ. saint, portsweep, mscan, nmap κ.λπ.

2.2 Ταξινόμηση της ανίχνευσης εισβολής

Όπως αναφέρθηκε και στο Κεφάλαιο 1 της παρούσας διπλωματικής εργασίας, η ανίχνευση εισβολών μπορεί να είναι είτε Ανίχνευση εισβολής υπολογιστικού συστήματος (Host Based Intrusion Detection - HIDS) είτε Ανίχνευση εισβολής βάσει δικτύου (Network Based Intrusion Detection - NIDS).

Τα HIDS αξιολογούν τις πληροφορίες που βρίσκονται σε ένα σύστημα υπολογιστή, συμπεριλαμβανομένου του περιεχομένου των λειτουργικών συστημάτων, των συστημάτων και των αρχείων εφαρμογών [16]. Βασίζουν τις αποφάσεις τους σε πληροφορίες που λαμβάνονται από έναν υπολογιστή ή ακόμη και μια εφαρμογή. Οι αποφάσεις λαμβάνονται βάσει των δεδομένων που μπορούν να εντοπιστούν από το λειτουργικό σύστημα του υπολογιστή, συνήθως αρχεία καταγραφής, χρήση πόρων, κίνηση δικτύου προς και από τον υπολογιστή ή πληροφορίες σχετικά με τις διεργασίες που εκτελούνται στον κεντρικό υπολογιστή. Από την άλλη, Τα NIDS αξιολογούν τις πληροφορίες που συλλέγονται από τις επικοινωνίες δικτύου, αναλύοντας τη ροή των πακέτων που κυκλοφορούν στο δίκτυο [16].

Τα NIDS λαμβάνουν δεδομένα παρακολουθώντας την κίνηση του δικτύου μεταξύ κεντρικών υπολογιστών. Εκτελούνται σε ξεχωριστό μηχάνημα, προσπαθώντας να ανιχνεύσουν πακέτα που θα μπορούσαν να είναι μέρος μιας επίθεσης.



Σχήμα 6. Δίκτυο υπολογιστών με συνδυασμό συστημάτων NIDS και HIDS

Τα συστήματα ανίχνευσης εισβολών μπορούν επίσης να κατηγοριοποιηθούν με βάση διαφορετικές τεχνικές ανίχνευσης: (i) ανίχνευση βάσει ανωμαλιών, (ii) ανίχνευση βάσει υπογραφής - κακής χρήσης και (iii) ανίχνευση βάσει προδιαγραφών.

Τα συστήματα ανίχνευσης εισβολών με βάση την ανίχνευση ανωμαλιών προσπαθούν να δημιουργήσουν ένα μοντέλο κανονικής συμπεριφοράς και θεωρούν ύποπτη οποιαδήποτε συμπεριφορά αποκλίνει από την κανονική συμπεριφορά και η απόκλιση υπερβαίνει ένα συγκεκριμένο όρια. Εάν ξεπεραστεί το όριο, οι ενέργειες θεωρούνται ύποπτες και καταγράφονται. Τα συστήματα ανίχνευσης εισβολών με βάση την ανίχνευση ανωμαλιών θεωρούνται αξιόπιστα αφού έχουν καλή ικανότητα ανίχνευσης επιθέσεων ακόμη και σε επιθέσεις που παρόμοιες με αυτές δεν είχαν εμφανιστεί στο παρελθόν. Παρόλα αυτά, θεωρούν πολλές ενέργειες ύποπτες χωρίς στην πραγματικότητα να είναι. Αυτό συμβαίνει λόγω της δυσκολίας δημιουργίας μιας συνεκτικής "κανονικής" συμπεριφοράς σε ένα σύγχρονο περιβάλλον υπολογιστικού συστήματος. Για την κατηγοριοποίηση μιας συμπεριφοράς ως κανονική ή μη χρησιμοποιούνται τεχνικές στατιστικής, γνώσης και αλγόριθμοι εξόρυξης γνώσης και μηχανικής μάθησης.

Οι στατιστικές τεχνικές ανίχνευσης περιλαμβάνουν ανάλυση χρονοσειρών. Από την άλλη, οι τεχνικές με βάση τη γνώση χρησιμοποιούν μηχανές πεπερασμένων καταστάσεων και κανόνων όπως συστήματα βασισμένα σε περιπτώσεις, βασισμένα σε εξειδικευμένα συστήματα και γλώσσες περιγραφής. Τέλος, οι αλγόριθμοι εξόρυξης γνώσης και μηχανικής μάθησης περιλαμβάνουν εκπαίδευση τεχνητών νευρωνικών δικτύων, εφαρμογή αλγορίθμων συσταδοποίησης κ.ο.κ. Τέλος, τα συστήματα ανίχνευσης βάσει υπογραφής ή αλλιώς με βάση την κακή χρήση, χρησιμοποιούν μια βάση γνώσεων, δηλαδή ένα σύνολο υπογραφών που προέκυψαν από προηγούμενες ανιχνεύεις εισβολών, για να αναγνωρίσουν άμεσα τις προσπάθειες εισβολής. Πρακτικά, αυτό σημαίνει ότι αντί να προσπαθεί να περιγράψει την κανονική συμπεριφορά, προσπαθεί να περιγράψει τις ανώμαλες συμπεριφορές. Αναμφίβολα, Πλεονέκτημα αυτής της τεχνικής είναι η υψηλή ακρίβεια ανίχνευσης για γνωστές επιθέσεις. Από την άλλη, ένα τέτοιο σύστημα δεν μπορεί να ανιχνεύσει νέες απειλές ή πολυμορφικές απειλές. Αξίζει να σημειωθεί ότι πολλά συστήματα ανίχνευσης συνδυάζουν τις ιδιότητες τόσο της τεχνικής βάσει ανωμαλίας όσο και της τεχνικής βάσει υπογραφής για να σχηματίσουν ένα υβριδικό μοντέλο, έχοντας ως σκοπό την καλύτερη απόδοση, είτε στον καλύτερο εντοπισμό των επιθέσεων, είτε στην αποδοτικότερη λειτουργίας τους από πλευράς κατανάλωσης πόρων και χρόνου, είτε ιδανικά στην επίτευξη και των δύο.

2.3 Συστατικά στοιχεία Συστημάτων ανίχνευσης εισβολής

Ένα Σύστημα ανίχνευσης εισβολής (IDS-Intrusion Detection System) είτε αυτό είναι HIDS είτε είναι NIDS αποτελείται από τρια λειτουργικά στοιχεία [17]. Το πρώτο στοιχείο ενός συστήματος ανίχνευσης εισβολής, επίσης γνωστό ως γεννήτρια συμβάντων (event generator), είναι μια πηγή δεδομένων (data source). Οι πηγές δεδομένων μπορούν να κατηγοριοποιηθούν

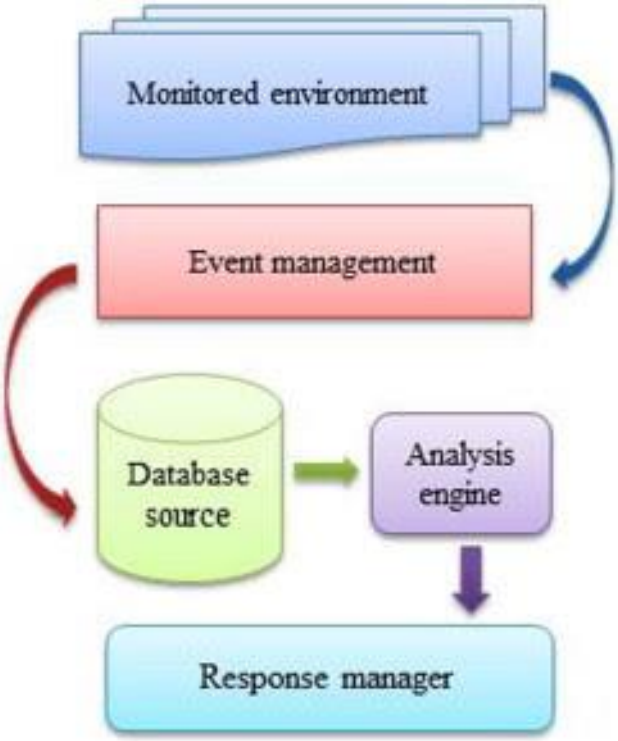
σε τέσσερις κατηγορίες, συγκεκριμένα Host-based monitors, Network-based monitors, Application-based monitors και Target-based monitors.

Το δεύτερο στοιχείο ενός συστήματος ανίχνευσης εισβολής είναι γνωστό ως μηχανή ανάλυσης (analysis engine). Αυτό το στοιχείο λαμβάνει πληροφορίες από την πηγή δεδομένων και εξετάζει τα δεδομένα για συμπτώματα επιθέσεων ή άλλων παραβιάσεων πολιτικής. Η μηχανή ανάλυσης μπορεί να χρησιμοποιήσει μία ή και τις δύο από τις ακόλουθες προσεγγίσεις ανάλυσης:

Κακή χρήση / Ανίχνευση βάσει υπογραφής (Misuse/Signature-Based Detection): Αυτός ο τύπος μηχανής ανίχνευσης εντοπίζει εισβολές που ακολουθούν γνωστά πρότυπα επιθέσεων (ή υπογραφών) που εκμεταλλεύονται γνωστά τρωτά σημεία λογισμικού [24, 25]. Ο κύριος περιορισμός αυτής της προσέγγισης είναι ότι αναζητά μόνο τις γνωστές αδυναμίες και μπορεί να μην ενδιαφέρεται για τον εντοπισμό άγνωστων μελλοντικών εισβολών [26].

Ανωμαλία / Στατιστική Ανίχνευση (Anomaly/Statistical Detection): Μια μηχανή ανίχνευσης που βασίζεται σε ανωμαλίες θα αναζητήσει κάτι σπάνιο ή ασυνήθιστο [26]. Αναλύει τις ροές συμβάντων του συστήματος, χρησιμοποιώντας στατιστικές τεχνικές για να βρει μοτίβα δραστηριότητας που πιθανόν να μην είναι φυσιολογικά. Τα κύρια μειονεκτήματα αυτού του συστήματος είναι ότι είναι πολύ ακριβό και μπορεί να αναγνωρίσει μια παρεμβατική συμπεριφορά ως φυσιολογική συμπεριφορά λόγω ανεπαρκών δεδομένων

Το τρίτο στοιχείο ενός συστήματος ανίχνευσης εισβολών είναι ο διαχειριστής απόκρισης (response manager). Εν συντομία, ο διαχειριστής απόκρισης θα ενεργήσει μόνο όταν εντοπιστούν ανακρίβειες (πιθανές επιθέσεις εισβολής) στο σύστημα, εκτελώντας τις κατάλληλες ενέργειες προστασίας εφόσον κριθεί απαραίτητο και ενημερώνοντας κάποιον ή κάτι με τη μορφή απάντησης.



Σχήμα 7. Λειτουργία συστήματος εντοπισμού εισβολών

2.4 Υπάρχοντα συστήματα εντοπισμού εισβολών

Εδώ περιγράφουμε μερικά από τα πιο σημαντικά και γνωστά συστήματα ανίχνευσης εισβολής και τα προβλήματά που αυτά παρουσιάζουν.

Συστήματα ανίχνευσης υφιστάμενης εισβολής (Existing Intrusion Detection Systems)

- **Snort**: Το Snort αποτελεί ένα δωρεάν και ανοιχτού κώδικα σύστημα που ανιχνεύει και προλαμβάνει εισβολές στο δίκτυο. Δημιουργήθηκε από τον Martin Roesch το 1998 και πλέον αναπτύσσεται από την Sourcefire. Το 2009, το Snort μπήκε στο InfoWorld's Open Source Hall of Fame ως ένα από τα "μεγαλύτερα λογισμικά ανοιχτού κώδικα όλων των εποχών" [19, 20]. Μέσω ανάλυσης πρωτοκόλλου, αναζήτησης περιεχομένου και διάφορων προ-επεξεργασιών, το Snort ανιχνεύει χιλιάδες κακόβουλες ενέργειες σκουλήκια, προσπάθειες εκμετάλλευσης ευπαθειών, σαρώσεις θυρών και άλλες ύποπτες συμπεριφορές [21, 22]. Το Snort έχει τη δυνατότητα να πραγματοποιεί ανάλυση κίνησης σε πραγματικό χρόνο και καταγραφή πακέτων σε δίκτυα Πρωτοκόλλου Internet (IP). Το Snort μπορεί να διαμορφωθεί σε τρεις κύριες λειτουργίες: 1. sniffer, 2. packet logger και 3. ανίχνευση εισβολής δικτύου:
  - Λειτουργία Sniffer: Το πρόγραμμα θα διαβάσει πακέτα δικτύου και θα τα εμφανίσει στην κονσόλα.

- Λειτουργία καταγραφής πακέτων: Στη λειτουργία καταγραφής πακέτων, το πρόγραμμα θα καταγράψει πακέτα στο δίσκο.
- Λειτουργία συστήματος ανίχνευσης εισβολής δικτύου: Στη λειτουργία ανίχνευσης εισβολής, το πρόγραμμα θα παρακολουθεί την κίνηση του δικτύου και θα την αναλύει με βάση έναν κανόνα που έχει οριστεί από τον χρήστη. Στη συνέχεια, το πρόγραμμα θα εκτελέσει μια συγκεκριμένη ενέργεια με βάση αυτό που έχει προσδιοριστεί.

- **OSSEC**: Το OSSEC είναι επίσης ανοιχτού κώδικα, σύστημα ανίχνευσης εισβολής βασισμένο σε κεντρικό υπολογιστή (HIDS). Πραγματοποιεί ανάλυση καταγραφής, έλεγχο ακεραιότητας, εντοπισμό rootkit, προειδοποίηση βάσει χρόνου και ενεργή απόκριση [21, 22]. Εκτός από τη λειτουργικότητά του συστήματος ανίχνευσης εισβολών, χρησιμοποιείται συνήθως ως λύση SEM/SIM. Λόγω της ισχυρής μηχανής ανάλυσης αρχείων καταγραφής, οι πάροχοι υπηρεσιών διαδικτύου, τα πανεπιστήμια και τα κέντρα δεδομένων εκτελούν OSSEC HIDS για να παρακολουθούν και να αναλύουν τα τείχη προστασίας, τα συστήματα ανίχνευσης εισβολών, τους εξυπηρετητές ιστού και τα αρχεία καταγραφής ελέγχου ταυτότητας. Το OSSEC αποτελείται από τρία βασικά στοιχεία. Μια κύρια εφαρμογή, έναν πράκτορα και μια διεπαφή ιστού:
  - Διαχειριστής (ή διακομιστής), που απαιτείται για κατανεμημένο δίκτυο ή αυτόνομες εγκαταστάσεις.
  - Agent, ένα μικρό πρόγραμμα εγκατεστημένο στα συστήματα που πρέπει να παρακολουθούνται.
  - Λειτουργία Agentless, μπορεί να χρησιμοποιηθεί για την παρακολούθηση τείχους προστασίας, δρομολογητές, ακόμη και συστήματα Unix.

  Τον Ιούνιο του 2008, το έργο OSSEC και όλα τα πνευματικά δικαιώματα που ανήκουν στον Daniel B. Cid, ο οποίος ήταν ο επικεφαλής του έργου, αποκτήθηκαν από την Third Brigade, Inc. Υποσχέθηκαν ότι θα συνεισφέρουν στην κοινότητα ανοιχτού κώδικα και να επεκτείνουν την εμπορική υποστήριξη και εκπαίδευση στους Κοινότητα ανοιχτού κώδικα OSSEC. Τον Μάιο του 2009, η Trend Micro απέκτησε το OSSEC, με υποσχέσεις ότι θα τη διατηρήσει ανοιχτή και δωρεάν. Το 2018, η Trend κυκλοφόρησε το domain name και τον πηγαίο κώδικα στο ιδρυμα OSSEC. Το έργο OSSEC διατηρείται επί του παρόντος από την Atomicorp που διαχειρίζεται την δωρεάν και ανοιχτού κώδικα έκδοση και προσφέρει επίσης μια βελτιωμένη εμπορική έκδοση.
- **OSSIM**: Ο στόχος του Συστήματος Διαχείρισης Πληροφοριών Ασφαλείας Ανοικτού Κώδικα (Open Source Security Information Management-OSSIM) είναι να παρέχει μια ολοκληρωμένη συλλογή εργαλείων τα οποία, όταν συνεργάζονται, παρέχουν στους

διαχειριστές δικτύου/ασφάλειας μια λεπτομερή εικόνα για κάθε πτυχή δικτύων, κεντρικών υπολογιστών, συσκευών φυσικής πρόσβασης, και διακομιστών [22]. Το OSSIM ενσωματώνει πολλά άλλα εργαλεία, συμπεριλαμβανομένων των Nagios και OSSEC HIDS. Το έργο ξεκίνησε το 2003 ως συνεργασία μεταξύ του Dominique Karg, Julio Casal και αργότερα του Alberto Román. Το 2008 έγινε η βάση για την εταιρεία τους AlienVault η όποια άρχισε να πουλά ένα εμπορικό παράγωγο του OSSIM ("AlienVault Unified Security Management"). Το AlienVault εξαγοράστηκε από την AT&T Communications και μετονομάστηκε σε AT & T Cybersecurity το 2019.

Το OSSIM είχε τέσσερις εκδόσεις. Το έργο έχει περίπου 7,4 εκατομμύρια γραμμές κώδικα. Το OSSIM αποσκοπεί να δώσει στους αναλυτές και τους διαχειριστές ασφάλειας μια πιο ολοκληρωμένη εικόνα όλων των πτυχών που σχετίζονται με την ασφάλεια του συστήματός τους, συνδυάζοντας τη διαχείριση αρχείων καταγραφής, η οποία μπορεί να επεκταθεί με πρόσθετα και διαχείριση περιουσιακών στοιχείων και ανακάλυψη με πληροφορίες από αποκλειστική ασφάλεια πληροφοριών συστήματα ελέγχου και ανίχνευσης. Αυτές οι πληροφορίες στη συνέχεια συσχετίζονται μαζί για να δημιουργήσουν πλαίσια για τις πληροφορίες που δεν είναι ορατές μόνο από ένα κομμάτι. Παρέχονται προβολές συναγερμού και διαθεσιμότητας μαζί με τις δυνατότητες αναφοράς για τη βελτίωση των δυνατοτήτων του εργαλείου και της χρησιμότητάς του στους μηχανικούς ασφάλειας και συστημάτων. Το OSSIM εκτελεί αυτές τις λειτουργίες χρησιμοποιώντας άλλα γνωστά στοιχεία ασφάλειας λογισμικού ανοιχτού κώδικα, ενώνοντάς τα κάτω από μια ενιαία διεπαφή ιστού. Η διασύνδεση παρέχει εργαλεία γραφικής ανάλυσης για πληροφορίες που συλλέγονται από το υποκείμενο λογισμικό ανοιχτού κώδικα (πολλά από τα οποία είναι εργαλεία μόνο της γραμμής εντολών που κατά τα άλλα καταγράφονται μόνο σε ένα απλό αρχείο κειμένου) και επιτρέπει την κεντρική διαχείριση των επιλογών διαμόρφωσης.

Το λογισμικό διανέμεται ελεύθερα υπό την GNU General Public License. Σε αντίθεση με τα μεμονωμένα στοιχεία που μπορούν να εγκατασταθούν σε ένα υπάρχον σύστημα, το OSSIM διανέμεται ως αρχείο ISO που έχει σχεδιαστεί για να αναπτυχθεί σε έναν φυσικό ή εικονικό κεντρικό υπολογιστή ως το βασικό λειτουργικό σύστημα του κεντρικού υπολογιστή. Το OSSIM έχει δημιουργηθεί χρησιμοποιώντας το Debian ως το υποκείμενο λειτουργικό του σύστημα. Λόγω αυτής της κεντρικής πλατφόρμας που είναι ανοιχτή, μπορούν να προστεθούν και να επεκταθούν πρόσθετες δυνατότητες από τους διαχειριστές ασφαλείας.

- **Suricata**: Πρόκειται για ένα σύστημα ανίχνευσης εισβολής βασισμένο σε ανοιχτό κώδικα το οποίο αναπτύχθηκε από το Open Information Security Foundation (OISF) [23]. Στον διαδικτυακό τόπο του Suricata αναφέρεται ότι πρόκειται για ένα σύστημα

εντοπισμού απειλών ανοιχτού κώδικα που συνδυάζει ανίχνευση εισβολής, πρόληψη εισβολής και παρακολούθηση ασφάλειας δικτύου.

- **Zeek** (πρώην Bro): Ένα σύστημα ανίχνευσης εισβολής ανοιχτού κώδικα, βασισμένο στο Unix [24]. Το Bro ανιχνεύει τις εισβολές αναλύοντας πρώτα την κυκλοφορία του δικτύου για να εξαγάγει τη σημασιολογία του σε επίπεδο εφαρμογής και στη συνέχεια εκτελώντας αναλυτές προσανατολισμένους σε συμβάντα που συγκρίνουν τη δραστηριότητα με μοτίβα που θεωρούνται ενοχλητικά. Αναπτύχθηκε για πρώτη φορά το 1994 από τον Vern Paxson και ονομάστηκε αρχικά σε σχέση με τον Big Brother. Έχει σχεδιαστεί για να είναι μια οθόνη ασφαλείας δικτύου, αλλά μπορεί επίσης να χρησιμοποιηθεί ως σύστημα ανίχνευσης εισβολής δικτύου (NIDS) σε συνδυασμό με πρόσθετη ζωντανή ανάλυση συμβάντων δικτύου. Σημειώνεται ότι κυκλοφορεί με άδεια BSD. Το Zeek προβλει ότι τα δικτυακά πακέτα μεταφέρονται σε μια μηχανή συμβάντων η οποία τα αποδέχεται ή τα απορρίπτει. Η μηχανή συμβάντων αναλύει την κίνηση δικτύου και παράγει συμβάντα. Δημιουργεί συμβάντα όταν συμβαίνει "κάτι". Το Zeek χρησιμοποιεί κοινές θύρες και δυναμική ανίχνευση πρωτοκόλλου για να κάνει την καλύτερη εικασία στην ερμηνεία των πρωτοκόλλων δικτύου. Τα γεγονότα είναι ουδέτερα ως προς την πολιτική. Απλώς σηματοδοτούν το σενάριο ότι κάτι συνέβη.
  Ο χειρισμός των συμβάντων γίνεται με σενάρια πολιτικής. Τα σενάρια είναι γραμμένα στην πλήρη γλώσσα Turing. Από προεπιλογή, το Zeek απλώς καταγράφει πληροφορίες σχετικά με συμβάντα σε αρχεία. Ωστόσο, μπορεί να διαμορφωθεί για να πραγματοποιεί άλλες ενέργειες, όπως αποστολή μηνύματος ηλεκτρονικού ταχυδρομείου, αύξηση ειδοποίησης, εκτέλεση εντολής συστήματος, ενημέρωση εσωτερικής μέτρησης, ακόμη και κλήση άλλου σεναρίου Zeek. Η προεπιλεγμένη συμπεριφορά παράγει έξοδο τύπου NetFlow (αρχείο καταγραφής σύνδεσης) καθώς και πληροφορίες συμβάντων εφαρμογής. Τα σενάρια Zeek είναι σε θέση να διαβάζουν σε δεδομένα από εξωτερικά αρχεία, όπως black lists, για χρήση εντός των σεναρίων πολιτικής Zeek.
- **Fragroute/Fragrouter**: Μια εργαλειοθήκη εντοπισμού διείσδυσης δικτύου [21]. Το Fragrouter βοηθά έναν εισβολέα να ξεκινήσει επιθέσεις που βασίζονται σε IP αποφεύγοντας τον εντοπισμό από συστήματα εντοπισμού εισβολών. Συγκεκριμένα εκτελεί τη δρομολόγηση της κίνησης του δικτύου με τέτοιο τρόπο ώστε να αποφύγει τα περισσότερα συστήματα ανίχνευσης εισβολής δικτύου. Είναι μέρος της σουίτας εργαλείων NIDSbench του Dug Song. Το fragroute υποκλέπτει, τροποποιεί και ξαναγράφει την έξοδο κυκλοφορίας που προορίζεται για έναν συγκεκριμένο κεντρικό υπολογιστή, εφαρμόζοντας τις περισσότερες από τις επιθέσεις που περιγράφονται στο έγγραφο Secure Networks "Insertion, Evasion, and Denial of Service: Eluding Network Intrusion Detection" του Ιανουαρίου 1998. Αυτό το εργαλείο γράφτηκε με καλή πίστη

για να βοηθήσει στη δοκιμή συστημάτων ανίχνευσης εισβολής δικτύου, τείχη προστασίας και βασική συμπεριφορά στοίβας TCP/IP.

- **BASE**: Η BASE είναι μια μηχανή ανάλυσης βασισμένη σε PHP για αναζήτηση και επεξεργασία μιας βάσης δεδομένων συμβάντων ασφαλείας που δημιουργούνται από διάφορα IDS, τείχη προστασίας και εργαλεία παρακολούθησης δικτύου [21].
- **Sguil**: Το Sguil κατασκευάζεται από αναλυτές ασφάλειας δικτύου για αναλυτές ασφάλειας δικτύου [21] [22]. Το κύριο συστατικό του είναι ένα διαισθητικό GUI που παρέχει γεγονότα (events) σε πραγματικό χρόνο από το Snort/barnyard. Περιλαμβάνει επίσης άλλα στοιχεία που διευκολύνουν την πρακτική της παρακολούθησης της ασφάλειας του δικτύου και την ανάλυση με βάση τα γεγονότα των ειδοποιήσεων IDS. Πρακτικά, πρόκειται για μια συλλογή από δωρεάν στοιχεία λογισμικού για την παρακολούθηση της ασφάλειας δικτύου (NSM) και την ανάλυση των ειδοποιήσεων IDS βάσει γεγονότων. Το πρόγραμμα πελάτης είναι γραμμένο σε Tcl/Tk και μπορεί να εκτελεστεί σε οποιοδήποτε λειτουργικό σύστημα. Το Sguil ενσωματώνει δεδομένα ειδοποιήσεων από το Snort, δεδομένα περιόδου σύνδεσης από το SANCP και πλήρη δεδομένα περιεχομένου από μια δεύτερη παρουσία Snort που εκτελείται σε κατάσταση καταγραφής πακέτων.

2.5 Προβλήματα με τα υπάρχοντα συστήματα εντοπισμού εισβολών

Τα περισσότερα υπάρχοντα συστήματα ανίχνευσης εισβολής πάσχουν από τουλάχιστον δύο από τα ακόλουθα προβλήματα [25]:

Πρώτον, οι πληροφορίες που χρησιμοποιούνται από το σύστημα ανίχνευσης εισβολών λαμβάνονται από μονοπάτια ελέγχου (audit trails) ή από πακέτα σε ένα δίκτυο. Τα δεδομένα πρέπει να διασχίσουν μια μεγαλύτερη διαδρομή από την προέλευσή τους προς το IDS και κατα την διάρκεια αυτής της διαδικασίας μπορεί δυνητικά να καταστραφούν ή να τροποποιηθούν από έναν εισβολέα. Επιπλέον, το σύστημα ανίχνευσης εισβολής πρέπει να συμπεράνει τη συμπεριφορά του συστήματος από τα δεδομένα που συλλέγονται, γεγονός που μπορεί να οδηγήσει σε παρερμηνείες ή χαμένα συμβάντα. Αυτό αναφέρεται ως πρόβλημα πιστότητας (fidelity problem).

Δεύτερον, το σύστημα ανίχνευσης εισβολής χρησιμοποιεί συνεχώς πρόσθετους πόρους στο σύστημα που παρακολουθεί ακόμη και όταν δεν υπάρχουν εισβολές, επειδή τα στοιχεία του συστήματος ανίχνευσης εισβολής πρέπει να λειτουργούν συνεχώς. Αυτό είναι το πρόβλημα χρήσης πόρων (resource usage problem).

Τρίτον, επειδή τα στοιχεία του συστήματος ανίχνευσης εισβολής υλοποιούνται ως ξεχωριστά προγράμματα, είναι επιρρεπή σε παραποιήσεις. Ένας εισβολέας μπορεί να απενεργοποιήσει ή να τροποποιήσει τα προγράμματα που εκτελούνται σε ένα σύστημα, καθιστώντας το σύστημα ανίχνευσης εισβολών άχρηστο ή αναξιόπιστο. Αυτό είναι το πρόβλημα αξιοπιστίας (reliability problem).

3

Αλγόριθμοι κατηγοριοποίησης για εντοπισμό εισβολών

Διάφοροι αλγόριθμοι κατηγοριοποίησης έχουν χρησιμοποιηθεί για τον εντοπισμό εισβολών σε υπολογιστικά και δικτυακά συστήματα. Πολλοί από αυτούς έχουν ενσωματωθεί στα συστήματα εντοπισμού εισβολών που καταγράφηκαν στο Κεφάλαιο 2 της παρούσας εργασίας. Τα επόμενα υποκεφάλαια περιγράφουν τους αλγόριθμους αναγνώρισης προτύπων που έχουν χρησιμοποιηθεί στο παρελθόν για την ανίχνευση εισβολών.

3.1 Κατηγοριοποίηση σε δεδομένα με άνιση κατανομή κλάσεων

Σε αυτό το σημείο αυτό το σημείο, αξίζει να αναφερθούν κάποια στοιχεία που αφορούν την κατηγοριοποίηση συμβάντων ως «Εισβολή» και «Μη εισβολή». Τα σύνολα δεδομένων εκπαίδευσης που αφορούν την κατηγοριοποίηση συμβάντων σε «Εισβολή» ή «Μη εισβολή» χαρακτηρίζονται συνήθως από άνιση κατανομή. Δηλαδή, έχουν άνιση κατανομή των στιγμιότυπων εκπαίδευσης στις δύο κλάσεις [32], [33]. Η κλάση «Εισβολή» είναι σπάνια ενώ η κλάση «Μη εισβολή» περιλαμβάνει την συντριπτική πλειοψηφία των στιγμιότυπων εκπαίδευσης.

Κατά την εκτέλεση κατηγοριοποίησης σε τέτοια σύνολα δεδομένων, η μέτρηση της ακρίβειας είναι ανεπαρκής. Ας υποθέσουμε ότι ένα σύστημα κατηγοριοποίησης έχει εκπαιδευτεί για την πρόβλεψη εισβολών. Η εκπαίδευση που χρησιμοποιείται περιλαμβάνει τις κλάσεις: «Μη εισβολή» και «Εισβολή». Σίγουρα, η κλάση «εισβολή» είναι σπάνια και ασφαλώς σπανιότερη από την κλάση «Μη εισβολή». Μια λανθασμένη πρόβλεψη για «Μη εισβολή» μπορεί να

οδηγήσει σε ολέθρια αποτελέσματα για τον οργανισμό ή την επιχείρηση. Από την άλλη πλευρά, το κόστος μιας λανθασμένης πρόβλεψης για τον «Εισβολή» δεν είναι τόσο υψηλό. Στην πραγματικότητα, το κόστος είναι κάποιες περιττές προστατευτικές ενέργειες για μια εισβολή που τελικά δεν θα συμβεί.

Ο κατηγοριοποιητής ZeroR, ο οποίος προβλέπει πάντα «Μη εισβολή» γιατί σε αυτή ανήκει η συντριπτική πλειοψηφία των στιγμιότυπων εκπαίδευσης, επιτυγχάνει υψηλή ακρίβεια. Ωστόσο, είναι εντελώς μη αποδεκτός κατηγοριοποιητής επειδή δεν προβλέπει καμία εισβολή (δηλαδή προβλέπει ψευδώς αρνητικά). Από την άλλη πλευρά, ένα σύστημα κατηγοριοποίησης που πολύ συχνά προβλέπει λανθασμένα «Εισβολή» (δηλαδή τα ψευδώς θετικά) είναι αναξιόπιστο αλλά σίγουρα είναι προτιμότερο από το προηγούμενο.

Στην περίπτωση μη ισορροπημένων δεδομένων εκπαίδευσης, πρέπει να ληφθούν υπόψη οι μετρήσεις της ορθότητας (precision), της ανάκλησης (recall), το F-Score και άλλες [34]. Οι μετρήσεις υπολογίζονται λαμβάνοντας υπόψη τον πίνακα σύγχυσης (confusion matrix) που περιέχει τον αριθμό των αληθώς θετικών (TP), των αληθώς ψευδών (TN), των ψευδώς θετικών (FP) και των ψευδώς αρνητικών (FN).

Η ορθότητα υπολογίζεται ως εξής:

$$\frac{|TP|}{|TP|+|FP|}$$

και μετράει πόσες θετικές προβλέψεις είναι σωστές. Από την άλλη, η ανάκληση υπολογίζεται από τον τύπο:

$$\frac{|TP|}{(|TP|+|FN|)}$$

και μετρά πόσα θετικά προβλέπονται σωστά. Στην περίπτωση του συστήματος ταξινόμησης για τις προβλέψεις εισβολών, Η ορθότητα εκτιμά πόσες προβλέψεις για εισβολή είναι σωστές ενώ η ανάκληση εκτιμά από τις εισβολές που πραγματοποιήθηκαν πόσες είχαν προβλέφθηκαν εκ των προτέρων. Συνεπώς, η ανάκληση είναι πιο σημαντική από την ορθότητα σε ένα σύστημα ανίχνευσης εισβολών, διότι ακόμη και λίγες λανθασμένες προβλέψεις για "μη εισβολή" (δηλαδή, FN) μπορεί να είναι καταστροφικές.

Αντίθετα, σε ένα σύστημα κατηγοριοποίησης που προβλέπει πάντα θετικά η ανάκληση θα είναι ίση με ένα (δεν υπάρχουν FNs - όλες οι εισβολές προβλέπονται σωστά). Ωστόσο το σύστημα θα είναι αναξιόπιστο αφού προβλέπει πολλά FP και, ως εκ τούτου, η ορθότητα είναι εξαιρετικά χαμηλή. Κατά συνέπεια, στην περίπτωση συστημάτων που προβλέπουν εισβολές, η ανάκληση είναι πιο σημαντική από την ορθότητα, αλλά η ορθότητα δεν πρέπει να αγνοηθεί πλήρως. Διαφορετικά, υπάρχει ο κίνδυνος οι χρήστες να πάψουν να εμπιστεύονται το μοντέλο πρόβλεψης και εντοπισμού εισβολών.

Από την άλλη πλευρά, ας υποθέσουμε ότι ένας χρήστης ιστού υποβάλλει ένα ερώτημα σε μια μηχανή αναζήτησης. Ας υποθέσουμε ότι η μηχανή αναζήτησης ευρετηριάζει χιλιάδες ιστοσελίδες. Δέκα ιστοσελίδες περιέχουν τις πληροφορίες που αναζητά ο χρήστης. Στην ιδανική περίπτωση, η μηχανή αναζήτησης επιστρέφει τις δέκα σχετικές ιστοσελίδες και όχι περισσότερες. Σε αυτή την περίπτωση, τόσο η ορθότητα όσο και η ανάκληση θα είναι ίσα με ένα. Αντίθετα, η ορθότητα και η ανάκληση θα είναι μηδέν εάν η μηχανή αναζήτησης επιστρέψει μόνο άσχετες ιστοσελίδες. Στην περίπτωση των μηχανών αναζήτησης, η ορθότητα είναι πιο σημαντική από την ανάκληση, αλλά η ανάκληση δεν πρέπει να αγνοηθεί. Ο χρήστης δεν μπορεί να αντέξει χαμηλή ορθότητα και υψηλή ανάκληση επειδή η μηχανή αναζήτησης επιστρέφει πολλές σχετικές ιστοσελίδες και πολλές ακόμη άσχετες. Εδώ, υπάρχει ο κίνδυνος να χαθούν οι σχετικές ιστοσελίδες μεταξύ των πολλών άσχετων ιστοσελίδων που επέστρεψε η μηχανή αναζήτησης και έτσι ο χρήστης να μην βρει αυτό που ψάχνει. Από την άλλη πλευρά, ο χρήστης μπορεί να αντέξει μια σχετικά χαμηλή ανάκληση με υψηλή ακρίβεια. Αυτό σημαίνει ότι ο χρήστης έχει την ευκαιρία να βρει αυτό που ψάχνει από τις λίγες σχετικές ιστοσελίδες που ανακτήθηκαν, αλλά ταυτόχρονα οι άσχετες ιστοσελίδες που εμφάνισε η μηχανή αναζήτησης και παραπλανούν τον χρήστη διατηρούνται στο ελάχιστο επίπεδο.

Αν και είναι ξεκάθαρο ότι για τον εντοπισμό εισβολών, η ανάκληση είναι σημαντικότερη από την ορθότητα στα συστήματα εντοπισμού εισβολών, ένα ακόμη μέτρο που πρέπει να λαμβάνεται υπόψιν είναι το F-Score το οποίο είναι το αρμονικό μέσο μεταξύ της ορθότητας και της ανάκλησης. Υπολογίζεται ως εξής:

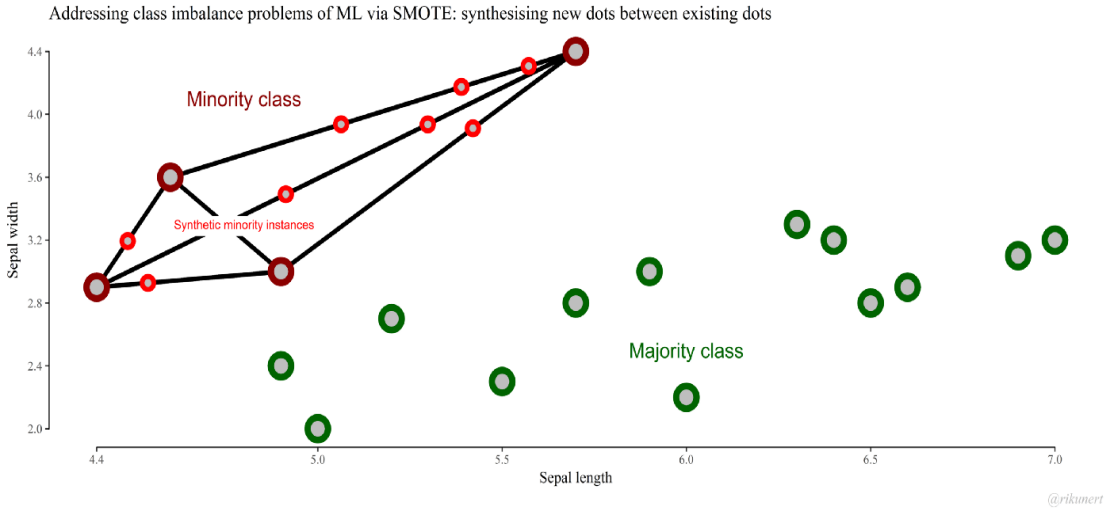
$$F - score = 2 * \frac{\frac{|TP|}{|TP|+|FP|} * \frac{|TP|}{|TP|+|FN|}}{\frac{|TP|}{|TP|+|FP|} + \frac{|TP|}{|TP|+|FN|}}$$

3.2 Αντιμετώπιση της άνισης κατανομής σε σύνολα δεδομένων που αφορούν εισβολές

Η αντιμετώπιση της άνισης κατανομής των κλάσεων μπορεί να εφαρμοστεί χρησιμοποιώντας είτε μια τεχνική υποδειγματοληψίας [35] είτε μια τεχνική υπερδειγματοληψίας. Οι τεχνικές που ανήκουν στην πρώτη κατηγορία προσπαθούν να αντιμετωπίσουν τις άνισες κατανομές των κλάσεων μειώνοντας τα στιγμιότυπα που ανήκουν στην “ισχυρή” κλάση. Στην περίπτωση μας προσπαθούν να μειώσουν τα στιγμιότυπα εκπαίδευσης που αφορούν την κλάση “Εισβολή”. Αντίθετα, οι τεχνικές υπερδειγματοληψίας [35, 36] προσπαθούν να “ενισχύσουν” την σπάνια κλάση. Στην περίπτωση μας προσπαθούν να ενισχύσουν την κλάση “Εισβολή”. Αυτό πραγματοποιείται δημιουργώντας τεχνητά στιγμιότυπα εκπαίδευσης της σπάνιας κλάσης και

προσθέτοντας τα στο σύνολο δεδομένων εκπαίδευσης. Τόσο η υπερδειγματοληψία όσο και η υποδειγματοληψία στοχεύουν στην εξισορρόπηση των κατανομών κλάσεων. Σημειώνεται, ότι η υπερδειγματοληψία, εξαιτίας της μεθόδου Synthetic Minority Over-Sampling Technique (SMOTE) [37], είναι πιο δημοφιλής προσέγγιση.

Η τεχνική SMOTE λειτουργεί ως εξής: Σε κάθε επανάληψη, επιλέγεται ένα τυχαίο στιγμιότυπο, έστω x, που ανήκει στη σπάνια κλάση. Στη συνέχεια, ανακτώνται τα κ εγγύτερα στιγμιότυπα τα οποία επίσης ανήκουν στην σπάνια κλάση. Συνήθως, χρησιμοποιείται κ = 5. Η τεχνική SMOTE συνεχίζει επιλέγοντας τυχαία έναν από τους γείτονες, έστω y και δημιουργεί ένα τεχνητό στιγμιότυπο σε ένα τυχαία σημείο μεταξύ του x και του y. Στην πραγματικότητα, τα x και y δημιουργούν μια γραμμή στο πολυδιάστατο χώρο χαρακτηριστικών και τα τεχνητά στιγμιότυπα δημιουργούνται σε τυχαίο σημείο σε αυτήν τη γραμμή. Οι επαναλήψεις σταματούν όταν τα δεδομένα είναι ισορροπημένα ή όταν δημιουργείται ένας προκαθορισμένος αριθμός τεχνητών στιγμιότυπων. Έτσι, το SMOTE μπορεί να χρησιμοποιηθεί για να δημιουργήσει όσα τεχνητά στιγμιότυπα σπάνια κλάσης απαιτούνται.



Σχήμα 8. Γραφική απεικόνιση της Τεχνικής SMOTE

Συνεπώς, χρησιμοποιώντας την Τεχνική SMOTE, μπορεί να χρησιμοποιηθεί σε συστήματα εντοπισμού εισβολών για να δημιουργήσει νέα τεχνητά στιγμιότυπα της κλάσης "εισβολή" και έτσι, το αντίστοιχο σύνολο δεδομένων εκπαίδευσης να μην περιλαμβάνει ακραίες ανισοκατανομές.

3.3 Απλοί (single) κατηγοριοποιητές εντοπισμού εισβολών

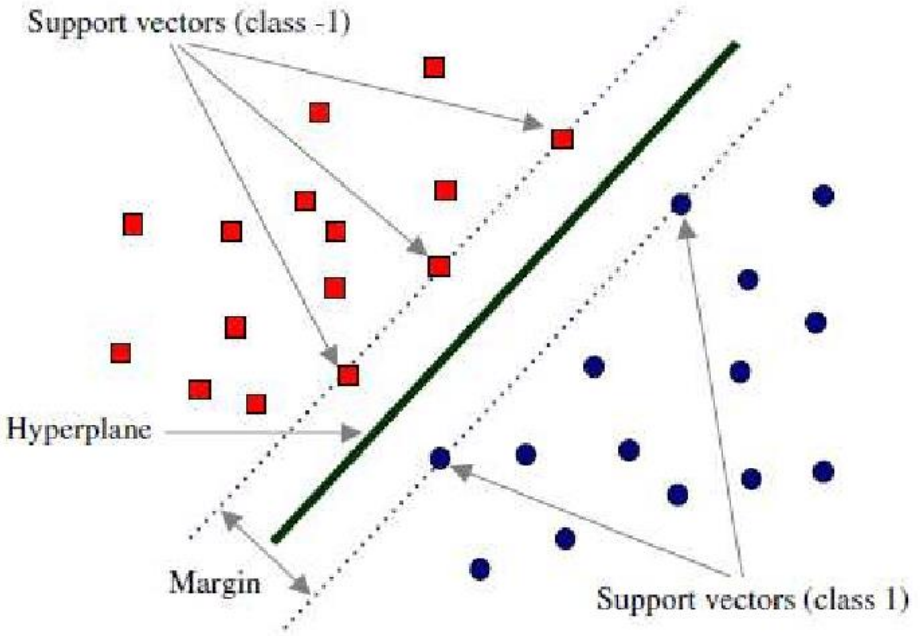
Το πρόβλημα ανίχνευσης εισβολής μπορεί να προσεγγιστεί με τη χρήση ενός μόνο αλγορίθμου εξόρυξης γνώσης και μηχανικής μάθησης. Στη βιβλιογραφία τεχνικές και αλγόριθμοι εξόρυξης γνώσης και μηχανικής μάθησης π.χ. Κατηγοριοποιητής κ εγγύτερων γειτόνων (k-nearest

neighbor), μηχανές διανυσμάτων υποστήριξης (support vector machines), τεχνητά νευρωνικά δίκτυα (artificial neural networks), δέντρα αποφάσεων (decision trees), αυτοοργανωτικοί χάρτες (self-organizing maps) κ.λπ. έχουν χρησιμοποιηθεί για τον εντοπισμό εισβολών σε υπολογιστικά και επικοινωνιακά συστήματα. Σε αυτή την ενότητα περιγράφονται οι αλγόριθμοι αυτοί.

3.3.1 Μηχανές διανυσμάτων υποστήριξης (Support vector machines)

Οι μηχανές διανυσμάτων υποστήριξης προτάθηκαν αρχικά από τον Vapnik [3]. Οι μηχανές αυτές "χαρτογραφούν" πρώτα το διάνυσμα εισόδου σε έναν χώρο χαρακτηριστικών πολλών διαστάσεων και στη συνέχεια αποκτά το βέλτιστο διαχωριστικό υπερ-επίπεδο που χωρίζει τον χώρο αυτόν. Επιπλέον, ένα όριο απόφασης, δηλαδή το διαχωριστικό υπερ-επίπεδο, καθορίζεται από διανύσματα υποστήριξης και όχι από ολόκληρα δείγματα εκπαίδευσης και, ως εκ τούτου, είναι εξαιρετικά ανθεκτικό στην ύπαρξη ακραίων τιμών.

Συγκεκριμένα, ένας κατηγοριοποιητής μηχανών διανυσμάτων υποστήριξης έχει σχεδιαστεί για δυαδική κατηγοριοποίηση (binary classification). Δηλαδή, στόχος είναι να διαχωρίσουμε ένα σύνολο διανυσμάτων εκπαίδευσης που ανήκουν σε δύο διαφορετικές κλάσεις. Σημειώστε ότι τα διανύσματα υποστήριξης είναι τα δείγματα εκπαίδευσης κοντά σε ένα όριο απόφασης μεταξύ των διαφορετικών κλάσεων του συνόλου δεδομένων εκπαίδευσης. Οι μηχανές διανυσμάτων υποστήριξης περιλαμβάνουν μια παράμετρο καθορισμένη από τον χρήστη που ονομάζεται παράγοντας ποινής. Επιτρέπει στους χρήστες να κάνουν μια αντιστάθμιση μεταξύ του αριθμού των εσφαλμένα ταξινομημένων δειγμάτων και του πλάτους ενός ορίου απόφασης.



Σχήμα 9. Εφαρμογή αλγορίθμου μηχανής διανύσματος υποστήριξης για τον διαχωρισμό ενός σετ δεδομένων σε δύο κλάσεις

Πολλές έρευνες που αφορούν τα NIDS εφάρμοσαν κατηγοριοποιητές βασισμένους σε μηχανές διανυσμάτων υποστήριξης επειδή, οι συγκεκριμένοι κατηγοριοποιητές καταφέρνουν υψηλές επιδόσεις γενίκευσης.

Έχοντας σαν βάση τις μηχανές διανυσμάτων υποστήριξης, δημιουργήθηκε ένα σύστημα ανίχνευσης εισβολής, το οποίο συνδυάζει έναν ιεραρχικό αλγόριθμο συσταδοποίησης, μια απλή διαδικασία επιλογής χαρακτηριστικών και την τεχνική μηχανών διανυσμάτων υποστήριξης [4]. Συγκεκριμένα, για την προεπεξεργασία δεδομένων χρησιμοποιήθηκε ο ιεραρχικός αλγόριθμος ομαδοποίησης BIRCH [38], ο οποίος παρέχει λιγότερη υποστήριξη στους κατηγοριοποιητές διανυσματικών μηχανών, αφηρημένα και ανώτερα στιγμιότυπα εκπαίδευσης (higher-qualified training instances), αντί για το αρχικό, μεγάλο σύνολο δεδομένων, τα οποία προέρχονται από το πολύ γνωστό σετ εκπαίδευσης KDD Cup 99. Το σύστημα αυτό ήταν σε θέση να συντομεύσει σημαντικά τον χρόνο εκπαίδευσης, αλλά και να βελτιώσει την απόδοση του προκύπτοντος κατηγοριοποιητή μηχανών διανυσμάτων υποστήριξης.

Ο αλγόριθμος BIRCH (balanced iterative reducing and clustering using hierarchies) είναι αλγόριθμος εξόρυξης δεδομένων χωρίς επίβλεψη με την χρήση του οποίο μπορεί να επιτευχθεί ιεραρχική κατηγοριοποίηση σε ιδιαίτερα μεγάλα σύνολα δεδομένων [48]. Επίσης δύναται να επιταχύνει σημαντικά την κατηγοριοποίηση κ-μέσων (k-means clustering) και την Γκαουσιανό μείγμα μοντελοποίησης (Gaussian mixture modeling) με τον αλγόριθμο αναμονής-μεγιστοποίησης (expectation-maximization algorithm). Ένα πλεονέκτημα του BIRCH είναι η δυνατότητα που έχει να κατηγοριοποιεί σταδιακά και δυναμικά, εισερχόμενα, πολυδιάστατα σημεία μετρικών δεδομένων προσπαθώντας με αυτόν τον τρόπο να δημιουργήσει την κατά το δυνατόν καλύτερη ποιοτικά ομαδοποίηση για ένα δεδομένο σύνολο πόρων (περιορισμοί μνήμης και χρόνου). Στις περισσότερες περιπτώσεις, το BIRCH απαιτεί μόνο μία σάρωση της βάσης δεδομένων. Η απόδοση των προηγούμενων αλγόριθμων κατηγοριοποίησης ήταν σημαντικά μικρότερη σε πολύ μεγάλες βάσεις δεδομένων. Επίσης οι αλγόριθμοι εκείνοι δεν εξέταζαν επαρκώς την περίπτωση κατά την οποία ένα σύνολο δεδομένων ήταν πολύ μεγάλο ώστε να είναι δυνατόν να χωρέσει στην κύρια μνήμη. Ως εκ τούτου, η διατήρηση της υψηλής ποιότητας της κατηγοριοποίησης δημιουργούσε μεγάλη επιβάρυνση, ενώ έγινε εφικτή η μείωση του κόστους των πρόσθετων λειτουργιών εισόδου/εξόδου (IO-input/output). Επιπλέον, οι περισσότεροι από τους προκατόχους του BIRCH ελέγχουν όλα τα σημεία δεδομένων (ή όλες τις υπάρχουσες συστάδες (clusters)) εξίσου για κάθε «απόφαση κατηγοριοποίησης» και δεν εκτελούν ευριστική στάθμιση βάσει της απόστασης μεταξύ αυτών των σημείων δεδομένων. Με την χρήση του αλγόριθμου BIRCH κάθε απόφαση ομαδοποίησης λαμβάνεται χωρίς να

πραγματοποιείται σάρωση όλων των σημείων δεδομένων και των υφιστάμενων συστάδων. Ουσιαστικά γίνεται εκμετάλλευση της παρατήρησης ότι ο χώρος των δεδομένων συνήθως δεν καταλαμβάνεται ομοιόμορφα και ότι κάθε σημείο δεδομένων δεν είναι εξίσου σημαντικό. Χρησιμοποιεί πλήρως τη διαθέσιμη μνήμη για να αντλήσει τις καλύτερες δυνατές υποσυστάδες, ενώ ελαχιστοποιεί το κόστος εισόδου/εξόδου (I/O). Είναι επίσης μια μέθοδος που λειτουργεί σταδιακά και άρα δεν απαιτεί ολόκληρο το σύνολο δεδομένων εκ των προτέρων.

Η απλή διαδικασία επιλογής χαρακτηριστικών εφαρμόστηκε για να εξαλειφθούν ασήμαντα χαρακτηριστικά από το σετ εκπαίδευσης, ώστε το ληφθέν μοντέλο SVM να μπορεί να ταξινομήσει τα δεδομένα κίνησης δικτύου με μεγαλύτερη ακρίβεια. Το σύνολο δεδομένων KDD Cup 1999 χρησιμοποιήθηκε για την αξιολόγηση του προτεινόμενου συστήματος. Σε σύγκριση με άλλα συστήματα ανίχνευσης εισβολής που βασίζονται στο ίδιο σύνολο δεδομένων, αυτό το σύστημα έδειξε καλύτερη απόδοση στην ανίχνευση επιθέσεων DoS και Probe και την συνολικά καλύτερη απόδοση στον τομέα της ακρίβειας. Συγκεκριμένα σύμφωνα με το πείραμα στο KDD Cup 1999, το προτεινόμενο σύστημα θα μπορούσε να φτάσει σε ακρίβεια 95,72% με ψευδώς θετικό ποσοστό 0,7%. Σε σύγκριση με άλλα NIDS που χρησιμοποίησαν επίσης το KDD Cup 1999 ως σύνολο δεδομένων, αυτό το σύστημα παρουσίασε ανώτερη απόδοση σε επιθέσεις DoS και Probe, αν και δεν ήταν το καλύτερο για επιθέσεις U2R και R2L. Ωστόσο, όσον αφορά την ακρίβεια, το προτεινόμενο σύστημα θα μπορούσε να έχει την καλύτερη απόδοση στο 95,72%. Αυτό το πείραμα πραγματοποιήθηκε σε ολόκληρο το σύνολο δεδομένων KDD Cup 1999 χωρίς δειγματοληψία. Ορισμένες νέες περιπτώσεις επίθεσης στο σύνολο δεδομένων δοκιμής, οι οποίες δεν εμφανίστηκαν ποτέ στην εκπαίδευση, θα μπορούσαν επίσης να ανιχνευθούν από αυτό το συγκεκριμένο σύστημα.

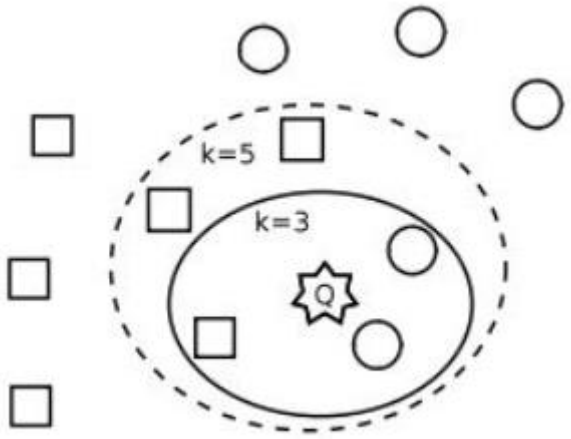
3.3.2 Κατηγοριοποιητής κ εγγύτερων γειτόνων

Ο αλγόριθμος k-εγγύτερων γειτόνων (k-NN) [5, 6]. αποτελεί έναν από τους πιο απλούς, κατανοητούς και εύκολους στη χρήση κατηγοριοποιητές. Επιπλέον, είναι ικανός να αντιμετωπίσει προβλήματα τόσο κατηγοριοποίησης όσο και παλινδρόμησης (regression). Λόγω της απλότητας του, καθώς και του γεγονότος ότι παράγει υψηλότερη ακρίβεια από πολλούς άλλους κατηγοριοποιητές, χρησιμοποιείται σε πολλές εφαρμογές. Όπως αναφέρθηκε και στο Κεφάλαιο 1, ο k-NN ανήκει στους οκνηρούς κατηγοριοποιητές, επομένως δεν χτίζει κάποιο μοντέλο κατηγοριοποίησης και χρησιμοποιεί τα δεδομένα εκπαίδευσης για να κατηγοριοποιήσει νέα δεδομένα. Ο αλγόριθμος ονομάζεται επίσης ως μάθηση βασισμένη σε παραδείγματα (instance based learning) και διαφέρει από την προσέγγιση της επαγωγικής μάθησης και τους πρόθυμους (eager) κατηγοριοποιητές [7]. Έτσι, δεν περιέχει το στάδιο

εκπαίδευσης, αλλά αναζητά μόνο στα δεδομένα εκπαίδευσης για να κατηγοριοποιήσει νέα στιγμιότυπα.

Ο k-εγγύτερων γειτόνων λειτουργεί ως εξής: κάθε φορά που πρέπει να κατηγοριοποιήσει ένα νέο στιγμιότυπο x, αναζητά να βρει τα k πιο κοντινά στιγμιότυπα εκπαίδευσης τα οποία ονομάζονται γείτονες, σύμφωνα με μια μετρική απόστασης. Μετά αναλύει σε ποια κλάση ανήκουν οι περισσότεροι από αυτούς τους k γείτονες και το x τελικά κατηγοριοποιείται σε αυτήν. Η ίδια διαδικασία επαναλαμβάνεται για κάθε στιγμιότυπο που πρέπει να κατηγοριοποιηθεί.

Στο παρακάτω σχήμα μπορούμε να δούμε με γραφικό τρόπο παράδειγμα της εκτέλεσης του κατηγοριοποιητή των k εγγύτερων γειτόνων. Έστω ότι έχουμε ένα σύνολο δεδομένων με στιγμιότυπα που ανήκουν σε δύο διαφορετικές κλάσεις: την κλάση Τετράγωνο και την κλάση Κύκλος. Επιπρόσθετα, έχουμε ένα αντικείμενο Q που πρέπει να κατηγοριοποιηθεί σε μία από τις δύο κλάσεις. Στην μία περίπτωση θέτουμε k=3 και από τους τρεις κοντινότερους γείτονες, οι δύο είναι κύκλοι. Επομένως το Q θα κατηγοριοποιηθεί ως κύκλος. Αντίθετα, αν θέσουμε k=5, τότε οι περισσότεροι από τους πέντε εγγύτερους γείτονες είναι τετράγωνα. Συνεπώς το Q θα κατηγοριοποιείται ως τετράγωνο.



Σχήμα 10. Λειτουργία του κατηγοριοποιητή κ εγγύτερων γειτόνων για κ=3 και για κ=5

Βέβαια, το ερώτημα που τίθεται είναι ποια είναι η κατάλληλη τιμή για το k. Η καλύτερη τιμή που μπορούμε να θέσουμε μπορεί να διαφέρει ανάλογα με το σύνολο δεδομένων που χρησιμοποιείται. Για παράδειγμα, αν έχουμε ένα σύνολο δεδομένων όπου παρατηρείται ένας μεγάλος αριθμός δεδομένων που αποτελούν θόρυβο, τότε προτιμάται μία μεγάλη τιμή για το k, αφού θα μας επιτρέψει να αναλύσουμε περισσότερους γείτονες και πιθανόν να ξεχωρίσουμε τα δεδομένα που είναι ``σωστά'' από τα αυτά που αποτελούν θόρυβο. Ωστόσο, επιλέγοντας μία υψηλή τιμή για το k, δεν μπορούμε να ξεχωρίσουμε το ίδιο καλά τα σύνορα μεταξύ των

κλάσεων. Αυτή είναι μόνο μία από τις πολλές πιθανές περιπτώσεις. Η βέλτιστη τιμή για το k συνήθως βρίσκεται δοκιμάζοντας πολλές τιμές και διαλέγοντας αυτήν που παράγει το καλύτερο αποτέλεσμα. Αξίζει να σημειωθεί ότι ένας παρόμοιος αλγόριθμος, ο 1-NN, είναι στην πραγματικότητα ο k-NN με k=1.

Στην περίπτωση που κατά την κατηγοριοποίηση ενός στιγμιότυπου, βρούμε ίσο αριθμό γειτόνων για δύο ή περισσότερες διαφορετικές κλάσεις, ο k-NN επιλέγει το σε ποια κλάση θα κατηγοριοποιηθεί το νέο στιγμιότυπο είτε τυχαία, είτε διαλέγοντας την κλάση που έχει ο πιο κοντινός γείτονας από όλους. Η μόνη περίσταση όπου δεν ισχύει αυτό είναι όταν έχουμε δυαδικό πρόβλημα κατηγοριοποίησης. Σε αυτήν την περίπτωση, θα πρέπει να θέσουμε έναν περιττό αριθμό ως τιμή για το k έτσι ώστε να αποφύγουμε κάθε περίπτωση "ισοπαλίας".

Ο τρόπος με τον οποίο υπολογίζεται η απόσταση μεταξύ του προς κατηγοριοποίηση στιγμιότυπου και των δεδομένων εκπαίδευσης εξαρτάται από την μετρική απόσταση που χρησιμοποιείται. Συνήθως χρησιμοποιείται η Ευκλείδεια απόσταση. Ωστόσο, μπορεί να χρησιμοποιηθεί οποιαδήποτε άλλη μετρική (π.χ. Manhatan). Ένα ακόμη σημείο που λαμβάνεται υπόψη είναι το εύρος των τιμών των χαρακτηριστικών. Έστω ότι έχουμε δύο διαφορετικά χαρακτηριστικά, ο μισθός και ο αριθμός παιδιών ενός ατόμου. Σε αυτήν την περίπτωση το εύρος τιμών του μισθού θα είναι πολύ μεγαλύτερο και λόγω αυτού θα συνεισφέρει περισσότερο στον υπολογισμό της απόστασης. Για αυτόν τον λόγο, συνιστάται η εκτέλεση μία διαδικασίας κανονικοποίησης ώστε τα δεδομένα να είναι στο ίδιο εύρος τιμών. Για παράδειγμα όλα τα χαρακτηριστικά να παίρνουν τιμές στο διάστημα [0-1] ώστε κάθε ένα από αυτά να συνεισφέρει το ίδιο στον υπολογισμό της απόστασης.

Ο κατηγοριοποιητής k εγγύτερων γειτόνων έχει χρησιμοποιηθεί για να κατηγοριοποιήσει τη συμπεριφορά ενός προγράμματος ως φυσιολογική ή επικίνδυνη [8]. Σύντομες ακολουθίες των κλήσεων του συστήματος έχουν χρησιμοποιηθεί από άλλους για να χαρακτηρίσουν την κανονική συμπεριφορά ενός προγράμματος στο παρελθόν. Ωστόσο, πρέπει να δημιουργηθούν ξεχωριστές βάσεις δεδομένων με σύντομες ακολουθίες κλήσεων συστήματος για διαφορετικά προγράμματα και τα προφίλ εκμάθησης προγραμμάτων περιλαμβάνουν χρονοβόρες διαδικασίες εκπαίδευσης και δοκιμών. Με τον κατηγοριοποιητή kNN, οι συχνότητες των κλήσεων του συστήματος χρησιμοποιούνται για να περιγράψουν τη συμπεριφορά του προγράμματος. Υιοθετούνται τεχνικές κατηγοριοποίησης κειμένου για τη μετατροπή κάθε διαδικασίας σε διάνυσμα και τον υπολογισμό της ομοιότητας μεταξύ δύο δραστηριοτήτων προγράμματος. Το γεγονός ότι δεν χρειάζεται να μάθουμε ανεξάρτητα προφίλ προγραμμάτων ξεχωριστά, μειώνει σε μεγάλο βαθμό τον χρόνο υπολογισμού. Διεξαγωγή πειραμάτων χρησιμοποιώντας το σετ δεδομένων DARPA BSM 1998 δείχνουν ότι ο κατηγοριοποιητής kNN μπορεί να ανιχνεύσει αποτελεσματικά παρεμβατικές επιθέσεις και να επιτύχει χαμηλό ποσοστό ψευδώς θετικών.

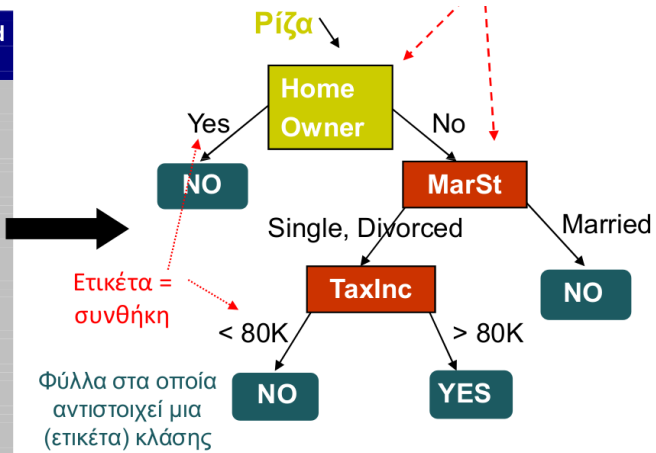
Αυτό το αποτέλεσμα ενδέχεται να μην ισχύει για ένα πιο εξελιγμένο σύνολο δεδομένων, ωστόσο ο κατηγοριοποιητής k-Nearest Neighbor φαίνεται να εφαρμόζεται με επιτυχία στον τομέα της ανίχνευσης εισβολής. Η τεχνική στάθμισης κατηγοριοποίησης tf idf υιοθετήθηκε για να μετατρέψει κάθε διαδικασία σε διάνυσμα. Με τη μέθοδο στάθμισης συχνότητας, όπου κάθε καταχώριση είναι ίση με τον αριθμό των εμφανίσεων μιας κλήσης συστήματος κατά την εκτέλεση της διαδικασίας, κάθε διάνυσμα διεργασίας δεν μεταφέρει πληροφορίες για άλλες διεργασίες.

Μια νέα διαδικασία κατάρτισης θα μπορούσε εύκολα να προστεθεί στο σύνολο δεδομένων εκπαίδευσης χωρίς να αλλάξει το βάρος των υπαρχόντων στιγμιότυπων εκπαίδευσης. Αυτό θα μπορούσε να καταστήσει τη μέθοδο κατηγοριοποίησης k εγγύτερων γειτόνων πιο κατάλληλη για δυναμικά περιβάλλοντα που απαιτούν συχνή ενημέρωση των δεδομένων εκπαίδευσης. Στην συγκεκριμένη εφαρμογή, χρησιμοποιήθηκαν όλες οι κλήσεις συστήματος για να αντιπροσωπευθεί η συμπεριφορά του προγράμματος. Η διάσταση των διανυσμάτων διαδικασίας (process vectors), και ως εκ τούτου το κόστος ταξινόμησης, μπορεί να μειωθεί περαιτέρω χρησιμοποιώντας μόνο τις πιο σχετικές κλήσεις συστήματος.

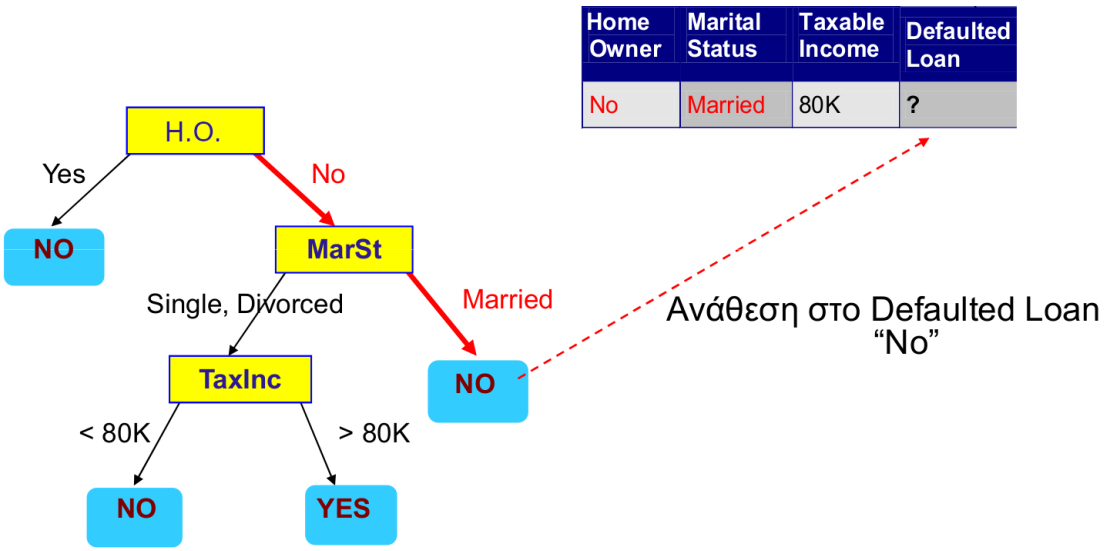
3.3.3 Δέντρα αποφάσεων (Decision trees)

Ένα δέντρο αποφάσεων κατηγοριοποιεί ένα στιγμιότυπο μέσω μιας ακολουθίας αποφάσεων, στην οποία η τρέχουσα απόφαση βοηθά στη λήψη της επόμενης απόφασης. Μια τέτοια ακολουθία αποφάσεων αντιπροσωπεύεται σε μια δομή δέντρου. Ουσιαστικά μπορούμε να πούμε πως πρόκειται για μια αλληλουχία “if then else” κανόνων, αλλά πολύ πιο ισχυρή. Η ταξινόμηση ενός δείγματος προέρχεται από τον κόμβο της ρίζας σε έναν κατάλληλο κόμβο τελικού φύλλου, όπου κάθε κόμβος τελικού φύλλου αντιπροσωπεύει μια κατηγορία ταξινόμησης. Τα χαρακτηριστικά των δειγμάτων εκχωρούνται σε κάθε κόμβο και η τιμή κάθε κλάδου αντιστοιχεί στα χαρακτηριστικά [7].

Tid	Home Owner	Marital Status	Taxable Income	Defaulted Loan
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



Σχήμα 11. Κατασκευή ενός δέντρου αποφάσεων



Σχήμα 12. Ταξινόμηση δείγματος με την βοήθεια ενός δέντρου αποφάσεων

Γνωστοί αλγόριθμοι κατασκευής τους δέντρου απόφασης είναι ο ID3 και η επέκταση του C4.5. Συγκεκριμένα, ο αλγόριθμος κατασκευής δέντρου απόφασης C4.5 είναι παραλλαγή/βελτίωση του αλγορίθμου ID3.

Ο αλγόριθμος ID3 (Iterative Dichotomiser 3) εφευρέθηκε από τον Ross Quinlan [49] και χρησιμοποιείται για τη δημιουργία ενός δέντρου αποφάσεων από ένα σύνολο δεδομένων. Ο ID3 είναι ο πρόδρομος του αλγορίθμου C4.5 και συνήθως χρησιμοποιείται στους τομείς μηχανικής μάθησης και επεξεργασίας φυσικής γλώσσας. Ο αλγόριθμος ID3 ξεκινά με το αρχικό σύνολο S ως κόμβο ρίζας. Σε κάθε επανάληψη του αλγορίθμου, επαναλαμβάνεται σε κάθε αχρησιμοποίητο χαρακτηριστικό του συνόλου S και υπολογίζει την εντροπία H (S) ή το κέρδος πληροφορίας IG(S) αυτού του χαρακτηριστικού. Στη συνέχεια, επιλέγει το χαρακτηριστικό που έχει τη μικρότερη τιμή εντροπίας (ή το μεγαλύτερο κέρδος πληροφορίας). Το σύνολο S στη συνέχεια χωρίζεται ή διαμερίζεται από το επιλεγμένο χαρακτηριστικό για να παράγει υποσύνολα των δεδομένων. (Για παράδειγμα, ένας κόμβος μπορεί να χωριστεί σε θυγατρικούς κόμβους με βάση τα υποσύνολα του πληθυσμού των οποίων οι ηλικίες είναι μικρότερες από 50, μεταξύ 50 και 100 και μεγαλύτερες από 100.) Ο αλγόριθμος συνεχίζει να επαναλαμβάνεται σε κάθε υποσύνολο, λαμβάνοντας υπόψη μόνο όσα χαρακτηριστικά δεν επιλέχθηκαν προηγουμένως.

Η επανάληψη σε ένα υποσύνολο μπορεί να σταματήσει όταν συντρέχει κάποιος από τους παρακάτω λόγους:

Κάθε στοιχείο της υποομάδας ανήκει στην ίδια κατηγορία. Σε αυτή την περίπτωση ο κόμβος μετατρέπεται σε κόμβο φύλλου και επισημαίνεται με την κλάση των παραδειγμάτων.

Δεν υπάρχουν άλλα χαρακτηριστικά προς επιλογή, αλλά τα παραδείγματα εξακολουθούν να μην ανήκουν στην ίδια κατηγορία. Σε αυτή την περίπτωση, ο κόμβος γίνεται κόμβος φύλλου και επισημαίνεται με την πιο κοινή κλάση των παραδειγμάτων στο υποσύνολο.

Δεν υπάρχουν παραδείγματα στο υποσύνολο, κάτι που συμβαίνει όταν κανένα παράδειγμα στο γονικό σύνολο δεν βρέθηκε να ταιριάζει με μια συγκεκριμένη τιμή του επιλεγμένου χαρακτηριστικού. Ένα παράδειγμα θα μπορούσε να είναι η απουσία ατόμου μεταξύ του πληθυσμού με ηλικία άνω των 100 ετών. Στη συνέχεια, δημιουργείται ένας κόμβος φύλλων και επισημαίνεται με την πιο κοινή κλάση των παραδειγμάτων στο σύνολο του γονικού κόμβου.

Σε όλο τον αλγόριθμο, το δέντρο αποφάσεων κατασκευάζεται με κάθε μη τερματικό κόμβο (εσωτερικό κόμβο) που αντιπροσωπεύει το επιλεγμένο χαρακτηριστικό στο οποίο χωρίστηκαν τα δεδομένα και με τερματικούς κόμβους (κόμβους φύλλων) που αντιπροσωπεύουν την ετικέτα κλάσης του τελικού υποσυνόλου αυτού του κλάδου.

Ο ID3 δεν εγγυάται τη βέλτιστη λύση. Μπορεί να συγκλίνει σε τοπικά βέλτιστα. Χρησιμοποιεί μια στρατηγική που χαρακτηρίζεται ως άπληστη, επιλέγοντας κάθε φορά το τοπικά καλύτερο χαρακτηριστικό για να χωρίσει το σύνολο δεδομένων σε κάθε επανάληψη. Η βελτιστοποίηση του αλγορίθμου μπορεί να καταστεί εφικτή εφόσον χρησιμοποιηθεί το backtracking κατά την αναζήτηση του βέλτιστου δέντρου αποφάσεων με πιθανό κόστος τον μεγαλύτερο χρόνο εκτέλεσης.

Ο ID3 μπορεί να υπερκαλύψει τα δεδομένα εκπαίδευσης. Για να αποφευχθεί η υπερβολική προσαρμογή (overfitting), θα πρέπει να προτιμώνται τα μικρότερα δέντρα αποφάσεων έναντι των μεγαλύτερων. Ενώ αυτός ο αλγόριθμος παράγει συνήθως μικρά δέντρα, δεν παράγει πάντα το μικρότερο δυνατό δέντρο αποφάσεων.

Ο ID3 είναι πιο δύσκολο να χρησιμοποιηθεί σε συνεχή δεδομένα από ό,τι σε δεδομένα που έχουν ήδη συνυπολογιστεί (τα δεδομένα που έχουν συνυπολογιστεί έχουν έναν διακριτό αριθμό πιθανών τιμών, μειώνοντας έτσι τα πιθανά σημεία διακλάδωσης). Εάν οι τιμές οποιουδήποτε χαρακτηριστικού είναι συνεχείς, τότε υπάρχουν πολλά περισσότερα μέρη για να χωριστούν τα δεδομένα με βάση το συγκεκριμένο χαρακτηριστικό και η αναζήτηση της καλύτερης τιμής για διαίρεση μπορεί να είναι χρονοβόρα.

Ένας γνωστός αλγόριθμος για την κατασκευή δέντρων αποφάσεων είναι ο CART (Classification and Regressing Tree) [9]. Ένα δέντρο αποφάσεων με μια σειρά από διακριτές (συμβολικές) ετικέτες κλάσης ονομάζεται δέντρο ταξινόμησης, ενώ ένα δέντρο αποφάσεων με ένα εύρος συνεχών (αριθμητικών) τιμών ονομάζεται δέντρο παλινδρόμησης.

Τα δέντρα αποφάσεων μπορούν να αναλύσουν δεδομένα και να εντοπίσουν σημαντικά χαρακτηριστικά στο δίκτυο που υποδεικνύουν κακόβουλες δραστηριότητες. Μπορεί να προσδώσουν επιπρόσθετη αξία σε πολλά συστήματα ασφαλείας πραγματικού χρόνου, αναλύοντας ένα μεγάλο σύνολο δεδομένων ανίχνευσης εισβολών. Επίσης έχουν την

δυνατότητα να αναγνωρίσουν τάσεις και μοτίβα που υποστηρίζουν περαιτέρω διερεύνηση, την ανάπτυξη υπογραφών επίθεσης και άλλες δραστηριότητες παρακολούθησης. Το κύριο πλεονέκτημα της χρήσης δέντρων αποφάσεων αντί άλλων τεχνικών ταξινόμησης είναι ότι παρέχουν ένα πλούσιο σύνολο κανόνων το οποίο είναι εύκολα κατανοητό και μπορούν να συνδυαστούν αβίαστα με τεχνολογίες πραγματικού χρόνου [10]

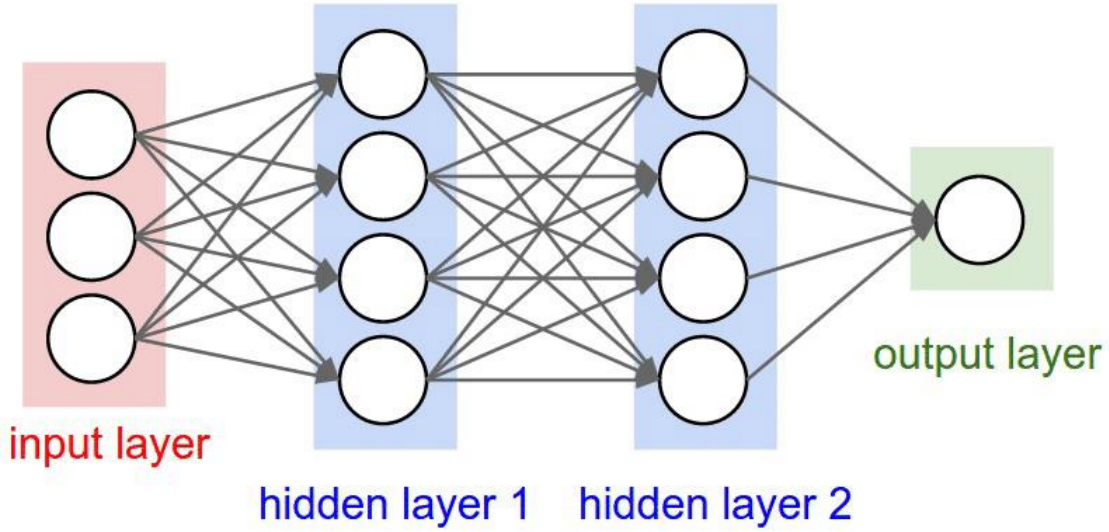
Ένας αλγόριθμος για την ανίχνευση επιθέσεων που βασίστηκε σε δέντρο αποφάσεων αναπτύχθηκε με βάση την προσέγγιση του δέντρου αποφάσεων C4.5 [11]. Η επιλογή χαρακτηριστικών και η τιμή διαίρεσης (split value) είναι σημαντικά ζητήματα για τη δημιουργία ενός δέντρου αποφάσεων. Ο αναφερόμενος αλγόριθμος σχεδιάστηκε προκειμένου να αντιμετωπίσει αυτά τα ζητήματα. Τα πιο συναφή χαρακτηριστικά επιλέγονται με τη χρήση του κέρδους πληροφοριών (information gain) και η τιμή διαίρεσης επιλέγεται με τέτοιο τρόπο που καθιστά τον κατηγοριοποιητή αμερόληπτο προς τις πιο συχνές τιμές. Ο πειραματισμός πραγματοποιείται στο σύνολο δεδομένων NSL-KDD (Network Security Laboratory Knowledge Discovery and Data Mining) με βάση τον αριθμό των χαρακτηριστικών.

Από τα αποτελέσματα των πειραμάτων, συμπεραίνεται ότι ο προτεινόμενος αλγόριθμος για την ανίχνευση εισβολής βάσει υπογραφών είναι πιο αποτελεσματικός σε σχέση με την εύρεση επιθέσεων στο δίκτυο με μικρότερο αριθμό χαρακτηριστικών και χρειάζεται λιγότερος χρόνος για την κατασκευή του μοντέλου. Συμπεραίνεται επίσης ότι η αποτελεσματικότητα εξαρτάται από το μέγεθος του συνόλου δεδομένων και τον αριθμό των χαρακτηριστικών που χρησιμοποιούνται για την κατασκευή του δέντρου αποφάσεων. Ο τύπος που χρησιμοποιείται στο DTS για τον υπολογισμό του λόγου κέρδους μπορεί επίσης να χρησιμοποιηθεί στην επιλογή χαρακτηριστικών για μείωση χαρακτηριστικών. Αναλύεται ο χρόνος που χρειάζεται ο ταξινομητής για την κατασκευή του μοντέλου και η ακρίβεια που επιτυγχάνεται. Τελικά συμπεραίνεται ότι ο προτεινόμενος αλγόριθμος Decision Tree Split (DTS) μπορεί να χρησιμοποιηθεί για ανίχνευση εισβολής βάσει υπογραφής.

3.3.4 Τεχνητά νευρωνικά δίκτυα - Artificial neural networks

Το νευρωνικό δίκτυο είναι μονάδες επεξεργασίας πληροφοριών που μιμούνται τους νευρώνες του ανθρώπινου εγκεφάλου [12]. Το πολυστρωματικό perceptron (MLP) είναι μια ευρέως χρησιμοποιούμενη αρχιτεκτονική νευρωνικών δικτύων σε πολλά προβλήματα αναγνώρισης προτύπων. Ένα δίκτυο MLP αποτελείται από ένα επίπεδο εισόδου που περιλαμβάνει ένα σύνολο αισθητήριων κόμβων ως κόμβους εισόδου, ένα ή περισσότερα κρυμμένα στρώματα υπολογιστικών κόμβων και ένα επίπεδο εξόδου κόμβων υπολογισμού. Κάθε διασύνδεση έχει συσχετίσει με ένα κλιμακωτό βάρος το οποίο προσαρμόζεται κατά τη διάρκεια της εκμάθησης (training). Επιπλέον, ο αλγόριθμος εκμάθησης backpropagation χρησιμοποιείται συνήθως για την εκπαίδευση ενός MLP, τα οποία επίσης ονομάζονται και νευρωνικά δίκτυα

backpropagation. Πρώτα απ 'όλα, δίνονται τυχαία βάρη στην αρχή της εκμάθησης. Στη συνέχεια, ο αλγόριθμος εκτελεί συντονισμό βαρών για να καθορίσει ποια παράσταση κρυφής μονάδας είναι πιο αποτελεσματική στην ελαχιστοποίηση του σφάλματος της λανθασμένης ταξινόμησης.



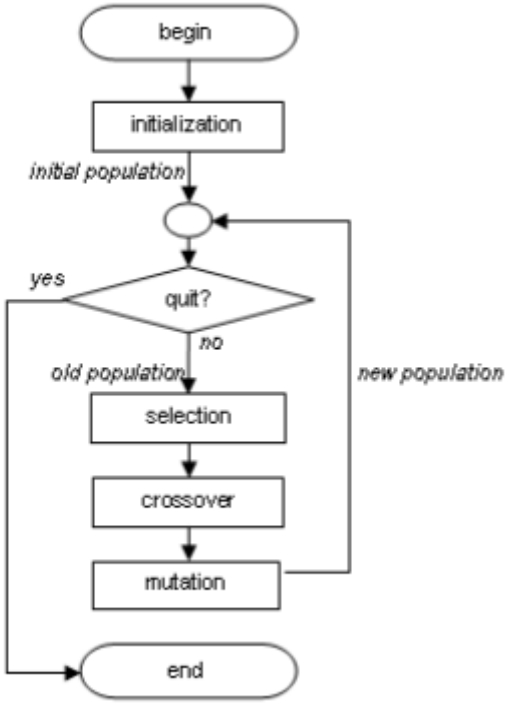
Σχήμα 13. Γραφική αναπαράσταση της λειτουργίας ενός τεχνητού νευρωνικού δικτύου

Μια τέτοια προσέγγιση για τον εντοπισμό κακόβουλης κίνησης δικτύου χρησιμοποιεί τεχνητά νευρωνικά δίκτυα κατάλληλα για χρήση σε συστήματα ανίχνευσης εισβολών βασισμένων σε επιθεώρηση πακέτων βαθιάς βάσης [13]. Πειραματικά αποτελέσματα χρησιμοποιώντας μια σειρά τυπικών καλοηθών δεδομένων κίνησης δικτύου (εικόνες, αρχεία βιβλιοθήκης δυναμικής σύνδεσης και μια επιλογή από άλλα διάφορα αρχεία όπως αρχεία καταγραφής, αρχεία μουσικής και έγγραφα επεξεργασίας κειμένου) και κακόβουλα αρχεία κώδικα κελύφους που προέρχονται από την online βιβλιοθήκη κενών ασφαλείας και ευπαθειών exploitdb, έχουν δείξει ότι η προτεινόμενη αρχιτεκτονική τεχνητού νευρωνικού δικτύου είναι σε θέση να διακρίνει με ακρίβεια μεταξύ της καλοήθους και της κακόβουλης κίνησης δικτύου. Η προτεινόμενη αρχιτεκτονική τεχνητού νευρωνικού δικτύου αποκτά μια μέση ακρίβεια 98%, μια μέση περιοχή κάτω από τη χαρακτηριστική καμπύλη του χειριστή δέκτη 0,98, και ένα μέσο ψευδώς θετικό ποσοστό μικρότερο από 2% σε ένα σχήμα 10-fold cross validation. Αυτό δείχνει ότι η συγκεκριμένη τεχνική ταξινόμησης είναι στιβαρή, ακριβής και λεπτομερής. Η προσέγγιση αυτή για την ανίχνευση κακόβουλης κίνησης δικτύου έχει τη δυνατότητα να ενισχύσει σημαντικά τη χρησιμότητα συστημάτων ανίχνευσης εισβολών που εφαρμόζονται τόσο στη συμβατική ανάλυση κίνησης δικτύου όσο και στην ανάλυση κίνησης δικτύου για κυβερνο-φυσικά συστήματα όπως τα έξυπνα δίκτυα.

3.3.5 Γενετικοί αλγόριθμοι Genetic algorithms

Οι γενετικοί αλγόριθμοι (GA) χρησιμοποιούν τον υπολογιστή για να εφαρμόσουν τη φυσική επιλογή και εξέλιξη [14]. Αυτή η έννοια προέρχεται από την "προσαρμοστική επιβίωση σε φυσικούς οργανισμούς". Ο αλγόριθμος ξεκινά δημιουργώντας τυχαία έναν μεγάλο πληθυσμό υποψηφίων προγραμμάτων. Χρησιμοποιείται κάποιος τύπος μέτρησης ικανότητας για την αξιολόγηση της απόδοσης κάθε ατόμου σε έναν πληθυσμό. Στη συνέχεια εκτελείται ένας μεγάλος αριθμός επαναλήψεων κατά τον οποίο τα προγράμματα χαμηλής απόδοσης αντικαθίστανται από γενετικούς ανασυνδυασμούς προγραμμάτων υψηλής απόδοσης. Δηλαδή, ένα πρόγραμμα με χαμηλό μέτρο ικανότητας διαγράφεται και δεν επιβιώνει για την επόμενη επανάληψη του υπολογιστή.

Στο σχήμα 1 παρουσιάζεται ένα σύστημα ανίχνευσης εισβολής (IDS), στο οποίο εφαρμόζεται γενετικός αλγόριθμος (GA) για την αποτελεσματική ανίχνευση διαφόρων τύπων εισβολών δικτύου.



Σχήμα 14. Διάγραμμα ροής του Γενετικού Αλγόριθμου

Αυτή η προσέγγιση [26] χρησιμοποιεί τη θεωρία της εξέλιξης στην εξέλιξη των πληροφοριών προκειμένου να φιλτράρει τα δεδομένα κίνησης και έτσι να μειώσει την πολυπλοκότητα. Το σύστημα αυτό μπορεί να χωριστεί σε δύο κύριες φάσεις: τη φάση του προκαταρκτικού υπολογισμού και τη φάση της ανίχνευσης. Η λίστα 1 απεικονίζει σημαντικά βήματα στη φάση του προκαταρκτικού υπολογισμού, όπου δημιουργείται ένα σύνολο χρωμοσωμάτων

χρησιμοποιώντας δεδομένα εκπαίδευσης. Αυτό το σετ χρωμοσωμάτων θα χρησιμοποιηθεί στην επόμενη φάση για λόγους σύγκρισης.

Λίστα 1 Σημαντικά βήματα στον προκαταρκτικό υπολογισμό

Αλγόριθμος: Αρχικοποίηση χρωμοσωμάτων για σύγκριση

Εισαγωγή: Δεδομένα ελέγχου δικτύου (για εκπαίδευση)

Έξοδος: Ένα σύνολο χρωμοσωμάτων

1. Range = 0.125
2. For each training data
3. If it has neighboring chromosome within Range
4. Merge it with the nearest chromosome
5. Else
6. Create new chromosome with it
7. End if
8. End for

Η λίστα 2 απεικονίζει σημαντικά βήματα της φάσης ανίχνευσης, όπου δημιουργείται ένας πληθυσμός για δεδομένα δοκιμής και περνά από ορισμένες διαδικασίες αξιολόγησης (επιλογή, διασταύρωση, μετάλλαξη) και προβλέπεται ο τύπος των δεδομένων δοκιμής. Το προκαθορισμένο σύνολο χρωμοσωμάτων χρησιμοποιείται σε αυτή τη φάση για να διαπιστωθεί η καταλληλότητα κάθε χρωμοσώματος του πληθυσμού.

Λίστα 2 Σημαντικά βήματα στην ανίχνευση

Αλγόριθμος: Προβλέψτε δεδομένα/τύπο εισβολής (χρησιμοποιώντας GA)

Εισαγωγή: Δεδομένα ελέγχου δικτύου (για δοκιμές), προκαθορισμένο σύνολο χρωμοσωμάτων

Έξοδος: Τύπος δεδομένων.

1. Initialize the population
2. CrossoverRate = 0.15, MutationRate = 0.35

3. While number of generation is not reached

4. For each chromosome in the population

5. For each precalculated chromosome

6. Find fitness

7. End for

8. Assign optimal fitness as the fitness of that chromosome

9. End for

10. Remove some chromosomes with worse fitness

11. Apply crossover to the selected pair of chromosomes of the population

12. Apply mutation to each chromosome of the population

13. End while

Για την εφαρμογή και τη μέτρηση της απόδοσης του συστήματος αυτού, χρησιμοποιήθηκε το σύνολο δεδομένων αναφοράς KDD99, το οποίο είχε σαν αποτέλεσμα ένα λογικό ποσοστό ανίχνευσης.

3.3.6 Αυτοοργανωτικοί χάρτες (SOM - Self-organizing maps)

Ο αυτοοργανωτικός χάρτης (Self-organizing maps - SOM) [27] εκπαιδεύεται από έναν χωρίς επίβλεψη ανταγωνιστικό αλγόριθμο μάθησης, μια διαδικασία αυτοοργάνωσης. Ο στόχος του SOM είναι να μειώσει τη διάσταση της απεικόνισης δεδομένων. Δηλαδή, το SOM προβάλλει και συγκεντρώνει διανύσματα εισόδου υψηλής διάστασης σε έναν οπτικοποιημένο χάρτη χαμηλής διάστασης, συνήθως 2 για οπτικοποίηση. Συνήθως αποτελείται από ένα στρώμα εισόδου και το στρώμα Kohonen το οποίο έχει σχεδιαστεί ως δισδιάστατη διάταξη νευρώνων που χαρτογραφεί είσοδο n διαστάσεων σε δύο διαστάσεις. Το SOM του Kohonen συσχετίζει καθένα από τα διανύσματα εισόδου σε μια αντιπροσωπευτική έξοδο. Το δίκτυο βρίσκει τον κόμβο που βρίσκεται πιο κοντά σε κάθε περίπτωση προπόνησης και μετακινεί τον νικητήριο κόμβο, ο οποίος είναι ο πλησιέστερος νευρώνας (δηλαδή ο νευρώνας με την ελάχιστη απόσταση) στην προπόνηση. Δηλαδή, το SOM χαρτογραφεί παρόμοια διανύσματα εισόδου στις ίδιες ή παρόμοιες μονάδες εξόδου σε έναν τέτοιο δισδιάστατο χάρτη. Επομένως, οι μονάδες εξόδου θα αυτο-οργανωθούν σε έναν διατεταγμένο χάρτη και οι μονάδες εξόδου με παρόμοια βάρη τοποθετούνται επίσης κοντά μετά την προπόνηση.

Ο αλγόριθμος "Self Organizing Maps" (SOM) είναι μια πολλά υποσχόμενη τεχνική που έχει χρησιμοποιηθεί σε πολλά προβλήματα ταξινόμησης και στην συνέχεια θα παρουσιαστεί η

δυνατότητα χρησιμοποίησής του σε συστήματα εντοπισμού εισβολής (IDS). Το στοιχείο νευρωνικού δικτύου θα υλοποιήσει τη νευρωνική προσέγγιση, η οποία βασίζεται στην υπόθεση ότι κάθε χρήστης είναι μοναδικός και αφήνει ένα μοναδικό αποτύπωμα σε ένα σύστημα υπολογιστή όταν το χρησιμοποιεί. Εάν το αποτύπωμα ενός χρήστη δεν ταιριάζει με το αποτύπωμα αναφοράς που βασίζεται σε συνήθεις δραστηριότητες του συστήματος, ο διαχειριστής του συστήματος ή ο υπεύθυνος ασφαλείας μπορεί να ειδοποιηθεί για πιθανή παραβίαση της ασφάλειας [28].

Η συγκεκριμένη εφαρμογή του αλγόριθμου περιλαμβάνει τα παρακάτω τρία στάδια:

Συλλογή δεδομένων (data collection): Εάν η κίνηση του δικτύου έχει εξεταστεί προσεκτικά για διαφορετικούς τύπους συμβάντων, όπως λήψη (downloading), σάρωση θυρών (port scanning), σερφάρισμα (surfing) κ.λπ., είναι δυνατό να εντοπιστούν οι τυπικές διαφορές μεταξύ τους. Σκοπός της ενέργειας αυτής είναι η συλλογή ξεχωριστών και διαφορετικών ειδών πακέτων δικτύου. Για τη συλλογή δεδομένων μπορούμε να χρησιμοποιήσουμε οποιοδήποτε πακέτο sniffer που είναι διαθέσιμο άμεσα. Στην εξεταζόμενη περίπτωση αναπτύχθηκε ένα πακέτο sniffer. Εκτός από τη λήψη ζωντανών (live) πακέτων, χρησιμοποιήθηκε ένα τυπικό σύνολο δεδομένων DARPA για εκπαιδευτικούς σκοπούς. Το σύνολο δεδομένων περιέχει πακέτα με και χωρίς εισβολή.

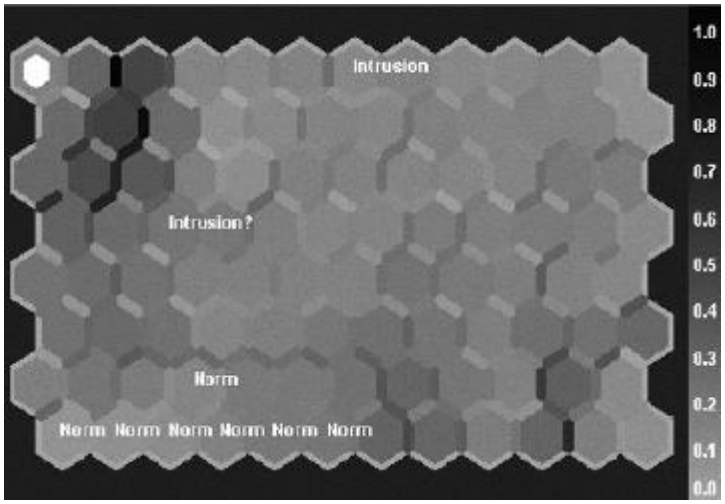
Εξαγωγή διανυσμάτων (vector extraction): Μετά τη διαδικασία συλλογής δεδομένων, τα χαρακτηριστικά θα πρέπει να εξάγονται από τα δεδομένα προκειμένου να επιτευχθούν καλύτερα αποτελέσματα ταξινόμησης. Έτσι εξάγονται ένα προς ένα προκειμένου να έχουμε την δημιουργία νέων διανυσμάτων τα οποία θα χρησιμοποιηθούν για την παροχή εισόδου στον αυτοοργανωμένο χάρτη. Στον Πίνακα 1. εμφανίζονται παραδείγματα ληφθέντων διανυσμάτων για το χρονικό διάστημα παραθύρου είναι ίσο με 5. Για την συγκεκριμένη εφαρμογή γίνεται η παραδοχή πως το μήκος παραθύρου είναι ίσο με 20. Δεδομένου ότι τα δεδομένα συλλέγονται σε κάθε 20 δευτερόλεπτα, ένα διάνυσμα εισόδου αντιστοιχεί σε χρονικό διάστημα 400 δευτερολέπτων.

V1	V2	V3	V4	V5
16.1	14.3	24	125	128
14.6	25	125	133	11.2
23	122	134	10.5	16
122	129	10.3	18	19
132	10.6	16	19	15.5

Πίνακας 1. Πίνακας που δείχνει τα εξαγόμενα διανύσματα

Εκπαίδευση του χάρτη αυτοδιοργάνωσης (training self organizing map): Για εκπαιδευτικούς σκοπούς κατασκευάστηκε ένας Χάρτη αυτοοργάνωσης 30 × 30 για να εκτελεστεί η ομαδοποίηση (clustering). Τα δεδομένα που χρησιμοποιήθηκαν για αυτό ήταν το σύνολο δεδομένων DARPA. Χρησιμοποιήθηκε ο αλγόριθμο κατάρτισης κατά παρτίδες με μήκος εκπαίδευσης 100 και ακτίνα έναρξης 15. Ο αυτοοργανωτικός χάρτης ήταν σε μεγάλο βαθμό επιτυχής στην ταξινόμηση των πακέτων IP.

Μετά τη συλλογή δεδομένων, την εξαγωγή διανυσμάτων και την εκπαίδευση των Χαρτών Αυτοδιοργάνωσης τα πακέτα πέρασαν μέσω του SOM. Το αποτέλεσμα φαίνεται στο σχήμα 15.



Σχήμα 15. Τα αποτελέσματα του πειράματος

Τα αποτελέσματα του σχήματος 15 δείχνουν την ταξινόμηση διανυσμάτων εισόδου, η οποία αντιπροσωπεύει τη συμπεριφορά και τη χαρτογράφησή της σε συγκεκριμένους νευρώνες, οι οποίοι σχηματίζουν μεμονωμένες πιθανές καταστάσεις συμπεριφοράς του χρήστη. Δημιουργούνται καταστάσεις όπως εισβολή - εισβολή, πιθανή εισβολή - Εισβολή; Κανονικό - Κανονικό. Από το αποτέλεσμα της δοκιμής εξάγεται το συμπέρασμα πως το δίκτυο SOM μπορεί να αποτελέσει έναν κατάλληλο πυρήνα για συστήματα IDS.

Πλεονεκτήματα του SOM:

- Απλός και κατανοητός αλγόριθμος που λειτουργεί.
- Τοπολογική ομαδοποίηση.
- Αλγόριθμος χωρίς επίβλεψη που λειτουργεί με μη γραμμικά σύνολα δεδομένων.
- Η εξαιρετική ικανότητα απεικόνισης δεδομένων υψηλής διάστασης σε χώρο 1 ή 2 διαστάσεων το καθιστά μοναδικό, ειδικά όταν πρόκειται για μείωση διαστάσεων.

Μειονεκτήματα του SOM:

- Χρονοβόρος αλγόριθμος, αυτό συμβαίνει επειδή ο αριθμός. των νευρώνων επηρεάζει την απόδοση του αλγορίθμου. Και καθώς ο αριθμός αυτός αυξάνεται, ο υπολογισμός αυξάνεται, με αποτέλεσμα την αύξηση του χρόνου υπολογισμού.

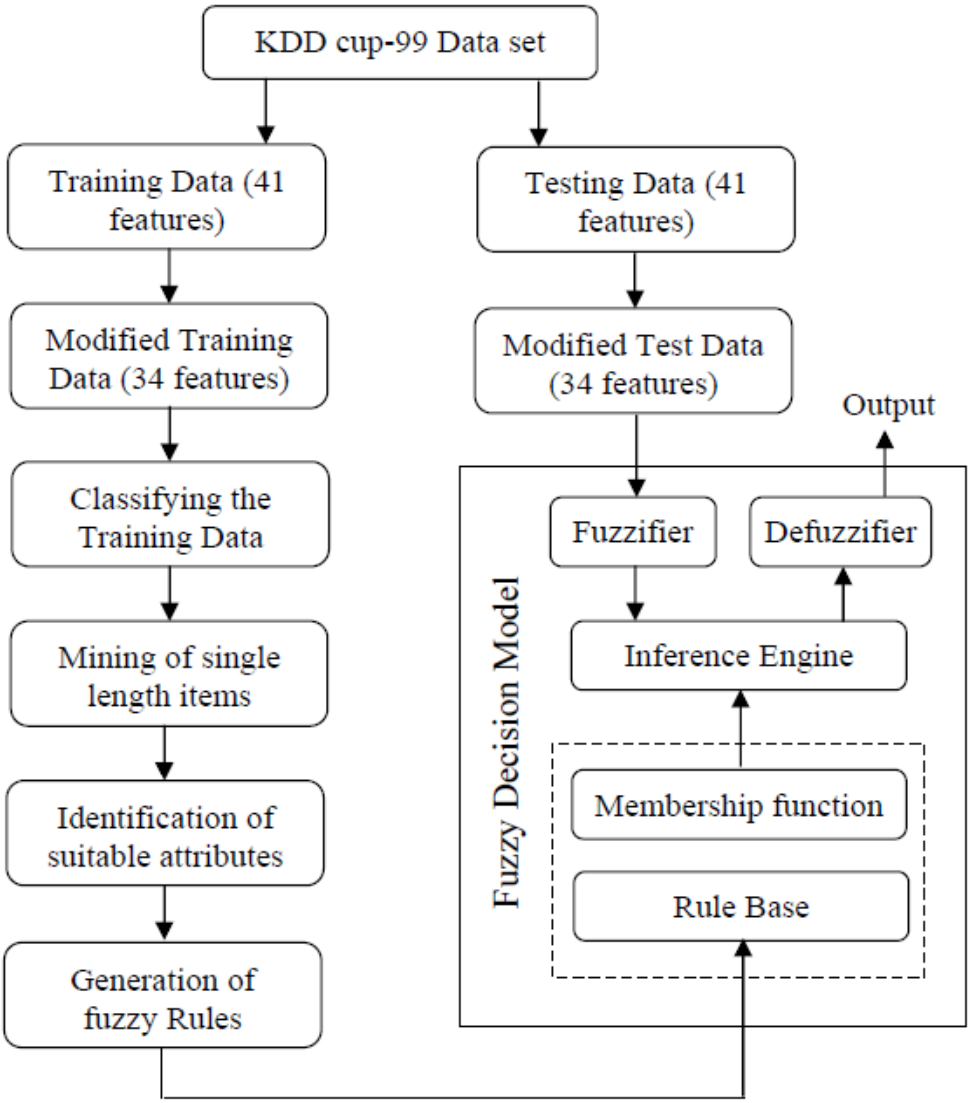
Γενικά ο αυτοοργανωτικός χάρτης είναι ένας εξαιρετικά ισχυρός μηχανισμός για αυτόματο μαθηματικό χαρακτηρισμό αποδεκτής δραστηριότητας συστήματος. Τα πραγματικά πειράματά έδειξαν ότι ακόμη και ένας απλός χάρτης, όταν εκπαιδευτεί σε κανονικά δεδομένα, θα ανιχνεύσει τα ανώμαλα χαρακτηριστικά και των δύο εισβολών υπερχείλισης buffer στα οποία εκτέθηκε. Αυτή η προσέγγιση είναι ιδιαίτερα ισχυρή επειδή ο χάρτης αυτοοργάνωσης λειτουργεί χωρίς να γνωρίζει εκ των προτέρων πώς μοιάζει η παρεμβατική συμπεριφορά. Μαθαίνοντας να χαρακτηρίζει την κανονική συμπεριφορά, προετοιμάζεται σιωπηρά για τον εντοπισμό τυχόν παρεκκλίνουσας δραστηριότητας δικτύου.

3.3.7 Ασαφής λογική - Fuzzy logic

Η ασαφής λογική (ή η θεωρία ασαφών συνόλων) βασίζεται στην έννοια του ασαφούς φαινομένου που συμβαίνει συχνά στον πραγματικό κόσμο. Η θεωρία ασαφών συνόλων λαμβάνει υπόψη τις τιμές μέλους συνόλου για το σκεπτικό και οι τιμές κυμαίνονται μεταξύ 0 και 1. Δηλαδή, στην ασαφή λογική ο βαθμός αλήθειας μιας δήλωσης μπορεί να κυμαίνεται μεταξύ 0 και 1 και δεν περιορίζεται στις δύο τιμές αλήθειας (π.χ. Σωστό Λάθος). Για παράδειγμα, η «βροχή» είναι ένα κοινό φυσικό φαινόμενο και μπορεί να έχει πολύ έντονες αλλαγές. Η βροχή μπορεί να μετατρέψει τις συνθήκες από ελαφρές σε βίαιες [29].

Παρακάτω θα γίνει αναφορά σε ένα σύστημα ανίχνευσης εισβολής το οποίο σχεδιάστηκε χρησιμοποιώντας ασαφή λογική [30]. Η είσοδος στο προτεινόμενο σύστημα είναι το σύνολο δεδομένων KDD Cup 1999, το οποίο χωρίζεται σε δύο υποσύνολα, το σύνολο δεδομένων εκπαίδευσης και το σύνολο δεδομένων δοκιμής. Αρχικά, το σύνολο δεδομένων κατάρτισης ταξινομείται σε πέντε υποσύνολα, ώστε να διαχωρίζονται τέσσερις τύποι επιθέσεων (DoS-Denial of Service (άρνηση υπηρεσίας), R2L-Remote to Local (απομακρυσμένο σε τοπικό), U2R-user to root (χρήστης σε ρίζα), probe (διερεύνηση)) και κανονικά δεδομένα. Μετά από αυτό, διαχωρίζονται τα συχνά αντικείμενα μήκους 1 από τα δεδομένα επίθεσης καθώς και κανονικά δεδομένα. Αυτά τα συχνά εξαγόμενα στοιχεία χρησιμοποιούνται για την εύρεση των σημαντικών χαρακτηριστικών του συνόλου δεδομένων εισόδου και τα προσδιορισμένα αποτελεσματικά χαρακτηριστικά χρησιμοποιούνται για τη δημιουργία ενός συνόλου ορισμένων και αόριστων κανόνων χρησιμοποιώντας τη μέθοδο απόκλισης. Στη συνέχεια, δημιουργείται ο ασαφής κανόνας σύμφωνα με τον καθορισμένο κανόνα, ο οποίος

διαμοιράζεται με τέτοιο τρόπο, ώστε στο τέλος να δημιουργηθεί ένα σύνολο ασαφών κανόνων if-then με επακόλουθα μέρη τα οποία υποδεικνύουν αν πρόκειται για κανονικά ή μη φυσιολογικά δεδομένα. Αυτοί οι κανόνες δίνονται στη βάση ασαφών κανόνων προκειμένου να εκπαιδευτεί αποτελεσματικά το ασαφές σύστημα. Στη φάση της δοκιμής, τα δεδομένα δοκιμής ταιριάζουν με ασαφείς κανόνες για να ανιχνευθεί εάν τα δεδομένα δοκιμής είναι μη φυσιολογικά δεδομένα ή κανονικά δεδομένα. Το σύνολο των σταδίων του παραπάνω συστήματος παρουσιάζονται στο παρακάτω σχήμα.



Σχήμα 16. Στάδια λειτουργίας ενός συστήματος ανίχνευσης εισβολών με την χρήση κανόνων ασαφούς λογικής (Fuzzy logic)

3.3.8 Αλγόριθμος Naive Bayes

Ο αλγόριθμος Naive Bayes έχει την βάση του στο θεώρημα του Bayes και η λειτουργία του περιλαμβάνει τον υπολογισμό της πιθανότητας για κάθε πιθανή κλάση βάση των χαρακτηριστικών που δίνονται και την κατάταξη του συγκεκριμένου στιγμιότυπου στην κλάση που παρουσιάζει την μεγαλύτερη πιθανότητα.

Στο πρώτο του βήμα ο αλγόριθμος βρίσκει την προκαταβολική πιθανότητα που έχει κάθε πιθανή τιμή που περιλαμβάνεται στην κλάση. Ο υπολογισμός αυτό πραγματοποιείται χρησιμοποιώντας τον αριθμό των συνολικών εμφανίσεων της τιμής αυτής μέσα στο δοθέν σύνολο δεδομένων. Μετά, για κάθε χαρακτηριστικό ξεχωριστά, ο αλγόριθμος προχωράει στον υπολογισμό των εκ των προτέρων πιθανοτήτων. Ουσιαστικά υπολογίζεται η πιθανότητα εμφάνισης στο συγκεκριμένο σύνολο δεδομένων για κάθε τιμή που μπορεί να λάβει το συγκεκριμένο χαρακτηριστικό. Στην συνέχεια για κάθε συνδυασμό υπολογίζεται η πιθανότητα εμφάνισής του. Αυτό σημαίνει πως υπολογίζεται, ξεχωριστά για κάθε χαρακτηριστικό, η κάθε πιθανή τιμή που μπορεί αυτό να λάβει. Αντίστοιχα το ίδιο γίνεται και για κάθε πιθανή τιμή που μπορεί να λάβουν οι κλάσεις.

Όταν πλέον έχει εκπαιδευτεί ο αλγόριθμος, όπως περιγράφηκε παραπάνω, είναι δυνατόν συγκεκριμένα στιγμιότυπα, δηλαδή σύνολα συγκεκριμένων τιμών για κάθε χαρακτηριστικό να καταταγούν στην κλάση που εμφανίζει την μεγαλύτερη πιθανότητα.

Πλεονεκτήματα – Μειονεκτήματα του κατηγοριοποιητή Naïve Bayes

Όπως όλοι οι αλγόριθμοι κατηγοριοποίησης, ο κατηγοριοποιητής Naïve Bayes, εμφανίζει πλεονεκτήματα και μειονεκτήματα:

Πλεονεκτήματα

Είναι δυνατή η χρησιμοποίηση του για εκπαίδευση και σε μεγάλα σύνολα δεδομένων

Χρειάζεται ένα μόνο πέρασμα στα δεδομένα, γεγονός που τον καθιστά ιδιαίτερα γρήγορο στην εκπαίδευσή του

Κατανοείται εύκολα

Μειονεκτήματα

Όταν σε μια συγκεκριμένη κλάση η πιθανότητα εμφάνισης για την τιμή ενός χαρακτηριστικού είναι μηδέν, στην περίπτωση που αυτή η τιμή περιλαμβάνεται σε κάποιο από τα στιγμιότυπα που δίνονται προς κατηγοριοποίηση, η τιμή που θα πάρει τελικά η πιθανότητα για την συγκεκριμένη κλάση θα είναι επίσης ίση με μηδέν.

Ο αλγόριθμος Naive Bayes κάνει την παραδοχή πως όλα τα υπό μελέτη χαρακτηριστικά δεν παρουσιάζουν κανενός είδους αλληλεξάρτηση. Στην πραγματικότητα όμως κάτι τέτοιο είναι σχεδόν αδύνατον να συμβεί. Το γεγονός αυτό καθιστά απαραίτητη την εκτέλεση κάποιας

τεχνικής που θα επιτρέψει την επιλογή χαρακτηριστικών, πριν αρχίσει η εκπαίδευση του μοντέλου.

Δεν έχει την δυνατότητα χειρισμού συνεχών τιμών, κάτι που έχει ως αποτέλεσμα η εκτέλεση διακριτοποίηση πριν την εκκίνηση της εκπαίδευσης του μοντέλου να κρίνεται επιβεβλημένη.

Παράδειγμα

Έστω το σύνολο δεδομένων του παρακάτω πίνακα:

KBytes	Category	Εισβολή
20	C	Όχι
30	C	Όχι
25	B	Ναι
30	A	Όχι
40	A	Όχι
20	B	Ναι
30	C	Όχι
25	C	Όχι
40	C	Όχι
20	A	Ναι

Πίνακας 2. Σύνολο δεδομένων με δύο ιδιότητες και μια κλάση

Βήμα 1ο:Υπολογισμός των εκ των προτέρων πιθανότητες της κλάσης.

P(Όχι) = 7/10

P(Ναι) = 3/10

Βήμα 2ο:Υπολογισμός των εκ των προτέρων πιθανότητες των χαρακτηριστικών.

P(C) = 5/10 , P(B) = 2/10 , P(A) = 3/10

P(20) = 3/10 , P(25) = 2/10 , P(30) = 3/10 , P(40) = 2/10

Βήμα 3ο:Υπολογισμός πιθανότητας του κάθε συνδυασμού.

P(C | Όχι) = 5/7 , P(B | Όχι) = 0/7, P(A | Όχι) = 2/7

P(C | Ναι) = 0/3 , P(B | Ναι) = 2/3, P(A | Ναι) = 1/3

P(20 | Όχι) = 1/7 , P(25 | Όχι) = 1/7 , P(30 | Όχι) = 3/7, P(40 | Όχι) = 2/7

P(20 | Ναι) = 2/3 , P(25 | Ναι) = 1/3 , P(30 | Ναι) = 0/3, P(40 | Ναι) = 0/3

Βήμα 4ο:Κατάταξη στιγμιότυπου.

Στιγμιότυπο: (30,C)

Στιγμιότυπο: (25,B)

Στιγμιότυπο: (20,A)

Για το 1ο στιγμιότυπο:

• P(Όχι | 30,C) = P(C | Όχι) *P(30 | Όχι) * P(Όχι) = (5/7) * (3/7) * (7/10) = 0.214

• P(Ναι | 30,C) = P(C | Ναι) * P(30 | Ναι) * P(Ναι) = (0/3) * (0/3) * (3/10) = 0.0

P(Όχι | 30,C) > P(Ναι | 30,C), οπότε τα καταταχθεί στην κλάση «Όχι»

Για το 2ο στιγμιότυπο:

• P(Όχι | 25,B) = P(Έγγαμος | Όχι) * P(25 | Όχι) * P(Όχι) = (0/7) * (1/7) * (7/10) = 0.0

• P(Ναι | 25,B) = P(Έγγαμος | Όχι) * P(25 | Όχι)* P(Όχι) = (2/3) * (1/3) * (3/10) = 0.067

P(Όχι | 25,B) < P(Ναι | 25,B), οπότε τα καταταχθεί στην κλάση «Ναι»

Για το 3ο στιγμιότυπο:

• P(Όχι | 20,C) = P(A | Όχι) * P(20 | Όχι) * P(Όχι) = (2/7) * (1/7) * (7/10) = 0.0285

• P(Ναι | 20,C) = P(A | Ναι) * P(20 |Ναι) * P(Ναι) = (1/3) * (2/3) * (3/10) = 0.0667

P(Όχι | 20,A) < P(Ναι | 20,A), οπότε τα καταταχθεί στην κλάση «Ναι»

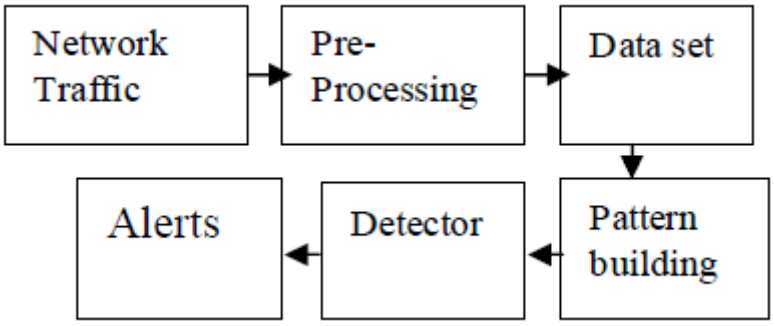
Οπότε έχουμε σαν αποτελέσματα τα εξής:

Στιγμιότυπο 1ο : κλάση «Όχι»,

Στιγμιότυπο 2ο : κλάση «Ναι»

Στιγμιότυπο 3ο : κλάση «Ναι»

Ο αλγόριθμος Naive Bayes χρησιμοποιήθηκε προκειμένου να δημιουργηθεί ένα σύστημα που εντοπίζει δικτυακές εισβολές (NIDS - Network Intrusion Detection System) [39]. Η δομή στην οποία στηρίχθηκε το σύστημα αυτό φαίνεται στο παρακάτω σχήμα.



Σχήμα 17. Η δομή του μοντέλου ανίχνευσης εισβολής που βασίστηκε στον αλγόριθμο Naïve Bayes

Το περιβάλλον που χρησιμοποιήθηκε ήταν το WEKA (Waikato Environment for Knowledge Analysis) και τα πειράματα εκτελέστηκαν στο 10% του συνόλου δεδομένων KDDCup'99. Η εκτίμηση της απόδοσης του συστήματός βασίστηκε στο ποσοστό ανίχνευσης (detection rate) και στο ψευδώς θετικό ποσοστό (false positive rate). Το ποσοστό ανίχνευσης λοιπόν με την χρήση αυτού του μοντέλου ανίχνευσης ήταν 95%, με ένα ποσοστό λάθους της τάξης του 5%. Επιπρόσθετα χρειάστηκε μόλις 1,89 δευτερόλεπτα προκειμένου να δημιουργήσει το μοντέλο. Αυτό το καθιστά πιο αποτελεσματικό και γρήγορο από το σύστημα που δημιουργήθηκε με την χρήση του κατηγοριοποιητή K-means [40] του οποίο το ποσοστό ανίχνευσης ήταν 92% και ο χρόνος εκτέλεσης 28 λεπτά και 21 δευτερόλεπτα. Συγκριτικά όμως με τα αποτελέσματα ενός δικτύου back propagation, ο αλγόριθμος Naive-Bayes δίνει περισσότερα εσφαλμένα θετικά αποτελέσματα, αλλά υπερέχει στους τομείς της απόδοσης, του κόστους και του χρόνου εκτέλεσης.

3.3.8.1 Naïve Baysian Networks

Υπάρχουν πολλές περιπτώσεις όπου γνωρίζουμε τις στατιστικές εξαρτήσεις ή τις αιτιώδεις σχέσεις μεταξύ των μεταβλητών του συστήματος. Ωστόσο, μπορεί να είναι δύσκολο να εκφραστούν με ακρίβεια οι πιθανολογικές σχέσεις μεταξύ αυτών των μεταβλητών. Με άλλα λόγια, η προηγούμενη γνώση για το σύστημα είναι απλώς ότι κάποια μεταβλητή μπορεί να επηρεάσει άλλες. Για την εκμετάλλευση αυτής της δομικής σχέσης ή των τυχαίων εξαρτήσεων μεταξύ των τυχαίων μεταβλητών ενός προβλήματος, μπορεί να χρησιμοποιηθεί ένα πιθανολογικό μοντέλο γράφου που ονομάζεται Naïve Baysian Networks (NB).

Το μοντέλο παρέχει μια απάντηση σε ερωτήσεις όπως «Ποια είναι η πιθανότητα να πρόκειται για συγκεκριμένο τύπο επίθεσης, δεδομένων κάποιων παρατηρούμενων συμβάντων του συστήματος;» χρησιμοποιώντας τον τύπο πιθανοτήτων υπό όρους. Η δομή ενός NB αντιπροσωπεύεται τυπικά από ένα κατευθυνόμενο ακυκλικό γράφο (Directed Acyclic Graph - DAG), όπου κάθε κόμβος αντιπροσωπεύει μία από τις μεταβλητές του συστήματος και κάθε σύνδεσμος κωδικοποιεί την επίδραση ενός κόμβου στον άλλο [31]. Έτσι, εάν υπάρχει σύνδεσμος από τον κόμβο Α στον κόμβο Β, το Α επηρεάζει άμεσα το Β.

3.4 Υβριδικοί (hybrid) κατηγοριοποιητές εντοπισμού εισβολών

Κατά την ανάπτυξη ενός Συστήματος Ανίχνευσης Εισβολής (IDS), ο υπέρτατος στόχος είναι να επιτευχθεί η καλύτερη δυνατή ακρίβεια για την συγκεκριμένη εργασία. Αυτός ο στόχος οδήγησε, όπως ήταν αναμενόμενο, στον σχεδιασμό υβριδικών προσεγγίσεων που έχουν ως σκοπό την αρτιότερη επίλυση του προβλήματος. Η ιδέα πίσω από έναν υβριδικό κατηγοριοποιητή είναι να συνδυάσει πολλές τεχνικές μηχανικής μάθησης, έτσι ώστε η

απόδοση του συστήματος να βελτιωθεί σημαντικά. Πιο συγκεκριμένα, μια υβριδική προσέγγιση αποτελείται συνήθως από δύο λειτουργικά στοιχεία. Το πρώτο λαμβάνει ακατέργαστα τα δεδομένα ως είσοδο και παράγει ενδιάμεσα αποτελέσματα. Το δεύτερο θα λάβει στη συνέχεια τα ενδιάμεσα αποτελέσματα και θα παράγει τα τελικά αποτελέσματα [41]. Πρακτικά, οι υβριδικοί κατηγοριοποιητές είναι απλή κατηγοριοποιητές οι οποίοι συνδυάζονται με κάποιου είδους προεπεξεργασία.

Συγκεκριμένα, οι υβριδικοί κατηγοριοποιητές (hybrid classifiers) μπορούν να βασίζονται σε διαδοχικούς διαφορετικούς κατηγοριοποιητές, όπως οι νευρο-ασαφείς (neuro-fuzzy) τεχνικές. Από την άλλη πλευρά, οι υβριδικοί κατηγοριοποιητές μπορούν να χρησιμοποιήσουν κάποια προσέγγιση βασισμένη σε ομαδοποίηση για την προεπεξεργασία των δειγμάτων εισόδου, προκειμένου να εξαλειφθούν μη αντιπροσωπευτικά δείγματα εκπαίδευσης από κάθε κλάση. Στη συνέχεια, τα αποτελέσματα της ομαδοποίησης χρησιμοποιούνται ως παραδείγματα εκπαίδευσης για το σχεδιασμό των κατηγοριοποιητών. Επομένως, το πρώτο επίπεδο των υβριδικών κατηγοριοποιητών μπορεί να βασιστεί σε τεχνικές μάθησης είτε με, είτε χωρίς επίβλεψη.

Τέλος, οι υβριδικοί κατηγοριοποιητές μπορούν επίσης να βασίζονται στην ενσωμάτωση δύο διαφορετικών τεχνικών, στις οποίες η πρώτη στοχεύει στη βελτιστοποίηση της μαθησιακής απόδοσης (δηλαδή ρύθμιση παραμέτρων) του δεύτερου μοντέλου που θα χρησιμοποιηθεί για πρόβλεψη.

3.5 Συνδυαστικοί (ensemble) κατηγοριοποιητές εντοπισμού εισβολών

Οι συνδυαστικοί κατηγοριοποιητές (ensemble classifiers) [43,44,45] χρησιμοποιούν πολλαπλούς αλγόριθμους κατηγοριοποίησης για να επιτύχουν καλύτερη ακρίβεια κατά την κατηγοριοποίηση και για να έχουν γενικά καλύτερη απόδοση από αυτή που θα μπορούσε να έχει ένας οποιοσδήποτε απλός αλγόριθμος μάθησης. Ένας συνδυαστικός κατηγοριοποιητής αποτελείται μόνο από ένα συγκεκριμένο πεπερασμένο σύνολο εναλλακτικών μοντέλων, αλλά τυπικά επιτρέπει την ύπαρξη μιας πολύ πιο ευέλικτης δομής μεταξύ αυτών των εναλλακτικών λύσεων. Οι συνδυαστικοί κατηγοριοποιητές προτάθηκαν για τη βελτίωση της απόδοσης κατηγοριοποίησης ενός μεμονωμένου κατηγοριοποιητή [42]. Ο όρος «συνδυαστικός» αναφέρεται στον συνδυασμό πολλαπλών απλών κατηγοριοποιητών Οι απλοί κατηγοριοποιητές εκπαιδεύονται σε διαφορετικά τμήματα του συνόλου εκπαίδευσης, έτσι ώστε η συνολική απόδοση να μπορεί να βελτιωθεί σημαντικά.

Ο συνδυασμός αδύναμων κατηγοριοποιητών μπορεί να μην υπερβαίνει απαραίτητα την απόδοση των καλύτερων κατηγοριοποιητών στο σύνολο. Ωστόσο, μπορεί να ελαχιστοποιήσει

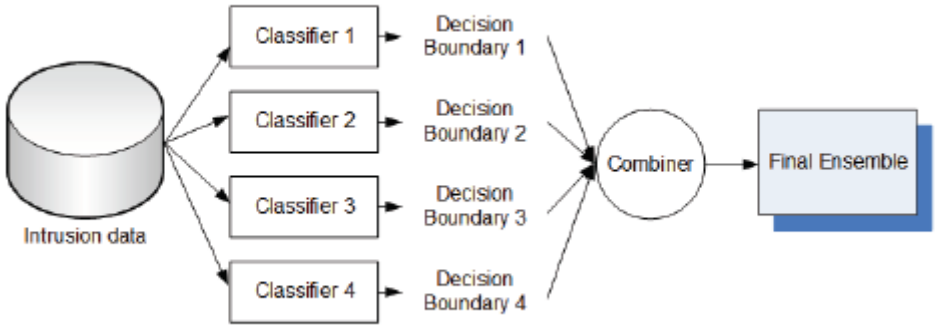
την ακαταλληλότητα της επιλογής των κατηγοριοποιητών που θα χρησιμοποιηθούν με νέα δεδομένα-στόχους. Εκπαιδεύονται αρκετά διαφορετικά υποσύνολα των σετ εκπαίδευσης και κάθε υποσύνολο παράγει διαφορετικά όρια σφάλματος, ωστόσο ο συνδυαστής μπορεί να δημιουργήσει το καλύτερο όριο απόφασης.

Μερικά από τα σχήματα συνδυασμού που δημιουργούνται με ετερογενείς κατηγοριοποιητές είναι τα παρακάτω:

Bagging. Το όνομα Bagging προκύπτει από τον συνδυασμό των λέξεων Booststrap Aggregating [51]. Ο αλγόριθμος Bagging υιοθετεί τον παράλληλο σχεδιασμό με βάση τον οποίο οι βασικοί κατηγοριοποιητές δημιουργούνται παράλληλα. Όπως υποδηλώνεται από το όνομα, εφαρμόζει δειγματοληψία bootstrap για τη λήψη των υποσυνόλων δεδομένων για την εκπαίδευση των βασικών κατηγοριοποιητών. Επιπλέον, ο bagging υιοθετεί στρατηγικές ψηφοφορίας με πλειοψηφία για κατηγοριοποίηση. Για να προβλέψει ένα στιγμιότυπο δοκιμής, ο Bagging τροφοδοτεί το παράδειγμα στους βασικούς κατηγοριοποιητές του και συλλέγει όλες τις εξόδους τους. Στην συνέχεια ψηφίζει τις ετικέτες και παίρνει την νικήτρια ετικέτα ως πρόβλεψη.

Ενίσχυση (boosting). Σε αντίθεση με το Bagging, ο αλγόριθμος ενίσχυσης υιοθετεί διαδοχικές συνδυαστικές μεθόδους στις οποίες οι βασικοί κατηγοριοποιητές δημιουργούνται διαδοχικά. Εν συντομία, η λειτουργία του αλγόριθμου της ενίσχυσης στηρίζεται στην διαδοχική εκπαίδευση ενός συνόλου κατηγοριοποιητών και εν συνεχεία στον συνδυασμό τους με σκοπό την πρόβλεψη. Στα πλαίσια αυτής της λειτουργίας οι μεταγενέστεροι κατηγοριοποιητές επικεντρώνονται περισσότερο στα λάθη των προηγούμενων κατηγοριοποιητών [54]. Στην βιβλιογραφία εμφανίζονται πολλές παραλλαγές ενίσχυσης [52], με πιο σημαντική από αυτές για την ερευνητική κοινότητα να φαίνεται να είναι ο Multiboost [53].

Ψηφοφορία της πλειοψηφίας (majority voting). Στον αλγόριθμο αυτόν, όπως απεικονίζεται στο σχήμα που ακολουθεί, κάθε κατηγοριοποιητής ψηφίζει υπέρ μιας συγκεκριμένης ετικέτας κλάσης και η τελική ετικέτα κλάσης εξόδου είναι αυτή που λαμβάνει περισσότερες από τις μισές ψήφους. Σε αντίθετη περίπτωση θα δοθεί μια επιλογή απόρριψης [50].

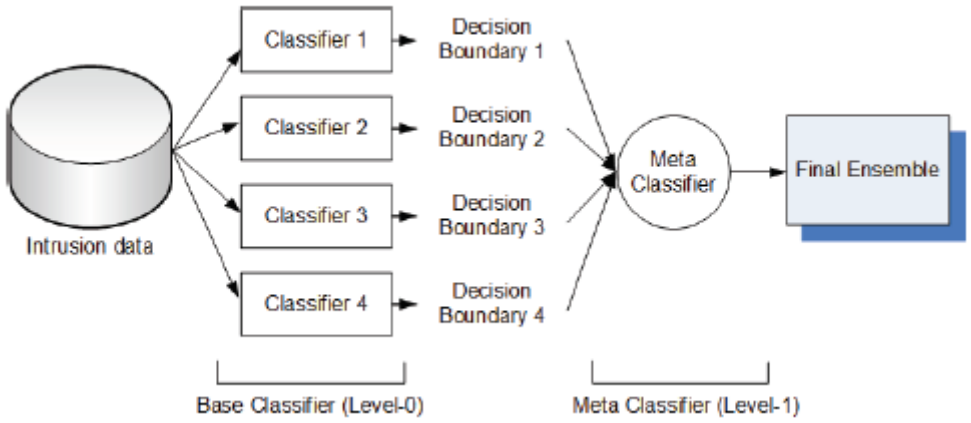


Σχήμα 18. Συνδυαστικός κατηγοριοποιητής με χρήση της ψηφοφορίας της πλειοψηφίας

Δίνονται T ανεξάρτητοι κατηγοριοποιητές $\{h_1, ..., h_T\}$ που σκοπός τους είναι ο συνδυασμός των hi προκειμένου να γίνει πρόβλεψη για την ετικέτα κλάσης, η οποία θα επιλεγεί από ένα σύνολο που περιέχει l ετικέτες τάξης $\{c_1, ..., c_l\}$. Θεωρείται ότι για ένα στιγμιότυπο x, οι τελικές έξοδοι του κατηγοριοποιητή h_i δίνονται ως ένα διάνυσμα με ετικέτα l -διαστάσεων $(h_i^1(x), \ldots, h_i^l(x))^T$, το οποίο $h_i^j(x)$ είναι η έξοδος του h_i για την ετικέτα κλάσης c_j. Στη συνέχεια, το $h_i^j(x) \in \{0, 1\}$ που παίρνει την τιμή ένα αν το h_i προβλέπει την c_j ως την ετικέτα της κλάσης, ενώ σε διαφορετική περίπτωση παίρνει την τιμή μηδέν. Η ετικέτα της κλάσης εξόδου της πλειοψηφικής ψήφου εκφράζεται ως εξής:

$$H(x) = \alpha(x) = \begin{cases} c_j & if \sum_{i=1}^{T} h_i^j(x) > \frac{1}{2} \sum_{k=1}^{l} \sum_{i=1}^{T} h_i^k(x) \\ rejection \end{cases}$$

Στοίβαξη (stacking). Όπως φαίνεται στο παρακάτω σχήμα , η στοίβαξη υιοθετεί την έννοια του μετα-κατηγοριοποιητή (κατηγοριοποιητής επιπέδου 1) για να συνδυάσει την ξεχωριστή έξοδο των βασικών κατηγοριοποιητών (κατηγοριοποιητές επιπέδου 0). Παρόλο που μπορούμε να επιλέξουμε οποιονδήποτε κατηγοριοποιητή ως κατηγοριοποιητή επιπέδου 1, ωστόσο, η στοίβαξη με γραμμική παλινδρόμηση (LR) έχει παρουσιάσει καλή απόδοση σε πολλούς τομείς εφαρμογών. Προκειμένου να αποφευχθεί η υπερβολική προσαρμογή (over-fitting), συνιστάται συχνά η διαδικασία εγκυρότητας για τη δημιουργία του μοντέλου κατηγοριοποιητή επιπέδου 1.



Σχήμα 19. Συνδυαστικός κατηγοριοποιητής με χρήση της στοίβαξης

Μεταξύ των στρατηγικών για τον συνδυασμό απλών κατηγοριοποιητών που αναφέρθηκαν παραπάνω, η «ψήφος της πλειοψηφίας» (majority voting) είναι αναμφισβήτητα η πιο συχνά χρησιμοποιούμενη στη βιβλιογραφία. Άλλες συνδυαστικές μέθοδοι, όπως η ενίσχυση και η αποθήκευση, βασίζονται στην αναμόρφωση των δεδομένων εκπαίδευσης και στη συνέχεια στη λήψη της πλειοψηφίας των ασθενών μαθητών που προκύπτουν.

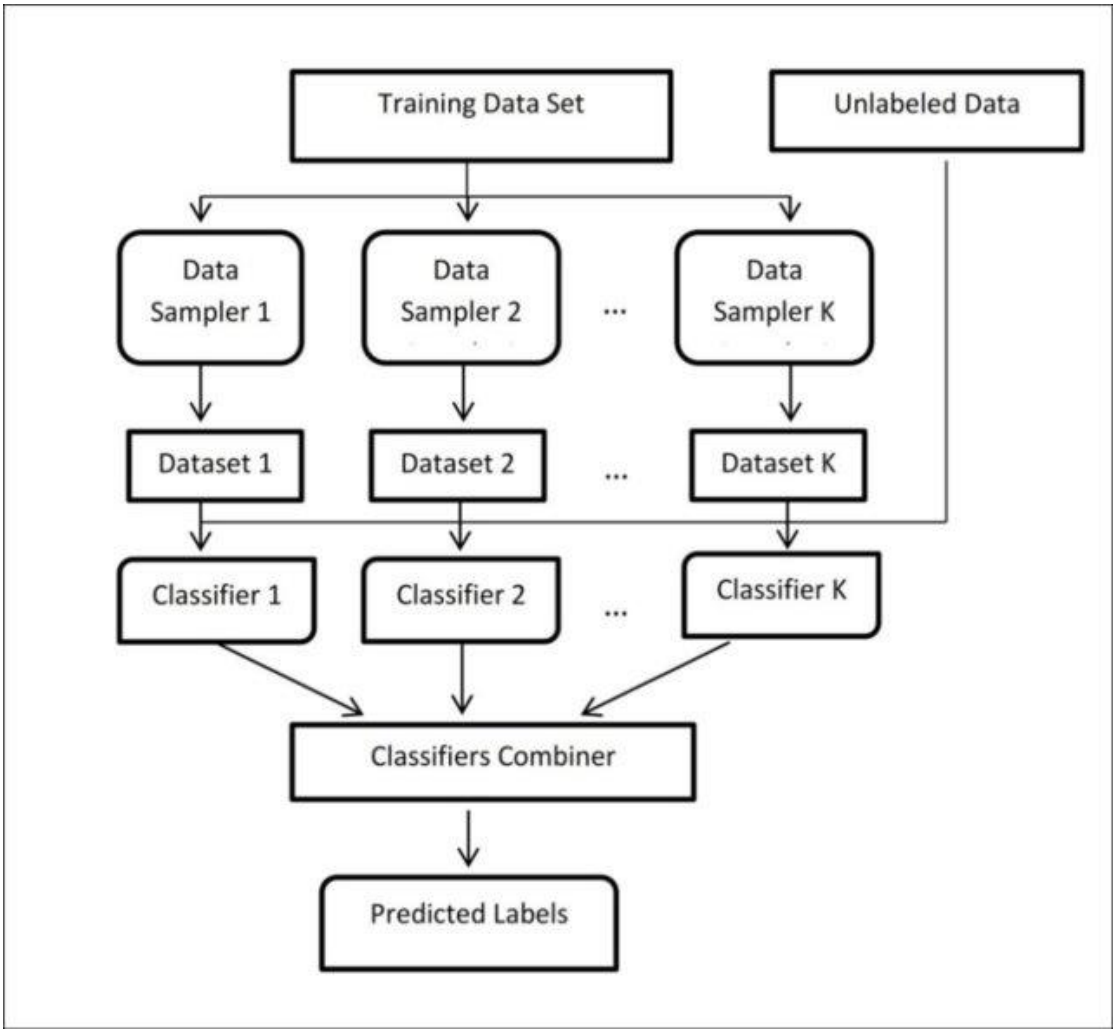
Η βασική ιδέα ανάπτυξης των συνδυαστικών κατηγοριοποιητών είναι ότι οι απλοί αλγόριθμοι κατηγοριοποίησης αναζητούν σε έναν χώρο υποθέσεων για να βρουν μια κατάλληλη υπόθεση που θα κάνει ακριβείς κατηγοριοποιήσεις. Ακόμα κι αν αυτός ο χώρος περιέχει υποθέσεις που είναι πολύ κατάλληλες για ένα συγκεκριμένο πρόβλημα, μπορεί να είναι δύσκολο να γίνει μια καλή πρόβλεψη. Οι συνδυαστικοί κατηγοριοποιητές συνδυάζουν πολλαπλές υποθέσεις για να σχηματίσουν μια καλύτερη υπόθεση.

Η αξιολόγηση της πρόβλεψης ενός συνδυαστικού κατηγοριοποιητή απαιτεί συνήθως περισσότερους υπολογισμούς από την αξιολόγηση της πρόβλεψης ενός απλού κατηγοριοποιητή. Ωστόσο, ένας συνδυαστικός κατηγοριοποιητής, με “σωστή” εκπαίδευση και την ανάλογη αύξηση των υπολογιστικών πόρων θα είναι πιο αποτελεσματικός στη βελτίωση της ακρίβειας από ότι ένας απλός κατηγοριοποιητής. Αξίζει να σημειωθεί ότι γρήγοροι αλγόριθμοι, όπως τα δέντρα αποφάσεων ή ο naive bayes, χρησιμοποιούνται συνήθως σε συνδυαστικούς κατηγοριοποιητές. Ωστόσο, συχνά χρησιμοποιούνται και αλγόριθμοι με μεγαλύτερο υπολογιστικό κόστος.

Εμπειρικά, οι συνδυαστικοί κατηγοριοποιητές τείνουν να αποδίδουν καλύτερα αποτελέσματα όταν υπάρχει σημαντική διαφοροποίηση μεταξύ των μοντέλων. Επομένως, πολλοί συνδυαστικοί κατηγοριοποιητές επιδιώκουν την προώθηση της διαφορετικότητας μεταξύ των μοντέλων που συνδυάζουν. Έτσι, περισσότεροι τυχαίοι αλγόριθμοι (όπως τα τυχαία δέντρα

αποφάσεων) μπορούν να χρησιμοποιηθούν για την παραγωγή ενός ισχυρότερου συνδυαστικού κατηγοριοποιητή. Ωστόσο, η χρήση ποικίλων ισχυρών αλγορίθμων μάθησης έχει αποδειχθεί ότι είναι πιο αποτελεσματική από τη χρήση τεχνικών που επιχειρούν να "χαλαρώσουν" τα μοντέλα προκειμένου να προωθήσουν τη διαφορετικότητα.

Ενώ ο αριθμός των κατηγοριοποιητών που αποτελούν συστατικά ενός συνολικού κατηγοριοποιητή έχει μεγάλο αντίκτυπο στην ακρίβεια της πρόβλεψης, υπάρχει περιορισμένος αριθμός μελετών που αντιμετωπίζουν αυτό το πρόβλημα. Ο καθορισμός εκ των προτέρων του πλήθους των κατηγοριοποιητών και του όγκου και της ταχύτητας των μεγάλων ροών δεδομένων το καθιστά ακόμη πιο κρίσιμο σημείο που χρήζει αντιμετώπισης από την επιστημονική κοινότητα. Κυρίως στατιστικές δοκιμές χρησιμοποιήθηκαν για τον προσδιορισμό του κατάλληλου αριθμού κατηγοριοποιητών. Πιο πρόσφατα, ένα θεωρητικό πλαίσιο πρότεινε ότι υπάρχει ένας ιδανικός αριθμός κατηγοριοποιητών για έναν συνδυαστικό κατηγοριοποιητή, έτσι ώστε η ύπαρξη περισσότερων ή λιγότερων από αυτόν τον αριθμό κατηγοριοποιητών να έχει ως αποτέλεσμα την επιδείνωση της ακρίβειας. Ονομάζεται "ο νόμος για τη μείωση των αποδόσεων στην κατασκευή συνδυαστικών κατηγοριοποιητών". Το θεωρητικό τους πλαίσιο δείχνει ότι η χρήση του ίδιου αριθμού ανεξάρτητων κατηγοριοποιητών με το πλήθος των κλάσεων του συνόλου δεδομένων εκπαίδευσης δίνει την υψηλότερη ακρίβεια κατά την κατηγοριοποίηση [46] [47].



Σχήμα 20. Στάδια λειτουργίας ενός συνδυαστικού (ensemble) κατηγοριοποιητή

Μια μελέτη με βάση πέντε βασικούς ταξινομητές και συγκεκριμένα τους RF-Random Forest, DT-Decision Tree, LR-Logistic Regression, FT-Functional Trees και CART, πραγματοποίησε μια συγκριτική αξιολόγηση διαφορετικών συνδυαστικών κατηγοριοποιητών που βασίζονται σε στρατηγικές όπως το bagging, την ενίσχυση, τη ψηφοφορία με πλειοψηφία και τη στοίβαξη [55]. Οι συνδυαστικοί κατηγοριοποιητές εφαρμόστηκαν σε δύο σύνολα δεδομένων ανίχνευσης εισβολής μεταξύ τομέων, όπως είναι σύνολο δεδομένων εισβολής δικτύου (NSL-KDD) και το σύνολο δεδομένων ασύρματης εισβολής (GPRS). Από το πειραματικό αποτέλεσμα, φάνηκε ότι οι συνδυαστικοί κατηγοριοποιητές είναι μια πολλά υποσχόμενη μέθοδος προς αξιοποίηση στην ανάπτυξη συστημάτων ανίχνευσης εισβολών. Στην ίδια πάντα έρευνα αποδείχθηκε ότι ο αλγόριθμος bagging αποδίδει καλύτερα από αυτόν της ενίσχυσης όσον αφορά τέσσερις δείκτες απόδοσης, και συγκεκριμένα την ακρίβεια (accuracy), την ακρίβεια (precision), την ανάκληση (recall) και το F score (F1). Επιπλέον, ένα ενδιαφέρον αποτέλεσμα που επισημάνθηκε είναι ότι μετά την εφαρμογή του αλγορίθμου bagging, η απόδοση του CART δεν μπορεί να ξεπεράσει

τον κατηγοριοποιητή βάσης όσον αφορά τρεις δείκτες απόδοσης, δηλαδή την ακρίβεια, την ανάκληση, και το F score. Μεταξύ των συνδυαστικών κατηγοριοποιητών , η στοίβαξη είναι ισχυρή μέθοδος για χρήση σε σύστημα εντοπισμού εισβολών (IDS), καθώς εμφανίζει την καλύτερη απόδοση από άποψη ακρίβειας, ακρίβειας και F score.

3.6 Σύνολα δεδομένων που χρησιμοποιούνται στη Βιβλιογραφία

KDD Cup 1999 Data

Το σύνολο δεδομένων KDD Cup χρησιμοποιήθηκε στον διαγωνισμό “The Third International Knowledge Discovery and Data Mining Tools”, ο οποίος πραγματοποιήθηκε σε συνδυασμό με το επιστημονικό συνέδριο “The Fifth International Conference on Knowledge Discovery and Data Mining”. Στόχος του διαγωνισμού ήταν η δημιουργία ενός ανιχνευτή εισβολής δικτύου. Δηλαδή ενός κατηγοριοποιητή ικανού να διακρίνει μεταξύ "κακών" συνδέσεων, που ονομάζονται εισβολές ή επιθέσεις και "καλές" κανονικές συνδέσεις. Το σύνολο δεδομένων περιλαμβάνει μια μεγάλη ποικιλία παρεμβολών που προσομοιώνονται σε περιβάλλον στρατιωτικού δικτύου. Το σύνολο δεδομένων περιλαμβάνει τέσσερις κλάσεις:

- **DOS**: άρνηση παροχής υπηρεσιών, π.χ. syn πλημμύρα?
- **R2L**: μη εξουσιοδοτημένη πρόσβαση από απομακρυσμένο μηχάνημα, π.χ. μαντέψτε τον κωδικό πρόσβασης.
- **U2R**: μη εξουσιοδοτημένη πρόσβαση σε τοπικά δικαιώματα υπερχρήστη (root), π.χ., διάφορες επιθέσεις "υπερχείλισης buffer".
- **Probing**: παρακολούθηση και άλλη ανίχνευση, π.χ. σάρωση θυρών.

Το μέγεθος των ακατέργαστων δεδομένων εκπαίδευσης είναι περίπου τέσσερα gigabytes. Πρόκειται για συμπιεσμένα δυαδικά δεδομένων απόρριψης TCP που προέκυψαν έπειτα από την παρακολούθηση της κυκλοφορίας του δικτύου για ένα διάστημα επτά εβδομάδων. Από την παρακολούθηση αυτή δημιουργήθηκαν και στην συνέχεια επεξεργάστηκαν περίπου πέντε εκατομμύρια αρχεία συνδέσεων. Αντιστοίχως, οι δύο εβδομάδες δεδομένων δοκιμής απέδωσαν περίπου δύο εκατομμύρια αρχεία συνδέσεων. Όταν γίνεται λόγος για σύνδεση, ουσιαστικά γίνεται αναφορά σε μια ακολουθία πακέτων TCP που σαν κύριο χαρακτηριστικό τους έχουν το γεγονός ότι ξεκινούν και τελειώνουν σε καθορισμένους χρόνους, μεταξύ των οποίων τα δεδομένα ρέουν από και προς μια διεύθυνση IP προέλευσης σε μια διεύθυνση IP προορισμού υπό κάποιο συγκεκριμένο και καλά καθορισμένο πρωτόκολλο. Κάθε σύνδεση

δύναται να χαρακτηριστεί είτε ως κανονική είτε ως επίθεση, με έναν ακριβώς συγκεκριμένο τύπο επίθεσης. Το μέγεθος κάθε εγγραφής σύνδεσης είναι περίπου 100 bytes.

Darpa Intrusion Detection Evaluation Dataset 1998, 1999 και 2000

Το DARPA είναι ένα σύνολο δεδομένων που αποτελείται από επικοινωνίες μεταξύ IP προέλευσης και IP προορισμού. Χρησιμοποιείται προκειμένου να δοκιμαστούν και να αξιολογηθούν τα συστήματα εντοπισμού εισβολών σε κατάσταση εκτός σύνδεσης. Τα συστήματα επεξεργάζονται αυτά τα δεδομένα μαζικά (batch mode) και προσπαθούν να διαχωρίσουν τις απόπειρες επίθεσης από την κανονική δραστηριότητα. Για την δημιουργία του συνόλου δεδομένων DARPA 1999 μεταφέρθηκαν στις εγκαταστάσεις του ερευνητικού εργαστηρίου της πολεμικής αεροπορίας των Η.Π.Α. συστήματα εντοπισμού επιθέσεων, τα οποία επιχείρησαν να εντοπίσουν εισβολές σε πραγματικό χρόνο κατά την διάρκεια κανονικής λειτουργίας [50]. Για την δημιουργία του συνόλου δεδομένων 1999 DARPA χρησιμοποιήθηκαν δεδομένα εκπαίδευσης που συλλέχθηκαν έπειτα από τρεις εβδομάδες λειτουργίας. Τα δεδομένα της πρώτης και της τρίτης εβδομάδες δεν περιέχουν καμία επίθεση. Ο σκοπός ύπαρξής τους είναι για να βελτιώσουν την εκπαίδευση των συστημάτων εντοπισμού ανωμαλιών στην κίνηση ενός δικτύου. Τα δεδομένα εκπαίδευσης της δεύτερης εβδομάδας περιέχουν ένα επιλεγμένο υποσύνολο των επιθέσεων από την αξιολόγηση του 1998, μαζί με αρκετές νέες επιθέσεις. Η παρουσία αυτών των επιθέσεων έχει σαν κύριο στόχο να παρέχει παραδείγματα αναφοράς των επιθέσεων που εντοπίζονται.

Στο σύνολο δεδομένων υπάρχουν τα παρακάτω αρχεία για κάθε μέρα ξεχωριστά:

- εξωτερικά δεδομένα ανίχνευσης (outside sniffing data)
- εσωτερικά δεδομένα ανίχνευσης (inside sniffing data)
- δεδομένα ελέγχου BSM (BSM audit data)
- δεδομένα ελέγχου NT (NT audit data)
- μεγάλες λίστες δέντρων καταλόγου (long listings of directory trees)
- αποθήκες επιλεγμένων καταλόγων (dumps of selected directories)
- μια αναφορά με πληροφορίες για το σύστημα αρχείων (a report of file system inode information)

UNM Dataset

Το σύνολο δεδομένων UNM αποτελείται από κλήσεις συστήματος που εκτελούνται από ενεργές διεργασίες. Αυτές περιλαμβάνουν διάφορα είδη προγραμμάτων. Τα προγράμματα ποικίλλουν σε μεγάλο βαθμό ως προς το μέγεθος και την πολυπλοκότητά τους και πρόκειται για διαφορετικά είδη εισβολών (υπερχειλίσεις buffer, συμβολικές συμβάσεις συνδέσμων και προγράμματα Trojan). Το σύνολο δεδομένων περιλαμβάνει μόνο εκείνα τα προγράμματα που λειτουργούν με προνόμια, επειδή η κακή χρήση αυτών των προγραμμάτων έχει τις μεγαλύτερες πιθανότητες να βλάψει το σύστημα.

NSL-KDD dataset

Το σύνολο δεδομένων NSL-KDD, σε αντίθεση με το KDDCup99, δεν περιλαμβάνει περιττές περιπτώσεις που οδηγούν τους κατηγοριοποιητές στην παραγωγή προκατειλημμένων αποτελεσμάτων. Το σύνολο δεδομένων διαθέτει 41 χαρακτηριστικά και μία ιδιότητα ετικέτας κλάσης. Το πλήρες σετ εκπαίδευσης NSL-KDD περιέχει 125927 στιγμιότυπα και χωρίζεται σε δύο κλάσεις, την κλάση επίθεσης με 58630 στιγμιότυπα και την κανονική κλάση με 67343 στιγμιότυπα [55].

GPRS Dataset

Το σύνολο δεδομένων GPRS (Grupo de Pesquisa em Redes e SeguranÃ§a) προτείνεται δεδομένου ότι ο αριθμός των διαθέσιμων δεδομένων που αφορούν τα ασύρματα δίκτυα είναι αρκετά περιορισμένος [56]. Αναπτύσσεται με βάση την ανίχνευση εισβολής στο περιβάλλον IEEE 802.11. Αποτελείται από δύο ξεχωριστές τοπολογίες δικτύου, π.χ. WPE/WPA και WPA2. Το σύνολο δεδομένων WPE/WPA ή WPA2 διαθέτει τα ίδια 15 χαρακτηριστικά και 1 ετικέτα κλάσης. Το πλήρες σετ εκπαίδευσης WPE/WPA αποτελείται από 2 κλάσεις, δηλαδή κανονική κλάση (6000 περιπτώσεις) και κλάση επίθεσης (3600 περιπτώσεις). Το πλήρες σετ εκπαίδευσης WPA2 περιέχει 4500 περιπτώσεις κανονικής κλάσης και 3000 στιγμιότυπα κλάσης επίθεσης.

4

Στατιστική ανάλυση βιβλιογραφίας

Στο κεφάλαιο αυτό θα παρουσιαστεί η στατιστική ανάλυση της σχετικής με την χρησιμοποίηση αλγορίθμων Μηχανικής Μάθησης σε συστήματα εντοπισμού εισβολών, βιβλιογραφίας.

4.1 Συνολική παρουσίαση της σχετικής βιβλιογραφίας

Στον πίνακα 3 παρουσιάζεται το σύνολο των άρθρων και δημοσιεύσεων που εντοπίστηκαν κατά την διάρκεια εκπόνησης της παρούσας διπλωματικής. Πρόκειται για εργασίες που αναφέρονται στην χρησιμοποίηση αλγορίθμων Μηχανικής Μάθησης σε διάφορες υλοποιήσεις συστημάτων εντοπισμού εισβολών. Στην πρώτη στήλη βρίσκεται ο αριθμός της κάθε σχετικής αναφοράς στη βιβλιογραφία της παρούσας εργασίας, στη δεύτερη το έτος κατά το οποίο αυτή δημοσιεύθηκε, στην τρίτη το είδος του αλγορίθμου στον οποίο γίνεται αναφορά στο paper και στην τέταρτη ο αριθμός των αναφορών στην εκάστοτε εργασία, όπως αυτές προέκυψαν από έρευνα στο Google Scholar με ημερομηνία ανάκτησης τον Σεπτέμβριο του 2021. Αξίζει να σημειωθεί πως στις περιπτώσεις που μια εργασία αναφέρεται σε παραπάνω από ένα είδος αλγόριθμου, τότε στην τρίτη στήλη αναφέρονται και οι δύο και θεωρούμε πως εκείνες αναφορές ανήκουν σε μια διαφορετική κατηγορία σε σχέση με τις αναφορές που αναφέρονται στους ίδιους αλγόριθμους αλλά μεμονωμένα.

Αναφορά	Έτος	Είδος αλγορίθμου στον οποίο αναφέρεται το paper	Αριθμός Αναφορών Google Scholar	Α/Α Αναφοράς	Έτος	Είδος αλγορίθμου στον οποίο αναφέρεται το paper	Αριθμός Αναφορών Google Scholar
57	2000	Single	175	138	2010	Single	47
58	2004	Single	111	139	2012	Single	404
59	2006	Single	93	140	2012	Single	240

60	2005	Single/ Hybrid	600	141	2010	Single	15
61	2002	Single/ Hybrid	1383	142	2012	Single	56
62	2004	Single	311	143	2012	Single	23
63	2003	Single	230	144	2011	Single	214
64	2007	Single	167	145	2012	Single	27
8	2002	Single	754	146	2009	Single	36
65	2004	Single	136	147	2012	Single	37
72	2004	Single	101	148	2011	Single	126
76	2005	Single	79	149	2012	Single	17
69	2005	Single	425	150	2008	Single	21
74	2001	Single	1360	151	2011	Single	106
67	2004	Single	132	152	2010	Single	5
87	2003	Single	665	153	2008	Single	165
79	2004	Single	23	154	2009	Single	73
68	2004	Single	1207	155	2009	Single	6
78	2006	Single	125	156	2009	Hybrid	85
84	2004	Single	87	157	2012	Hybrid	102
80	2006	Single	14	158	2012	Hybrid	46
70	2005	Single	115	159	2013	Hybrid	75
66	2007	Hybrid/ Ensemble	132	160	2012	Hybrid	36
77	2000	Hybrid	107	161	2010	Hybrid	2
86	2004	Hybrid	160	162	2009	Hybrid	137
82	2007	Hybrid	153	163	2009	Hybrid	14
85	2002	Hybrid	149	164	2012	Hybrid	39
92	2003	Hybrid / Ensemble	77	165	2011	Hybrid	73
89	2006	Hybrid	222	166	2011	Hybrid	27
73	2003	Hybrid	151	167	2010	Hybrid	24
75	2007	Hybrid	186	168	2012	Hybrid	230
71	2007	Hybrid	487	169	2010	Hybrid	5
88	1998	Hybrid	2051	170	2009	Hybrid	29
83	2000	Hybrid	1212	171	2010	Hybrid	29
91	2006	Hybrid	23	172	2012	Hybrid	124
97	2004	Hybrid	169	173	2011	Hybrid	20
94	2007	Hybrid	189	174	2012	Hybrid	196
81	2000	Hybrid	279	175	2010	Hybrid	37
95	2004	Hybrid	275	176	2008	Hybrid	152
90	2007	Hybrid	137	177	2012	Hybrid	11
99	2007	Hybrid / Ensemble	547	178	2012	Hybrid	25
100	2006	Hybrid	35	179	2011	Hybrid	134
96	2007	Hybrid	442	180	2010	Hybrid	103
93	2005	Hybrid	342	181	2009	Hybrid	112

102	2007	Hybrid	310	182	2010	Hybrid	294
98	2007	Hybrid	268	183	2012	Hybrid	8
104	2005	Hybrid	45	184	2010	Hybrid	560
103	2005	Hybrid	246	185	2008	Hybrid	21
101	2004	Hybrid	72	186	2010	Hybrid	12
105	2009	Ensemble	55	187	2008	Hybrid	187
106	2006	Ensemble	263	188	2010	Hybrid	7
109	2003	Ensemble	78	189	2009	Hybrid	6
108	2005	Ensemble	195	190	2011	Ensemble	117
107	2005	Ensemble	495	191	2011	Ensemble	15
111	2011	Single	550	192	2012	Ensemble	70
112	2011	Single	20	193	2009	Ensemble	103
113	2012	Single	139	194	2011	Ensemble	151
114	2011	Single	376	195	2012	Ensemble	330
115	2010	Single	39	196	2012	Single	73
116	2011	Single	11	197	2012	Single	13
117	2011	Single	207	198	2017	Single	50
118	2012	Single	90	199	2018	Hybrid	115
119	2009	Single	80	200	2017	Single	29
120	2012	Single	101	201	2017	Single	34
121	2009	Single	16	202	2016	Single	21
122	2008	Single	61	203	2017	Single	168
123	2010	Single	141	204	2017	Single	15
124	2010	Single	96	205	2017	Single	50
125	2009	Single	27	206	2018	Single / Hybrid	0
126	2009	Single	20	207	2018	Single/ Ensemble	19
127	2008	Single	28	208	2018	Single	26
128	2011	Single	3	209	2019	Single	5
129	2010	Single	22	210	2015	Single/ Ensemble	43
130	2009	Single	7	211	2015	Single	449
131	2012	Single	32	212	2020	Hybrid	108
132	2012	Single	319	213	2019	Hybrid	16
133	2011	Single	108	214	2020	Ensemble	2
134	2012	Single	338	215	2020	Hybrid	4
135	2010	Single	19	216	2020	Hybrid	2
136	2011	Single	21	217	2020	Ensemble	40
137	2012	Single	41	218	2016	Ensemble	2

Πίνακας 3. Η σχετική βιβλιογραφία με Α/Α αναφοράς, έτος δημοσίευσης, είδος αλγορίθμου Μηχανικής Μάθησης στον οποίο αναφέρεται και αναφορές στο Google Scholar

4.2 Αναφορές στο Google Scholar για την βιβλιογραφία συνολικά

Στον Πίνακα 4 παρουσιάζονται τα χαρακτηριστικά της βιβλιογραφίας που αφορούν τις αναφορές οι οποίες γίνονται σε αυτές από άλλες εργασίες. Πρόκειται για 162 σχετικές εργασίας με Μ.Ο. 169,2 αναφορές στο Google Scholar. Το συνολικό άθροισμα των αναφορών είναι ίσο με 27407. Ο μέγιστος αριθμός αναφορών στο Google Scholar είναι 2051 αναφορές και πρόκειται για την εργασία με Α/Α το 88, ενώ υπάρχει και μια εργασία που εμφανίζει ελάχιστο αριθμό αναφορών ίσο με το μηδέν, αυτή με Α/Α το 206. Η τυπική απόκλιση των αναφορών είναι ίση με 275.

Σύνολο Βιβλιογραφικών Αναφορών	Μ.Ο. αναφορών στο Google Scholar	Άθροισμα αναφορών στο Google Scholar	Μέγιστος αριθμός αναφορών στο Google Scholar	Ελάχιστος αριθμός αναφορών στο Google Scholar	Τυπική απόκλιση (STDDEV)
162	169,2	27407	2051	0	275

Πίνακας 4. Συνολικά χαρακτηριστικά της βιβλιογραφίας σχετικά με τις αναφορές στο Google Scholar

4.3 Αναφορές στο Google Scholar για τις βιβλιογραφικές αναφορές ανά είδος αλγορίθμου

Στον πίνακα 5 παρουσιάζουμε τις βιβλιογραφικές αναφορές με βάση το είδος του αλγόριθμου στον οποίο αναφέρονται και τα χαρακτηριστικά που αφορούν τις αναφορές στο Google Scholar. Έτσι παρατηρείται πως για το σύνολο των ετών που δημοσιεύθηκαν οι εργασίες στις οποίες αναφερόμαστε (1998-2020), περισσότερες ήταν οι εργασίες που αναφερόντουσαν στους απλούς (single) (77), ακολούθησαν οι υβριδικοί (hybrid) με 63 σχετικές εργασίες, ενώ οι συνδυαστικοί (ensemble) απασχόλησαν αποκλειστικά 14 δημοσιεύσεις. Έχουμε και 8 εργασίες που αναφέρονται σε περισσότερα από ένα είδη αλγορίθμων ταυτόχρονα. Από τις εργασίες που αναφέρονται σε ένα μόνο είδος αλγορίθμου πιο δημοφιλές είδος με βάση τον μέσο όρο αναφορών σε αυτό είναι οι υβριδικοί αλγόριθμοι με 173,3, έπονται οι απλοί με 152,9 και ακολουθούν οι συνδυαστικοί με 136,9. Η δημοσίευση με τον μεγαλύτερο αριθμό αναφορών στην οποία έγινε μνεία και παραπάνω αφορά υβριδικό αλγόριθμο.

Είδος αλγορίθμου	Σύνολο Βιβλιογραφικών Αναφορών	Μ.Ο. αναφορών στο Google Scholar	Άθροισμα αναφορών στο Google Scholar	Μέγιστος αριθμός αναφορών στο Google Scholar	Ελάχιστος αριθμός αναφορών στο Google Scholar	Τυπική απόκλιση (STDDEV)
Single	77	152,9	11773	1360	3	238,9
Hybrid	63	173,3	10917	2051	2	300,4
Ensemble	14	136,9	1916	495	2	136,8
Single /Hybrid	3	661	1983	1383	0	566,3
Single/ Ensemble	2	31	62	43	19	12
Hybrid/ Ensemble	3	252	756	547	77	209,8

Πίνακας 5. Αναφορές στο Google Scholar για τις βιβλιογραφικές αναφορές ανά είδος αλγορίθμου

4.4 Αναφορές στο Google Scholar για τις βιβλιογραφικές αναφορές ανά έτος

Στον πίνακα 6 εμφανίζονται στοιχεία που αφορούν τις αναφορές στις δημοσιεύσεις της βιβλιογραφίας όπως αυτές εμφανίζονται στο Google Scholar, συνολικά για όλα τα είδη των αλγόριθμων, ταξινομημένες ανά έτος. Παρατηρούμε πως ο μεγαλύτερος Μ.Ο. αναφορών εμφανίζεται για το έτος 2001 με 1360 (αν και αφορά μία μόνο εργασία) και ακολουθεί το 2000 με 443,3 αναφορές (σε τέσσερις εργασίες). Οι δύο εργασίες που περιλαμβάνονται στην βιβλιογραφία και δημοσιεύθηκαν το 2019 παρουσιάζουν τον μικρότερο μέσο όρο αναφορών στο Google Scholar με 10,5 αναμενόμενο μια και πρόκειται για καινούργιες που δεν πρόλαβαν να τις μελετήσουν άλλοι ερευνητές.

Έτος	Σύνολο Βιβλιογραφικών Αναφορών	Μ.Ο. αναφορών στο Google Scholar	Άθροισμα αναφορών στο Google Scholar	Μέγιστος αριθμός αναφορών στο Google Scholar	Ελάχιστος αριθμός αναφορών στο Google Scholar	Τυπική απόκλιση (STDDEV)
1998	1	2051	2051	2051	2051	0
2000	4	443,3	1773	1212	107	448
2001	1	1360	1360	1360	1360	0
2002	3	762	2286	1383	149	503,8
2003	5	240,2	1201	665	77	219,8

2004	12	232	2784	1207	23	304
2005	9	282,4	2542	600	45	184
2006	7	110,7	775	263	14	91,7
2007	11	274,4	3018	547	132	144,4
2008	7	90,7	635	187	21	68,7
2009	16	50,4	806	137	6	41,2
2010	18	80,9	1457	560	2	135,3
2011	18	126,6	2279	550	3	138,4
2012	28	113,1	3167	404	8	114,5
2013	1	75	75	75	75	0
2015	2	246	492	449	43	203
2016	2	11,5	23	21	2	9,5
2017	6	57,7	346	168	15	50,8
2018	4	40	160	115	0	44,3
2019	2	10,5	21	16	5	5,5
2020	5	31,2	156	108	2	41

Πίνακας 6. Αναφορές στο Google Scholar για τις βιβλιογραφικές αναφορές ανά έτος

4.5 Αναφορές στο Google Scholar για τις βιβλιογραφικές αναφορές ανά έτος και είδος αλγορίθμου

Στον πίνακα 7 εμφανίζονται στοιχεία που αφορούν τις αναφορές στις δημοσιεύσεις της βιβλιογραφίας όπως αυτές εμφανίζονται στο Google Scholar, συνολικά για όλα τα είδη των αλγόριθμων, ταξινομημένες ανά έτος και είδος. Προκειμένου όμως να έχουμε μια πιο εύκολη, εποπτική ματιά κρίθηκε χρήσιμο να κατασκευαστούν δύο διαγράμματα. Στο διάγραμμα του σχήματος 21 παρουσιάζεται με τρεις διαφορετικές γραμμές το πλήθος των δημοσιεύσεων στο Google Scholar ανά έτος και τύπο αλγορίθμου (απλός, υβριδικός ή συνδυαστικός). Στο σημείο αυτό και προκειμένου να εξηγήσουμε το σχήμα καλύτερα αξίζει να αναφέρουμε πως για την έρευνά μας στηριχτήκαμε στην [221] που παρουσίαζε συνοπτικά τις σχετικές εργασίες μέχρι και 2007. Στην συνέχεια έγινε προσπάθεια να εμπλουτιστεί η λίστα των αναφορών με περισσότερες και πιο καινούργιες εργασίες, γεγονός που φαίνεται να λειτούργησε και να μας δίνει μια καλή εικόνα για τις εξελίξεις στο συγκεκριμένο πεδίο μέχρι και το 2013. Μετά οι εργασίες που προστέθηκαν ήταν σχετικά λίγες με αποτέλεσμα στην γραφική απεικόνιση να φαίνεται σαν να μειώθηκε σημαντικά η έρευνα στο συγκεκριμένο αντικείμενο, κάτι που πιθανότατα δεν ισχύει. Μέχρι και το 2013, που φαίνεται να είναι και αντιπροσωπευτικότερα τα στοιχεία που έχουμε, εμφανίζεται μια συνεχής αύξηση στον αριθμό των δημοσιεύσεων, γεγονός που υποδηλώνει το ολοένα αυξανόμενο ενδιαφέρον της επιστημονικής κοινότητας για το πεδία αυτό. Σχετικά με το πλήθος των αναφορών όπως αυτές παρουσιάζονται στο διάγραμμα

του σχήματος 22, έχουμε να εντοπίσουμε την σημαντική πτώση που εμφάνισε ο μέσος όρων αυτών που σχετίζονται με τους απλούς αλγόριθμους, ενώ οι μέσοι όροι για τους υβριδικούς και τους συνδυαστικούς φαίνονται σχετικά σταθεροί.

Έτος	Είδος αλγορίθμου	Σύνολο Βιβλιογραφικών Αναφορών	Μ.Ο. αναφορών στο Google Scholar	Άθροισμα αναφορών στο Google Scholar	Μέγιστος αριθμός αναφορών στο Google Scholar	Ελάχιστος αριθμός αναφορών στο Google Scholar	Τυπική απόκλιση (STDDEV)
1998	Hybrid	1	2051	2051	2051	2051	0
2000	Hybrid	3	532,7	1598	1212	107	485,5
2000	Single	1	175	175	175	175	0
2001	Single	1	1360	1360	1360	1360	0
2002	Hybrid	1	149	149	149	149	0
2002	Single	1	754	754	754	754	0
2002	Single/ Hybrid	1	1383	1383	1383	1383	0
2003	Ensemble	1	78	78	78	78	0
2003	Hybrid	1	151	151	151	151	0
2003	Hybrid/ Ensemble	1	77	77	77	77	0
2003	Single	2	447,5	895	665	230	217,5
2004	Hybrid	4	169	676	275	72	72
2004	Single	8	263,5	2108	1207	23	364,8
2005	Ensemble	2	345	690	495	195	150
2005	Hybrid	3	211	633	342	45	123,7
2005	Single	3	206,3	619	425	79	155,3
2005	Single/ Hybrid	1	600	600	600	600	0
2006	Ensemble	1	263	263	263	263	0
2006	Hybrid	3	93,3	280	222	23	91,1
2006	Single	3	77,3	232	125	14	46,6
2007	Hybrid	8	271,5	2172	487	137	124,1
2007	Hybrid/ Ensemble	2	339,5	679	547	132	207,5
2007	Single	1	167	167	167	167	0
2008	Hybrid	3	120	360	187	21	71,4
2008	Single	4	68,8	275	165	21	57,6
2009	Ensemble	2	79	158	103	55	24
2009	Hybrid	6	63,8	383	137	6	50,3
2009	Single	8	33,1	265	80	6	26,7
2010	Hybrid	10	107,3	1073	560	2	172,8
2010	Single	8	48	384	141	5	44

2011	Ensemble	3	94,3	283	151	15	57,8
2011	Hybrid	4	63,5	254	134	20	45,5
2011	Single	11	158,4	1742	550	3	164,2
2012	Ensemble	2	200	400	330	70	130
2012	Hybrid	10	81,7	817	230	8	74,9
2012	Single	16	121,9	1950	404	13	125,3
2013	Hybrid	1	75	75	75	75	0
2015	Single	1	449	449	449	449	0
2015	Single/ Ensemble	1	43	43	43	43	0
2016	Ensemble	1	2	2	2	2	0
2016	Single	1	21	21	21	21	0
2017	Single	6	57,7	346	168	15	50,8
2018	Hybrid	1	115	115	115	115	0
2018	Single	1	26	26	26	26	0
2018	Single/ Ensemble	1	19	19	19	19	0
2018	Single/ Hybrid	1	0	0	0	0	0
2019	Hybrid	1	16	16	16	16	0
2019	Single	1	5	5	5	5	0
2020	Ensemble	2	21	42	40	2	19
2020	Hybrid	3	38	114	108	2	49,5

Πίνακας 7. Αναφορές στο Google Scholar για τις βιβλιογραφικές αναφορές ανά έτος και είδος αλγορίθμου

Σχήμα 21. Πλήθος δημοσιεύσεων στο Google Scholar ανά έτος και τύπο αλγορίθμου

Σχήμα 22. Μέσος όρος αναφορών στο Google Scholar ανά έτος και τύπο αλγορίθμου

5

Επίλογος

5.1 Σύνοψη και συμπεράσματα

Οι αλγόριθμοι Μηχανικής Μάθησης και Εξόρυξης Γνώσης έχουν επηρεάσει σε μεγάλο βαθμό την καθημερινότητα του ανθρώπου. Πολλές ερευνητικές προσπάθειες έχουν πραγματοποιηθεί στο παρελθόν και εξακολουθούν να πραγματοποιούνται σε αυτα τα πεδία της επιστήμης των υπολογιστών. Ως αποτέλεσμα, ένα πολύ μεγάλο ποσοστό ερευνητικών εργασιών της επιστήμης των υπολογιστών που δημοσιεύονται σε επιστημονικά συνέδρια και περιοδικά αφορούν τα συγκεκριμένα πεδία.

Απο τα αρχικά στάδια ανάπτυξης των αλγορίθμων μηχανικής μάθησης, οι αλγόριθμοι αυτοί αρχίσαν να εφαρμόζονται με επιτυχία και για την αναγνώριση εισβολών και παραβιάσεων σε πληροφοριακά και επικοινωνιακά συστήματα. Η άμεση αναγνώριση εισβολών είναι ζωτικής σημασίας για οποιονδήποτε οργανισμό. Έτσι, το πεδίο της ανίχνευσης εισβολών με αλγορίθμους μηχανικής μάθησης έχει προσελκύσει το ενδιαφέρον τόσο την ερευνητική κοινότητα της τεχνητής νοημοσύνης όσο και την ερευνητική κοινότητα της ασφάλειας πληροφοριακών και επικοινωνιακών συστημάτων και έτσι, πολλές ερευνητικές εργασίες δημοσιεύονται κάθε χρόνο που να αφορούν το συγκεκριμένο αντικείμενο. Συνεπώς, πολλοί αλγόριθμοι είναι διαθέσιμοι στη βιβλιογραφία που έχουν ως στόχο την αποτελεσματική κατηγοριοποίηση σε συστήματα που αφορούν την ανίχνευση εισβολών. Στις ημέρες μας, η ανίχνευση εισβολών με αλγορίθμους μηχανικής μάθησης αποτελεί ένα ενεργό και δημοφιλές πεδίο έρευνας της επιστήμης των υπολογιστών.

Η παρούσα διπλωματική εργασία πραγματοποίησε μια εκτεταμένη ανασκόπηση της βιβλιογραφίας που αφορά τους αλγορίθμους και τις τεχνικές εξόρυξης γνώσης και μηχανικής μάθησης για την αποτελεσματική αναγνώριση εισβολών και κακόβουλων ενεργειών σε πληροφοριακά και επικοινωνιακά συστήματα καθώς και της επισκόπησης των αντίστοιχων συστημάτων ανίχνευσης εντοπισμού εισβολών. Κατηγοριοποίησε τους αλγορίθμους

ανίχνευσης εισβολών σε τρεις βασικές κατηγορίες, οι οποίες είναι: απλοί κατηγοριοποιητές, υβριδικοί κατηγοριοποιητές και συνδυαστικοί κατηγοριοποιητές. Επιπρόσθετα, κατέγραψε τις εργασίες που δημοσιεύτηκαν σε επιστημονικά συνέδρια και περιοδικά τις τελευταίες δεκαετίες, τις κατηγοριοποίησε, παρουσίασε τον αντίκτυπο που είχε η κάθε μια στην επιστημονική κοινότητα ενώ παράλληλα παρουσίασε και ενδιαφέροντα στατιστικά.

Οι συνδυαστικοί κατηγοριοποιητές φαίνεται ότι είναι οι πιο δημοφιλείς προσεγγίσεις τα τελευταία χρόνια αφού φαίνεται να επιτυγχάνουν καλύτερα αποτελέσματα. Ωστόσο, η ερευνητική κοινότητα αναγνωρίζει το βασικό μειονέκτημα των συγκεκριμένων προσεγγίσεων το οποίο είναι το μεγάλο υπολογιστικό κόστος που απαιτείται για την εκπαίδευση των συγκεκριμένων κατηγοριοποιητών. Παρόλα αυτά, οι συγκεκριμένοι κατηγοριοποιητές φαίνεται ότι είναι πιο αποτελεσματικοί στην ανίχνευση εισβολών σε πληροφοριακά και επικοινωνιακά συστήματα τόσο από τους υβριδικούς όσο και από τους απλούς κατηγοριοποιητές αφού αυξάνουν την ανάκληση (recall) ενώ κρατούν την ορθότητα (precision) σε υψηλά επίπεδα. Έτσι, την τελευταία δεκαετία, οι συνδυαστικοί κατηγοριοποιητές έχουν κυριαρχήσει.

5.2 Μελλοντικές επεκτάσεις

Στη συνέχεια θα γίνει αναφορά σε κάποιες ιδέες και δυνατότητες που φαίνεται να υπάρχουν για την επέκταση της διπλωματικής. Αρχικά φαίνεται πως θα είχε ενδιαφέρον να μελετηθεί περαιτέρω αυτή η αυξητική τάση που εμφανίστηκε αναφορικά με το ερευνητικό ενδιαφέρον σχετικά με τη χρησιμοποίηση συνδυαστικών (ensemble) αλγορίθμων στα συστήματα ανίχνευσης εισβολών (IDS). Ουσιαστικά τα ερωτήματα που μένουν να απαντηθούν είναι αν θα παραμείνουν οι αλγόριθμοι αυτοί στο επίκεντρο του ενδιαφέροντος των ερευνητών, αν θα επανακάμψει κάποιο από τα άλλα δύο υπάρχοντα είδη, απλό (single) και υβριδικό (hybrid), ή αν θα εμφανιστεί κάποιος άλλος καινούργιος αλγόριθμος που θα κερδίσει το ενδιαφέρον των ερευνητών. Επιπλέον τα ζητήματα των ροών δεδομένων (data streams) με την παραγωγή μεγάλου όγκων δεδομένων σε πολύ μικρό χρονικό διάστημα και της εννοιολογικής απόκλισης (concept drift) φαίνεται πως ήδη αποτελούν μια νέα τάση που απασχολεί τους ερευνητές [219,220]. Είναι ιδιαίτερα σημαντικό να ερευνηθεί το αν και κυρίως πόσο αποτελεσματικά μπορούν οι υπάρχοντες αλγόριθμοι να λειτουργήσουν σε συστήματα ανίχνευσης εισβολών έχοντας σαν είσοδο τέτοιες ροές.

6

Βιβλιογραφία

1. Michalski, R. S., Bratko, I., & Kubat, M. (1998). Machine learning and data mining methods and applications. Chichester, New York, Weinheim, Brisbane, Toronto, Singapore: Wiley.

2. Theodoridis, S., & Koutroumbas, K. (2006). Pattern recognition. Amsterdam, Boston, Heidelberg, London, New York, Oxford, Paris, San Diego, San Francisco, Singapore, Sydney, Tokyo: Academic Press.

3. Vapnik, V. (1998). Statistical learning theory. New York: John Wiley.

4. S. J. Horng, M. Y. Su, Y. H. Chen, T. W. Kao, R. J. Chen, J. L. Lai and C. D. Kara, "A novel intrusion detection system based on hierarchical clustering and support vector machines," Expert Systems with Applications, 38(1): 306-313. 2011.

5. Bishop, C. M. (1995). Neural networks for pattern recognition. England: Oxford University

6. Manocha, S., & Girolami, M. A. (2007). An empirical analysis of the probabilistic K-nearest neighbour classifier. Pattern Recognition Letters, 28, 1818–1824.

7. Mitchell, T. (1997). Machine learning. New york: McGraw Hill.

8. Liao, Yihua & Vemuri, Rao. (2002). Use of K-Nearest Neighbor classifier for intrusion detection. Computers & Security. 21. 439-448. 10.1016/S0167-4048(02)00514-X.

9. Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, P. J. (1984). Classification and regressing trees. California: Wadsworth International Group.

10. J. Markey, Using Decision Tree Analysis for Intrusion Detection: A How-To Guide, SANS Institute InfoSec Reading Room, June, 2011

11. Rai, Kajal & Devi, Mandalika & Guleria, Ajay. (2016). Decision Tree Based Algorithm for Intrusion Detection. International Journal of Advanced Networking and Applications. 7. 2828-2834.

12. Haykin, S. (1999). Neural networks: A comprehensive foundation (2nd ed.). New Jersey: Prentice Hall.

13. Alex Shenfield, David Day, Aladdin Ayesh. Intelligent intrusion detection systems using artificial neural networks. ICT Express, Volume 4, Issue 2 (2018) Pages 95-99.

14. Koza, J. R. (1992). Genetic programming: On the programming of computers by means of natural selection. Massachusetts: MIT.

15. A. Sung, S. Mukkamala, “Identifying important features for intrusion detection using support vector machines and neural networks” in Symposium on Applications and the Internet, pp. 209–216. 2003.

16. J. P. Planquart, “Application of Neural Networks to Intrusion Detection”, SANS Institute Reading Room.

17. R. G. Bace, “Intrusion Detection”, Macmillan Technical Publishing. 2000.

18. Snort (software); http://en.wikipedia.org/wiki/Snort_%28software%29

19. InfoWorld, The greatest open source software of all time, 2009;

20. http://www.infoworld.com/d/open-source/greatest-open-source-software-all-time-776?source=fssr

21. Sectools.Org: 2006 Results; http://sectools.org/tools2006.html

22. SecTools.Org: Top 125 Network Security Tools; http://sectools.org/tag/ids/

23. Suricata (software); http://en.wikipedia.org/wiki/Suricata_(software)

24. The Bro Network Security Monitor; http://bro-ids.org/

25. R. Graham, “FAQ: Network Intrusion Detection Systems”. March 21, 2000.

26. Mohammad Sazzadul Hoque, Md. Abdul Mukit and Md. Abu Naser Bikas, “AN IMPLEMENTATION OF INTRUSION DETECTION SYSTEM USING GENETIC ALGORITHM”, International Journal of Network Security & Its Applications (IJNSA), Vol.4, No.2, March 2012 109-120

27. Kohonen, T. (1982). Self-organized formation of topologically correct feature maps.Biological Cybernetics, 43, 59–69.

28. V. K. Pachghare, P. Kulkarni and D. M. Nikam, "Intrusion Detection System using Self Organizing Maps," *2009 International Conference on Intelligent Agent & Multi-Agent Systems*, 2009, pp. 1-5, doi: 10.1109/IAMA.2009.5228074.

29. Zimmermann, H. (2001). Fuzzy set theory and its applications. Kluwer Academic Publishers.

30. R, Shanmugavadivu & Dr.N.Nagarajan,. (2011). Network Intrusion Detection System using Fuzzy Logic. Indian Journal of Computer Science and Engineering. 2.

31. Pearl, J. (1988). Probabilistic reasoning in intelligent systems. Morgan Kaufmann.

32. Haibo He and E.A. Garcia. Learning from imbalanced data.Knowledgeand Data Engineering, IEEE Transactions on, 21(9):1263–1284, Sept2009.

33. Haibo He and Yunqian Ma.Imbalanced Learning: Foundations, Algo-rithms, and Applications. Wiley-IEEE Press, 1st edition, 2013

34. Amalia Luque, Alejandro Carrasco, Alejandro Martn, and Ana de lasHer as. The impact of class imbalance in classification performance metrics based on the binary confusion matrix.Pattern Recognition,91:216–231, 2019

35. V. S. Spelmen and R. Porkodi. A review on handling imbalanced data. In2018 International Conference on Current Trends towards ConvergingTechnologies (ICCTCT), pages 1–11, 2018

36. Gosain and S. Sardana. Handling class imbalance problem usingoversa mpling techniques: A review. In2017 International Conferenceon Advances in Computing, Communications and Informatics (ICACCI),pages 79–85, 2017

37. Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. PhilipKegelmeyer. Smote: Synthetic minority over-sampling technique.J.Artif. Int. Res., 16(1):321357, June 2002

38. Zhang, T.; Ramakrishnan, R.; Livny, M. (1996). "BIRCH: an efficient data clustering method for very large databases". Proceedings of the 1996 ACM SIGMOD international conference on Management of data - SIGMOD '96. pp. 103–114. doi:10.1145/233269.233324.

39. Panda, Mrutyunjaya & Patra, Manas. (2007). Network intrusion detection using naive bayes. 7.

40. K.M.Faroun, A.Boukelif, "Neural network learning improvement using K-means clustering algorithm to detect network intrusions", April 17, 2006, http://www.dcc.ufla.br/infocomp/artigos/v5.3/art04.pdf

41. Jang, J.-S., Sun, C.-T., & Mizutani, E. (1996). Neuro-fuzzy and soft computing: A computational approach to learning and machine intelligence. New Jersey: Prentice Hall.

42. Kittler, J., Hatef, M., Duin, R. P. W., & Matas, J. (1998). On combining classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(3), 226–239.

43. Opitz, D.; Maclin, R. (1999). "Popular ensemble methods: An empirical study". Journal of Artificial Intelligence Research. 11: 169–198. doi:10.1613/jair.614.

44. Polikar, R. (2006). "Ensemble based systems in decision making". IEEE Circuits and Systems Magazine. 6 (3): 21–45. doi:10.1109/MCAS.2006.1688199. S2CID 18032543.

45. Rokach, L. (2010). "Ensemble-based classifiers". Artificial Intelligence Review. 33 (1–2): 1–39. doi:10.1007/s10462-009-9124-7. S2CID 11149239.

46. R. Bonab, Hamed; Can, Fazli (2016). A Theoretical Framework on the Ideal Number of Classifiers for Online Ensembles in Data Streams. CIKM. USA: ACM. p. 2053.

47. R. Bonab, Hamed; Can, Fazli (2019). Less Is More: A Comprehensive Framework for the Number of Components of Ensemble Classifiers. TNNLS. USA: IEEE. arXiv:1709.02925.

48. https://en.wikipedia.org/wiki/BIRCH#cite_note-birch-1

49. https://en.wikipedia.org/wiki/ID3_algorithm

50. https://www.ll.mit.edu/r-d/datasets/1999-darpa-intrusion-detection-evaluation-dataset

51. L. Breiman, “Bagging predictors,” Machine learning, vol. 24, no. 2, pp. 123–140, 1996.

52. L. I. Kuncheva, Combining pattern classifiers: methods and algorithms-2nd Edition. John Wiley & Sons, 2014.

53. G. I. Webb, “Multiboosting: A technique for combining boosting and wagging,”Machine learning, vol. 40, no. 2, pp. 159–196, 2000.

54. Z.-H. Zhou, Ensemble methods: foundations and algorithms. CRC Press, 2012.

55. Adhi Tama, Bayu & Rhee, Kyung Hyune. (2017). An extensive empirical evaluation of classifier ensembles for intrusion detection task. Computer Systems Science and Engineering. 32. 149-158.

56. D. W. Vilela, E. Ferreira, A. A. Shinoda, N. V. de Souza Araujo, R. de Oliveira, and V. E. Nascimento, “A dataset for evaluating intrusion detection systems in IEEE 802.11 wireless networks,” in IEEE Colombian Conference on Communications and Computing (COLCOM). IEEE, 2014, pp. 1–5.

57. Balajinath, B., & Raghavan, S. V. (2000). Intrusion detection through behavior model. Computer Communication, 24, 1202–1212.

58. Bouzida, Y., Cuppens, F., Cuppens-Boulahia, N., & Gombault, S. (2004). Efficient intrusion detection using principal component analysis. In Paper presented at the proceedings of the 3eme conference surla securite et architectures reseaux (SAR). Orlando, FL, USA.

59. Chimphlee, W., Addullah, A. H., Sap, M. N. M., Srinoy, S., & Chimphlee, S. (2006). Anomaly-based intrusion detection using fuzzy rough clustering. In Paper presented at the international conference on hybrid information technology (ICHIT’06).

60. Depren, O., Topallar, M., Anarim, E., & Ciliz, M. K. (2005). An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks. Expert Systems with Applications, 29, 713–722.

61. Eskin, E., Arnold, A., Prerau, M., Portnoy, L., & Stolfo, S. (2002). A geometric framework for unsupervised anomaly detection: Detecting intrusions in unlabeled data. Kluwer.

62. Fan, W., Lee, W., Miller, M., Stolfo, S. J., & Chan, P. K. (2004). Using artificial anomalies to detect unknown and known network intrusions. Knowledge and Information Systems, 507–527.

63. Heller, K. A., Svore, K. M., Keromytis, A. D., & Stolfo, S. J. (2003). One class support vector machines for detecting anomalous window registry accesses. In Paper presented at the 3rd IEEE conference data mining workshop on data mining for computer security. Florida.

64. Li, Y., & Guo, L. (2007). An active learning based TCM-KNN algorithm for supervised network intrusion detection. Computer and Security, 26, 459–467.

65. Mukkamala, S., Sung, A. H., & Abraham, A. (2004). Modeling intrusion detection systems using linear genetic programming approach. In Paper presented at the proceedings of innovations in applied artificial intelligence, 17th international conference on industrial and engineering applications of artificial intelligence and expert systems (IEA/AIE). Lecture notes in computer science (Vol. 3029). Springer.

66. Abadeh, M. S., Habibi, J., Barzegar, Z., & Sergi, M. (2007). A parallel genetic local search algorithm for intrusion detection in computer networks. Engineering Applications of Artificial Intelligence, 20, 1058–1069.

67. Scott, S. L. (2004). A Bayesian paradigm for designing intrusion detection systems. Computational Statistics and Data Analysis, 45, 69–83.

68. Wang, K., & Stolfo, S. J. (2004). Anomalous Payload-based network intrusion detection. In Paper presented at the proceedings of recent advance in intrusion detection (RAID). Sophia Antipolis, France.

69. Chen, W.-H., Hsu, S.-H., & Shen, H.-P. (2005). Application of SVM and ANN for intrusion detection. Computer and Operations Research, 32, 2617–2634.

70. Zhang, Z., & Shen, H. (2005). Application of online-training SVMs for real-time intrusion detection with different considerations. Computer Communications, 28, 1428–1442.

71. Khan, L., Awad, M., & Thuraisingham, B. (2007). A new intrusion detection system using support vector machines and hierarchical clustering. The VLDB Journal, 16, 507–521.

72. Peddabachigari, S., Abraham, A., & Thomas, J. (2004). Intrusion detection systems using decision trees and support vector machines. International Journal of Applied Science and Computations.

73. Joo, D., Hong, T., & Han, I. (2003). The neural network models for IDS based on the asymmetric costs of false negative errors and false positive errors. Expert System with Applications, 25, 69–75.

74. Schultz, M. G., Eskin, E., Zadok, E., & Stolfo, S. J. (2001). Data mining methods for detection of new malicious executables. In Paper presented at the proceedings of the 2001 IEEE symposium on security and privacy (SP'01).

75. Kayacik, H. G., Nur, Z.-H., & Heywood, M. I. (2007). A hierarchical SOM-based intrusion detection system. Engineering Applications of Artificial Intelligence, 20, 439–451.

76. Ramos, V., & Abraham, A. (2005). ANTIDS: Self organized ant based clustering model for intrusion detection system. In Paper presented at the proceedings of the fourth IEEE international workshop on soft computing as transdisciplinary science and technology (WSTST'05). Berlin: Springer-Verlag.

77. Bridges, S. M., & Vaughn, R. B. (2000). Intrusion detection via fuzzy data mining. In Paper presented at the twelfth annual Canadian information technology security symposium. Ottawa, USA.

78. Wang, W., & Battiti, R. (2006). Identifying intrusions in computer networks with principal component analysis. In Paper presented at the proceedings of the first international conference on availability, reliability and security (ARES'06).

79. Tian, M., Chen, S. -C., Zhuang, Y., & Liu, J. (2004). Using statistical analysis and support vector machine classification to detect complicated attacks. In Paper presented at the proceedings of the third international conference on machine learning and cybernetics. Shanghai.

80. Wang, Y., Kim, I., Mbateng, G., & Ho, S.-Y. (2006). A latent class modeling approach to detect network intrusion. Computer Communications, 30, 93–100.

81. Luo, J., & Bridgest, S. M. (2000). Mining fuzzy association rules and fuzzy frequency episodes for intrusion detection. International Journal of Intelligent Systems, 15, 687–703.

82. Chen, Y., Abraham, A., & Yang, B. (2007). Hybrid flexible neural-tree-based intrusion detection systems. International Journal of Intelligent Systems, 22, 337–352.

83. Lee, W., & Stolfo, S. (2000). A framework for constructing features and models for intrusion detection systems. ACM Transactions on Information and System Security (TISSEC), 3(4), 227–261.

84. Wang, W., Guan, X., & Zhang, X. (2004). A novel intrusion detection method based on principle component analysis in computer security. In Paper presented at the proceedings of the international symposium on neural networks. Dalian, China.

85. Florez, G., Bridges, S. M., & Vaughn, R. B. (2002). An improved algorithm for fuzzy data mining for intrusion detection. In Paper presented at the proceedings of the North American fuzzy information processing society conference (NAFIPS 2002). New Orleans, LA.

86. Chavan, S., Shah, K. D. N., & Mukherjee, S. (2004). Adaptive neuro-fuzzy intrusion detection systems. In Paper presented at the in proceedings of the international conference on information technology: Coding and computing (ITCC'04).

87. Shyu, M., Chen, S., Sarinnapakorn, K., & Chang, L. (2003). A novel anomaly detection scheme based on principal component classifier. In Paper presented at the proceedings of ICDM'03.

88. Lee, W., & Stolfo, S. (1998). Data mining approaches for intrusion detection. In Paper presented at the proceedings of the seventh USENIX security symposium (SECURITY'98). San Antonio, TX.

89. Jiang, S. Y., Song, X., Wang, H., Han, J.-J., & Li, Q.-H. (2006). A clustering-based method for unsupervised intrusion detections. Pattern Recognition Letters, 27, 802–810.

90. Ozyer, T., Alhajj, R., & Barker, K. (2007). Intrusion detection by integrating boosting genetic fuzzy classifier and data mining criteria for rule pre-screening. Journal of Network and Computer Applications, 30, 99–113.

91. Liu, G., & Yi, Z. (2006). Intrusion detection using PCASOM neural networks. In Paper presented at the proceeding of ISNN2006. Lecture notes in computer science. Berlin, Heidelberg.

92. Giacinto, G., & Roli, F. (2003). Intrusion detection in computer networks by multiple classifier systems. In Paper presented at the proceeding of ICPR 2002, 16th international conference on pattern recognition. Quebec City, Canada.

93. Stein, G., Chen, B., Wu, A. S., & Hua, K. A. (2005). Decision tree classifier for network intrusion detection with GA-based feature selection. In Paper presented at the proceedings of the 43rd annual Southeast regional conference. Kennesaw, Georgia.

94. Liu, G., Yi, Z., & Yang, S. (2007). A hierarchical intrusion detection model based on the PCA neural networks. Neurocomputing, 70, 1561–1568.

95. Moradi, M., & Zulkernine, M. (2004). A neural network based system for intrusion detection and classification of attacks. In Paper presented at the proceeding of the 2004 IEEE international conference on advances in intelligent systems – Theory and applications. Luxembourg.

96. Shon, T., & Moon, J. (2007). A hybrid machine learning approach to network anomaly detection. Information Sciences, 177, 3799–3821.

97. Liu, Y., Chen, K., Liao, X., & Zhang, W. (2004). A genetic clustering method for intrusion detection. Pattern Recognition, 37, 927–942.

98. Tsang, C.-H., Kwong, S., & Wang, H. (2007). Genetic-fuzzy rule mining approach and evaluation of feature selection techniques for anomaly intrusion detection. Pattern Recognition, 40, 2373–2391.

99. Peddabachigari, S., Abraham, A., Grosan, C., & Thomas, J. (2007). Modeling intrusion detection system using hybrid intelligent systems. Journal of Network and Computer Applications, 30, 114–132.

100. Shon, T., Kovah, X., & Moon, J. (2006). Applying genetic algorithm for classifying anomalous TCP/IP packets. Neurocomputing, 69, 2429–2433.

101. Zhang, L.-H., Zhang, G.-H., Yu, L., Zhang, J., & Bai, Y.-C. (2004). Intrusion detection using rough set classification. Journal of Zhejiang University Science, 5(9), 1076–1086.

102. Toosi, A. N., & Kahani, M. (2007). A new approach to intrusion detection based on an evolutionary soft computing model using neuro-fuzzy classifiers. Computer Communication, 30, 2201–2212.

103. Zhang, C., Jiang, J., & Kamel, M. (2005). Intrusion detection using hierarchical neural network. Pattern Recognition Letters, 26, 779–791.

104. Xiang, C., & Lim, S. M. (2005). Design of multiple-level hybrid classifier for intrusion detection system. In Paper presented at the proceeding of the IEEE workshop machine learning for signal processing.

105. C. Dartigue, J. Hyun Ik and W. Zeng, A New Data- Mining Based Approach for Network Intrusion Detection," Communication Networks and Services Research Conference, 2009. CNSR '09. Seventh Annual. 2009.

106. Giacinto, G., Perdisci, R., Rio, M. D., & Roli, F. (2006). Intrusion detection in computer networks by a modular ensemble of one-class classifiers. Information Fusion, 9, 69–82.

107. Mukkamala, S., Sung, A. H., & Abraham, A. (2005). Intrusion detection using an ensemble of intelligent paradigms. Network and Computer Applications, 28, 167–182.

108. Kang, D. K., Fuller, D., & Honavar, V. (2005). Learning classifiers for misuse and anomaly detection using a bag of system calls representation. In Paper presented at the proceeding of the 2005 IEEE.

109. Han, S.-J., & Cho, S.-B. (2003). Detecting intrusion with ruled-based integration of multiple models. Computers and Security, 22(7), 613–623.

110. Fan, W., Lee, W., Miller, M., Stolfo, S. J., & Chan, P. K. (2001). Using artificial anomalies to detect unknown and known network intrusions. Knowledge and Information Systems, 507–527.

111. S. J. Horng, M. Y. Su, Y. H. Chen, T. W. Kao, R. J. Chen, J. L. Lai and C. D. Kara, "A novel intrusion detection system based on hierarchical clustering and support vector machines," Expert Systems with Applications, 38(1): 306-313. 2011.

112. P. Somwang, and W. Lilakiatsakun, “Computer network security based on Support Vector Machine approach,” 2011 11th International Conference on Control, Automation and Systems, (ICCAS 2011).

113. H. Altwaijry and S. Algarny, "Bayesian based intrusion detection system." Journal of King Saud University - Computer and Information Sciences, 24(1): 1-6, 2012.

114. F. Amiri, M. R. Yousefi, C. Lucas, A. Shakery, N. Yazdani, "Mutual information-based feature selection for intrusion detection systems," Journal of Network and Computer Applications, 34(4): 1184-1199. 2011.

115. N. Araújo, R. de Oliveira, E. Ferreira, A. A. Shinoda and B. Bhargava, "Identifying important characteristics in the KDD99 intrusion detection dataset by feature selection using a hybrid approach," 2010 17th International Conference on Telecommunications, 2010, pp. 552-558, doi: 10.1109/ICTEL.2010.5478852.

116. R. Ashok, A. J. Lakshmi, G. D. V. Rani, M. N. Kumar, “Optimized feature selection with k-means clustered triangle SVM for Intrusion Detection,” 2011 Third International Conference on. Advanced Computing (ICoAC), 2011.

117. V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso- Belanzos, "Feature selection and classification in multiple class datasets: An application to KDD Cup 99 dataset," Expert Systems with Applications 38(5): 5947-5957. 2011.

118. C. A. Catania, F. Bromberg and C. G. Garino, "An autonomous labeling approach to support vector machines algorithms for network traffic anomaly detection." Expert Systems with Applications, 39(2): 1822-1829, 2012.

119. R. C. Chen, K. F. Cheng C. F. Hsieh, “Using Rough Set and Support Vector Machine for Network Intrusion Detection System,” Proceedings of the 2009 First Asian Conference on Intelligent Information and Database Systems, IEEE Computer Society, 465-470, 2009.

120. C. Chi, T. Wee-Peng H. Guang-Bin, “Extreme learning machines for intrusion detection,” The 2012 International Joint Conference on Neural Networks (IJCNN), 2012.

121. G. Chunhua, and Z. Xueqin, “A Rough Set and SVM Based Intrusion Detection Classifier,” Second International Workshop on Computer Science and Engineering (WCSE '09), 2009.

122. L. Cohen, G. Avrahami M. Last, A. Kandel, "Info- fuzzy algorithms for mining dynamic data streams." Applied Soft Computing, 8(4): 1283-1294, 2008.

123. H.F. Eid, A. Darwish A. H. Ella and A. Abraham, “Principle components analysis and Support Vector Machine based Intrusion Detection System,” 2010, 10th International Conference on Intelligent Systems Design and Applications (ISDA), 2010.

124. D. M. Farid, and M. Z. Rahman, “Anomaly Network Intrusion Detection Based on Improved Self Adaptive Bayesian Algorithm,” 2010.

125. L. Feng, W. Wang, L. Zhu and Y. Zhang, "Predicting intrusion goal using dynamic Bayesian network with transfer probability estimation." Journal of Network and Computer Applications, 32(3): 721-732, 2009.

126. Z. Gengming and L. Junguo, “Research of Intrusion Detection Based on Support Vector Machine,” International Conference on Advanced Computer Theory and Engineering 2008 (ICACTE '08), 2008.

127. S.J. Horng, P. Fan, Y. P. Chou, Y. C. Chang Y. Pan, "A feasible intrusion detector for recognizing IIS attacks based on neural networks." Computers & Security, 27(3-4): 84-100, 2008.

128. J. Jiaqi, L. Ru, Z. Tianhang and S. Feigin, “A New Intrusion Detection System Using Class and Sample Weighted C-support Vector Machine,” 2011 Third International Conference on Communications and Mobile Computing (CMC), 2011.

129. Y. Jingbo, L. Haixiao D. Shunli and C. Limin, Intrusion Detection Model Based on Improved Support Vector Machine. 2010 Third International Symposium on Intelligent Information Technology and Security Informatics (IITSI), 2010.

130. Z. Kai-mei, Q. Xu Z. Vu and J. Li-juan, “Intrusion Detection Using Isomap and Support Vector Machine,” AICI '09, 2009 International Conference on Artificial Intelligence and Computational Intelligence, 2009.

131. N. Kausar, B. B. Samir S. B. Sulaiman, I. Ahmad and M. Hussain, “An approach towards intrusion detection using PCA feature subsets and SVM,” 2012 International Conference on Computer & Information Science (ICCIS), 2012.

132. L. Koc, T.A. Mazzuchi and S. Sarkani, "A network intrusion detection system based on a Hidden Naïve Bayes multiclass classifier." Expert Systems with Applications, 39(18): 13492-13500, 2012.

133. W. Li, and Z. Liu, "A method of SVM with Normalization in Intrusion Detection." Procedia Environmental Sciences 11, Part A(0): 256-262, 2011.

134. Y. Li, J. Xia, S. Zhang, J. Yan, X. Xi and K. Dai, "An efficient intrusion detection system based on support vector machines and gradually feature removal method." Expert Systems with Applications, 39(1): 424-430, 2012.

135. Z. L. Li, M. Z. Ya, B. Z. Yu, "Network intrusion detection method by least squares support vector machine classifier," 2010 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT), 2010.

136. M. N. Mohammad, N. Sulaiman and E. T. Khalaf, "A novel local network intrusion detection system based on support vector machine." Journal of Computer Science, 7(10): 1560-1564, 2011.

137. M.N. Mohammed and N. Sulaiman, "Intrusion Detection System Based on SVM for WLAN," Procedia Technology, 1(0): 313-317, 2012.

138. M. S. Mok, S. Y. Sohn and Y. H. Ju, "Random effects logistic regression model for anomaly detection," Expert Systems with Applications, 37(10): 7162-7166, 2010.

139. S. Mukherjee and N. Sharma, "Intrusion Detection using Naive Bayes Classifier with Feature Reduction," Procedia Technology, 4(0): 119-128, 2012.

140. A.P. Muniyandi, R. Rajeswari and R. Rajaram, "Network Anomaly Detection by Cascading K-Means Clustering and C4.5 Decision Tree algorithm," Procedia Engineering, 30(0): 174-182, 2012.

141. M. Muntean, H. Valean, L. Miclea and A. Incze, "A novel intrusion detection method based on support vector machines," 2010 11th International Symposium on Computational Intelligence and Informatics (CINTI), 2010.

142. C.R. Pereira, R.Y.M. Nakamura, K.A.P Costa, J.P. Papa, "An Optimum-Path Forest framework for intrusion detection in computer networks." Engineering Applications of Artificial Intelligence, 25(6): 1226-1234, 2012.

143. S. Saha, A.S. Sairam, A. Yadav and A. Ekbal, "Genetic algorithm combined with support vector machine for building an intrusion detection system," Proceedings of the International Conference on Advances in Computing, Communications and Informatics. Chennai, India, ACM: 566-572, 2012.

144. P. Sangkatsanee, N. Wattanapongsakorn and C. Charnsripinyo, "Practical real-time intrusion detection using machine learning approaches," Computer Communications, 34(18): 2227-2235. 2011.

145. N. Sharma and S. Mukherjee, "A Novel Multi-Classifier Layered Approach to Improve Minority Attack Detection in IDS." Procedia Technology, 6(0): 913-921. 2012.

146. H. M. Shirazi, "Anomaly Intrusion Detection System Using Information Theory, K-NN and KMC algorithms." Australian Journal of Basic & Applied Sciences, 3(3): 251-2597, 2009.

147. S. Suthaharan and T. Panchagnula, “Relevance feature selection with data cleaning for intrusion detection system,” 2012 Proceedings of IEEE Southeastcon, 2012.

148. P. Winter, E. Hermann and M. Zeilinger, “Inductive Intrusion Detection in Flow-Based Network Data Using One-Class Support Vector Machines,” 2011 4th IFIP International Conference on New Technologies, Mobility and Security (NTMS), 2011.

149. Y. Xie and Y. Zhang, “An intelligent anomaly analysis for intrusion detection based on SVM,” 2012 International Conference on Computer Science and Information Processing (CSIP), 2012.

150. D. Xuejun, Z. Guiling, Y. Ke, B. Ma and Z. LI, “High Efficient Intrusion Detection Methodology with Twin Support Vector Machines,” International Symposium on Information Science and Engineering 2008 (ISISE '08), 2008.

151. Y. Yi, J. Wu and W. Xu, "Incremental SVM based on reserved set for network intrusion detection," Expert Systems with Applications, 38(6): 7698-7707, 2011.

152. Z. Yongli, and Z. Yanwei, “Application of Improved Support Vector Machines in Intrusion Detection,” 2010 2nd International Conference on e-Business and Information System Security (EBISS), 2010.

153. J. Yu, H. Lee, M. Kim and D. Park, "Traffic flooding attack detection with SNMP MIB using SVM," Computer Communications, 31(17): 4212-4219, 2008.

154. S. Zaman, and F. Karray, “Features Selection for Intrusion Detection Systems Based on Support Vector Machines,” 6th IEEE Consumer Communications and Networking Conference 2009 (CCNC 2009), 2009.

155. R. Zhao, Y. Yu and M. Cheng, “An Intrusion Detection Algorithm Model Based on Extension Clustering Support Vector Machine,” International Conference on Artificial Intelligence and Computational Intelligence 2009 (AICI '09).

156. K. Shafi and H. A. Abbass , "An adaptive genetic-based signature learning system for intrusion detection." Expert Systems w ith Applications, 36(10): 12036-12043, 2009.

157. B. Agarwal and N. Mittal, "Hybrid Approach for Detection of Anomaly Network Traffic using Data Mining Techniques." Procedia Technology, 6(0): 996- 1003, 2012.

158. N. Devarakonda, S. Pamidi, V. V. Kumari and A. Govardhan, "Intrusion Detection System using Bayesian Network and Hidden Markov Model." Procedia Technology, 4(0): 506-514, 2012.

159. X. S. Gan, J. S. Duanmu, J. F. Wang and W. Cong, "Anomaly intrusion detection based on PLS feature extraction and core vector machine." Knowledge- Based Systems, 40(0): 1-6, 2013.

160. S. Ganapathy, K. Kulothungan, P. Vogesh and A. Kannan, "A Novel Weighted Fuzzy C –Means Clustering Based on Immune Genetic Algorithm for Intrusion Detection." Procedia Engineering, 38(0): 1750-1757, 2012.

161. Z. Guiling, K. Yongzhen, S. Liankun and L. Weixin, "An Improvement of Payload-Based Intrusion Detection Using Fuzzy Support Vector Machine," 2010 2nd International Workshop on Intelligent Systems and Applications (ISA), 2010.

162. X. D. Hoang, J. Hu, and P. Bertok, "A program-based anomaly intrusion detection scheme using multiple detection engines and fuzzy inference." Journal of Network and Computer Applications 32(6): 1219- 1228, 2009.

163. L. Huike and G. Daquan, "A Novel Intrusion Detection Scheme Using Support Vector Machine Fuzzy Network for Mobile Ad Hoc Networks," Second Pacific-Asia Conference on Web Mining and Web- based Application, 2009. (WMWA '09).

164. F. Kuang, W. Xu, S. Zhang, Y. Wang and K Liu, "A novel approach of KPCA and SVM for intrusion detection," Journal of Computational Information Systems, 8(8): 3237-3244, 2012.

165. S. Lee, G. Kim and S. Kim, "Self-adaptive and dynamic clustering for online anomaly detection." Expert Systems with Applications, 38(12): 14891- 14898, 2011.

166. L. Lei, and Z. Ke-nan, "A New Intrusion Detection System Based on Rough Set Theory and Fuzzy Support Vector Machine," 3rd International Workshop on Intelligent Systems and Applications (ISA), 2011.

167. L. Lei, G. Zhi-ping, D. Wen-Yan, "Fuzzy Multi-class Support Vector Machine Based on Binary Tree in Network Intrusion Detection," International Conference on Electrical and Control Engineering (ICECE), 2010.

168. S.W. Lin, K.C. Ying C. Y. Lee and Z. J. Lee, "An intelligent algorithm with feature selection and decision rules applied to anomaly intrusion detection." Applied Soft Computing, 12(10): 3285-3290, 2012.

169. Z. Liu, J. Kang and Y. Li, "A hybrid method of rough set and support vector machine in network intrusion detection," 2nd International Conference on Signal Processing Systems (ICSPS), 2010.

170. G. Meijuan, T. Jingwen and X. Mingping, "Intrusion Detection Method Based on Classify Support Vector Machine," Second International Conference on Intelligent Computation Technology and Automation, 2009. ICICTA '09.

171. S.A. Mulay, P.R. Devale and G.V. Garje, "Decision tree based Support Vector Machine for Intrusion Detection," International Conference on Networking and Information Technology (ICNIT), 2010.

172. H. Om and A. Kundu, “A hybrid system for reducing the false alarm rate of anomaly intrusion detection system,” 1st International Conference on Recent Advances in Information Technology (RAIT), 2012.

173. V.K. Pachghare and P. Kulkarni, “Pattern based network security using decision trees and support vector machine,” 3rd International Conference on Electronics Computer Technology (ICECT), 2011.

174. M. Panda, A. Abraham and M. R. Patra, "A Hybrid Intelligent Approach for Network Intrusion Detection," Procedia Engineering, 30(0): 1-9, 2012.

175. T. Pingjie, J. Rong-an and Z. Mingwei, “Feature Selection and Design of Intrusion Detection System Based on k-Means and Triangle Area Support Vector Machine,” Second International Conference on Future Networks, 2010, ICFN '10.

176. S.T. Powers and J. He, "A hybrid artificial immune system and Self Organising Map for network intrusion detection." Information Sciences, 178(15): 3024-3042, 2008.

177. K. Qazanfari, M. S. Mirpouryan and H. Gharaee, “A novel hybrid anomaly based intrusion detection method,” Sixth International Symposium on Telecommunications (IST), 2012.

178. P. Srinivasu and P. S. Avadhani, "Genetic Algorithm based Weight Extraction Algorithm for Artificial Neural Network Classifier in Intrusion Detection." Procedia Engineering, 38(0): 144-153, 2012.

179. M.Y. Su, "Real-time anomaly detection systems for Denial-of-Service attacks by weighted k-nearest- neighbor classifiers." Expert Systems with Applications, 38(4): 3492-3498, 2011.

180. G. C. Tjhai, S. M. Furnell, M. Papadaki and N. L. Clarke, "A preliminary two-stage alarm correlation and filtering system using SOM neural network and K-means algorithm," Computers & Amp; Security, 29(6): 712-723. 2010.

181. X. Tong, Z. Wang and H. Yu, "A research using hybrid RBF/Elman neural networks for intrusion detection system secure model." Computer Physics Communications, 180(10): 1795-1801, 2009.

182. C. F. Tsai, and C. Y. Lin, "A triangle area based nearest neighbors approach to intrusion detection," Pattern Recognition, 43(1): 222-229. 2010.

183. J. Visumathi and K. L. Shunmuganathan, "An Effective IDS for MANET Using Forward Feature Selection and Classification Algorithms," Procedia Engineering, 38(0): 2816-2823, 2012.

184. G. Wang, J. Hao, J. Hao and L. Huang, "A new approach to intrusion detection using Artificial Neural Networks and fuzzy clustering." Expert Systems with Applications, 37(9): 6225-6232, 2010.

185. Y.X. Wei and M. Q. Wu, "KFDA and clustering based multiclass SVM for intrusion detection." The Journal of China Universities of Posts and Telecommunications, 15(1): 123-128, 2008.

186. Z. Wei, T. Shaohua, Z. Haibin, H. Du and X. Li, “Fuzzy Multi-Class Support Vector Machines for cooperative network intrusion detection,” 9th IEEE International Conference on Cognitive Informatics (ICCI), 2010.

187. C. Xiang, P. C. Yong and L. S. Meng, "Design of multiple-level hybrid classifier for intrusion detection system using Bayesian clustering and decision trees." Pattern Recognition Letters, 29(7): 918-924, 2008.

188. F. Xiaozhao, Z. Wei, T. Shaohua and H. Na, “A Research on Intrusion Detection Based on Support Vector Machines,” International Conference on Communications and Intelligence Information Security (ICCIIS), 2010.

189. C. Zhenguo and Z. Guanghua, “Support Vector Machines Improved by Artificial Immunisation Algorithm for Intrusion Detection,” International Conference on Information Engineering and Computer Science, 2009, ICIECS 2009.

190. M. Govindarajan and R.M. Chandrasekaran, "Intrusion detection using neural bas ed hybrid classification methods," Computer Networks, 55(8): 1662-1671, 2011.

191. S. Guanghui, G. Jiankang, N. Yan, “An Intrusion Detection Method Based on Multiple Kernel Support Vector Machine,” International Conference on Network Computing and Information Security (NCIS), 2011.

192. B. Kavitha, D.S. Karthikeyan and P. S. Maybell, "An ensemble design of intrusion detection system for handling uncertainty using Neutrosophic Logic Classifier," Knowledge-Based Systems, 28(0): 88-96, 2012.

193. Y. Li, J.L. Wang, Z. H. Tian, T. B. Lu and C. Young, "Building lightweight intrusion detection system using wrapper-based feature selection mechanisms." Computers & Security, 28(6): 466-475, 2009.

194. P. A. Raj Kumar and S. Selvakumar, "Distributed denial of service attack detection using an ensemble of neural classifier," Computer Communications, 34(11): 1328- 1341.2011.

195. S. S. S. Sindhu, S. Geetha, A. Kannan, "Decision tree based light weight intrusion detection using a wrapper approach." Expert Systems with Applications, 39(1): 129-141, 2012.

196. S. M. Lee, D. S. Kim J. H. Lee and J. S. Park, "Detection of DDoS attacks using optimized traffic matrix." Computers & amp; Mathematics with Applications, 63(2): 501-510, 2012.

197. P. K. Sujatha, C. S. Priya and A. Kannan, “Network intrusion detection system using genetic network programming with support vector machine,” Proceedings of the International Conference on Advances in Computing, Communications and Informatics. Chennai, India, ACM: 645-649. 2012.

198. A. R. Syarif and W. Gata, “Intrusion detection system using hybrid binary PSO and K-nearest neighborhood algorithm,” Proc. 11th Int. Conf. Inf. Commun.

Technol. Syst. ICTS 2017, vol. 2018-Janua, pp. 181–186, 2018, doi: 10.1109/ICTS.2017.8265667.

199. V. Hajisalem and S. Babaie, “A hybrid intrusion detection system based on ABC-AFS algorithm for misuse and anomaly detection,” Comput. Networks, vol. 136, pp. 37–50, 2018, doi: 10.1016/j.comnet.2018.02.028.

200. M. A. Jabbar, R. Aluvalu, and S. Sai Satyanarayana Reddy, “Cluster based ensemble classification for intrusion detection system,” ACM Int. Conf. Proceeding Ser., vol. Part F1283, pp. 253–257, 2017, doi: 10.1145/3055635.3056595.

201. L. P. Dias, J. J. F. Cerqueira, K. D. R. Assis, and R. C. Almeida, “Using artificial neural network in intrusion detection systems to computer networks,” 2017 9th Comput. Sci. Electron. Eng. Conf. CEEC 2017 - Proc., pp. 145–150, 2017, doi: 10.1109/CEEC.2017.8101615.

202. M. A. Manzoor and Y. Morgan, “Real-time Support Vector Machine based Network Intrusion Detection system using Apache Storm,” 7th IEEE Annu. Inf. Technol. Electron. Mob. Commun. Conf. IEEE IEMCON 2016, pp. 1–5, 2016, doi: 10.1109/IEMCON.2016.7746264.

203. Akashdeep, I. Manzoor, and N. Kumar, “A feature reduced intrusion detection system using ANN classifier,” Expert Syst. Appl., vol. 88, pp. 249–257, 2017, doi: 10.1016/j.eswa.2017.07.005.

204. D. G. Mogal, S. R. Ghungrad, and B. B. Bhusare, “NIDS using Machine Learning Classifiers on UNSW-NB15 and KDDCUP99 Datasets,” Ijarcce, vol. 6, no. 4, pp. 533–537, 2017, doi: 10.17148/ijarcce.2017.64102.

205. A. S. Amira, S. E. O. Hanafi, and A. E. Hassanien, “Comparison of classification techniques applied for network intrusion detection and classification,” J. Appl. Log., vol. 24, pp. 109–118, 2017, doi: 10.1016/j.jal.2016.11.018.

206. J. Hrabovsky, P. Segec, M. Moravcik, and J. Papan, Trends in application of machine learning to network-based intrusion detection systems, vol. 863. Springer International Publishing, 2018.

207. M. Alkasassbeh and M. Almseidin, (2018). “Machine Learning Methods for Network Intrusion Detection.”

208. P. Verma, S. Anwar, S. Khan, and S. B. Mane, “Network Intrusion Detection Using Clustering and Gradient Boosting,” 2018 9th Int. Conf. Comput. Commun. Netw. Technol. ICCCNT 2018, pp. 1–7, 2018, doi: 10.1109/ICCCNT.2018.8494186.

209. N. S. Naganhalli and S. Terdal, “Network intrusion detection using supervised machine learning technique,” Int. J. Sci. Technol. Res., vol. 8, no. 9, pp. 345–350, 2019.

210. D. P. Gaikwad and R. C. Thool, “Intrusion detection system using Bagging with Partial Decision Tree base classifier,” Procedia Comput. Sci., vol. 49, no. 1, pp. 92–98, 2015, doi: 10.1016/j.procs.2015.04.231.

211. W. C. Lin, S. W. Ke, and C. F. Tsai, “CANN: An intrusion detection system based on combining cluster centers and nearest neighbors,” Knowledge-Based Syst., vol. 78, no. 1, pp. 13–21, 2015, doi: 10.1016/j.knosys.2015.01.009.

212. Y. Zhou, G. Cheng, S. Jiang, M. Dai, “Building an efficient intrusion detection system based on feature selection and ensemble classifier”, Computer Networks, Volume 174, 2020, 107247.

213. A. Iqbal and S. Aftab, “A Feed-Forward and Pattern Recognition ANN Model for Network Intrusion Detection,” no. April, 2019, doi: 10.5815/ijcnis.2019.04.03.

214. Y. V. Kumar and K. Kamatchi, “Anomaly Based Network Intrusion Detection Using Ensemble Machine Learning Technique,” no. 4, 2020.

215. P. Maniriho, L. J. Mahoro, E. Niyigaba, Z. Bizimana, T. Ahmad, “Detecting Intrusions in Computer Network Traffic with Machine Learning Approaches Detecting Intrusions in Computer Network Traffic with Machine Learning Approaches,” no. April, 2020, doi: 10.22266/ijies2020.0630.39.

216. P. Raviteja, M. Satya Venkata Sarojini Devi, M. Gowri, M. Vamsi S. Krishna, P. V. S. Prabhakar, “Implementation of Machine Learning Algorithms For Detection Of Network Intrusion,” vol. 8, no. 2, pp. 163–169, 2020.

217. S. Rajagopal, P. P. Kundapur, and K. S. Hareesha, “A Stacking Ensemble for Network Intrusion Detection Using Heterogeneous Datasets,” vol. 2020, 2020.

218. A. V. Mankar, T. C. Ravekar, “A Study of Intrusion Detection System using Advanced Genetic Algorithm,” vol. 3, no. 11, pp. 7–12, 2016.

219. X. Yuan, R. Wang, Y. Zhuang, K. Zhu and J. Hao, "A Concept Drift Based Ensemble Incremental Learning Approach for Intrusion Detection," 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), 2018, pp. 350-357, doi: 10.1109/Cybermatics_2018.2018.00087.

220. F. Breve and L. Zhao, "Semi-supervised Learning with Concept Drift Using Particle Dynamics Applied to Network Intrusion Detection Data," 2013 BRICS Congress on Computational Intelligence and 11th Brazilian Congress on Computational Intelligence, 2013, pp. 335-340, doi: 10.1109/BRICS-CCI-CBIC.2013.63.

221. Chih-Fong Tsai, Yu-Feng Hsu, Chia-Ying Lin, Wei-Yang Lin, Intrusion detection by machine learning: A review, Expert Systems with Applications, Volume 36, Issue 10, 2009, Pages 11994-12000, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2009.05.029.