ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ & ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΑΝΑΛΥΣΗ ΚΥΚΛΟΦΟΡΙΑΣ ΣΕ ΑΣΥΡΜΑΤΑ ΔΙΚΤΥΑ ΜΕ ΤΕΧΝΙΚΕΣ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ

TRAFFIC ANALYSIS IN WIRELESS NETWORKS USING MACHINE LEARNING TECHNIQUES

Διπλωματική Εργασία

Καποδίστρια Αγγελική

Επιβλέποντες: Δρ. Αργυρίου Αντώνιος
Δρ. Κοράκης Αθανάσιος

Βόλος, Σεπτέμβριος 2015

Στην οικογένειά μου

ΕΥΧΑΡΙΣΤΙΕΣ

Με την περάτωση της παρούσας εργασίας, θα ήθελα αρχικά να ευχαριστήσω θερμά τους επιβλέποντες καθηγητές Δρ. Αγυρίου Αντώνιο και Δρ. Κοράκη Αθανάσιο για τη βοήθειά τους και την άριστη συνεργασία καθ' όλη τη διάρκεια εκπόνησης της εργασίας αυτής, αλλά και για τη στήριξη και τη βοήθειά τους σε όλη τη διάρκεια των σπουδών.

Ένα μεγάλο ευχαριστώ στην οικογένεια μου που ήταν δίπλα μου, στηρίζοντας με και βοηθώντας όλα αυτά τα χρόνια και ένα ακόμη ευχαριστώ στον Χάρη και τους φίλους μου.

ΠΕΡΙΕΧΟΜΕΝΑ

ΕΥΡΕΤΗΡΙΟ ΠΙΝΑΚΩΝ

ΕΥΡΕΤΗΡΙΟ ΣΧΗΜΑΤΩΝ

ΠΕΡΙΛΗΨΗ

Η ταξινόμηση της κίνησης ενός δικτύου, είτε ενσύρματου είτε ασύρματου, είναι ένας τομέας που απασχολεί πολύ τους ερευνητές την τελευταία δεκαετία. Η παρούσα εργασία, χρησιμοποιεί τεχνικές Μηχανικής Μάθησης για την μελέτη και την ανάλυση της κίνησης σε ασύρματα δίκτυα. Αρχικά, ανιχνεύονται τα πακέτα που κινούνται στο δίκτυο και εν συνεχεία ταξινομούνται σε κλάσεις – κατηγορίες, ανάλογα με την εφαρμογή τους. Οι κλάσεις – κατηγορίες που μελετώνται εδώ είναι κυρίως Torrent και You Tube, λόγω της συχνής χρήσης των δύο αυτών εφαρμογών στην καθημερινότητα. Στόχος της έρευνας αυτής είναι να μπορούμε μέσω παθητικών μετρήσεων να γνωρίζουμε τι κινείται στο δίκτυο, καθώς επίσης και να παρέχεται μία καλύτερη ποιότητα υπηρεσιών (Quality of Services -QoS) στους χρήστες των δικτύων.

ABSTRACT

Traffic classification in a network, whether wired or wireless, is an area that employs many researchers in the last decade. This thesis uses machine learning techniques to study and analyze traffic on wireless networks. Initially, we detect the packets traveling on the network and then we sort them into classes, depending on their application. Classes studied here are mainly Torrent and You Tube, because of the frequent use of these two applications in everyday life. The objective of this research is to know through passive measurements what is moving on the network, as well as to provide a better quality of service (Quality of Services -QoS) to network users.

ΚΕΦΑΛΑΙΟ 1. ΕΙΣΑΓΩΓΗ

1.1. Το πρόβλημα

Η *ταξινόμηση της κίνησης* (traffic classification) σε ένα δίκτυο δεδομένων αποτελεί σημαντικό πρόβλημα και χρησιμεύει στην επίλυση δύσκολων προβλημάτων διαχείρισης δικτύου και αφορά τον φορέα (carrier), τους παρόχους υπηρεσιών διαδικτύου (Internet Service Providers (ISPs)) και τους διαχειριστές του δικτύου. Η *ταξινόμηση κίνησης* είναι μια τεχνική που αντιστοιχεί τις ροές κυκλοφορίας με τις εφαρμογές ή τα πρωτόκολλα που τις δημιουργούν. Έτσι, ο διαχειριστής του δικτύου έχει την εποπτεία παρατηρώντας τις ροές των πακέτων που κινούνται στο δίκτυο και διαχειρίζεται την ασφάλεια, την κίνηση (π.χ. αποτρέπει ροές ή ορίζει προτεραιότητες) και γενικότερα τις παραμέτρους που επιτυγχάνουν τους επιθυμητούς στόχους απόδοσης.

Η ταξινόμηση κίνησης επιτυγχάνεται με διάφορους τρόπους, όπως [1]:

- Έλεγχος αριθμού θύρας (Port number Check)
- Σε βάθος εξέταση πακέτων (Deep Packet Inspection)
- Στατιστική Ταξινόμηση (Statistical Classification)

Ο *έλεγχος αριθμού θύρας* είναι η απλούστερη και ταχύτερη παραδοσιακή τεχνική που βασίζεται στα ports (well-known ports) του επιπέδου μεταφοράς (TCP και UDP) για να αποφανθεί σε ποια εφαρμογή αντιστοιχούν τα πακέτα. Ο ταξινομητής χρησιμοποιεί τον αριθμό θύρας για να κατατάξει τα πακέτα σε αντίστοιχες κατηγορίες. Για παράδειγμα πακέτα με αριθμό θύρας 80 χαρακτηρίζονται ως HTTP

πακέτα. Οι συγκεκριμένες μέθοδοι λειτουργούν για πολύ γνωστές εφαρμογές και δεν μπορούν να χρησιμοποιηθούν σε κρυπτογραφημένα πακέτα. Επιπλέον δεν θεωρούνται αξιόπιστες.

Στην σε *βάθος εξέταση πακέτων* τεχνική συγκρίνεται το περιεχόμενο του πακέτου (payload) αν ταιριάζει με συγκεκριμένα πρότυπα (pattern matching) που ουσιαστικά αποτελούν την υπογραφή της εκάστοτε εφαρμογής. Η διαδικασία μοιάζει με το ταίριασμα με κανονική έκφραση. Αν και είναι ευρέως διαδεδομένες, δεν είναι αρκετά ακριβείς και όταν η κίνηση είναι κρυπτογραφημένη, η ακρίβεια μειώνεται ακόμη πιο πολύ. Επίσης εγείρονται ζητήματα όπως η ιδιωτικότητα (privacy) και το αυξημένο υπολογιστικό κόστος για την υλοποίησης τους.

Η *στατιστική ταξινόμηση* βασίζεται σε χαρακτηριστικά της κυκλοφορίας για να αντιστοιχίσει τις ροές της σε εφαρμογές. Συγκεκριμένα αξιοποιεί στατιστικές ιδιότητες των ροών κυκλοφορίας όπως η διάρκεια (duration), ο ανενεργός χρόνος (idle time), το μήκος του πακέτου, κα, για να διακρίνει διαφορετικές μεταξύ τους ροές και να τις κατηγοριοποιήσει. Μία από τις πιο αποτελεσματικές μεθόδους, που χρησιμοποιείται συχνά στην στατιστική ταξινόμηση, είναι η Μηχανική Μάθηση (Machine Learning) καθώς επιτρέπει τον χειρισμό διαφορετικών προτύπων (pattern) κίνησης, τη διαχείριση μεγάλου όγκου δεδομένων, και τη διαχείριση προβλημάτων που προκύπτουν από εφαρμογές που η επικοινωνία τους είναι κρυπτογραφημένη.

Γιατί όμως το πρόβλημα της ταξινόμησης της κίνησης σε δίκτυα επικοινωνιών έχει τόσο εξέχουσα σημασία ιδιαίτερα σήμερα; Η ικανότητα να εντοπίζονται δυναμικά και να ταξινομούνται οι ροές κίνησης δίνει πολύτιμες πληροφορίες για:

- Εποπτεία ασφάλειας: Για παράδειγμα, αποτρέπεται η διάδοση κακόβουλης κίνησης ενώ είναι δυνατή η αυτοματοποιημένη ανίχνευση εισβολών και αναγνωρίζονται πρότυπα κίνησης όπως επιθέσεις άρνησης υπηρεσίας (Denial of Service - DoS). Επίσης είναι δυνατό να προσδιοριστεί αν οι πελάτες παραβιάζουν κατά κάποιο τρόπο τους όρους του φορέα εκμετάλλευσης της υπηρεσίας.
- Διαχείριση λειτουργιών δικτύωσης [2]: έχοντας γνώση για την κίνηση στο δίκτυο είναι δυνατό να: α)επιτευχθεί καλύτερος σχεδιασμός και πρόβλεψη σύμφωνα με τις εκάστοτε ανάγκες για ποιότητα υπηρεσίας (Quality of Service - QoS). Έτσι μπορεί να δοθούν προτεραιότητες στις ροές κυκλοφορίας ή η διαχείρισή τους να γίνει με διαφορετικά κριτήρια αφού με την ταξινόμηση σε

κατηγορίες αναγνωρίζονται οι ροές που αντιστοιχούν σε κρίσιμες εφαρμογές, β)ενεργοποιηθούν διαφοροποιημένα τέλη ή συμβόλαια με τον χρήστη (Service Level Agreements -SLA), γ)πραγματοποιηθεί νόμιμη παρακολούθηση κίνησης (*Lawful Interception-LI*) για παράνομη κυκλοφορία σε περίπτωση που ένα κράτος απαιτεί να γνωρίζουν οι πάροχοι το περιεχόμενο που μεταδίδεται στα δίκτυα τους.

- Διαχείριση και εκχώρηση πόρων.

Αν και οι εφαρμογές ταξινόμησης κίνησης είναι πολλές, εντούτοις, οι ερευνητές, που εξετάζουν την ταξινόμηση κίνησης από την πλευρά της δημιουργίας ταξινομητή, έρχονται αντιμέτωποι με πολλά ζητήματα και προκλήσεις. Είναι γεγονός ότι οι χρήστες του δικτύου αυξήθηκαν εντυπωσιακά και νέες, πιο απαιτητικές, εφαρμογές έκαναν την εμφάνιση τους. Όμως παράλληλα αυξάνονται οι ρυθμοί μετάδοσης και η κίνηση στα δίκτυα, η οποία πλέον είναι ιδιαίτερα σύνθετη.

1.2. Ανασκόπηση Βιβλιογραφίας

Κατά τις δύο τελευταίες δεκαετίες το ενδιαφέρον της ερευνητικής κοινότητας για ταξινόμηση κίνησης συνεχώς εντείνεται.

Παραδοσιακά, οι τεχνικές ταξινόμησης κίνησης, γνωστές και ως port-based, βασίστηκαν στον έλεγχο της θύρας (well-known ports). Αυτό είχε επιτυχία, καθώς πολλές ευρέως χρησιμοποιούμενες εφαρμογές χρησιμοποιούν συγκεκριμένο αριθμό θύρας που ανατίθεται από την αρχή Internet Assigned Numbers Authority (IANA). Η διαδικασία ήταν σχετικά απλή: από την επικεφαλίδα του πακέτου εξαγόταν η συγκεκριμένη τιμή της θύρας και μέσω ενός πίνακα με τις γνωστές θύρες γινόταν η αντιστοίχιση σε συγκεκριμένη εφαρμογή. Καθώς όμως, σύγχρονες εφαρμογές αποφεύγουν να χρησιμοποιούν συγκεκριμένες θύρες (π.χ. P2P) είτε γιατί χρησιμοποιούν τυχαίους αριθμούς είτε γιατί "κρύβονται" πίσω από άλλα πρωτόκολλα, οι τεχνικές αυτές παύουν να είναι αξιόπιστες [1][2][3]. Παρά την ανακρίβεια η συγκεκριμένη τεχνική μπορεί να χρησιμοποιηθεί σε περιπτώσεις που απαιτείται παρακολούθηση κίνησης όχι ιδιαίτερα απαιτητική ως προς την ακρίβεια[1].

Για να ανταπεξέλθει στα προβλήματα της παραπάνω τεχνικής, η ερευνητική κοινότητα στράφηκε σε πιο εξελιγμένες προσεγγίσεις που κάνουν σε βάθος ανίχνευση πακέτων ελέγχοντας για το ταίριασμα με συγκεκριμένα πρότυπα (payload-based τεχνικές). Τα πρότυπα συνήθως αποτελούν σύνολο χαρακτηριστικών και θεωρούνται ως υπογραφές γνωστών εφαρμογών. Τα αποτελέσματα αυτών των τεχνικών είναι ικανοποιητικά ακόμη και για P2P μετάδοση δεδομένων. Όμως έρχονται αντιμέτωπες με πλήθος ζητημάτων όπως: α) η ιδιωτικότητα καθώς το περιεχόμενο του πακέτου μπορεί να γίνει ορατό στον ταξινομητή ή σε τρίτους [4], β) η κρυπτογράφηση ή η ενθυλάκωση πρωτοκόλλου δεν επιτρέπει την ταξινόμηση του πακέτου [1], γ) το κόστος επεξεργασίας και αποθήκευσης είναι υπολογίσιμο.

Τα ζητήματα αυτά οδήγησαν τους ερευνητές σε νέες προσεγγίσεις, όπου η ταξινόμηση της κίνησης γίνεται με την αναγνώριση στατιστικών μοτίβων (προτύπων), τα οποία είναι δυνατό να συλλεγούν από εξωτερικές παρατηρήσιμες ιδιότητες της κίνησης (π.χ. μήκος πακέτου). Σε αντίθεση με τις payload-based τεχνικές, οι συγκεκριμένες προσεγγίσεις μπορούν να χαρακτηριστούν ως lightweight ενώ χειρίζονται και κρυπτογραφημένη κίνηση. Το κύριο μειονέκτημα τους είναι η μειωμένη ακρίβεια.

Για την υλοποίηση τους χρησιμοποιούνται αλγόριθμοι από το πεδίο της αναγνώρισης προτύπων, χρησιμοποιώντας τεχνικές Μηχανικής Μάθησης. Η χρήση τους έχει αυξηθεί τα τελευταία χρόνια και γίνονται αρκετές ερευνητικές δημοσιεύσεις για την εφαρμογή των τεχνικών αυτών σε διαφόρους τύπους κίνησης. Τα ερευνητικά αποτελέσματα καταδεικνύουν την αποτελεσματικότητα των τεχνικών Μηχανικής Μάθησης, ακόμη και σε περιπτώσεις που οι μέθοδοι των δύο προηγούμενων κατηγοριών δεν ήταν ικανοποιητική. Μάλιστα σε μία σύγκριση μεθόδων και για τις τρεις κατηγορίες με κρυπτογραφημένη κίνηση, η ακρίβεια των τεχνικών που έκαναν χρήση Μηχανικής Μάθησης έφτασε το 98% [6].

Μια τεχνική Μηχανικής Μάθησης προϋποθέτει πρώτα την δημιουργία του μοντέλου και μετά την εφαρμογή του για ταξινόμηση [3]. Όταν δημιουργείται το μοντέλο εκπαιδεύεται με εμπειρικά δεδομένα πριν εφαρμοσθεί για ταξινόμηση ενός συνόλου δεδομένων.

Ο Nguyen [4], μεταξύ άλλων, διακρίνει τέσσερις κατηγορίες για την ταξινόμηση της κίνησης βασιζόμενες στη Μηχανική Μάθηση:

- Ομαδοποίησης: χρησιμοποιούν τεχνικές μάθησης χωρίς επίβλεψη και εφαρμόζουν αλγορίθμους που εντάσσονται σε αυτή την κατηγορία μάθησης (π.χ. Bayesian classifier, K-Means)
- Εποπτευόμενης Μάθησης: χρησιμοποιούν αλγορίθμους επιτηρούμενης μάθησης (π.χ. κοντινότερου γείτονα, Naïve Bayes)
- Υβριδικές: χρησιμοποιούν συνδυασμό τεχνικών μάθησης με επίβλεψη ή χωρίς επίβλεψη.
- Σύγκρισης: προσεγγίσεις που συγκρίνουν τεχνικές ταξινόμησης που βασίζονται είτε στη Μηχανική Μάθηση είτε στις άλλες κατηγορίες.

1.3. Αντικείμενο και διάρθρωση της εργασίας

Στην παρούσα εργασία μελετάτε το πρόβλημα της ταξινόμησης της κίνησης εστιάζοντας σε ασύρματο δικτυακό περιβάλλον και εφαρμόζοντας τεχνικές Μηχανικής Μάθησης.

Στόχος είναι να παρουσιάσουμε το σύνθετο πρόβλημα της ταξινόμησης κίνησης και να εξετάσουμε πιθανές λύσεις για ταξινόμησης κίνησης που δημιουργείται από δημοφιλείς δικτυακές εφαρμογές με την χρήση τεχνικών Μηχανικής Μάθησης σε ασύρματα δίκτυα. Για το σκοπό αυτό εφαρμόζονται οι τεχνικές Μηχανικής Μάθησης, Naive Bayes classifier και Support Vector Machines (SVMs) για την κατηγοριοποίηση δύο κλασικών υπηρεσιών διαμοίρασης περιεχομένου: η πρώτη (utorrent) χρησιμοποιεί το πρωτόκολλο BitTorrent για την μεταφορά των δεδομένων μέσω του διαδικτύου, η δεύτερη (You Tube) χρησιμοποιεί το πρωτόκολλο TCP με κρυπτογράφηση και ο χρήστης μπορεί να παρακολουθεί βίντεο μετά από αίτηση. Το ζητούμενο είναι κατά πόσο ο ταξινομητής θα μπορεί να αποφανθεί αν τα εισερχόμενα πακέτα ανήκουν σε μια συγκεκριμένη εφαρμογή ή όχι. Επίσης ελέγχεται η ακρίβεια και η αξιοπιστίας της κάθε μεθόδου.

Το υπόλοιπο αυτής της εργασίας διαρθρώνεται ως εξής:

Στο δεύτερο κεφάλαιο αναφέρεται στη Μηχανικής Μάθησης και παρουσιάζονται αναλυτικά οι τεχνικές Naive Bayes και Support Vector Machines που εφαρμόζονται στην συγκεκριμένη εργασία. Στο τρίτο κεφάλαιο περιγράφεται το πλαίσιο εργασίας που χρησιμοποιείται στο πρακτικό μέρος της εργασίας για την ταξινόμηση της κίνησης. Συγκεκριμένα παρουσιάζεται η διαδικασία συλλογής δεδομένων,

εκπαίδευσης και εξαγωγής συμπερασμάτων (ταξινόμηση). Επίσης, παρατίθενται τα αναλυτικά αποτελέσματα των πειραμάτων και αποτιμάται η ακρίβεια και η αξιοπιστία των μεθόδων ταξινόμησης, τόσο συγκριτικά μεταξύ των δύο τεχνικών όσο και γενικότερα. Τέλος, στο τέταρτο κεφάλαιο, αναφέρονται συνοπτικά τα συνολικά συμπεράσματα που εξήχθησαν και προτείνονται πιθανές βελτιώσεις και κατευθύνσεις μελλοντικής έρευνας.

ΚΕΦΑΛΑΙΟ 2. ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ (MACHINE LEARNING)

2.1. Βασικές Έννοιες

Η Μηχανική Μάθηση είναι ένας τομέας της τεχνητής νοημοσύνης, που ασχολείται με αλγορίθμους και μεθόδους που παρέχουν τη δυνατότητα σε μια μηχανή να "μαθαίνει", δηλαδή αποκτά την ικανότητά στην πρόσκτηση επιπλέον γνώσης κατά την αλληλεπίδρασή της με το περιβάλλον στο οποίο δραστηριοποιείται και την ικανότητά του να βελτιώνει με την επανάληψη τον τρόπο με τον οποίο εκτελεί μία ενέργεια.

Η χρήση της, καθιστά εφικτή την κατασκευή προσαρμόσιμων προγραμμάτων, τα οποία λειτουργούν με βάση την αυτοματοποιημένη ανάλυση δεδομένων και όχι τη διαίσθηση των μηχανικών που τα προγραμμάτισαν.

Ο Άρθουρ Σάμουελ (1959), θεωρεί τη Μηχανική Μάθηση πεδίο μελέτης το οποίο δίνει τη δυνατότητα στους υπολογιστές να μαθαίνουν χωρίς να έχουν προγραμματίσει.

Ένας πιο γενικός ορισμός δίνεται από τον Τομ Μ. Μίτσελ (1997), σύμφωνα με τον οποίο: «*Ένα πρόγραμμα υπολογιστή θεωρείται ότι μαθαίνει από μία εμπειρία Ε, σε σχέση με μία σειρά από έργα Τ και απόδοση μετρημένη με Ρ, αν η απόδοση του στα έργα Τ, μετρημένη με Ρ, βελτιώνεται με την εμπειρία Ε*».

Για παράδειγμα, το πρόβλημα της ταξινόμησης κίνησης θα μπορούσε να προσδιοριστεί σύμφωνα με τον παραπάνω ορισμό ως εξής:

- *Έργο Τ*: Η κατάταξη των πακέτων σε ένα προκαθορισμένο σύνολο (συγκεκριμένη εφαρμογή).
- *Μέτρο Απόδοσης Ρ*: Το ποσοστό των πακέτων που ταξινομούνται σωστά.
- *Εμπειρία Ε*: Ένα σύνολο από πακέτα με γνωστή κατηγοριοποίηση.

Οι αλγόριθμοι Μηχανικής Μάθησης κατηγοριοποιούνται σε τρεις, επικρατέστερες, κατηγορίες βάσει του αποτελέσματος τους. Οι κατηγορίες αυτές είναι:

- *Επιτηρούμενη μάθηση* (Supervised Learning), στην οποία ο αλγόριθμος κατασκευάζει μία συνάρτηση που απεικονίζει δεδομένες εισόδους σε γνωστές – επιθυμητές εξόδους (σύνολο εκπαίδευσης – training set), με στόχο τη γενίκευση της συνάρτησης και για εισόδους με άγνωστη έξοδο (σύνολο ελέγχου – test set).
- *Μη επιτηρούμενη μάθηση* (Unsupervised Learning), όπου το μοντέλο κατασκευάζεται για κάποιο σύνολο εισόδων, χωρίς όμως να είναι γνωστές οι επιθυμητές έξοδοι για το το σύνολο εκπαίδευσης.
- *Ενισχυτική μάθηση* (Reinforcement learning), στην οποία ο αλγόριθμος μαθαίνει μια στρατηγική ενεργειών για μια δεδομένη παρατήρηση.

2.2. Ταξινόμηση (Classification)

Η ταξινόμηση είναι η διαδικασία της αντιστοίχισης ενός αντικειμένου σε μία κατηγορία – κλάση, ανάμεσα σε διάφορες προκαθορισμένες. Τα δεδομένα εισόδου για την διαδικασία της ταξινόμησης είναι μία συλλογή από εγγραφές. Κάθε εγγραφή, γνωστή και ως δείγμα, αποτελείται από μία πλειάδα (tuple) της μορφής (x, y), όπου x είναι το σύνολο των γνωρισμάτων και y μία ετικέτα για την κλάση. Ουσιαστικά, η ταξινόμηση είναι η διαδικασία μάθησης μίας συνάρτησης-στόχου (f) ώστε να αντιστοιχεί κάθε σύνολο γνωρισμάτων x σε μία προκαθορισμένη κλάση με ετικέτα y, όπως φαίνεται και στο Σχήμα 2.1.

Η ταξινόμηση είναι διαδικασία δύο σταδίων:

- Κατασκευή Μοντέλου. Περιγράφεται το σύνολο των προκαθορισμένων κλάσεων (σύνολο εκπαίδευσης). Το μοντέλο αναπαρίσταται είτε ως κανόνες ταξινόμησης (classification rules), είτε ως δέντρο απόφασης (decision trees), είτε ως μαθηματική έκφραση (mathematical formula).

- Χρήση Μοντέλου. Το μοντέλο χρησιμοποιείται για μελλοντική κατάταξη. Πρώτα εκτιμάται η ακρίβεια του μοντέλου: γνωστά δείγματα (σύνολα ελέγχου) δίνονται προς ταξινόμηση και εξετάζεται το αποτέλεσμα. Αν το αποτέλεσμα είναι αποδεκτό το μοντέλο για την ταξινόμηση δεδομένων που η κατηγορίας τους δεν είναι γνωστή.

Σχήμα 2.1: Η ταξινόμηση ως διαδικασία αντιστοίχισης συνόλου γνωρισμάτων (x) σε κλάση ετικέτας y.

Η ταξινόμηση μπορεί να είναι είτε περιγραφική είτε προβλεπτική. Στην περιγραφική μορφή, το μοντέλο ταξινόμησης λειτουργεί ως εργαλείο για τη διάκριση αντικειμένων διαφορετικών κλάσεων. Στην προβλεπτική μορφή, το μοντέλο χρησιμοποιείται για να προβλέψει σε ποια κλάση ανήκουν άγνωστα δείγματα.
Η ταξινόμηση απαιτεί οι ετικέτες των τάξεων να έχουν διακριτά χαρακτηριστικά και για αυτό το λόγο είναι καταλληλότερη για πρόβλεψη ή περιγραφή συνόλων δεδομένων με δυαδικές ή ονομαστικές κλάσεις [5].

2.3. Μοντέλα Ταξινόμησης

Στην παρούσα εργασία χρησιμοποιήθηκαν δύο μοντέλα ταξινόμησης, ο Naïve Bayes και οι Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machines - SVM). Στη συνέχεια ακολουθεί η παρουσίαση και ανάλυση τους.

2.3.1. Ταξινομητής Naïve Bayes

Το Θεώρημα Bayes

Αν X είναι ένα σύνολο χαρακτηριστικών και Y η κλάση και αν υπάρχει μια μη-ντετερμινιστική σχέση μεταξύ αυτών των δύο, τότε μπορούμε να τις θεωρήσουμε σαν τυχαίες μεταβλητές και να υπολογιστεί πιθανοτικά η σχέση τους με τη χρήση της:

$$P(Y|X) = \frac{P(X|Y) * P(Y)}{P(X)}$$

Η υπό συνθήκη πιθανότητα P(Y|X) είναι γνωστή και ως εκ των υστέρων πιθανότητα (posterior probability) του Y και η $P(Y)$ είναι γνωστή ως εκ των προτέρων πιθανότητα.

Naïve Bayes

Οι ταξινομητές Naïve Bayes είναι μια γενική κατηγορία πιθανοτικών ταξινομητών, που βασίζονται στην εφαρμογή του θεωρήματος Bayes, θεωρώντας ότι υπάρχει ισχυρή ανεξαρτησία μεταξύ των χαρακτηριστικών. Η ταξινόμηση των δεδομένων γίνεται σε δύο βήματα:

- *Εκπαίδευση*: χρησιμοποιούνται τα δεδομένα εκπαίδευσης (training set) έτσι ώστε η μέθοδος να εκτιμήσει τις παραμέτρους της πιθανοτικής κατανομής, υποθέτοντας πως οι προβλέψεις είναι ανεξάρτητες, δεδομένης της κλάσης.
- *Πρόβλεψη*: για κάθε νέο δεδομένο – τεστ υπολογίζεται η εκ των υστέρων πιθανότητα του να ανήκει σε κάθε κλάση. Στη συνέχεια το δείγμα ταξινομείται στην κλάση με τη μεγαλύτερη εκ των υστέρων πιθανότητα.

Ένα από τα σημαντικά πλεονεκτήματά του είναι πως χρειάζεται λίγα δεδομένα εκπαίδευσης για να υπολογίσει τις απαιτούμενες, για την ταξινόμηση, παραμέτρους.

Έστω X είναι ένα σύνολο χαρακτηριστικών και Y η κλάση. Στο πλαίσιο της εκπαίδευσης του ταξινομητή, χρειάζεται να μάθουμε όλες τις εκ των υστέρων πιθανότητες P(Y|X) για κάθε συνδυασμό μεταξύ X, Y βασιζόμενοι στα δεδομένα εκπαίδευσης. Γνωρίζοντας αυτές τις πιθανότητες, ένα συνόλο ελέγχου X' μπορεί να ταξινομηθεί αν βρεθεί η κλάση Y' η οποία μεγιστοποιεί την εκ των υστέρων πιθανότητα P(Y'|X'). Το θεώρημα Bayes είναι χρήσιμο, γιατί επιτρέπει τον υπολογισμό της εκ των υστέρων πιθανότητας με τη χρήση της εκ των προτέρων πιθανότητας $P(Y)$, της υπό συνθήκης πιθανότητας $P(X|Y)$ και της γνωστής πιθανότητας $P(X)$.

$$P(Y|X) = \frac{P(X|Y)*P(Y)}{P(X)}.$$

Στη σύγκριση των εκ των υστέρων πιθανοτήτων για διαφορετικές τιμές του Y, ο παρονομαστής $P(X)$, είναι σταθερός και μπορεί να αγνοηθεί. Η εκ των προτέρων πιθανότητα $P(Y)$, μπορεί εύκολα να υπολογισθεί από το σύνολο εκπαίδευσης αν υπολογιστεί το ποσοστό των εκπαιδευόμενων εγγραφών που αντιστοιχεί σε κάθε κλάση. Για τον υπολογισμό της υπό συνθήκης πιθανότητας $P(X|Y)$, ο Naïve Bayes υποθέτει πως τα χαρακτηριστικά είναι υπό συνθήκη ανεξάρτητα μεταξύ τους, δεδομένης της ετικέτας της κλάσης y. Η ανεξαρτησία δίνεται από τον τύπο:

$$P(X|Y = y) = \prod_{i=1}^{d} P(X_i|Y = y),$$

όπου κάθε σύνολο χαρακτηριστικών $X = \{X_1, X_2, \dots X_d\}$, αποτελείται από d χαρακτηριστικά.

Με βάση τα παραπάνω, για να κάνει την ταξινόμηση υπολογίζει την εκ των υστέρων πιθανότητα για κάθε κλάση Y ως εξής:

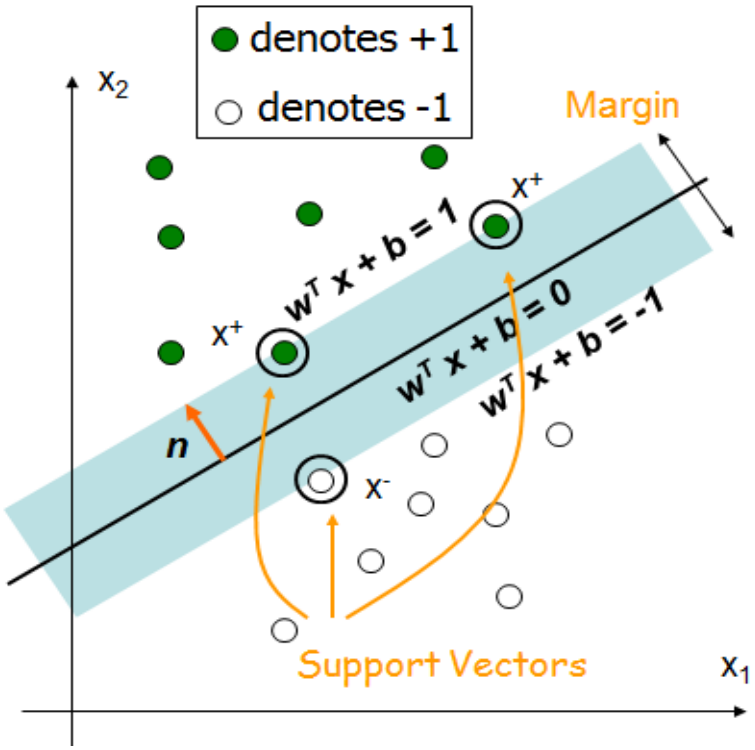
$$P(Y|X) = \frac{P(Y)\prod_{i=1}^{d} P(X_i|Y)}{P(X)}.$$

Η $P(X)$ είναι ίδια για όλες τις κλάσεις Y, οπότε είναι αρκετό να επιλεχθεί η κλάση στην οποία ο όρος του αριθμητή είναι μέγιστος [5].

2.3.2. Support Vector Machines-SVM

Οι Μηχανές Διανυσμάτων Υποστήριξης (SVM), είναι μη πιθανοτικοί, δυαδικοί, γραμμικοί ταξινομητές, καθώς δεδομένου ενός συνόλου εκπαίδευσης, το οποίου τα δείγματα ανήκουν είτε στη μία είτε στην άλλη κλάση, δημιουργείται ένα μοντέλο που ταξινομεί τα δεδομένα είτε στη μία είτε στην άλλη κατηγορία. Αποτελεί την αναπαράσταση των δειγμάτων σαν σημεία στο χώρο, τα οποία είναι χωρισμένα με τέτοιο τρόπο ώστε να υπάρχει ένα ευδιάκριτο όριο μεταξύ τους, το οποίο είναι όσο το δυνατόν πλατύτερο γίνεται. Δηλαδή, η ταξινόμηση γίνεται αφού βρεθεί το καλύτερο υπερ-επίπεδο που θα διαχωρίζει την μία κλάση από την άλλη, με το μεγαλύτερο δυνατό όριο που συνεπάγεται και μικρότερο λάθος για τον ταξινομητή. Μπορούν να κάνουν και μη γραμμική ταξινόμηση αρκεί να αντιστοιχίσουν τα δεδομένα τους σε χώρους μεγαλύτερων διαστάσεων.

Υποθέτοντας πως έχουμε δύο κλάσεις, τις 1 και -1, με τα – και + να είναι αντίστοιχα δείγματα της κάθε κλάσης. Στο Σχήμα 2.2 βλέπουμε ένα παράδειγμα ενός SVM.



Σχήμα 2.2: Support Vector Machine για κλάσεις 1 και -1. (Πηγή: Jinwei Gu, "An Introduction of Support Vector Machine", 2008)

Μαθηματικός Ορισμός: Βασικός

Τα δεδομένα του συνόλου εκπαίδευσης είναι ένα σύνολο από σημεία (διανύσματα) x_i με κατηγορίες y_i. Για μία διάσταση d, το $x_i \in R^d$ και τα $y_i = \pm 1$. Η εξίσωση του υπερ-επιπέδου είναι:

$$< w, x > + b = 0, < w, x > + b = 0$$

όπου το $w \in R^d$, το <w,x> είναι το εσωτερικό γινόμενο των w, x και το b ένας πραγματικός αριθμός.

Το βέλτιστο διαχωριστικό υπερ-επίπεδο, ορίζεται από το εξής πρόβλημα: να βρεθεί το w το οποίο ελαχιστοποιεί το ||w||, έτσι ώστε για όλα τα σημεία (x_i, y_i) να ισχύει:

$$y_i(< w, x_i > + b) \geq 1.$$

Τα support vectors είναι τα x_i για τα οποία $y_i(< w, x_i > + b) = 1$.

Για μαθηματική ευκολία, το πρόβλημα θωρείται ισοδύναμο με την ελαχιστοποίηση του $\frac{<w,w>}{2}$, το οποίο είναι πρόβλημα τετραγωνικού προγραμματισμού.

Μαθηματικός Ορισμός: Δϋικός

Η δϋική μορφή του προβλήματος, χρησιμοποιεί τους θετικούς συντελεστές Lagrange, της μορφής α_i, πολλαπλασιασμένους με τους περιορισμούς και στη συνέχεια αφαιρεί το αποτέλεσμα από την κύρια συνάρτηση:

$$L_P = \frac{1}{2} < w, w > - \sum_i a_i \, (y_i(< w, x_i > + b) - 1).$$

Το ζητούμενο είναι η εύρεση ενός σταθερού L_p σε σχέση με τα w και b. Αν θέσουμε το $L_P = 0$, τότε:

$$w = \sum_i a_i y_i x_i \qquad \text{Εξίσωση 2.1}$$

$$0 = \sum_i a_i y_i$$

$$w = \sum_i a_i y_i x_i$$

Αντικαθιστώντας τα παραπάνω στο L_p, παίρνουμε το δϋικό L_D:

$$L_D = \sum_i a_i - ½ \sum_i \sum_j a_i a_j y_i y_j < x_i x_j >,$$

και μεγιστοποιούμε για $\alpha_i \geq 0$. Συνήθως, τα περισσότερα α_i είναι 0 στο μέγιστό τους. Το μη μηδενικό είναι και η λύση του δϋικού προβλήματος που ορίζει το υπερ-επίπεδο, όπως φαίνεται και στην Εξίσωση 2.1, που δίνει το w ως άθροισμα των $a_i y_i x_i$. Τα σημεία – δεδομένα x_i που αντιστοιχούν στο μη μηδενικό α_i είναι τα support vectors.

Παραγωγίζοντας την L_D ως προς ένα μη μηδενικό α_i είναι το βέλτιστο το οποίο και μας δίνει:

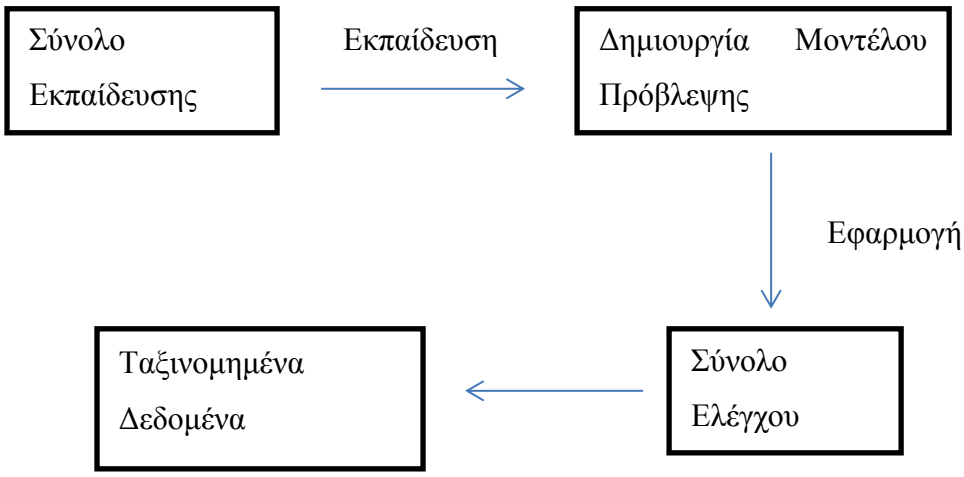
$$y_i(< w, x_i > + \, b) - 1 = 0 \, .$$

Συγκεκριμένα, αυτό μας δίνει την τιμή του b στη λύση, επιλέγοντας οποιοδήποτε i με μη μηδενικό α_i [7].

ΚΕΦΑΛΑΙΟ 3. ΠΕΙΡΑΜΑΤΙΚΗ ΑΞΙΟΛΟΓΗΣΗ

3.1. Παρουσίαση Πειραμάτων

Στην παρούσα εργασία χρησιμοποιήθηκαν δύο μοντέλα ταξινόμησης, ο Naïve Bayes και τα Support Vector Machines. Στο Σχήμα 3.1 παρουσιάζεται το μοντέλο που υλοποιήθηκε και χρησιμοποιήθηκε και για τους δύο ταξινομητές. Σε πρώτο στάδιο, χρησιμοποιώντας το σύνολο εκπαίδευσης, δημιουργείται ένα μοντέλο πρόβλεψης. Σε δεύτερο στάδιο στο μοντέλο αυτό εφαρμόζεται το σύνολο ελέγχου για να ταξινομήσει τα δεδομένα.



Σχήμα 3.1: Μοντέλο Υλοποίησης Ταξινομητών

Τα μοντέλα εφαρμόστηκαν για την διάκριση των κλάσεων BitTorrent και You Tube. Τα πειράματα έγιναν σε Matlab R2014a. Τα σύνολα εκπαίδευσης και ελέγχου καθώς και οι συνθήκες των πειραμάτων παρουσιάζονται αναλυτικά παρακάτω.

3.2. Σύνολα Εκπαίδευσης και Σύνολα Ελέγχου

Για την υλοποίηση των ταξινομητών, χρησιμοποιήθηκαν δύο διαφορετικά σύνολα εκπαίδευσης και δύο διαφορετικά σύνολα ελέγχου. Ένα σύνολο ελέγχου και ένα εκπαίδευσης για τον Naïve Bayes ταξινομητή και αντίστοιχα για τα SVM.

Από το σύνολο των δεδομένων, το ενδιαφέρον μας εστιάζεται στις εφαρμογές διαμοίρασης περιεχομένου που χρησιμοποιούν το πρωτόκολλο BitTorrent ή τον δικτυακό τόπο You Tube. Έτσι ορίζουμε δύο κλάσεις, τις BitTorrent και You Tube στις οποίες θέλουμε να ταξινομήσουμε τα δεδομένα που συλλέγονται. Για την κλάση You Tube, εκτελέστηκε και μία δεύτερη σειρά πειραμάτων στην οποία λήφθηκε υπόψη η διαφορά μεταξύ του χρόνου άφιξης των πακέτων αντί για τον κανονικό χρόνο άφιξης των πακέτων.

Τα σύνολα των πακέτων, είναι δεδομένα κίνησης ενός ασύρματου δικτύου ταχύτητας 4Mbps που συλλέχθηκαν με την χρήση του εργαλείου Wireshark [10]. Το Wireshark είναι ένα εργαλείο για την καταγραφή της εισερχόμενης / εξερχόμενης κίνησης μέσω μιας διεπαφής (π.χ. Ethernet κάρτα δικτύου) σε πραγματικό χρόνο, με σκοπό την περαιτέρω ανάλυση των πακέτων.

Τα δεδομένα είναι πακέτα, και για κάθε πακέτο συλλέγονται τα ακόλουθα χαρακτηριστικά: ο χρόνος άφιξης, η διεύθυνση πηγής (source ip), η διεύθυνση προορισμού (destination ip), το πρωτόκολλο, το μέγεθος του πακέτου και πληροφορίες σχετικές με το κάθε πακέτο.

Όλα τα δεδομένα έχουν την ίδια μορφή και αποθηκεύονται σε πίνακα. Κάθε γραμμή του πίνακα αντιστοιχεί σε ένα διαφορετικό πακέτο, και κάθε στήλη αντιστοιχεί σε διαφορετικό χαρακτηριστικό. Πιο συγκεκριμένα, η πρώτη στήλη είναι ο αύξων αριθμός, η δεύτερη ο χρόνος άφιξης, η τρίτη η διεύθυνση πηγής, η τέταρτη η διεύθυνση προορισμού, η πέμπτη το πρωτόκολλο που χρησιμοποιείται, η έκτη το μέγεθος του πακέτου και η έβδομη περιέχει πληροφορίες για το κάθε πακέτο.

Για να είναι δυνατή η επεξεργασία των δεδομένων αυτών, χρειάστηκε να γίνει η μετατροπή των μη αριθμητικών δεδομένων σε αριθμητικά. Για την μετατροπή αυτή

χρησιμοποιήθηκαν οι συναρτήσεις *categorical* και *double* του Matlab. Παρακάτω, φαίνεται μία ενδεικτική χρήση των συναρτήσεων για τη μετατροπή των διευθύνσεων πηγής και προορισμού, του πρωτοκόλλου και των πληροφοριών έτσι ώστε να γίνει στη συνέχεια η εκπαίδευση του ταξινομητή.

```
% Make non numeric attributes numeric
source_ip = categorical(C{3});
numericSource_ip = double(source_ip);

dest_ip = categorical(C{4});
numericDest_ip = double(dest_ip);

protocol = categorical(C{5});
numericProtocol = double(protocol);

info = categorical(C{7});
numericInfo = double(info);
```

Τα συλλεγόμενα δεδομένα χρησιμοποιούνται είτε ως σύνολα εκπαίδευσης είτε ως σύνολα ελέγχου. Τα σύνολα εκπαίδευσης χρησιμοποιούνται έτσι ώστε να μπορεί ο ταξινομητής να δημιουργήσει το μοντέλο που θα ταξινομεί τα σύνολα ελέγχου. Τα σύνολα ελέγχου χρησιμοποιούνται για τον έλεγχο της αποτελεσματικότητας των ταξινομητών και ουσιαστικά, αποτελούν τα δεδομένα τα οποία θέλουμε να ταξινομήσουμε και να μελετήσουμε.

3.3. Naive Bayes

Για τον *Naïve Bayes* ταξινομητή χρησιμοποιήθηκαν:

- *Σύνολα Εκπαίδευσης*: Αποτελούνται από δεδομένα μίας μόνο κλάσης, αυτής που ελέγχεται κάθε φορά (BitTorrent / You Tube). Τα δεδομένα είναι λιγότερα σε σχέση με αυτά που χρησιμοποιήθηκαν στα SVMs, λόγω της ιδιότητας του αλγορίθμου να μην χρειάζεται πολλά δεδομένα για εκπαίδευση. Αποτελούνται από 5158 πακέτα για την κίνηση που αφορά το BitTorrent και από 4270 πακέτα για την κίνηση του You Tube.
- *Σύνολα Ελέγχου:* Είναι δεδομένα που δεν έχουν υποστεί «φιλτράρισμα» και περιλαμβάνουν διάφορα πακέτα από διάφορες εφαρμογές. Για τον έλεγχο του

BitTorrent τα πακέτα είναι 10000 και 20000 αντίστοιχα και για του You Tube 53781 και 63336 πακέτα.

Μετά τη μετατροπή των δεδομένων σε αριθμητικά, χρησιμοποιήθηκε η συνάρτηση fitNaiveBayes(), όπως φαίνεται παρακάτω, για να γίνει η εκπαίδευση του ταξινομητή.

```
% Classifier's training
NBmodel = fitNaiveBayes(train_mat,class, 'Distribution', 'kernel');
```

Χρησιμοποιώντας τη συνάρτηση predict() με ορίσματα τα δεδομένα ελέγχου και το αποτέλεσμα της εκπαίδευσης, γίνεται η ταξινόμηση του συνόλου ελέγχου:

```
% Classifier's testing
label = predict(NBmodel, test_mat_1);
```

3.4. Support Vector Machines

Για τον *SVM* ταξινομητή χρησιμοποιήθηκαν:

- *Σύνολα Εκπαίδευσης:* Έχουν δεδομένα από διαφορετικές εφαρμογές, τα οποία και αντιστοιχούν σε δύο κλάσεις, αυτή που μας ενδιαφέρει κάθε φορά και μία ακόμη. Γι'αυτό το λόγο χρειάζονται πιο πολλά δεδομένα για την καλύτερη εκπαίδευση του ταξινομητή. Αποτελούνται από 6758 πακέτα για την εκπαίδευση του BitTorrent και από 7500 για αυτή του You Tube (για την περίπτωση του YouTube που λαμβάνεται υπόψη η διαφορά του χρόνου άφιξης των πακέτων – inter arrival time, τα πακέτα είναι 6500).
- *Σύνολα Ελέγχου:* Όπως και τα δεδομένα ελέγχου του Naïve Bayes, έτσι και εδώ, τα δεδομένα ελέγχου είναι δεδομένα από την παρακολούθηση της γενικής κίνησης σε ένα ασύρματο δίκτυο τα οποία δεν έχουν υποστεί «φιλτράρισμα». Για το BitTorrent αποτελούνται 10000 και 20000 πακέτα και για το You Tube 20000 και 30000. (Λόγω περιορισμών στη μνήμη ο έλεγχος στα SVM έγινε σε λιγότερα δεδομένα.)

Για την εκπαίδευση του ταξινομητή, αφού είχαν γίνει οι απαραίτητες μετατροπές στα δεδομένα, χρησιμοποιήθηκε η συνάρτηση fitcsvm(), όπως φαίνεται πιο κάτω:

```
% Classifier's training
SVMModel = fitcsvm(SVM_mat,SVMclass,'Standardize',true);
```

Χρησιμοποιούμε τη συνάρτηση predict() στον εκπαιδευμένο ταξινομητή με ορίσματα τα δεδομένα ελέγχου για την ταξινόμηση τους.

```
% Classifier's testing
SVMLabel = predict(SVMModel, SVMtest_mat_1);
```

Κατά τον ίδιο τρόπο γίνεται η εφαρμογή του ταξινομητή και στο δεύτερο σύνολο δεδομένων.

3.5. Αξιολόγηση Αποτελεσμάτων

Για την *αποτελεσματικότητα* και την ακρίβεια των ταξινομητών, χρησιμοποιήθηκε η εκ των υστέρων πιθανότητα (posterior probability) στον Naïve Bayes και το λάθος ταξινόμησης στα SVM (resubstitution loss / loss).

Η εκ των υστέρων πιθανότητα (posterior probability) δηλώνει την πιθανότητα να ανατεθεί μία παρατήρηση i στην κλάση j. Η συνάρτηση posterior(), επιστρέφει απλά έναν πίνακα στήλη με τις πιθανότητες ένα δείγμα να ανήκει στην αντίστοιχη κλάση και NaN (Not a Number) σε περίπτωση που αντιστοιχεί σε άδεια κλάση.

Το *λάθος ταξινόμησης* είναι ένα μέτρο για το πόσο καλά ταξινομεί τα δεδομένα ο ταξινομητής. Εφαρμόζονται δύο συναρτήσεις υπολογισμού του λάθους ταξινόμησης που έχουν να κάνουν με το ποσοστό λάθους στην εκπαίδευση (συνάρτηση resubLoss) και το ποσοστό λάθους στην ταξινόμηση του συνόλου ελέγχου (συνάρτηση loss).

Η resubLoss χρησιμοποιεί για τον υπολογισμό του λάθους το μέσο τετραγωνικό σφάλμα (mean squared error - mse):

$$mse = \frac{\sum_{j=1}^{n} w_j (f(x_j) - y_j)^2}{\sum_{j=1}^{n} w_j},$$

όπου:

- n, το πλήθος των γραμμών των δεδομένων,
- x_j, η j-οστή σειρά των δεδομένων,
- y_j, η πραγματική απάντηση για το x_j,
- $f(x_j)$, η απάντηση-πρόβλεψη του SVM για το x_j
- w, ένας πίνακας με βάρη, τα οποία εξ' ορισμού είναι ίσα με τη μονάδα [8].

Αντίστοιχα, η loss χρησιμοποιεί το λάθος ταξινόμησης (classification error):

$$L = \frac{\sum_{j=1}^{n} w_j e_j}{\sum_{j=1}^{n} w_j},$$

όπου:

- w_j, το βάρος της παρατήρησης j. Το λογισμικό τα κανονικοποιεί ώστε να αθροίζουν στη μονάδα.
- $e_j = 1$, αν η προβλεπόμενη κλάση j διαφέρει από την πραγματική κλάση και 0 αλλιώς.

Ουσιαστικά, το λάθος ταξινόμησης είναι το ποσοστό των παρατηρήσεων, οι οποίες ταξινομούνται λανθασμένα [1].

Εκτελώντας τα πειράματα για τα SVM, παρατηρήθηκαν τα παρακάτω λάθη:

Για την κλάση BitTorrent το λάθος στην εκπαίδευση ήταν ίσο με L_ = 1.4810e-04, ενώ και στα δύο σύνολα ελέγχου το λάθος ταξινόμησης ήταν ίσο με 0. Να σημειωθεί πως παίζει ρόλο στον υπολογισμό του σφάλματος το αν θα συμπεριληφθούν όλα τα πρωτόκολλα που συγκαταλλέγονται στις περιγραφές των πακέτων στην συνάρτηση categorical(), καθώς αν κάποιο δεν υπάρχει, θα αντιστοιχηθεί σε BitTorrent και κατά συνέπεια θα ταξινομηθεί λάθος.

Για την κλάση You Tube, το λάθος ταξινόμησης στην εκπαίδευση είναι ίσο με L_youTInAr = 0.3477, ενώ στα δύο σύνολα ελέγχου είναι LyouTinAr = 0.3109 και LyouTinAr_2 = 0.3426 αντίστοιχα. Τα παραπάνω, σημαίνουν πως υπάρχει η πιθανότητα να γίνει λάθος στην εκπαίδευση σε ποσοστό 34% και στα σύνολα ελέγχου, οι πιθανότητες να ταξινομηθούν λάθος τα δεδομένα είναι 31% και 34% αντίστοιχα.

Τέλος, αν στην κλάση YouTube λάβουμε υπόψη την διαφορά μεταξύ του χρόνου άφιξης των πακέτων, παρατηρούμε το εξής λάθος στην εκπαίδευση L_youT = 0.0612 και τα LyouT = 0.1102 και LyouT_2 = 0.3071 στα δύο σύνολα ελέγχου. Σε σύγκριση με τα παραπάνω πειράματα, βλέπουμε πως αν αντί για τον κανονικό χρόνο άφιξης των πακέτων χρησιμοποιήσουμε τη διαφορά μεταξύ των χρόνων άφιξης, τότε το λάθος στην εκπαίδευση είναι μόνο 6%, στο πρώτο σύνολο ελέγχου είναι 11% αλλά στο δεύτερο, το ποσοστό ανέρχεται σε 30%. Τα δύο πρώτα ποσοστά είναι εμφανώς μικρότερα σε σχέση με αυτά του προηγούμενου πειράματος, το δεύτερο όμως αν και μικρότερο δεν έχει την ίδια διαφορά.

Στους παρακάτω πίνακες, παρουσιάζονται συνοπτικά τα λάθη. Πιο συγκεκριμένα, ο Πίνακας 3.1 περιλαμβάνει το λάθος στην εκπαίδευση, ενώ οι Πίνακας 3.2 και Πίνακας 3.3 τα λάθη στα σύνολα ελέγχου 1 και 2.

Πίνακας 3.1: Λάθη Εκπαίδευσης

Κλάση	Λάθος
BitTorrent	1.4810e-04
You Tube	0.3477
You Tube (inter arrival time)	0.0612

Πίνακας 3.2: Λάθη Ταξινόμησης Πρώτου Συνόλου Ελέγχου

Κλάση	Λάθος
BiTorrent	0
You Tube	0.3109
You Tube (inter arrival time)	0.1102

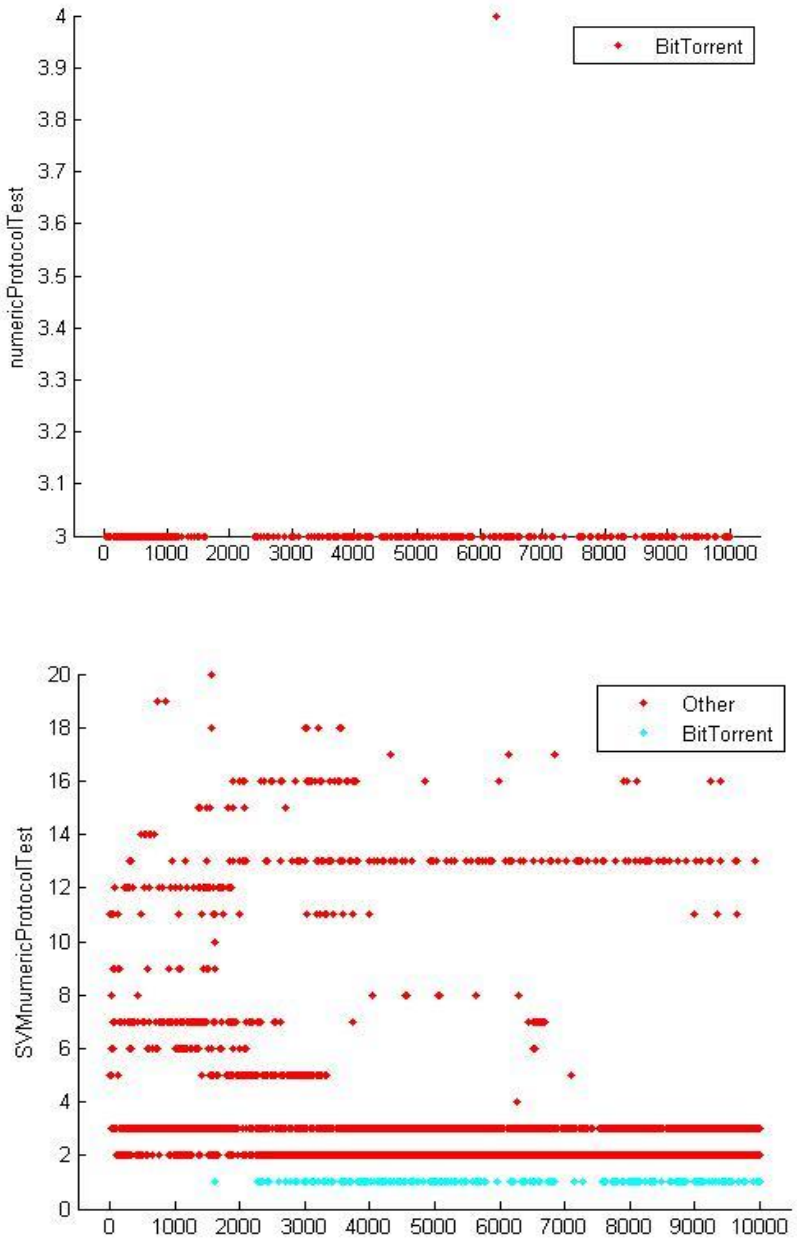
Πίνακας 3.3: Λάθη Ταξινόμησης Δεύτερου Συνόλου Ελέγχου

Κλάση	Λάθος
BitTorrent	0
You Tube	0.3426
You Tube (inter arrival time)	0.3071

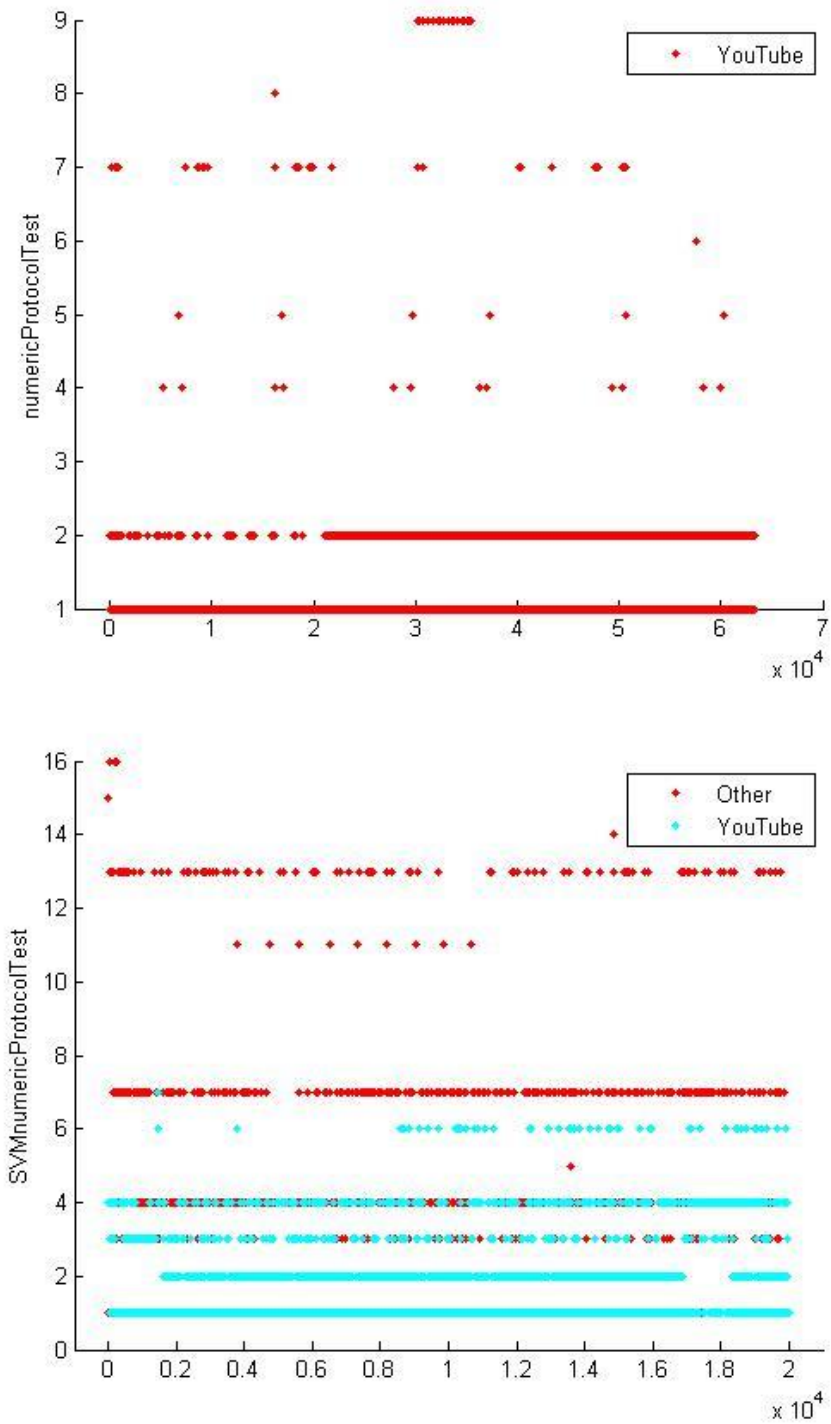
Στη συνέχεια, ακολουθούν γραφικές παραστάσεις που απεικονίζουν τα αποτελέσματα των δύο ταξινομητών (Naïve Bayes, SVM) στα δύο σύνολα ελέγχου. Στον άξονα των

x αντιστοιχίζεται ο αύξων αριθμός του πακέτου και στον άξονα y αντιστοιχίζεται το πρωτόκολλο (σε αριθμητική μορφή).

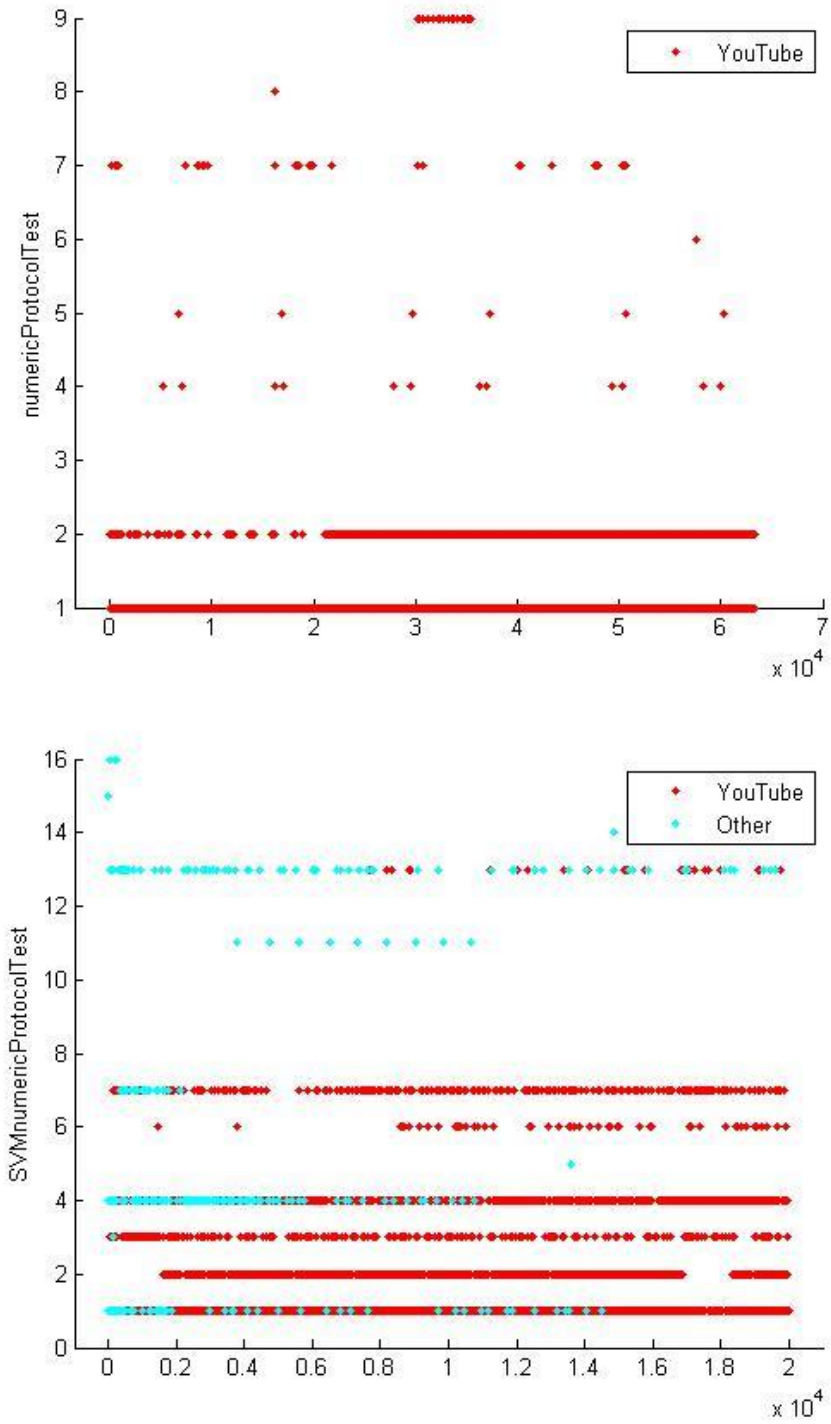
Τα σχήματα Σχήμα 3.2, Σχήμα 3.3, Σχήμα 3.4 δείχνουν τις γραφικές παραστάσεις και των δύο ταξινομητών για το πρώτο σύνολο ελέγχου.



Σχήμα 3.2: Αποτελέσματα Naive Bayes και SVM για την κλάση BitTorrent

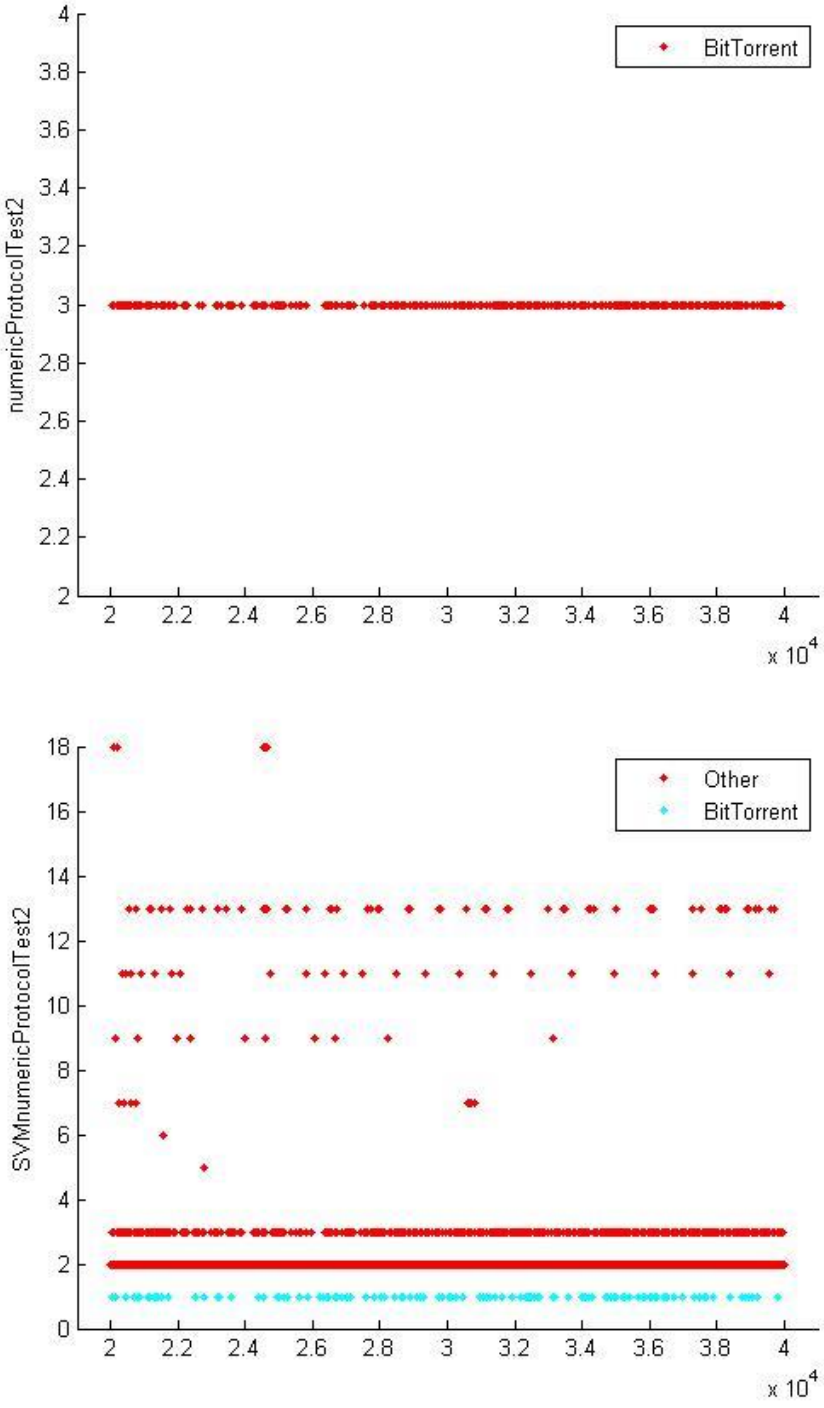


Σχήμα 3.3: Αποτελέσματα Naive Bayes και SVM για την κλάση YouTube

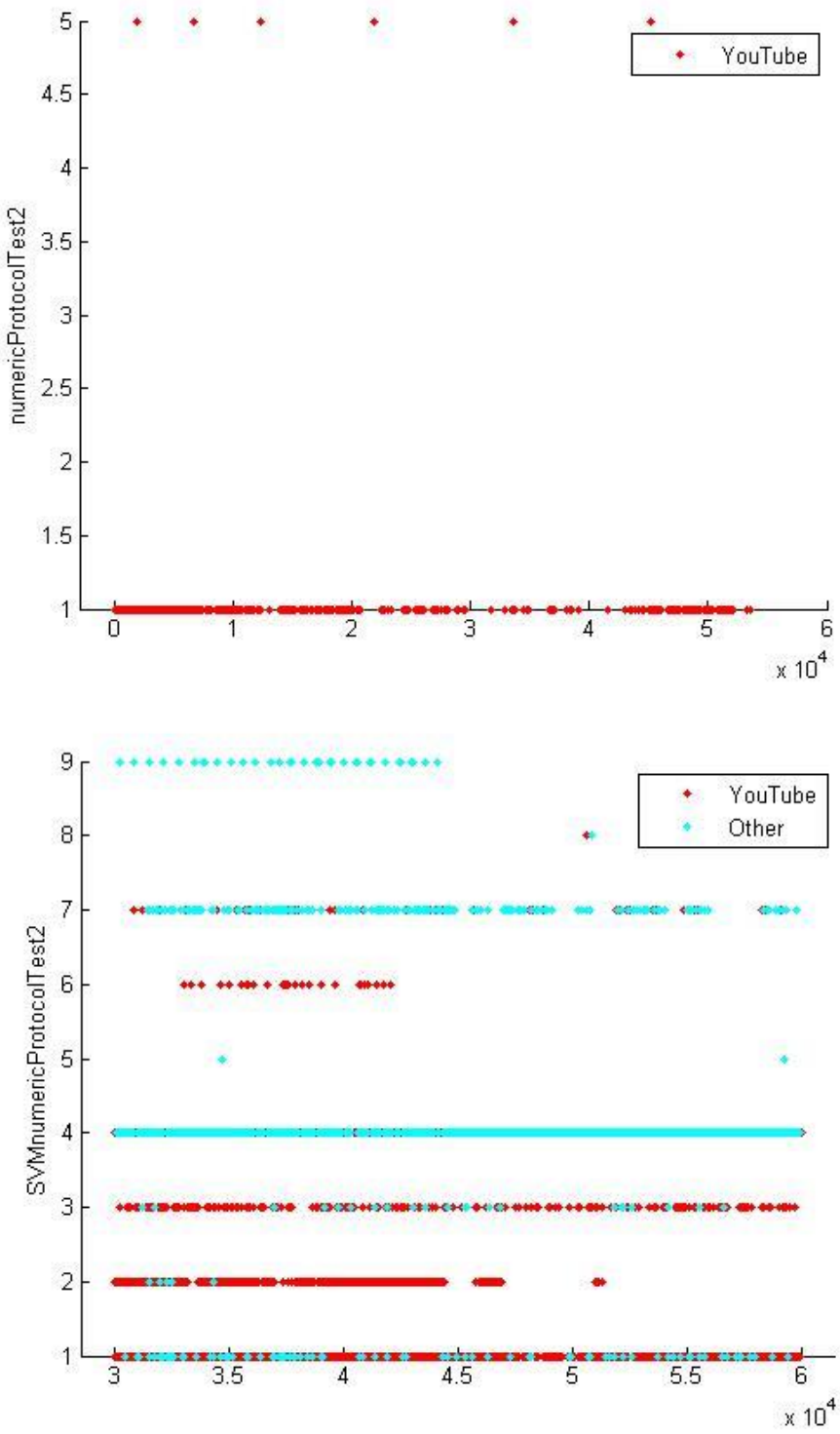


Σχήμα 3.4: Αποτελέσματα Naive Bayes και SVM για την κλάση You Tube (inter arrival time)

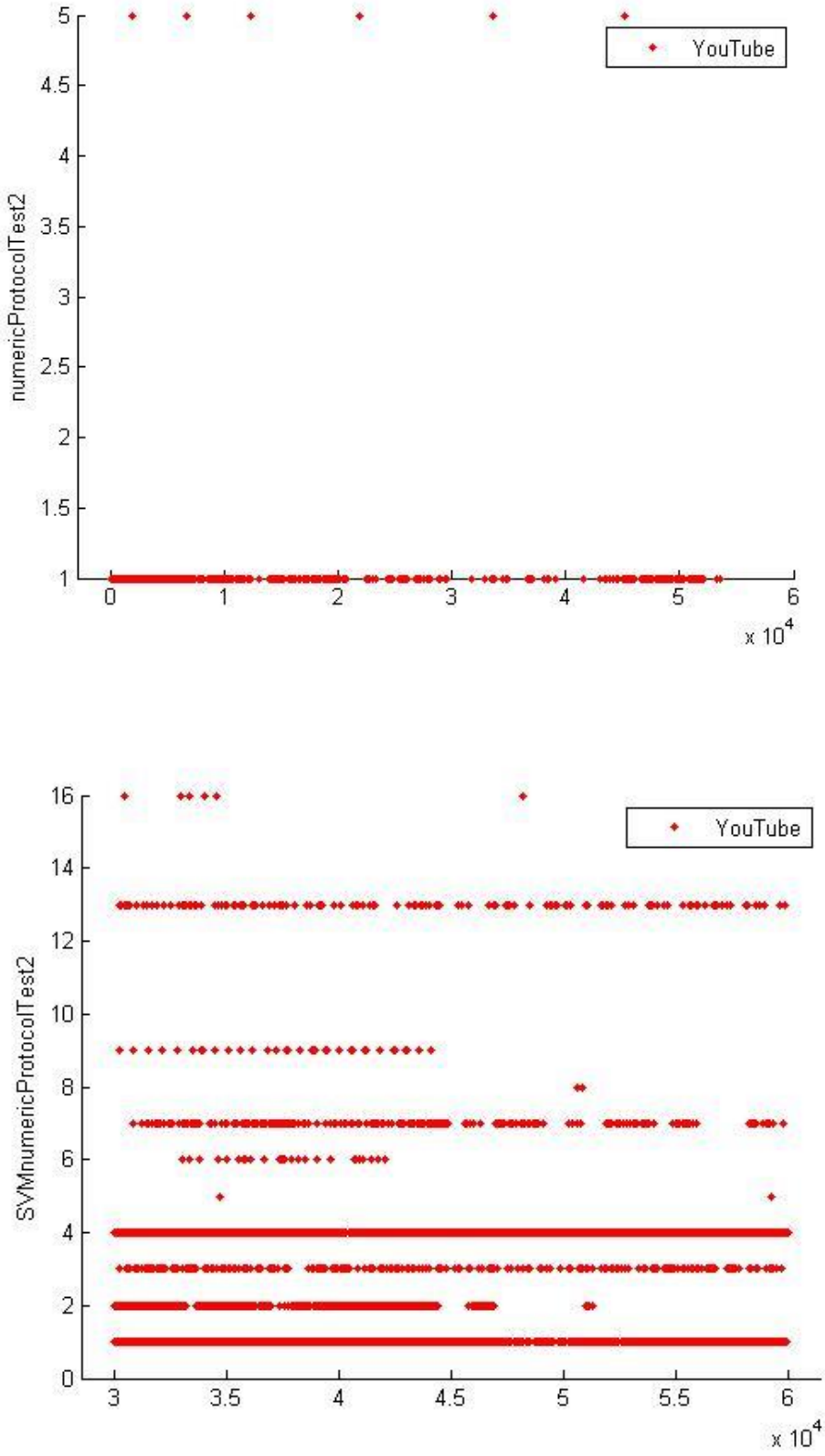
Αντίστοιχα, τα σχήματα Σχήμα 3.5, Σχήμα 3.6, Σχήμα 3.7 δείχνουν τις γραφικές παραστάσεις των δύο ταξινομητών για το δεύτερο σύνολο ελέγχου.



Σχήμα 3.5: Αποτελέσματα Naive Bayes και SVM για την κλάση BitTorrent



Σχήμα 3.6: Αποτελέσματα Naive Bayes και SVM για την κλάση You Tube



Σχήμα 3.7: Αποτελέσματα Naive Bayes και SVM για την κλάση You Tube (inter arrival time)

Στο σύνολο των γραφικών παραστάσεων, παρατηρείται πως γενικά τα SVM έχουν καλύτερα αποτελέσματα από τον Naïve Bayes. Παρόλα αυτά, ειδικότερα στην κλάση You Tube υπάρχουν πολλά πακέτα που ταξινομούνται λάθος. Αυτό οφείλεται στο γεγονός ότι η κίνηση από το You Tube δεν μπορεί να αναπαρασταθεί με την ίδια ευκολία, πρώτον γιατί χρησιμοποιεί 2 είδη πρωτοκόλλων (TCP και TLSv1.2) και δεύτερον γιατί αυτά τα πρωτόκολλα χρησιμοποιούνται και από αρκετές άλλες εφαρμογές.

ΚΕΦΑΛΑΙΟ 4. ΣΥΜΠΕΡΑΣΜΑΤΑ – ΠΙΘΑΝΕΣ ΠΡΟΕΚΤΑΣΕΙΣ

4.1. Συμπεράσματα Έρευνας

Έχοντας ολοκληρώσει τα πειράματα των δύο ταξινομητών και για τα δύο σύνολα ελέγχου, είναι προφανές πως τα SVM έχουν καλύτερη απόδοση (ταξινομούν μεγαλύτερο ποσοστό δεδομένων σωστά) συγκριτικά με τον Naive Bayes, για τα συγκεκριμένα δεδομένα. Σημαντικό ρόλο παίζει ο τρόπος λειτουργίας τους, καθώς χρειάζονται δεδομένα δύο κλάσεων (αυτής που θέλουμε κάθε φορά και μίας ακόμη), κάτι το οποίο «διευκολύνει» τον ταξινομητή σε περιπτώσεις που τα πακέτα είναι παρόμοια.

Παρατηρούμε όμως, πως η συγκεκριμένη υλοποίηση λειτουργεί καλύτερα, όταν η εφαρμογή χρησιμοποιεί ένα πολύ συγκεκριμένο πρωτόκολλο, όπως πχ το BitTorrent, και αυτό γιατί τα πρωτόκολλα που χρησιμοποιούνται από πολλές εφαρμογές, όπως το TCP που χρησιμοποιεί You Tube, ενδέχεται να θεωρηθούν (λανθασμένα) πως χρησιμοποιούνται από την εφαρμογή που μας ενδιαφέρει και αυτό να έχει ως αποτέλεσμα την λάθος ταξινόμησή τους.

Επίσης, όσον αφορά τα πειράματα, παρατηρούμε πως δεν είναι όλες οι στήλες απαραίτητες για την εκπαίδευση και για την τελική ταξινόμηση. Η έβδομη στήλη, που περιέχει πληροφορίες για το κάθε πακέτο, δεν είναι απαραίτητη καθώς δεν παρέχει σημαντικές για την ταξινόμηση πληροφορίες. Ως εκ τούτου, μπορεί να

αφαιρεθεί από τα σύνολα εκπαίδευσης και ελέγχου, γεγονός που θα ωφελήσει συνολικά την ταχύτητα των πειραμάτων.

4.2. Πιθανές προεκτάσεις

Μελλοντικά, η παρούσα υλοποίηση μπορεί να βελτιστοποιηθεί έτσι ώστε να λειτουργεί και για πρωτόκολλα που χρησιμοποιούνται από περισσότερες εφαρμογές. Επίσης, σε συνδυασμό με τον τομέα της ανίχνευσης καινοτομιών (novelty detection) μπορεί να αναγνωρίζει κίνηση η οποία προέρχεται από άγνωστα δεδομένα ή δεδομένα για τα οποία ο ταξινομητής δεν έχει εκπαιδευτεί.

Πιο γενικά, καθώς η ταξινόμηση της κίνησης είναι ένα από τους ερευνητικούς τομείς που κερδίζουν όλο και περισσότερο ενδιαφέρον, θα είχε ενδιαφέρον σε μελλοντικές έρευνες να δοθεί αρκετή βαρύτητα σε ταξινομητές, ή ακόμη και να γίνουν προσπάθειες δημιουργίας πολυταξινομητών (multiclassifiers), για την ταξινόμηση κίνησης την ώρα που κάποιος χρησιμοποιεί το δίκτυο (on the fly). Αυτό θα έχει ως αποτέλεσμα και την καλύτερη παροχή υπηρεσιών, αλλά και την καλύτερη χρήση του δικτύου. Θα συμβάλλει επίσης αρκετά στην ασφάλεια των δικτύων, γεγονός που θα ωφελήσει τόσο τους παρόχους όσο και τους χρήστες.

Επίσης είναι σημαντικό να γίνει ένας διαχωρισμός στο τι μελετάται κάθε φορά, καθώς δεν είναι το ίδιο να ταξινομούνται παθητικές μετρήσεις με το να ταξινομούνται δεδομένα που έχουν υποστεί κάποιο είδος φιλτραρίσμα.

ΑΝΑΦΟΡΕΣ

[1] Dainotti, A.; Pescape, A.; Claffy, K.C., "Issues and future directions in traffic classification," inNetwork, IEEE , vol.26, no.1, pp.35-40, January-February 2012

[2] Silvio Valenti, Dario Rossi, Alberto Dainotti, Antonio Pescapè, Alessandro Finamore, and Marco Mellia. Reviewing traffic classification. In DataTraffic Monitoring and Analysis, Ernst Biersack, Christian Callegari, and Maja Matijasevic (Eds.). Springer-Verlag, Berlin, Heidelberg 123-147. 2013.

[3] Erman, J.; Mahanti, A.; Arlitt, M., "QRP05-4: Internet Traffic Identification using Machine Learning," in Global Telecommunications Conference, 2006. GLOBECOM '06. IEEE , vol., no., pp.1-6, Nov. 27 2006-Dec. 1 2006

[4] Nguyen, T.T.T.; Armitage, G., "A survey of techniques for internet traffic classification using machine learning," in Communications Surveys & Tutorials, IEEE , vol.10, no.4, pp.56-76, Fourth Quarter 2008

[5] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. "Introduction to Data Mining", (First Edition). Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2005.

[6] H. Kim, K.C. Claffy, M. Fomenkov, D. Barman, M. Faloutsos, and K.Y. Lee, "Internet traffic classification demystified: Myths, caveats, and the best practices", Proceedings of the 2008 ACM CoNEXT conference, 2008.

[7] "Understanding Support Vector Machines", http://www.mathworks.com/help/stats/support-vector-machines-svm.html#bsr5b42, [Online].

[8] "resubLoss, R2015a" http://www.mathworks.com/help/stats/regressionsvm.resubloss.html, [Online].

[9] "loss, R2015a" http://www.mathworks.com/help/stats/compactclassificationsvm.loss.html, [Online]

[10] Wireshark, https://www.wireshark.org/