



**ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ**  
**ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ**

**ΣΤΗΝ ΕΠΙΣΤΗΜΗ ΤΩΝ ΥΠΟΛΟΓΙΣΤΩΝ**

**Διπλωματική Εργασία  
Μεταπτυχιακού Διπλώματος Ειδίκευσης**

ΟΙΚΟΝΟΜΙΚΟ  
ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΑΘΗΝΩΝ  
ΒΙΒΛΙΟΘΗΚΗ  
εισ. 81897  
Αρ. ΚΩΣ  
ταξ.

**Θέμα:**

*Διήθηση ανεπιθύμητης ηλεκτρονικής αλληλογραφίας με διάφορες μορφές του απλοϊκού  
ταξινομητή Bayes και διαμοιρασμό φίλτρων μεταξύ χρηστών*

**Άρης Κοσμόπουλος**

**Επιβλέπων: Ιων Ανδρουτσόπουλος**

**Μάιος 2007**





ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ  
ΚΑΤΑΛΟΓΟΣ

0 000000627627

# Περιεχόμενα

<b>ΠΕΡΙΛΗΨΗ.....</b>	<b>1</b>
<b>1 ΕΙΣΑΓΩΓΗ.....</b>	<b>2</b>
1.1 Ευχαριστίες.....	3
<b>2 ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ.....</b>	<b>4</b>
2.1 Δυνατά λάθη κατάταξης.....	4
2.2 Γενική μορφή φύλτρων που χρησιμοποιούν απλοϊκούς ταξινομητές Bayes.....	4
2.3 Μέτρα αξιολόγησης και διαγράμματα ROC.....	6
2.4 Αρχική επιλογή ιδιοτήτων βάσει αριθμού εμφανίσεων στα μηνύματα εκπαίδευσης.....	7
2.5 Αξιολόγηση ιδιοτήτων βάσει πληροφοριακού κέρδους.....	7
2.6 Μορφές απλοϊκού ταξινομητή Bayes που χρησιμοποιήθηκαν.....	8
2.6.1 Πολυμεταβλητή μορφή Bernoulli του απλοϊκού ταξινομητή Bayes.....	8
2.6.2 Πολυωνυμικός απλοϊκός ταξινομητής Bayes.....	9
2.6.3 Πολυμεταβλητή μορφή Gauss του απλοϊκού ταξινομητή Bayes.....	10
2.6.4 Flexible Bayes.....	11
2.6.5 Το φύλτρο του Paul Graham.....	11
2.7 Ανταλλαγή φύλτρων.....	12
<b>3 ΠΕΙΡΑΜΑΤΙΚΗ ΑΞΙΟΛΟΓΗΣΗ.....</b>	<b>14</b>
3.1 Συλλογές μηνυμάτων.....	15
3.2 Πειράματα επιβεβαίωσης της ορθότητας του λογισμικού.....	16
3.3 Πειράματα επιλογής της καλύτερης μορφής απλοϊκού ταξινομητή Bayes.....	23
3.4 Ιδιότητες που αντιστοιχούν σε n-γράμματα χαρακτήρων.....	29
3.5 Πειράματα με ανταλλαγή φύλτρων και ομαδικό φύλτρο.....	33
<b>4 ΕΠΙΛΟΓΟΣ.....</b>	<b>39</b>
4.1 Σύνοψη.....	39
4.2 Μελλοντικές επεκτάσεις.....	40
<b>ΒΙΒΛΙΟΓΡΑΦΙΑ.....</b>	<b>41</b>



## Περίληψη

Το πρόβλημα των ανεπιθύμητων διαφημιστικών μηνυμάτων έχει λάβει ανησυχητικές διαστάσεις τα τελευταία χρόνια. Ένας από τους πιο επιτυχημένους τρόπους αντιμετώπισης του προβλήματος είναι η χρήση φίλτρων που διαχωρίζουν τα επιθυμητά από τα ανεπιθύμητα μηνύματα χρησιμοποιώντας αλγορίθμους μηχανικής μάθησης. Μεταξύ αυτών των αλγορίθμων, ιδιαίτερα δημοφιλής σε φίλτρα ανεπιθύμητης αλληλογραφίας είναι ο αλγόριθμος του απλοίκου ταξινομητή Bayes, λόγω της απλότητας και του μικρού υπολογιστικού του κόστους. Σε αυτή την εργασία υλοποιήσαμε και αξιολογήσαμε μέσω πειραμάτων φίλτρα που χρησιμοποιούν διάφορες μορφές του απλοϊκού ταξινομητή Bayes, με διαφορετικές διανυσματικές αναπαραστάσεις των μηνυμάτων. Στη συνέχεια συγκρίναμε, πάλι μέσω πειραμάτων, τη χρήση ενός ξεχωριστού ατομικού φίλτρου για κάθε χρήστη, τη χρήση ενός κοινού φίλτρου για όλους του χρήστες, και μια τρίτη προσέγγιση στην οποία κάθε χρήστης χρησιμοποιεί τόσο το δικό του φίλτρο όσο και τα φίλτρα των συνεργατών του.



## 1. Εισαγωγή

Τα τελευταία χρόνια πολύς κόσμος χρησιμοποιεί το ηλεκτρονικό ταχυδρομείο ως μέσο επικοινωνίας. Τα μηνύματα όμως που καταφθάνουν στον καθένα μας καθημερινώς δεν περιορίζονται μόνο σε αυτά που πραγματικά μας ενδιαφέρουν. Ένα μεγάλο ποσοστό, το οποίο υπολογίζεται ότι ξεπερνάει το 60%, ανήκει στην κατηγορία των λεγομένων ανεπιθύμητων μηνυμάτων (spam), συνήθως διαφημιστικού περιεχομένου. Ένας από τους πιο αποτελεσματικούς τρόπους περιορισμού του προβλήματος είναι η χρήση ειδικών φίλτρων που διαχωρίζουν αυτόμata τα μηνύματα σε επιθυμητά (ham) και μη (spam).

Μια ιδιαίτερα επιτυχημένη κατηγορία φίλτρων αυτού του είδους χρησιμοποιεί αλγορίθμους μηχανικής μάθησης, ιδιαίτερα τον αλγόριθμο του απλοϊκού ταξινομητή Bayes (Naive Bayes), ο οποίος συχνά προτιμάται στα φίλτρα ανεπιθύμητης αλληλογραφίας λόγω της απλότητάς του και του μικρού υπολογιστικού του κόστους [1, 2, 3]. Για να εκπαιδευτούν τα φίλτρα αυτά πρέπει να χρησιμοποιηθούν συλλογές μηνυμάτων που ήδη έχουν διαχωριστεί σε επιθυμητά και μη. Μετά την εκπαίδευσή τους, τα φίλτρα είναι σε θέση να αποφανθούν αν ένα νέο μήνυμα, του οποίου δεν γνωρίζουμε την κατηγορία, είναι επιθυμητό ή όχι.

Μια εναλλακτική προσέγγιση είναι η προώθηση από τους χρήστες των ανεπιθύμητων μηνυμάτων που λαμβάνουν προς κεντρικές ή κατανεμημένες βάσεις δεδομένων. Οι βάσεις αυτές, χρησιμοποιώντας «αποτυπώματα» (π.χ. κωδικούς κατακερματισμού), μπορούν να αποφανθούν αν ένα μήνυμα που καταφθάνει σε έναν χρήστη είναι αντίγραφο ή πολύ παρόμοιο με μηνύματα που άλλοι χρήστες έχουν ήδη αναφέρει ως ανεπιθύμητα [4, 5, 6]. Αυτή η μέθοδος, όμως, έχει δύο βασικά μειονεκτήματα. Πρώτον, κακόβουλοι χρήστες ενδέχεται να αναφέρουν στις βάσεις πολλά επιθυμητά μηνύματα ως ανεπιθύμητα, «δηλητηριάζοντάς» τις, και δεύτερον μπορεί κάποιος να εισαγάγει αυτομάτως μεγάλα τμήματα διαφορετικών τυχαίων κειμένων στα πολλά αντίγραφα του διαφημιστικού μηνύματος που στέλνει, κάτι που είναι γνωστό ως «μετάλλαξη» (mutation), ώστε κάθε αντίγραφο να απεικονιστεί σε διαφορετικό «αποτύπωμα».

Τελικός σκοπός της εργασίας ήταν να προσεγγίσουμε το πρόβλημα με έναν υβριδικό τρόπο, που συνδυάζει χαρακτηριστικά από τις δυο παραπάνω μεθόδους. Ας υποθέσουμε ότι κάθε χρήστης έχει το προσωπικό του φίλτρο μηχανικής μάθησης, το οποίο έχει εκπαιδευθεί σε χειρωνακτικά διαχωρισμένα μηνύματα που έχει λάβει ο ίδιος στο παρελθόν, αλλά συγχρόνως μπορεί να «δανείζει» το φίλτρο του σε ένα σύνολο άλλων χρηστών (π.χ. σε συναδέλφους του ή άλλους χρήστες σε ένα δίκτυο ομοτίμων) [7]. Κάθε χρήστης θα μπορούσε τότε να εμπιστεύεται σε διαφορετικό βαθμό κάθε άλλο φίλτρο, ανάλογα με το ποσοστό ορθότητας (accuracy) που επιτυγχάνει το άλλο φίλτρο κατά την κατάταξη μηνυμάτων που έχει λάβει ο συγκεκριμένος χρήστης στο παρελθόν. Η προσέγγιση αυτή αποτελεί μία μορφή συλλογικής μάθησης [8, 9], που οδηγεί συνήθως σε βελτίωση του ποσοστού ορθότητας, όταν οι συνδυαζόμενοι ταξινομητές δεν κάνουν τα ίδια λάθη. Το τελευταίο είναι πολύ πιθανό, αν αναλογιστούμε πως στην περίπτωση ανταλλαγής προσωπικών φίλτρων, ο κάθε ταξινομητής (φίλτρο) είναι εκπαιδευμένος σε διαφορετικό σύνολο μηνυμάτων (ενός διαφορετικού χρήστη). Ακόμη, μια που ανταλλάσσονται φίλτρα και όχι μηνύματα, ενδέχεται να αυξάνεται η ανθεκτικότητα του συστήματος σε μεταλλάξεις μηνυμάτων, ενώ δεν τίθενται τόσο έντονα ζητήματα προσωπικών δεδομένων, όπως θα συνέβαινε αν ο κάθε χρήστης διέθετε στους υπόλοιπους ως δεδομένα εκπαίδευσης τα μηνύματα που έχει λάβει στο παρελθόν.

Μετά τη μελέτη της σχετικής βιβλιογραφίας, η εργασία πραγματοποιήθηκε σε τρία στάδια. Στο πρώτο στάδιο υλοποιήθηκαν διαφορετικές παραλλαγές του αλγορίθμου του απλοϊκού ταξινομητή Bayes, που χρησιμοποιούν διαφορετικές αναπαραστάσεις των μηνυμάτων. Στο δεύτερο στάδιο πραγματοποιήθηκαν πειράματα με τις παραλλαγές αυτές, για να επιλεγεί η καλύτερη. Τα

πειράματα έγιναν με δύο διαφορετικές συλλογές μηνυμάτων, στις οποίες θα αναφερθούμε αναλυτικά σε ακόλουθη ενότητα. Στο τρίτο στάδιο προσπαθήσαμε να διαπιστώσουμε μέσω πειραμάτων κατά πόσο αποδίδει η υβριδική μας προσέγγιση, σε σχέση με το να χρησιμοποιεί ο κάθε χρήστης μόνο το ατομικό του φίλτρο ή να έχουν όλοι ένα κοινό φίλτρο, εκπαιδευμένο στα μηνύματα που έχουν λάβει όλοι στο παρελθόν. Καταλήξαμε στο συμπέρασμα πως καλύτερες επιδόσεις ορθότητας επιτυγχάνονται με τη χρήση ομαδικού φίλτρου, αν και ενδέχεται η υβριδική μας προσέγγιση να έχει καλύτερα αποτελέσματα αν στο μέλλον χρησιμοποιηθούν βελτιώσεις που περιγράφονται στο τέλος της εργασίας.

Στη συνέχεια, στο κεφάλαιο 2, θα γίνει μία σύντομη θεωρητική ανάλυση του προβλήματος διαχωρισμού μηνυμάτων σε επιθυμητά και μη, καθώς και του τρόπου προσέγγισής του με αλγορίθμους βασισμένους στον απλοϊκό ταξινομητή Bayes. Κατόπιν θα περιγράψουμε αναλυτικά τις διάφορες μορφές του απλοϊκού ταξινομητή Bayes που υλοποιήσαμε και αξιολογήσαμε, καθώς και τους τρόπους αναπαράστασης των μηνυμάτων που χρησιμοποιούν αυτές οι μορφές.

Στο κεφάλαιο 3 θα παρουσιάσουμε τα πειράματα της εργασίας, τα αποτελέσματά τους και τα συμπεράσματα στα οποία μας οδήγησαν. Συγχρόνως θα αναφερθούμε στις δύο συλλογές μηνυμάτων που χρησιμοποιήσαμε στα πειράματα, καθώς και στις διαφορετικές δυνατότητες προσέγγισης που δίνει η κάθε μια.

Τέλος, στο κεφάλαιο 4 θα συνοψίσουμε τα συμπεράσματα της εργασίας και θα αναφερθούμε σε πιθανές μελλοντικές επεκτάσεις της.

## 1.1 Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κύριο Ίωνα Ανδρουτσόπουλο, για την καθοδήγησή του κατά τη διάρκεια της εκπόνησης της διπλωματικής μου εργασίας, αλλά και για την ευκαιρία που μου έδωσε να ασχοληθώ με αυτό το ενδιαφέρον θέμα. Επίσης ευχαριστώ τον κύριο Γιώργο Παλιούρα του Ε.Κ.Ε.Φ.Ε. «Δημόκριτος», συνεπιβλέποντα της εργασίας, τον οποίο και συμβουλευόμουν καθ' όλη τη διάρκεια της εργασίας και ο οποίος με βοήθησε να ξεπεράσω πολλά προβλήματα και δυσκολίες, αλλά και να προσεγγίσω κάποια θέματα από άλλες οπτικές γωνίες. Επίσης ένα μεγάλο ευχαριστώ στους πραγματικούς κατόχους των μηνυμάτων της συλλογής μηνυμάτων του Δημόκριτου, χωρίς την οποία δεν θα μπορούσαν να έχουν πραγματοποιηθεί τα πειράματα της εργασίας. Τέλος, ευχαριστώ τους Βαγγέλη Μέτση για τη βοήθειά του στην αρχή της εργασίας αυτής και Δημήτρη Μπόχτη, που ασχολείται με την ενσωμάτωση του λογισμικού της εργασίας στο Thunderbird.

## 2. ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ

Στο κεφάλαιο αυτό αρχικά θα αναφερθούμε στα λάθη που μπορεί να κάνει ένα φύλτρο ανεπιθύμητης αλληλογραφίας και στη σοβαρότητά τους. Κατόπιν θα περιγράψουμε συνοπτικά τη γενική μορφή του αλγορίθμου του απλοϊκού ταξινομητή Bayes και τα μέτρα που χρησιμοποιούμε για την αξιολόγηση των φύλτρων. Στη συνέχεια θα παρουσιάσουμε αναλυτικά τις διαφορετικές μορφές των απλοϊκών ταξινομητών Bayes που χρησιμοποιήσαμε, καθώς και τις αναπαραστάσεις των μηνυμάτων που δοκιμάσαμε. Τέλος, θα περιγράψουμε αναλυτικότερα την προσέγγιση ανταλλαγής φύλτρων μεταξύ χρηστών με την οποία πειραματιστήκαμε. Όλες οι μέθοδοι που θα περιγραφούν υλοποιήθηκαν σε C++ στη διάρκεια της εργασίας.

### 2.1 Δυνατά λάθη κατάταξης

Η κυριότερή μας απαίτηση από ένα φύλτρο ανεπιθύμητων μηνυμάτων είναι να διαχωρίζει με όσο μεγαλύτερη ακρίβεια γίνεται τα επιθυμητά μηνύματα από τα ανεπιθύμητα. Συνήθως το φύλτρο αφήνει στο φάκελο των εισερχομένων μηνυμάτων του χρήστη όλα τα μηνύματα που θεωρεί επιθυμητά, ενώ μεταφέρει αυτόματα τα μηνύματα που θεωρεί ανεπιθύμητα σε έναν ειδικό φάκελο ανεπιθύμητων μηνυμάτων, τον οποίο ο χρήστης μπορεί να ελέγχει και να αδειάζει περιοδικά (π.χ. σε ώρες που έχει άνεση χρόνου).

Τα δυνατά λάθη που μπορεί να κάνει ένα τέτοιο φύλτρο είναι δύο ειδών. Το πρώτο είδος (λάθη τύπου 1) είναι να κατατάξει κάποιο ανεπιθύμητο μήνυμα ως επιθυμητό. Στην περίπτωση αυτή, ο χρήστης θα αναγκαστεί να διαβάσει και να διαγράψει χειρωνακτικά το ανεπιθύμητο μήνυμα από το φάκελο εισερχομένων του, κάτι που ενδέχεται να έχει ως αποτέλεσμα απώλεια κρίσιμου χρόνου, εκνευρισμό κλπ. Το δεύτερο είδος (λάθη τύπου 2) είναι να κατατάξει κάποιο επιθυμητό μήνυμα ως ανεπιθύμητο. Το δεύτερο είδος λαθών είναι γενικά πιο σοβαρό, γιατί ενδέχεται να προκαλέσει καθυστερημένη απάντηση του χρήστη (αν ελέγχει το φάκελο ανεπιθύμητων μηνυμάτων του πολύ αραιά) ή και απώλεια επιθυμητού μηνύματος (αν δεν ελέγχει ποτέ το φάκελο ανεπιθύμητων μηνυμάτων). Η προτεραιότητα, λοιπόν, του φύλτρου είναι συνήθως να ελαχιστοποιήσει τα σφάλματα τύπου 2, μειώνοντας κατά το δυνατόν παράλληλα τα σφάλματα τύπου 1. Εδώ πρέπει να θυμόμαστε ότι η προσπάθεια περιορισμού των λαθών του ενός τύπου συνήθως προκαλεί αύξηση των λαθών του άλλου.

### 2.2 Γενική μορφή φύλτρων που χρησιμοποιούν απλοϊκούς ταξινομητές Bayes

Κάθε μήνυμα (γενικότερα, κάθε αντικείμενο προς κατάταξη) αναπαριστάνεται από ένα διάνυσμα  $\langle x_1, \dots, x_m \rangle$ , όπου κάθε  $x_i$  είναι η τιμή μιας ιδιότητας (τυχαίας μεταβλητής)  $X_i$ . Στην περίπτωσή μας, κάθε  $x_i$  παρέχει πληροφορίες για μία διαφορετική λεκτική μονάδα (token) ή ακολουθία χαρακτήρων (n-γράμματα, n-grams) του μηνύματος. Για παράδειγμα, η συμβολοσειρά «Hi,\_I\_am\_Aris» περιέχει τις λεκτικές μονάδες «Hi», «,», «I», «am», και «Aris», ενώ περιέχει τα 3-γράμματα «Hi», «i,», «,I», «\_I», «I\_a» κλπ. Σημειωτέον ότι, αντίθετα από άλλες προσεγγίσεις [14], δεν αγνοούμε τα σημεία στίξεως, αλλά θεωρούμε κάθε σημείο στίξεως διαφορετική λεκτική μονάδα. Τα σημεία στίξεως είναι ιδιαίτερα χρήσιμα, αφού τα ανεπιθύμητα μηνύματα συχνά περιέχουν περισσότερα σημεία στίξεως (π.χ. θαυμαστικά) από ό,τι τα επιθυμητά, συχνά οι αποστολείς ανεπιθύμητων μηνυμάτων εισάγουν σημεία στίξεως μεταξύ των γραμμάτων των λέξεων (π.χ. «H.i.,I.a.m.A.r.i.s.») προκειμένου να κάνουν δυσκολότερο τον εντοπισμό των λεκτικών μονάδων κλπ. Στην πιο απλή περίπτωση, οι ιδιότητες είναι δυαδικές (Boolean), δηλαδή

παίρνουν την τιμή 1 αν η αντίστοιχη λεκτική μονάδα ή ακολουθία χαρακτήρων εμφανίζεται στο μήνυμα και 0 αν δεν εμφανίζεται.<sup>1</sup> Εναλλακτικά, μπορεί οι τιμές των ιδιοτήτων να δείχνουν πόσες φορές εμφανίζονται στο μήνυμα οι αντίστοιχες λεκτικές μονάδες ή ακολουθίες χαρακτήρων (term frequencies, TF). Οι ιδιότητες TF προσφέρουν προφανώς περισσότερες πληροφορίες, αλλά η χρήση δυαδικών ιδιοτήτων συχνά οδηγεί σε καλύτερα αποτελέσματα [3], ενδεχομένως επειδή κατά τη χρήση ιδιοτήτων TF είναι δυνατόν να απαιτούνται υποθέσεις για τις κατανομές των τιμών τους που δεν ισχύουν.

Οι δύο προηγούμενοι τύποι ιδιοτήτων δεν λαμβάνουν υπόψιν το μήκος του μηνύματος. Για παράδειγμα, είναι διαφορετικό το να εμφανίζεται μια λέξη τρεις φορές σε ένα κείμενο δέκα λέξεων από το να εμφανίζεται τρεις φορές σε ένα κείμενο χιλίων λέξεων. Προκειμένου να αντιμετωπιστεί αυτό το πρόβλημα, είναι δυνατόν να χρησιμοποιηθούν ιδιότητες με κανονικοποιημένες τιμές TF (normalized term frequencies, normTF), όπου οι αριθμοί εμφανίσεων των λεκτικών μονάδων ή ακολουθιών χαρακτήρων διαιρούνται με το συνολικό πλήθος των λεκτικών μονάδων ή ακολουθιών του μηνύματος.

Μία πρόσθετη μορφή κανονικοποίησης, που μπορεί να χρησιμοποιηθεί είτε σε συνδυασμό με τιμές TF είτε με τιμές normTF, είναι να πολλαπλασιάζεται η τιμή TF ή normTF με την ανάστροφη συχνότητα εγγράφων (inverse document frequency, IDF) της λεκτικής μονάδας (ή ακολουθίας χαρακτήρων) στην οποία αντιστοιχεί η ιδιότητα (έστω  $X_i$ ), δηλαδή με το  $\log \frac{1}{\sum_j \delta_{ij}}$ , όπου  $j$  είναι δείκτης προς τα κείμενα μιας συλλογής (π.χ. μια συλλογή μηνυμάτων του παρελθόντος στην περίπτωσή μας) και  $\delta_{ij}$  είναι 1 αν η λεκτική μονάδα (ή ακολουθία χαρακτήρων) της ιδιότητας εμφανίζεται στο κείμενο  $j$  ή 0 αν δεν εμφανίζεται. Με την κανονικοποίηση αυτή πετυχαίνουμε να προσδίδουμε μεγαλύτερο βάρος σε σπάνιες λεκτικές μονάδες (ή ακολουθίες χαρακτήρων), από ότι στις συνηθισμένες. Η κανονικοποίηση αυτή προέρχεται από το χώρο της ανάκτησης πληροφοριών, αλλά έχει χρησιμοποιηθεί με επιτυχία και σε φίλτρα αλληλογραφίας [13].

Ανεξαρτήτως του είδους των ιδιοτήτων που θα χρησιμοποιήσουμε, όλες οι μορφές του απλοϊκού ταξινομητή Bayes αποφαίνονται ότι ένα εισερχόμενο μήνυμα με διάνυσμα  $\vec{x}$  είναι ανεπιθύμητο ανν  $p(c_s|\vec{x}) = \frac{p(c_s) \cdot p(\vec{x}|c_s)}{p(c_s) \cdot p(\vec{x}|c_s) + p(c_h) \cdot p(\vec{x}|c_h)} > T$ , όπου  $c_s$  και  $c_h$  παριστάνουν τις κατηγορίες των ανεπιθύμητων και επιθυμητών μηνυμάτων, αντίστοιχα, και  $T$  είναι ένα κατώφλι που ρυθμίζει πόσο εύκολα (χαμηλό  $T$ ) ή δύσκολα (υψηλό  $T$ ) κατατάσσει το φίλτρο τα εισερχόμενα μηνύματα ως ανεπιθύμητα. Όσο υψηλότερο είναι το  $T$ , τόσο λιγότερα σφάλματα τύπου 2 θα κάνει το φίλτρο, αλλά και τόσο περισσότερα σφάλματα τύπου 1. Αυτό που διαφοροποιείται σε κάθε μορφή του απλοϊκού ταξινομητή Bayes είναι ο τρόπος υπολογισμού του  $p(\vec{x}|c)$ , δηλαδή της πιθανότητας εμφάνισης του διανύσματος  $\vec{x}$  δεδομένης μιας κατηγορίας  $c$ . Οι πιθανότητες  $p(c_s)$  και  $p(c_h)$  εκτιμώνται πάντα με τον ίδιο τρόπο, μετρώντας το ποσοστό ανεπιθύμητων ή επιθυμητών μηνυμάτων, αντίστοιχα, σε μια συλλογή μηνυμάτων του παρελθόντος.

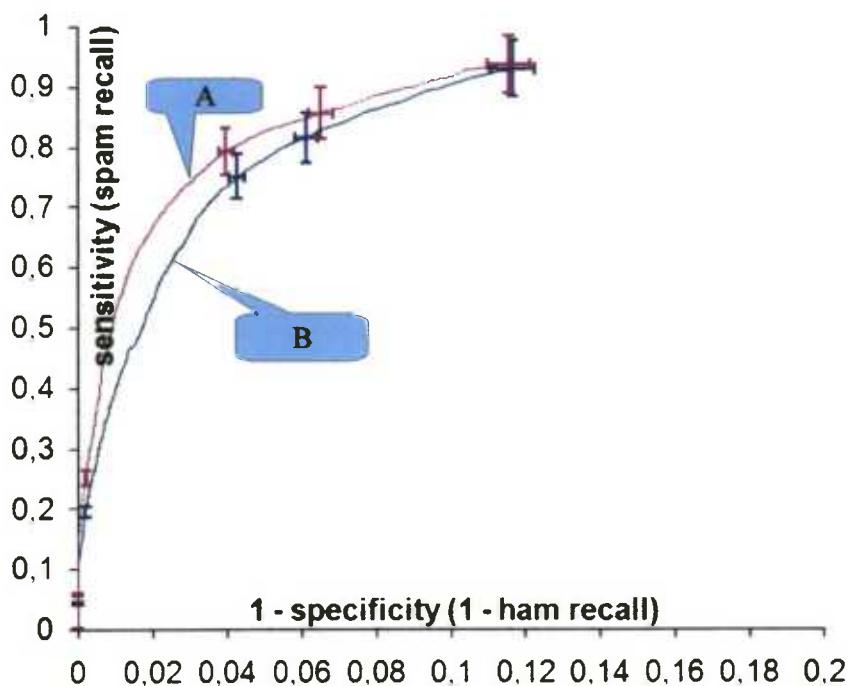
Εναλλακτικά, μπορούμε να διαιρούμε την  $p(c_s|\vec{x})$  με την  $p(c_h|\vec{x})$  (που υπολογίζεται αντίστοιχα) ή ισοδύναμα το  $p(c_s) \cdot p(\vec{x}|c_s)$  με το  $p(c_h) \cdot p(\vec{x}|c_h)$  και να κατατάσσουμε το μήνυμα ως ανεπιθύμητο αν ο λόγος υπερβαίνει ένα κατώφλι. Στην περίπτωση αυτή, μπορούμε ισοδύναμα να κατατάσσουμε το μήνυμα ως ανεπιθύμητο ανν

<sup>1</sup> Στην εργασία αυτή εξετάζουμε μόνο το κυρίως κείμενο (body) των μηνυμάτων. Γενικότερα, η διανυσματική αναπαράσταση των μηνυμάτων μπορεί να περιλαμβάνει και ιδιότητες που παρέχουν πληροφορίες για τα συνημμένα έγγραφα του μηνύματος (π.χ. αν υπάρχουν και τι είδους είναι), το χρόνο αποστολής του μηνύματος κλπ.

$\log(p(c_s)) + \log(p(\tilde{x}|c_s)) - \log(p(c_h)) - \log(p(\tilde{x}|c_h)) > \Delta$ , όπου  $\Delta$  ένα κατώφλι που καθορίζει πάλι πόσο εύκολα (μεγάλο  $\Delta$ ) ή δύσκολα (μικρό  $\Delta$ ) κατατάσσει το φίλτρο ένα εισερχόμενο μήνυμα ως ανεπιθύμητο. Όπως θα δούμε σε επόμενες ενότητες, η εκτίμηση των  $p(\tilde{x}|c)$  γίνεται πολλαπλασιάζοντας τις εκτιμήσεις των  $p(x_i|c)$ , για όλες τις τιμές  $x_i$  του  $\tilde{x}$ . Οι πολλαπλασιασμοί των  $p(x_i|c)$  οδηγούν συχνά σε πολύ μικρούς αριθμούς, οι οποίοι είναι εκτός των ορίων που μπορεί να χειριστεί η γλώσσα προγραμματισμού. Η χρήση λογαρίθμων αντιμετωπίζει αυτό το πρόβλημα.

### 2.3 Μέτρα αξιολόγησης και διαγράμματα ROC

Κατά την πειραματική αξιολόγηση των φίλτρων, μετράμε πόσα ανεπιθύμητα μηνύματα κατετάγησαν σωστά (true positives), πόσα επιθυμητά κατετάγησαν σωστά (true negatives), πόσα ανεπιθύμητα κατετάγησαν λανθασμένα (false positives) και πόσα επιθυμητά κατετάγησαν λανθασμένα (false negatives). Στη συνέχεια υπολογίζουμε το ποσοστό ανάκλησης των ανεπιθύμητων μηνυμάτων (spam recall), δηλαδή το ποσοστό ανεπιθύμητων μηνυμάτων που κατετάγησαν σωστά, ως  $SR = \frac{TP}{(TP + FN)}$ . Ομοίως, υπολογίζουμε το ποσοστό ανάκλησης των επιθυμητών μηνυμάτων (ham recall), δηλαδή το ποσοστό των επιθυμητών μηνυμάτων που κατετάγησαν σωστά, ως  $HR = \frac{TN}{(TN + FP)}$ . Τα δύο ποσοστά ανάκλησης επηρεάζονται από το  $T$  ή  $\Delta$  της προηγούμενης ενότητας. Για παράδειγμα, υψηλό  $\Delta$  οδηγεί σε χαμηλό SR και υψηλό HR.



Προκειμένου να συγκρίνουμε δύο φίλτρα για όλες τις δυνατές τιμές  $\Delta$  (ή  $T$ ), κατασκευάζουμε διαγράμματα ROC. Ο κατακόρυφος άξονας μετρά την «ευαισθησία» (sensitivity) του φίλτρου, που στην περίπτωσή μας είναι το SR, ενώ ο οριζόντιος τη «σαφήνειά» (specificity), που στην περίπτωσή μας είναι το HR. Για την ακρίβεια, ο οριζόντιος άξονας μετρά συνήθως το  $1 - \text{specificity} (1 - HR)$ . Για κάθε τιμή  $\Delta$  (ή  $T$ ), παίρνουμε ένα άλλο σημείο ( $1 - HR$ ,  $SR$ ) στο επόπεδο και ενώνοντας αυτά τα σημεία προκύπτει η καμπύλη ROC του φίλτρου. Αν η καμπύλη ενός φίλτρου A είναι πάντα πιο ψηλά από την καμπύλη ενός φίλτρου B, αυτό δείχνει ότι το A είναι καλύτερο, επειδή για οποιαδήποτε τιμή HR έχει υψηλότερο SR («κόβει» περισσότερα

ανεπιθύμητα, επιτρέποντας στο ίδιο ποσοστό επιθυμητών μηνυμάτων να περάσουν), όπως φαίνεται στο παρακάτω παράδειγμα. Επειδή ενδιαφερόμαστε μόνο για πολύ υψηλές τιμές HR (να περνούν σχεδόν όλα τα επιθυμητά, πολύ λίγα σφάλματα τύπου 2), περιορίζουμε στα διαγράμματα τις τιμές του οριζόντιου άξονα στο διάστημα [0, 0.2].

Για την δημιουργία των διαγραμμάτων ROC, χρειαζόμαστε ένα σύνολο αρκετών ζευγαριών από τιμές sensitivity και 1-specificity κάθε φύλτρου, αρκετών για τη δημιουργία μια γραφικής παράστασης. Για να μπορέσουμε να πάρουμε αυτά ζεύγη, υπολογίζουμε τα TP, FP, TN και FN κάθε φύλτρου για διάφορες τιμές Δ. Συγκεκριμένα, χρησιμοποιούμε 2000 διαφορετικές τιμές Δ, που φαίνονται στον ακόλουθο πίνακα:

Τιμές	Βήμα
0-5	0,01
5-755	0,5

## 2.4 Αρχική επιλογή ιδιοτήτων βάσει αριθμού εμφανίσεων στα μηνύματα εκπαίδευσης

Παρ' όλο που οι απλοϊκοί ταξινομητές Bayes έχουν γραμμική πολυπλοκότητα ως προς τον αριθμό των ιδιοτήτων των διανυσμάτων, στην πράξη η χρήση πολύ μεγάλου αριθμού ιδιοτήτων μπορεί να οδηγήσει σε μείωση της ταχύτητας του φύλτρου καθώς και σε αυξημένες απαιτήσεις μνήμης. Αυξάνει, επίσης, την πιθανότητα υπερ-εφαρμογής (over-fitting) στα παραδείγματα εκπαίδευσης. Ως ένα πρώτο βήμα περιορισμού του αριθμού των ιδιοτήτων, στην περίπτωση ιδιοτήτων που αντιστοιχούν σε λεκτικές μονάδες, αγνοούμε λεκτικές μονάδες που δεν εμφανίζονται σε τουλάχιστον  $df$  (π.χ.  $df = 4$  ή 5) μηνύματα εκπαίδευσης. Ομοίως, στην περίπτωση που χρησιμοποιούμε ιδιότητες που αντιστοιχούν σε  $n$ -γράμματα (ακολουθίες χαρακτήρων), αγνοούμε  $n$ -γράμματα που δεν εμφανίζονται σε τουλάχιστον  $df$  μηνύματα εκπαίδευσης. Λεκτικές μονάδες ή ακολουθίες χαρακτήρων που δεν υπερβαίνουν αυτό το κατώφλι αριθμού εμφανίσεων είναι τόσο σπάνιες που στην πράξη είναι άχρηστες. Το απλό αυτό πρώτο βήμα επιλογής ιδιοτήτων μειώνει κατά πολύ τον αριθμό των ιδιοτήτων, χωρίς να απαιτεί περίπλοκους υπολογισμούς.

## 2.5 Αξιολόγηση ιδιοτήτων βάσει πληροφοριακού κέρδους

Κάθε λεκτική μονάδα ή  $n$ -γράμμα που έχει «επιζήσει» από το προηγούμενο στάδιο επιλογής αντιστοιχίζεται σε μια διαφορετική ιδιότητα. Προαιρετικά μπορούμε κατόπιν να περιορίσουμε περαιτέρω τον αριθμό των ιδιοτήτων, εκτιμώντας το πληροφοριακό κέρδος (information gain) που παρέχει κάθε ιδιότητα  $X$ . Ο τύπος του πληροφοριακού κέρδους είναι:

$$IG(X, C) = \sum_{x \in \{0,1\}, c \in \{c_L, c_S\}} P(X = x \wedge C = c) \cdot \log_2 \frac{P(X = x \wedge C = c)}{P(X = x) \cdot P(C = c)}$$

όπου  $X$  η τυχαία μεταβλητή που παριστάνει την κατηγορία του μηνύματος. Αποδεικνύεται ότι το πληροφοριακό κέρδος εκτιμά την αναμενόμενη μείωση της εντροπίας της  $C$  που προσφέρει η γνώση της τιμής της  $X$ . Αφού αξιολογήσουμε όλες τις ιδιότητες, μπορούμε να κρατήσουμε τις  $m$  καλύτερες ως προς το πληροφοριακό κέρδος που παρέχουν.

Η χρήση του πληροφοριακού κέρδους έχει δυο μειονεκτήματα. Πρώτον, αν ο συνολικός αριθμός των ιδιοτήτων που έχουν απομείνει από την αρχική επιλογή της προηγούμενης ενότητας είναι ήδη κοντά στο  $m$ , τότε η αξιολόγηση των ιδιοτήτων με το μέτρο του πληροφοριακού κέρδους προφανώς δεν περιορίζει περαιτέρω τον αριθμό των ιδιοτήτων σε σημαντικό βαθμό. Δεύτερον, αξιολογεί όλες τις ιδιότητες σαν ήταν δυαδικές (στον παραπάνω τύπο, η  $X$  παίρνει μόνο τις τιμές 0 και 1), κάτι που δεν ισχύει σε όλες τις μορφές απλοϊκού ταξινομητή Bayes (π.χ. κάποιες από τις μορφές χρησιμοποιούν ιδιότητες με τιμές TF κλπ.). Τα πειραματικά αποτελέσματα του Schneider [15], όμως, που δοκίμασε εναλλακτικές μορφές μέτρησης του πληροφοριακού κέρδους για TF ιδιότητες, δείχνουν πως δεν χάνουμε τίποτα αξιολογώντας TF ιδιότητες σαν να ήταν δυαδικές, χρησιμοποιώντας τον παραπάνω τύπο, θεωρώντας δηλαδή ότι η τιμή της TF ιδιότητας είναι 1 όποτε η πραγματική της τιμή είναι μη μηδενική.

## 2.6 Μορφές απλοϊκού ταξινομητή Bayes που χρησιμοποιήθηκαν

Οι μορφές του απλοϊκού ταξινομητή Bayes που υλοποιήθηκαν και αξιολογήθηκαν διαφέρουν μεταξύ τους σε δύο κυρίως σημεία. Το πρώτο είναι ο τρόπος υπολογισμού του  $p(\vec{x}|c)$ , ενώ το δεύτερο είναι το είδος των ιδιοτήτων. Όλες οι μορφές υλοποιήθηκαν με χρήση λογαρίθμων (βλ. ενότητα 2.2). Πρέπει να θυμόμαστε ότι ο σκοπός είναι ο υπολογισμός των δυο παρακάτω όρων:

$$[\log(p(c_s)) + \log(p(\vec{x}|c_s))] - [\log(p(c_h)) + \log(p(\vec{x}|c_h))] > \Delta$$

P\_spam P\_ham

Επίσης, όπως θα δούμε, σε όλες τις μορφές του απλοϊκού ταξινομητή που χρησιμοποιούμε, το  $p(\vec{x}|c)$  είναι ένα γινόμενο  $m$  παραγόντων, οπότε ο λογαρίθμός του οδηγεί πάντα σε άθροισμα λογαρίθμων. Όποτε στο εξής αναφερόμαστε σε άθροισμα  $m$  λογαρίθμων, εννοούμε το άθροισμα αυτό.

### 2.6.1 Πολυμεταβλητή μορφή Bernoulli του απλοϊκού ταξινομητή Bayes

Η πρώτη και πιο απλή μορφή απλοϊκού ταξινομητή Bayes που υλοποιήθηκε στη διάρκεια της εργασίας είναι η πολυμεταβλητή μορφή Bernoulli (multivariate Bernoulli Naive Bayes). Κάθε μήνυμα παριστάνεται από ένα διάνυσμα  $\vec{x} = \{x_1, \dots, x_m\}$ , όπου τα  $x_i$  είναι οι τιμές των ιδιοτήτων. Οι ιδιότητες είναι δυαδικές και κάθε ιδιότητα έχει τιμή 1, ανν η αντίστοιχη λεκτική μονάδα ή  $n$ -γραμμα εμφανίζεται στο μήνυμα. Το  $p(\vec{x}|c)$  εκτιμάται ως εξής, όπου  $t_i$  η λεκτική μονάδα ή  $n$ -γραμμα που αντιστοιχεί στην  $i$ -στη ιδιότητα:

$$p(\vec{x}|c) = \prod_{i=1}^m p(t_i|c)^{x_i} \cdot (1 - p(t_i|c))^{(1-x_i)}$$

Ουσιαστικά θεωρούμε ότι κάθε μήνυμα  $d$  είναι το αποτέλεσμα  $m$  δοκιμών Bernoulli. Σε κάθε δοκιμή, αποφασίζουμε αν η λεκτική μονάδα ή  $n$ -γραμμα που αντιστοιχεί στην  $i$ -στη ιδιότητα θα εμφανίστει στο μήνυμα. Κάνουμε, δηλαδή, την απλοϊκή παραδοχή ότι οι τιμές των ιδιοτήτων είναι ανεξάρτητες δεδομένης της κατηγορίας  $c$  του μηνύματος, παραδοχή που ενώ δεν ισχύει στην πράξη, οδηγεί παρ' όλα αυτά σε καλά αποτελέσματα [16].

Το κάθε  $P(t | c)$  εκτιμάται ως  $\frac{1+M_{t,c}}{2+M_c}$  όπου  $M_{t,c}$  είναι ο αριθμός των μηνυμάτων της

κατηγορίας  $c$  που περιέχουν το  $t$  και  $M_c$  το σύνολο των μηνυμάτων της κατηγορίας  $c$ . Η πρόσθεση του 1 στον αριθμητή και του 2 στον παρανομαστή γίνεται για να αποφευχθούν μηδενικές εκτιμήσεις.

## 2.6.2 Πολυωνυμικός απλοϊκός ταξινομητής Bayes

Υλοποιήθηκαν τρεις διαφορετικές μορφές του πολυωνυμικού απλοϊκού ταξινομητή Bayes (multinomial Naive Bayes), με δυαδικές, TF και TF-IDF ιδιότητες.

### Πολυωνυμικός απλοϊκός ταξινομητής Bayes με δυαδικές ιδιότητες

Χρησιμοποιεί δυαδικές ιδιότητες, όπως και η πολυμεταβλητή μορφή Bernoulli.

Διαφοροποιείται, όμως, στον τρόπο εκτίμησης του  $p(t | c)$ , που γίνεται χρησιμοποιώντας τον τύπο  $\frac{1+N_{t,c}}{m+N_c}$  όπου  $N_{t,c}$  ο αριθμός εμφανίσεων του  $t$  στα κείμενα της κατηγορίας  $c$ , ενώ  $N_c$  είναι το

άθροισμα των  $m$  διαφορετικών  $N_{t,c}$ . Επίσης, στον τύπο υπολογισμού του  $p(\tilde{x}|c)$  δεν λαμβάνουμε υπόψιν την απουσία μιας λεκτικής μονάδας ή  $n$ -γράμματος, οπότε λείπει ο όρος  $(1 - p(t_i | c))^{(1-x_i)}$ . Περισσότερες πληροφορίες για το πώς προκύπτουν αυτούς οι τύποι δίνονται στην εργασία [3].

### Πολυωνυμικός απλοϊκός ταξινομητής Bayes με ιδιότητες TF

Στην περίπτωση αυτή δεν χρησιμοποιούνται δυαδικές αλλά TF ιδιότητες (βλ. ενότητα 2.2). Ο υπολογισμός του  $p(\tilde{x}|c)$  γίνεται με τον τύπο:

$$\prod_{i=1}^m p(t_i | c_s)^{x_i}$$

Τα  $p(t | c)$  υπολογίζονται όπως στην περίπτωση του πολυωνυμικού ταξινομητή με δυαδικές ιδιότητες. Και πάλι, περισσότερες πληροφορίες για το πώς προκύπτουν αυτοί οι τύποι δίνονται στην εργασία [3].

### Πολυωνυμικός απλοϊκός ταξινομητής Bayes με μετασχηματισμένες ιδιότητες TF

Στην περίπτωση αυτή, η TF τιμή κάθε ιδιότητας μετασχηματίζεται ως εξής:

1. Αρχικά λογαριθμίζουμε το άθροισμα της τιμής TF με τη μονάδα (για να αποφύγουμε μηδενικό όρισμα λογαρίθμου). Ο μετασχηματισμός αυτός προέρχεται από την εργασία [13] και θα μπορούσε να εφαρμοστεί και στην προηγούμενη μορφή του πολυωνυμικού απλοϊκού ταξινομητή Bayes, που χρησιμοποιεί ιδιότητες TF. Ο μετασχηματισμός αυτός επιχειρεί να αντιμετωπίσει το ότι η κατανομή του αριθμού εμφανίσεων μιας λεκτικής μονάδας σε κείμενα φυσικής γλώσσας στην πραγματικότητα δεν ακολουθεί πολυωνυμική κατανομή.
2. Στη συνέχεια πολλαπλασιάζουμε το αποτέλεσμα του βήματος 1 με την τιμή IDF, δηλαδή το

$$\log \frac{\sum_k 1}{\sum_k \delta_{kj}} \quad (\text{βλ. ενότητα 2.2}).$$

3. Στο τρίτο βήμα διαιρούμε το αποτέλεσμα του βήματος 2 με  $\sqrt{\sum_k (\mathbf{d}_{kj})^2}$ , όπου  $\mathbf{d}_{kj}$  οι αρχικές τιμές TF των ιδιοτήτων. Ο μετασχηματισμός αυτός είναι παρόμοιος με το μετασχηματισμό των ιδιοτήτων normTF (βλ. ενότητα 2.2). Όπως ο μετασχηματισμός του βήματος 1, προέρχεται από την εργασία [13] και θα μπορούσε να εφαρμοστεί και στην προηγούμενη μορφή του πολυωνυμικού απλοϊκού ταξινομητή Bayes, που χρησιμοποιεί ιδιότητες TF.

### **Κανονικοποίηση των $p(t_i | c)$**

Σε όλες τις μορφές του πολυωνυμικού απλοϊκού ταξινομητή Bayes, στον υπολογισμό του  $p(\vec{x}|c)$  εμπλέκονται οι πιθανότητες  $p(t_i | c)$ . Για την ακρίβεια, μια που χρησιμοποιούμε το λογάριθμο του  $p(\vec{x}|c)$ , εμπλέκονται τα  $\log p(t_i | c)$ . Τα τελευταία μπορούν να κανονικοποιηθούν, διαιρώντας το καθένα με  $\sum_{i=1}^m \log p(t_i | c)$ . Ο μετασχηματισμός αυτός προέρχεται από την εργασία [13] και αποσκοπεί στο να αντιμετωπίσει εν μέρει το γεγονός ότι οι εμφανίσεις διαφορετικών λεκτικών μονάδων ή  $n$ -γραμμάτων στα μηνύματα μιας κατηγορίας δεν είναι ανεξάρτητες, αντίθετα από τις παραδοχές του απλοϊκού ταξινομητή Bayes.

### **Δυναμική επιλογή ιδιοτήτων**

Υπάρχουν σχετικές έρευνες που υποστηρίζουν ότι, αντί ο υπολογισμός του  $p(\vec{x}|c)$  να γίνεται απ' όλες (τι συνολικά) τις ιδιότητες, είναι προτιμότερο να χρησιμοποιούμε μόνο τις ιδιότητες που παρέχουν τη μεγαλύτερη βεβαιότητα ως προς την κατηγορία του συγκεκριμένου μηνύματος, ένα είδος «τοπικής» επιλογής ιδιοτήτων σε κάθε μήνυμα. Στην περίπτωσή μας, κρατάμε τα κ μεγαλύτερα  $x_i \cdot \log P(t_i | c)$  κατά απόλυτη τιμή, για  $c = \text{spam}$  και  $c = \text{ham}$  συγχρόνως. Η τοπική αυτή μέθοδος επιλογής ιδιοτήτων δεν χρησιμοποιήθηκε σε κανένα πείραμα, γιατί ενσωματώθηκε αργά στην υλοποίηση, αλλά το λογισμικό της εργασίας επιτρέπει τη διερεύνησή της σε μελλοντικά πειράματα.

### **2.6.3 Πολυμεταβλητή μορφή Gauss του απλοϊκού ταξινομητή Bayes**

Στην περίπτωση αυτή θεωρούμε ότι κάθε ιδιότητα (από τις  $m$ ) ακολουθεί κανονική κατανομή  $g(x_i; \mu_{i,c}, \sigma_{i,c})$  στα μηνύματα της κατηγορίας  $c$ . Το  $\mu_{i,c}$  είναι η μέση τιμή και το  $\sigma_{i,c}$  η τυπική απόκλιση της κατανομής, που εκτιμώνται κατά την εκπαίδευση του ταξινομητή. Ο τύπος του  $g(x_i; \mu_{i,c}, \sigma_{i,c})$  είναι ο εξής:

$$g(x_i; \mu_{i,c}, \sigma_{i,c}) = \frac{1}{\sigma_{i,c} \sqrt{2\pi}} e^{-\frac{(x_i - \mu_{i,c})^2}{2\sigma_{i,c}^2}}$$

Το  $p(\vec{x}|c)$  υπολογίζεται ουσιαστικά ως το γινόμενο των  $m$   $g(x_i; \mu_{i,c}, \sigma_{i,c})$ . Χρησιμοποιήσαμε normTF τιμές των  $x_i$  (βλ. ενότητα 2.2). Επίσης πειραματιστήκαμε με ιδιότητες των οποίων οι τιμές προκύπτουν όπως στα βήματα 2-3 (χωρίς το μετασχηματισμό του βήματος 1 )

της περίπτωσης του πολυωνυμικού απλοϊκού ταξινομητή Bayes με μετασχηματισμένες ιδιότητες TF (βλ. ενότητα 2.6.2). Ένα πρόβλημα είναι ότι η τυπική απόκλιση πολλές φορές εκτιμάται ως μηδενική, κάτι που δημιουργεί προβλήματα στον παραπάνω τύπο.

#### 2.6.4 Flexible Bayes

Η διαφορά του Flexible Bayes από την προηγούμενη μορφή είναι πως κάθε ιδιότητα θεωρούμε ότι ακολουθεί σε κάθε κατηγορία  $c$  μια κατανομή  $p(x_i|c)$  που είναι μείγμα κανονικών κατανομών: 
$$p(x_i|c) = \frac{1}{|c|} \cdot \sum_{l=1}^{|c|} g(x_i; \mu_{i,c,l}, \sigma_c) .$$

Στον παραπάνω τύπο,  $|c|$  είναι ο αριθμός των παραδειγμάτων εκπαίδευσης της κατηγορίας  $c$  και  $\mu_{i,c,l}$  είναι η τιμή της ιδιότητας  $X_i$  στο  $l$ -στό παράδειγμα εκπαίδευσης της κατηγορίας  $c$ . Θεωρούμε, δηλαδή, ότι κάθε παράδειγμα εκπαίδευσης συνεισφέρει στο μείγμα μία κανονική κατανομή. Το  $\sigma_c$ , η τυπική απόκλιση κάθε κανονικής κατανομής του μίγματος είναι κοινή για όλες τις ιδιότητες της κατηγορίας  $c$  και ισούται με  $\frac{1}{\sqrt{|c|}}$ , δηλαδή οι κανονικές κατανομές των παραδειγμάτων εκπαίδευσης «στενεύουν» όσο αυξάνονται τα παραδείγματα εκπαίδευσης της κατηγορίας.

Όπως στην μορφή της προηγούμενης ενότητας, χρησιμοποιήσαμε ιδιότητες  $\text{normTF}$ , αλλά πειραματιστήκαμε και με ιδιότητες των οποίων οι τιμές προκύπτουν όπως στα βήματα 2-3 (χωρίς το μετασχηματισμό του βήματος 1) της περίπτωσης του πολυωνυμικού απλοϊκού ταξινομητή Bayes με μετασχηματισμένες ιδιότητες TF (βλ. ενότητα 2.6.2).

Ο Flexible Bayes δεν παρουσιάζει το πρόβλημα της μηδενικής τυπικής απόκλισης που εμφανίζεται στη μορφή της προηγούμενης ενότητας και θεωρητικά μπορεί να προσεγγίσει καλύτερα τις πραγματικές κατανομές των ιδιοτήτων, που ενδέχεται να μην είναι κανονικές. Όμως είναι πιο αργός από τους υπολούπους, αφού η πολυπλοκότητά του κατά την ταξινόμηση νέων μηνυμάτων είναι  $O(mN)$ , όπου  $N$  το σύνολο των μηνυμάτων εκπαίδευσης και  $m$  ο αριθμός των ιδιοτήτων, ενώ οι υπόλοιπες μορφές του απλοϊκού ταξινομητή Bayes που εξετάσαμε έχουν πολυπλοκότητα ταξινόμησης  $O(m)$ . Κατά την εκπαίδευση, όλες οι μορφές έχουν πολυπλοκότητα  $O(m)$ .

#### 2.6.5 Το φίλτρο του Paul Graham

Ο συγκεκριμένος αλγόριθμος [18, 19] δεν ανήκει ακριβώς στην οικογένεια των απλοϊκών ταξινομητών Bayes, αλλά χρησιμοποιείται σε πολλά φίλτρα και αναφέρεται συχνά ως αφελής ταξινομητής Bayes. Η υλοποίησή του έγινε με σκοπό να συγκριθούν οι επιδόσεις του με εκείνες των απλοϊκών ταξινομητών Bayes που περιγράφηκαν στις προηγούμενες ενότητες.

Ο αλγόριθμος αυτός χρησιμοποιεί έναν πίνακα κατακερματισμού, σε κάθε εγγραφή του οποίου αποθηκεύει ως κλειδί μία λεκτική μονάδα  $t$  και ως τιμή μια «πιθανότητα»  $p_t$ , η οποία υπολογίζεται ως εξής:

1. Αρχικά υπολογίζουμε τον αριθμό  $g$ , που ισούται με  $2 * \text{τον αριθμό των επιθυμητών μηνυμάτων εκπαίδευσης στα οποία εμφανίζεται η λεκτική μονάδα.}$

- Στη συνέχεια υπολογίζουμε το  $b$  ως τον αριθμό των ανεπιθύμητων μηνυμάτων εκπαίδευσης στα οποία εμφανίζεται η λεκτική μονάδα.
- Αν  $b+g \geq 5$ , υπολογίζουμε την «πιθανότητα»  $p_i$  από τον παρακάτω τύπο, αλλιώς δεν εισάγουμε τη λεκτική μονάδα στον πίνακα κατακερματισμού (πρωτοεμφανιζόμενες λεκτικές μονάδες σε ένα κείμενο έχουν πάντα πιθανότητα 0,4):

$$p_i = \max(0.1, \min(0.99, \frac{\min(1, b/nbad)}{\min(1, g/ngood) + \min(1, b/nbad)}))$$

όπου  $ngood$  είναι το πλήθος των επιθυμητών μηνυμάτων εκπαίδευσης και  $nbad$  το πλήθος των ανεπιθύμητων μηνυμάτων εκπαίδευσης. Ουσιαστικά το  $p_i$  είναι εκτίμηση του  $p(spam)$ .

Κατά την κατάταξη ενός νεοεισερχόμενου μηνύματος  $d$ , ο αλγόριθμος επιλέγει μεταξύ των λεκτικών μονάδων του μηνύματος τις 15 που έχουν τις μεγαλύτερες και μικρότερες πιθανότητες  $p_i$  (τις μεγαλύτερες αποστάσεις κατά απόλυτη τιμή από την τιμή 0,5), μια μορφή δυναμικής επιλογής ιδιοτήτων (βλ. ενότητα 2.6.2). Η τελική πιθανότητα να είναι το εισερχόμενο μήνυμα  $d$  ανεπιθύμητο εκτιμάται από τον τύπο:

$$p(spam|d) = \frac{p_{t_1} \cdot \dots \cdot p_{t_{15}}}{p_{t_1} \cdot \dots \cdot p_{t_{15}} + (1-p_{t_1}) \cdot \dots \cdot (1-p_{t_{15}})}$$

Ο αριθμητής του παραπάνω τύπου υπολογίζει ουσιαστικά το  $p(d|spam)$  θεωρώντας, όπως στην περίπτωση των απλοϊκών ταξινομητών Bayes, ότι οι πιθανότητες εμφάνισης των λεκτικών μονάδων είναι ανεξάρτητες δεδομένης της κατηγορίας. Ο παρανομαστής χρησιμοποιείται για λόγους κανονικοποίησης. Ουσιαστικά πρόκειται για την πολυμεταβλητή μορφή Bernoulli του απλοϊκού ταξινομητή Bayes (βλ. ενότητα 2.6.1), αλλά χρησιμοποιείται δυναμική επιλογή ιδιοτήτων, αγνοείται η εκ των προτέρων πιθανότητα  $p(spam)$  και τα  $p_i$ , δηλαδή τα  $p(t|spam)$  εκτιμώνται όπως στην περίπτωση του πολυωνυμικού απλοϊκού ταξινομητή Bayes (βλ. ενότητα 2.6.2), λαμβάνοντας δηλαδή υπόψη πόσες φορές εμφανίζονται συνολικά οι λεκτικές μονάδες στα μηνύματα κάθε κατηγορίας.

## 2.7 Ανταλλαγή φίλτρων

Στην περίπτωση αυτή, ο κάθε χρήστης χρησιμοποιεί τόσο το δικό του προσωπικό φίλτρο, δηλαδή ένα φίλτρο που έχει εκπαιδευθεί στα μηνύματα που έχει λάβει ο ίδιος στο παρελθόν, όσο και τα προσωπικά φίλτρα συνεργατών του. Στο τέλος κάθε ημέρας, ο κάθε χρήστης διορθώνει τα λάθη που έκανε ο συνδυασμός όλων των φίλτρων στα μηνύματα που έφτασαν στη διάρκεια της ημέρας, επανεκπαιδεύει το προσωπικό του φίλτρο σε όλα τα μηνύματα που έχει λάβει ως τότε (δηλαδή τα μηνύματα που έλαβε στη διάρκεια της ημέρας και τα παλαιότερα) και στέλνει το επανεκπαιδευμένο φίλτρο του (ουσιαστικά τις εκτιμήσεις πιθανοτήτων που χρησιμοποιεί, θεωρώντας ότι όλοι χρησιμοποιούν την ίδια μορφή απλοϊκού ταξινομητή Bayes) στους συνεργάτες του. Στη διάρκεια των πειραμάτων αξιολογούμε το κατά πόσον το να χρησιμοποιεί ο κάθε χρήστης τόσο το δικό του όσο και τα φίλτρα των συναδέλφων του οδηγεί σε καλύτερα αποτελέσματα από την περίπτωση όπου κάθε χρήστης χρησιμοποιεί μόνο το δικό του προσωπικό φίλτρο.

Το πιο σημαντικό κομμάτι στη διαδικασία αυτή είναι το πώς θα κρίνει κάθε χρήστης σε

τι βαθμό θα εμπιστεύεται τα φύλτρα των συνεργατών του. Στα πειφάματά μας, θεωρήσαμε ότι στο τέλος κάθε ημέρας, αφού επανεκπαιδεύσει το φύλτρο του και λάβει τα επανεκπαιδευμένα φύλτρα των υπολοίπων, ο κάθε χρήστης μετρά το ποσοστό ορθότητας (accuracy) που επιτυγχάνει κάθε φύλτρο στα μηνύματα που έχει λάβει ο συγκεκριμένος χρήστης στο παρελθόν. Κατόπιν, στη διάρκεια της επόμενης ημέρας, κατατάσσει τα εισερχόμενα μηνύματά του ακολουθώντας το αποτέλεσμα της «ψηφιοφορίας» όλων των φύλτρων, ζυγίζοντας όμως την ψήφο κάθε φύλτρου με το ποσοστό ορθότητάς του, όπως υπολογίστηκε στο τέλος της προηγούμενης ημέρας.

Η μέθοδος αυτή έχει το ελάττωμα ότι ο βαθμός εμπιστοσύνης (ποσοστό ορθότητας) του προσωπικού φύλτρου του κάθε χρήστη εκτιμάται πάνω σε μηνύματα στα οποία το προσωπικό φύλτρο έχει εκπαιδευτεί στο παρελθόν, κάτι που ενδέχεται να οδηγεί σε υπεραισιόδοξο βαθμό εμπιστοσύνης για το προσωπικό φύλτρο του κάθε χρήστη, αποδίδοντας μικρότερη βαρύτητα από ότι θα έπρεπε στα φύλτρα των υπολοίπων. Κανονικά θα έπρεπε να υπάρχει μια τρίτη συλλογή μηνυμάτων του κάθε χρήστη, που δεν θα τη χρησιμοποιούσαμε για την εκπαίδευση του προσωπικού του φύλτρου, στην οποία θα βλέπαμε πώς αποδίδει το κάθε φύλτρο.

Υπάρχουν πολλοί άλλοι τρόποι με τους οποίους θα μπορούσε να πραγματοποιηθεί η ανταλλαγή φύλτρων. Εμείς επιλέξαμε αυτόν αρχικά λόγο της απλότητάς του, αλλά υπάρχουν πολλά ακόμη περιθώρια βελτίωσης, τα οποία θα αναφέρουμε στις μελλοντικές επεκτάσεις (κεφάλαιο 4.2.)

### 3. ΠΕΙΡΑΜΑΤΙΚΗ ΑΞΙΟΛΟΓΗΣΗ

Στο κεφάλαιο αυτό θα περιγράψουμε αρχικά τις συλλογές μηνυμάτων που χρησιμοποιήθηκαν στα πειράματα της εργασίας και τους λόγους επιλογής τους. Στη συνέχεια θα περιγράψουμε τα πειράματα που πραγματοποιήσαμε και τα συμπεράσματα που βγάλαμε από αυτά. Τα πειράματα πραγματοποιήθηκαν σε τρία στάδια:

- i. Πειράματα με σκοπό την επιβεβαίωση της ορθότητας της υλοποίησης των απλοϊκών ταξινομητών Bayes της εργασίας.
- ii. Πειράματα με σκοπό την επιλογή της καλύτερης (για τους σκοπούς μας) μορφής απλοϊκού ταξινομητή Bayes, χρησιμοποιώντας ιδιότητες που αντιστοιχούν σε λεκτικές μονάδες.
- iii. Πειράματα με σκοπό την επιβεβαίωση ότι η χρήση ιδιοτήτων που αντιστοιχούν σε λεκτικές μονάδες δεν υστερεί έναντι της χρήσης ιδιοτήτων που αντιστοιχούν σε η-γράμματα χαρακτήρων.
- iv. Πειράματα με σκοπό την αξιολόγηση της ανταλλαγής φίλτρων μεταξύ συνεργατών, έναντι της χρήσης μόνο προσωπικών φίλτρων ή μόνο ενός κοινού ομαδικού φίλτρου.

Στο πρώτο στάδιο επαναλάβαμε πειράματα του άρθρου [3] με τα ίδια δεδομένα, προκειμένου να επιβεβαιώσουμε την ορθότητα του λογισμικού της εργασίας.

Στο δεύτερο στάδιο θέλαμε να επιλέξουμε μεταξύ των διαφορετικών μορφών των απλοϊκών ταξινομητών Bayes που υλοποιήσαμε εκείνον που παρουσιάζει τα εξής χαρακτηριστικά στο μεγαλύτερο βαθμό: Πρώτον, να παρουσιάζει σπάνια λάθη τύπου 2 (βλ. ενότητα 2.1) και δευτερευόντως κατά το δυνατόν λιγότερα λάθη τύπου 1. Ισοδύναμα, η καμπύλη ROC του (βλ. ενότητα 2.3) να βρίσκεται ψηλότερα από τις καμπύλες ROC των υπολοίπων μορφών (υψηλότερη ανάκληση ανεπιθύμητων μηνυμάτων, λιγότερα λάθη τύπου 1) σε μια περιοχή εξαιρετικά υψηλής ανάκλησης επιθυμητών μηνυμάτων (π.χ. μεγαλύτερη του 0.985, εξαιρετικά σπάνια λάθη τύπου 2) Δεύτερον, να μην απαιτεί περισσότερα από μερικά λεπτά κατά την επανεκπαίδευσή του (θεωρούμε ότι το φίλτρο επανεκπαίδευται στο τέλος κάθε ημέρας) και η ταξινόμηση εισερχόμενων μηνυμάτων να προκαλεί ανεπαίσθητη καθυστέρηση (της τάξης των msec).

Επειδή τα περισσότερα πειράματα έγιναν με ιδιότητες που αντιστοιχούν σε λεκτικές μονάδες, στο τρίτο στάδιο θέλαμε να σιγουρευτούμε ότι η χρήση ιδιοτήτων αυτού του είδους δεν υστερεί έναντι της χρήσης ιδιοτήτων που αντιστοιχούν σε η-γράμματα χαρακτήρων. Δοκιμάσαμε επίσης την ταυτόχρονη χρήση ιδιοτήτων που αντιστοιχούν τόσο σε λεκτικές μονάδες όσο και σε η-γράμματα.

Το τέταρτο στάδιο αποτελούσε το απώτερο στόχο της εργασίας. Θέλαμε να διερευνήσουμε το κατά πόσον η ανταλλαγή φίλτρων μεταξύ συνεργατών υπερτερεί της χρήσης προσωπικών φίλτρων ή/και της χρήσης ενός κοινού ομαδικού φίλτρου. Σε γενικές γραμμές, κάθε πείραμα εκτελείται ως εξής. Ταξινομούμε τα μηνύματα της συλλογής κατά το χρόνο αφίξεώς τους και τα χωρίζουμε σε περιόδους (π.χ. ανά ημέρα ή ανά 100 μηνύματα). Το φίλτρο εκπαίδευται αρχικά στα μηνύματα της πρώτης περιόδου και κατατάσσει τα μηνύματα της δεύτερης περιόδου.

Στη συνέχεια, επανεκπαιδεύεται στα μηνύματα τόσο της πρώτης όσο και της δεύτερης περιόδου (θεωρώντας ότι ο χρήστης έχει διορθώσει τα λάθη που έκανε το φίλτρο κατά την κατάταξη των μηνυμάτων της δεύτερης περιόδου) και κατατάσσει τα μηνύματα της τρίτης περιόδου κ.ο.κ. Σημειωτέον ότι σε κάθε επανεκπαιδευση φίλτρου (στο τέλος κάθε περιόδου), επαναλαμβάνουμε εκ νέου την επιλογή των ιδιοτήτων. Με αυτόν τον τρόπο, κάθε μήνυμα της συλλογής (εκτός των μηνυμάτων της πρώτης περιόδου) έχει καταταγεί τελικά από το φίλτρο (σωστά ή λάθος) ακριβώς μία φορά και μπορούμε να υπολογίσουμε αποτελέσματα όπως τα ποσοστά ανάκλησης των δύο κατηγορίων. Για την ακρίβεια, για κάθε μήνυμα που κατατάσσεται, το φίλτρο επιστρέφει μια τιμή  $P_{ham} - P_{spam}$  που πρέπει να υπερβαίνει ένα κατώφλι  $\Delta$  (βλ. ενότητα 2.6) για να καταταγεί το μήνυμα ως ανεπιθύμητο. Στη διάρκεια του πειράματος, αποθηκεύουμε για κάθε μήνυμα που κατατάσσεται την τιμή  $P_{ham} - P_{spam}$ , κάτι που μας επιτρέπει να υπολογίσουμε ποια μηνύματα των δύο κατηγορίων θα είχαν καταταγεί σωστά ή λανθασμένα για διαφορετικές τιμές του  $\Delta$ . Έτσι, μεταβάλλοντας τις τιμές του  $\Delta$  λαμβάνουμε διαφορετικά ζεύγη τιμών HR και SR (βλ. ενότητα 2.3), τα οποία παριστάνονται ως σημεία στις καμπύλες ROC.

### 3.1 Συλλογές μηνυμάτων

Η εύρεση συλλογών μηνυμάτων πραγματικών χρηστών που να μπορούν να χρησιμοποιηθούν σε πειράματα είναι πολύ δύσκολη. Πολύ λίγοι δίνουν άδεια να χρησιμοποιήσει κάποιος τα μηνύματά τους, έστω και για ερευνητικούς σκοπούς. Ακόμα όμως και όταν κάποιοι χρήστες δίνουν άδεια να χρησιμοποιηθούν συλλογές μηνυμάτων τους, δεν ξέρουμε κατά πόσο έχουν αφαιρεθεί από τις συλλογές μηνύματα με συγκεκριμένα θέματα (π.χ αποδείξεις αγορών ή προσωπικά μηνύματα).

Στα πειράματα της εργασίας χρησιμοποιήθηκαν δυο ελεύθερα διαθέσιμες συλλογές μηνυμάτων. Η πρώτη, που λέγεται Enron-Spam, είναι αυτή που χρησιμοποιήθηκε στα πειράματα του άρθρου [3]. Τα επιθυμητά μηνύματα της συλλογής αυτής προέρχονται από έξι εργαζομένους της εταιρίας Enron και έγιναν ελεύθερα διαθέσιμα, μαζί με τα μηνύματα πολλών άλλων εργαζομένων της εταιρίας, κατά τη διαδικασία της διερεύνησης του σκανδάλου της Enron. Τα ανεπιθύμητα μηνύματα της Enron-Spam προέρχονται από τέσσερις διαφορετικές πηγές ανεπιθύμητων μηνυμάτων (του Spam Assassin και του HoneyPot Project μαζί, του Bruce Guenter και ενός από τους συγγραφείς της εργασίας [3]). Η συλλογή περιλαμβάνει έξι υποσυλλογές, μία για τα επιθυμητά μηνύματα κάθε χρήστη, στα οποία έχουν προστεθεί ανεπιθύμητα μηνύματα μιας από τις παραπάνω πηγές με διαφορετικές αναλογίες επιθυμητών και ανεπιθύμητων μηνυμάτων (βλ. [3] για περισσότερες λεπτομέρειες). Η Enron-Spam χρησιμοποιήθηκε προκειμένου να βεβαιωθούμε για την ορθότητα της υλοποίησής μας (πειράματα πρώτου σταδίου) συγκρίνοντας με τα αποτελέσματα της εργασίας [3]. Επίσης χρησιμοποιήθηκε στα πειράματα του δευτέρου σταδίου (ιδιότητες που αντιστοιχούν σε η-γράμματα). Ο παρακάτω πίνακας συνοψίζει τα χαρακτηριστικά της συλλογής Enron-Spam.

Η δεύτερη συλλογή προέρχεται από έξι πραγματικούς χρήστες του E.K.E.Φ.Ε. «Δημόκριτος», τους οποίους και ευχαριστούμε για την προσφορά τους. Περιέχει όλα τα μηνύματα (επιθυμητά και ανεπιθύμητα) που έλαβαν σε διάστημα 213 ημερών. Τα μηνύματα αυτά πριν έρθουν στην κατοχή μας είχαν υποστεί την ακόλουθη επεξεργασία. Κάθε λεκτική μονάδα τους είχε αντικατασταθεί από ένα μοναδικό αριθμό, τον ίδιο αριθμό για όλες τις εμφανίσεις της ίδιας λεκτικής μονάδας, με διαφορετική αντιστοιχία λεκτικών μονάδων και αριθμών σε κάθε χρήστη. Με τη μετατροπή αυτή δεν παραβιάζουμε το ιδιωτικό απόρρητο των ιδιοκτητών των μηνυμάτων, αφού την αντιστοιχία λεκτικών μονάδων και αριθμών τη γνωρίζουν μόνο οι ιδιοκτήτες, ενώ δεν επηρεάζονται οι αλγόριθμοι μάθησης στην περίπτωση που οι ιδιότητες αντιστοιχούν σε λεκτικές



μονάδες. Μοναδικό πρόβλημα αυτής της μετατροπής είναι ότι δεν επιτρέπει την πραγματοποίηση πειραμάτων όπου οι ιδιότητες αντιστοιχούν σε π-γράμματα.

Σε κάθε μια από τις παραπάνω συλλογές, λαμβάνουμε υπόψιν μας μόνο το Θέμα (Subject) του κάθε μηνύματος και το κείμενο του κυρίως μέρους του (Body), αγνοώντας τα συνημμένα αρχεία και τα υπόλοιπα στοιχεία των κεφαλίδων, τις ετικέτες HTML κλπ.

ham + spam	ham:spam	ham, spam periods
<b>farmer-d + GP</b>	3672:1500	[12/99, 1/02], [12/03, 9/05]
<b>kaminski-v + SH</b>	4361:1496	[12/99, 5/01], [5/01, 7/05]
<b>kitchen-l + BG</b>	4012:1500	[2/01, 2/02], [8/04, 7/05]
<b>williams-w3 + GP</b>	1500:4500	[4/01, 2/02], [12/03, 9/05]
<b>beck-s + SH</b>	1500:3675	[1/00, 5/01], [5/01, 7/05]
<b>lokay-m + BG</b>	1500:4500	[6/00, 3/02], [8/04, 7/05]

Χαρακτηριστικά Συλλογής Enron

ham + spam	ham:spam	ham, spam periods
user 1	8134:1731	14.11.2005-15.06.2006
user 2	3045:7081	14.11.2005-15.06.2006
user 3	3562:9982	14.11.2005-15.06.2006
user 4	7785:9729	14.11.2005-15.06.2006
user 5	4670:3604	14.11.2005-15.06.2006
user 6	9214:11497	14.11.2005-15.06.2006

Χαρακτηριστικά Συλλογής Δημόκριτου

### 3.2 Πειράματα επιβεβαίωσης της ορθότητας του λογισμικού

Στα πειράματα αυτά χρησιμοποιήσαμε τη συλλογή Enron-Spam, που είχε χρησιμοποιηθεί στην εργασία [3], με ιδιότητες που αντιστοιχούσαν σε λεκτικές μονάδες. Κατά την επιλογή ιδιοτήτων, αγνοούσαμε λεκτικές μονάδες που δεν εμφανίζονταν σε τουλάχιστον πέντε μηνύματα εκπαίδευσης και επιλέγαμε στη συνέχεια τις ιδιότητες με τα υψηλότερα  $\pi$  πληροφοριακά κέρδη, για  $m = 500, 1000, 3000$ , ή όλες τις ιδιότητες (χωρίς επιλογή μέσω πληροφοριακού κέρδους).

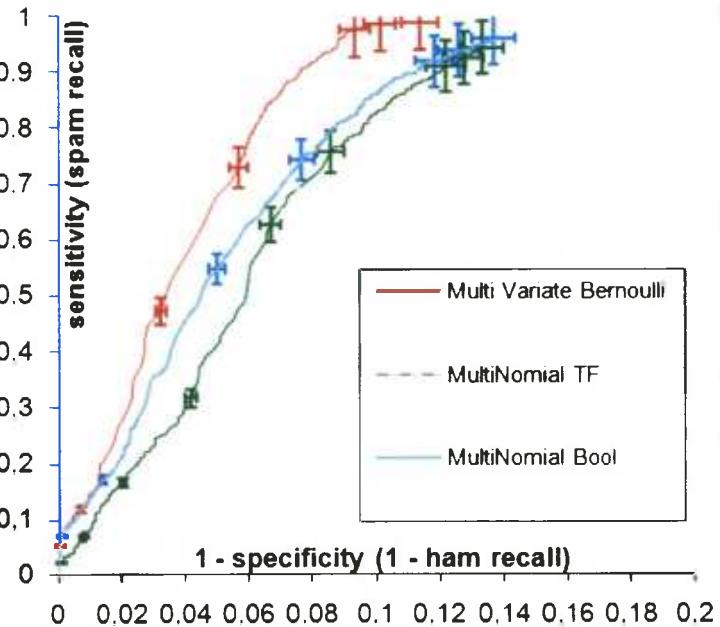
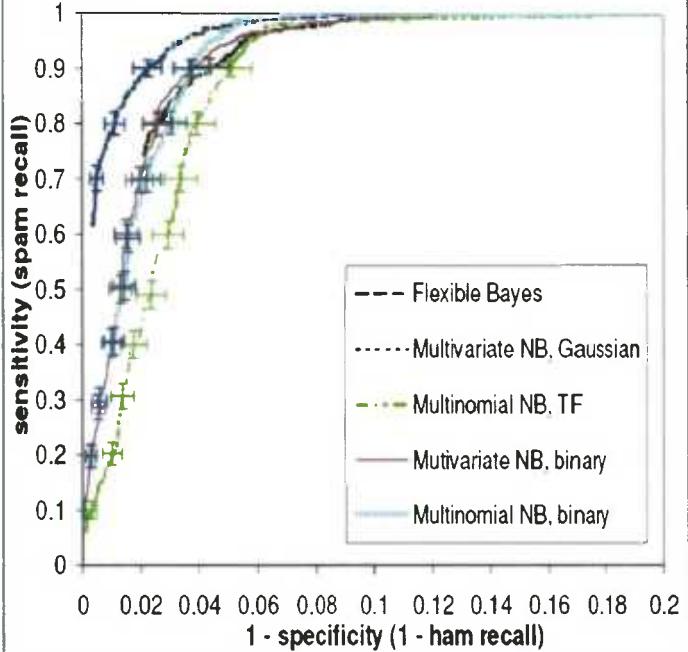
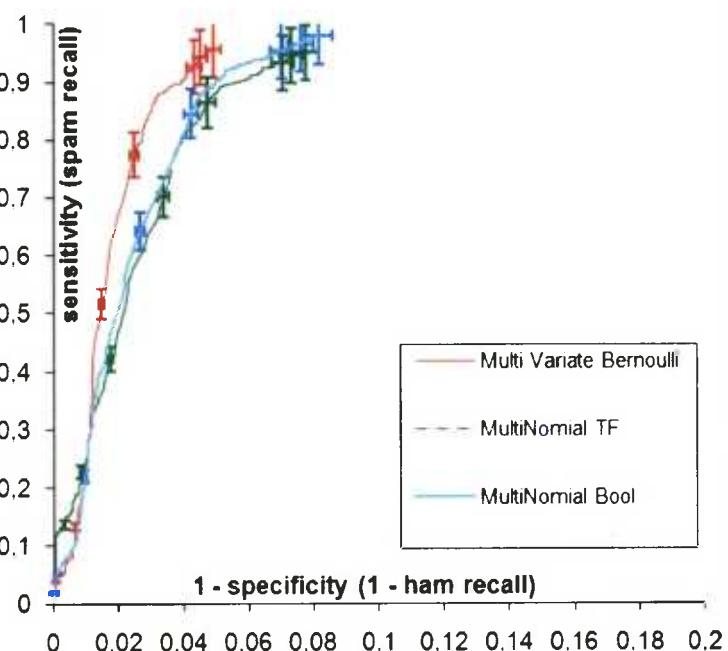
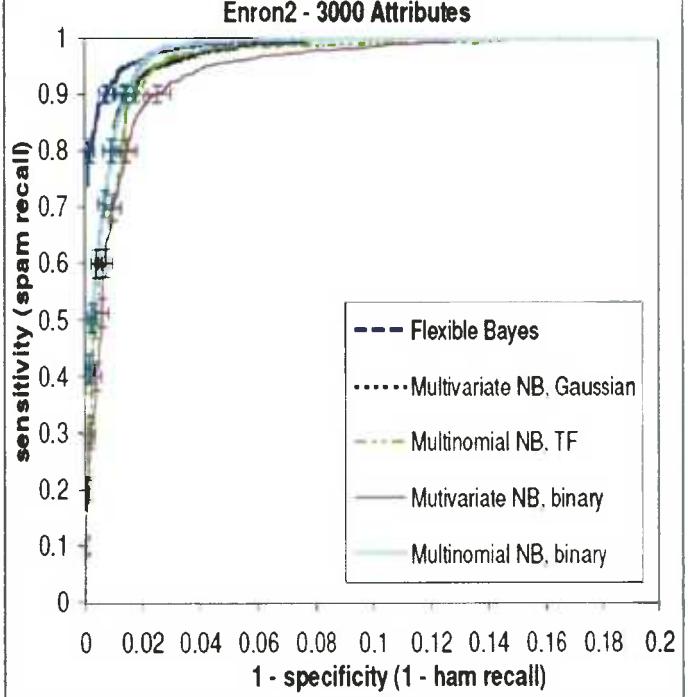
Στα πειράματα είχαμε αρχικά συμπεριλάβει την πολυμεταβλητή μορφή Gauss και τον Flexible Bayes, που είχαν χρησιμοποιηθεί στην εργασία [3] μαζί με άλλες μορφές του απλοϊκού ταξινομητή Bayes. Τελικά όμως απορρίψαμε τον Flexible Bayes, επειδή ήταν πολύ πιο αργός από τους υπολοίπους κατά την κατάταξη νέων μηνυμάτων. (Η πολυπλοκότητά του κατά την κατάταξη νέων μηνυμάτων είναι  $O(mN)$ , όπου  $N$  το σύνολο των μηνυμάτων εκπαίδευσης. Το  $N$  γίνεται πολύ μεγάλο κατά τη διάρκεια των πειραμάτων, ιδιαίτερα στη συλλογή του Δημόκριτου.)

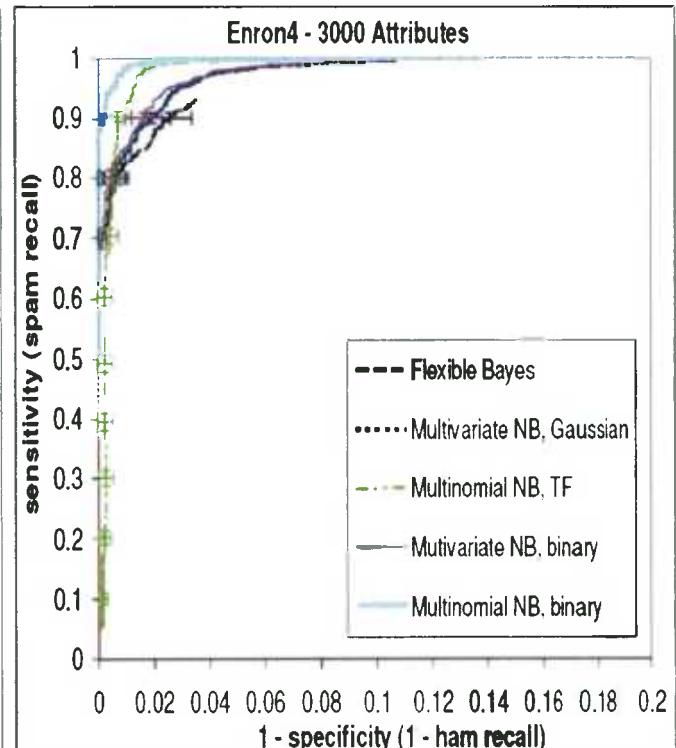
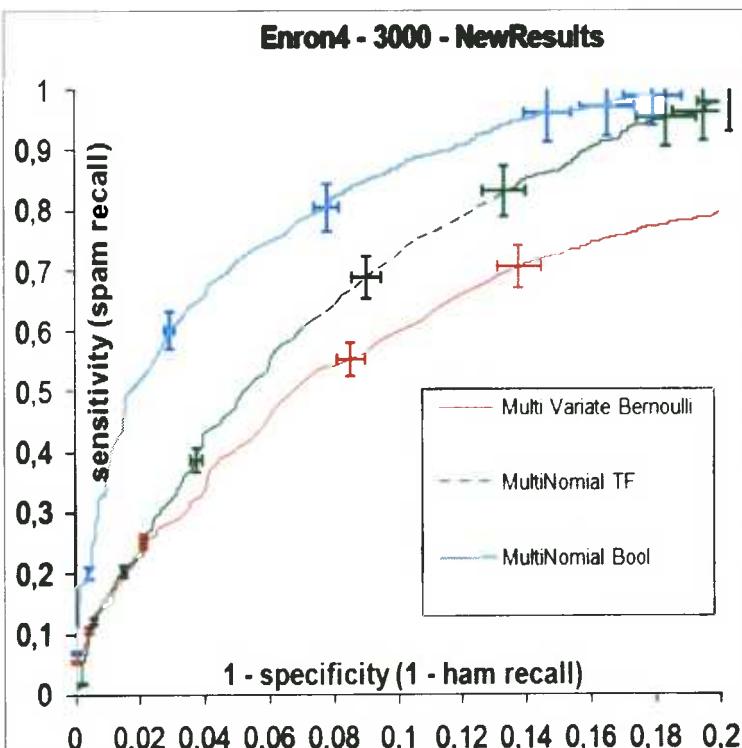
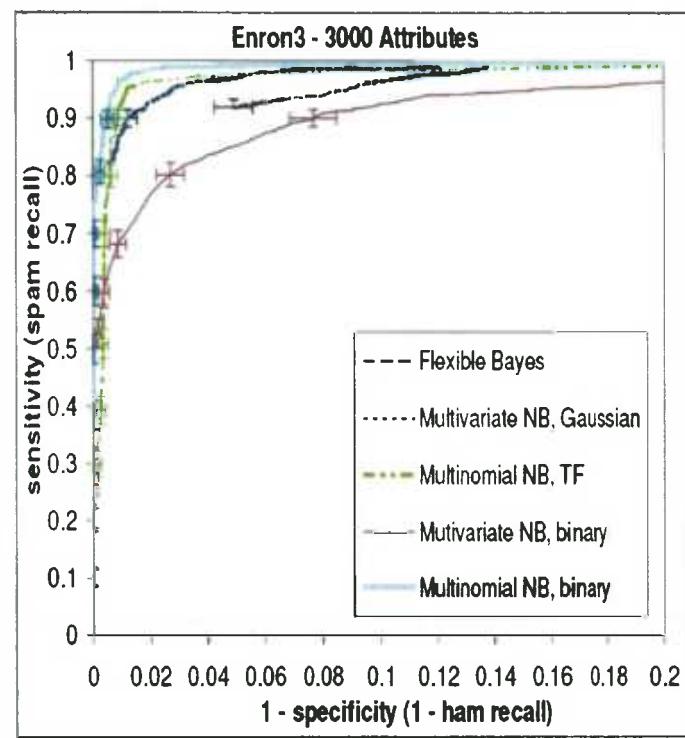
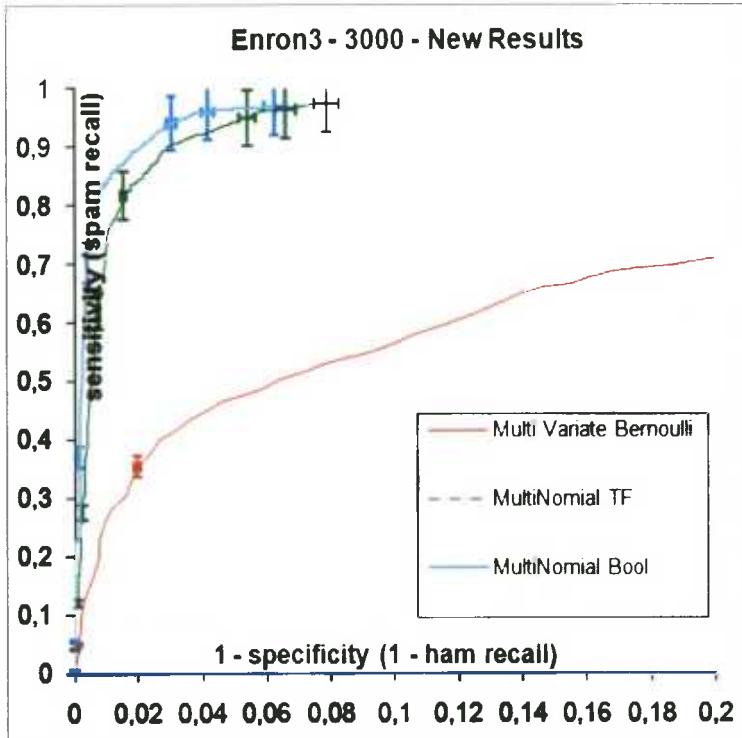
Η πολυμεταβλητή μορφή Gauss επίσης απορρίφθηκε, κυρίως λόγω του προβλήματος της τυπικής απόκλισης. Όπως αναφέραμε στην ενότητα 2.6.3, η μορφή αυτή χρησιμοποιεί μια εκτίμηση της τυπικής απόκλισης κάθε ιδιότητας  $X_i$  στα μηνύματα κάθε κατηγορίας c κατά τον υπολογισμό του  $g(x_i; \mu_{i,c}, \sigma_{i,c})$  και η εκτίμηση αυτή εμφανίζεται σε δύο παρανομαστές κλασμάτων. Είναι πολύ πιθανό η εκτίμηση να είναι μηδενική, ειδικά αν η λεκτική μονάδα που αντιστοιχεί στη  $X_i$  δεν έχει εμφανιστεί ποτέ σε μηνύματα της κατηγορίας c. Προσπαθήσαμε να αντιμετωπίσουμε αυτό το πρόβλημα δίνοντας μια πολύ μικρή τιμή στο  $\sigma_{i,c}$  όταν η εκτίμησή ήταν μηδενική. Τα πειραματικά αποτελέσματα, όμως, εξαρτώνταν σε πολύ μεγάλο βαθμό από την επιλογή της τιμής αυτής. Συγκεκριμένα, τα αποτελέσματα ήταν καλά μόνο σε μερικές περιόδους και το ποιες ακριβώς ήταν αυτές οι περίοδοι καθορίζοταν σε μεγάλο βαθμό από την τιμή που δίναμε στο  $\sigma_{i,c}$  όταν η εκτίμησή του ήταν μηδενική.

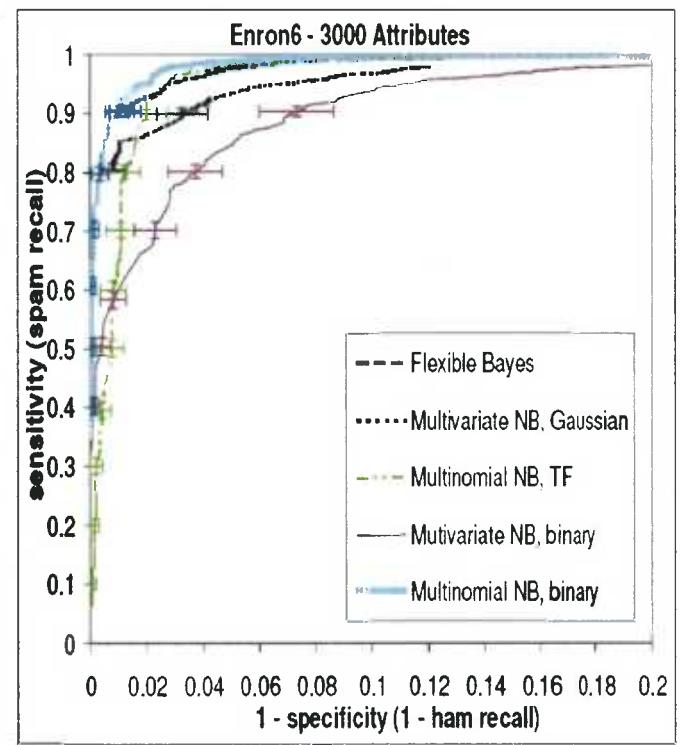
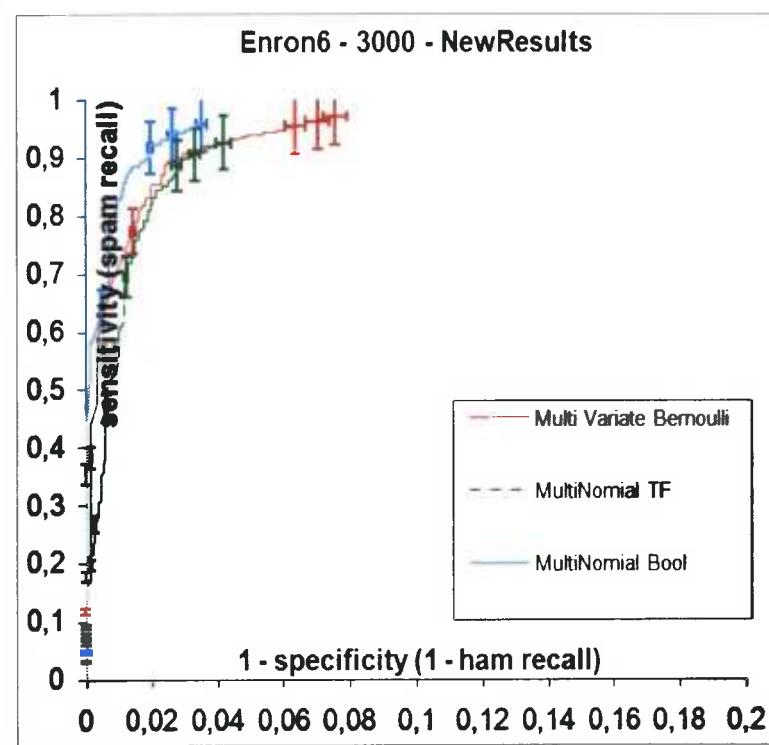
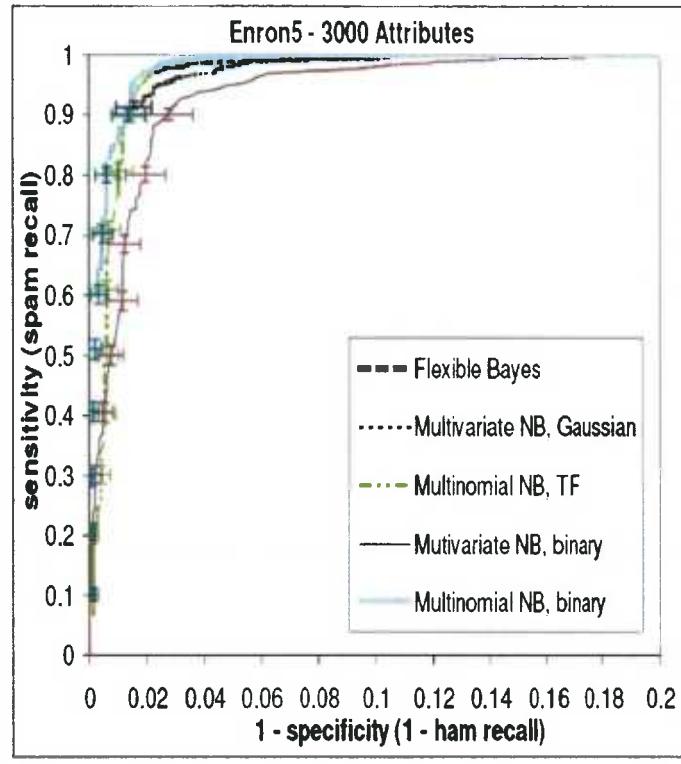
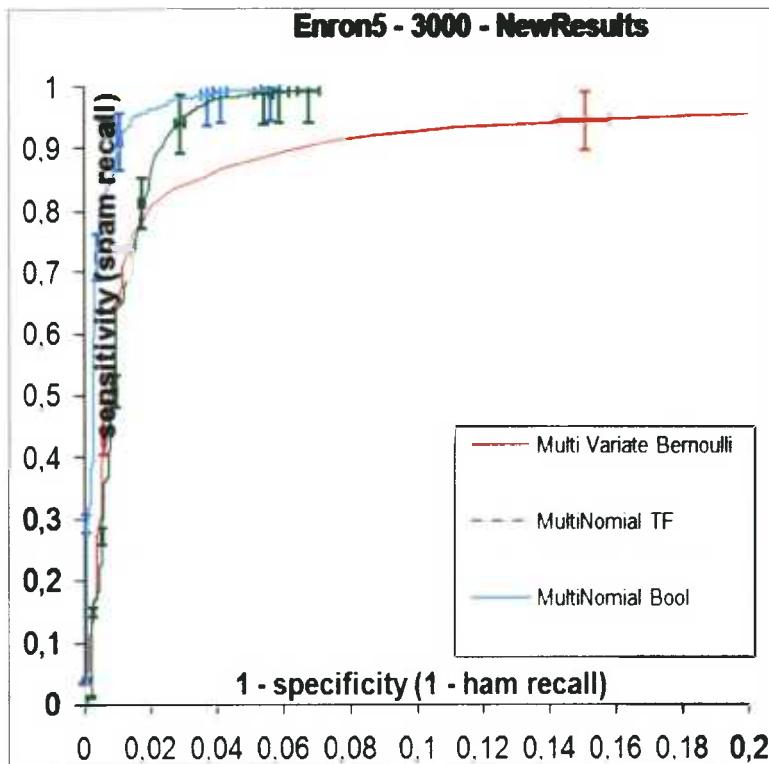
Τελικά οι μορφές του απλοϊκού ταξινομητή Bayes που κρατήσαμε για να συγκρίνουμε με το άρθρο [3] ήταν οι :

- πολυμεταβλητή μορφή Bernoulli (ενότητα 2.6.1),
- πολυωνυμική μορφή με ιδιότητες TF (ενότητα 2.6.2),
- πολυωνυμική μορφή με δυαδικές ιδιότητες (ενότητα 2.6.2).

Οι «περίοδοι» αποτελούνταν από 100 μηνύματα. Στα διαγράμματα των επομένων σελίδων παρουσιάζουμε τις καμπύλες ROC των παραπάνω μορφών των απλοϊκών ταξινομητών Bayes στις έξι υποσυλλογές μηνυμάτων της συλλογής Enron-Spam για  $m = 3000$ . Τα πειράματα έγιναν και για  $m = 500$  και  $m = 1000$ , αλλά τα αποτελέσματα και για αυτές τις τιμές  $m$  οδηγούν σε παρόμοια συμπεράσματα. Όλα τα διαγράμματα δείχνουν ενδεικτικά και τα διαστήματα εμπιστοσύνης 95% σε μερικά σημεία των καμπυλών ROC. Στην περίπτωση όπου δεν χρησιμοποιείται επιλογή ιδιοτήτων βάσει πληροφοριακού κέρδους ( $m = \text{all}$ ), παρατηρήσαμε ότι σε καμία περίπτωση οι ιδιότητες δεν ήταν περισσότερες από 3000, οπότε τα αποτελέσματα είναι τα ίδια με εκείνα της περίπτωσης  $m = 3000$ . Αυτό όμως δεν συνέβαινε πάντα στα πειράματα των επόμενων ενοτήτων.

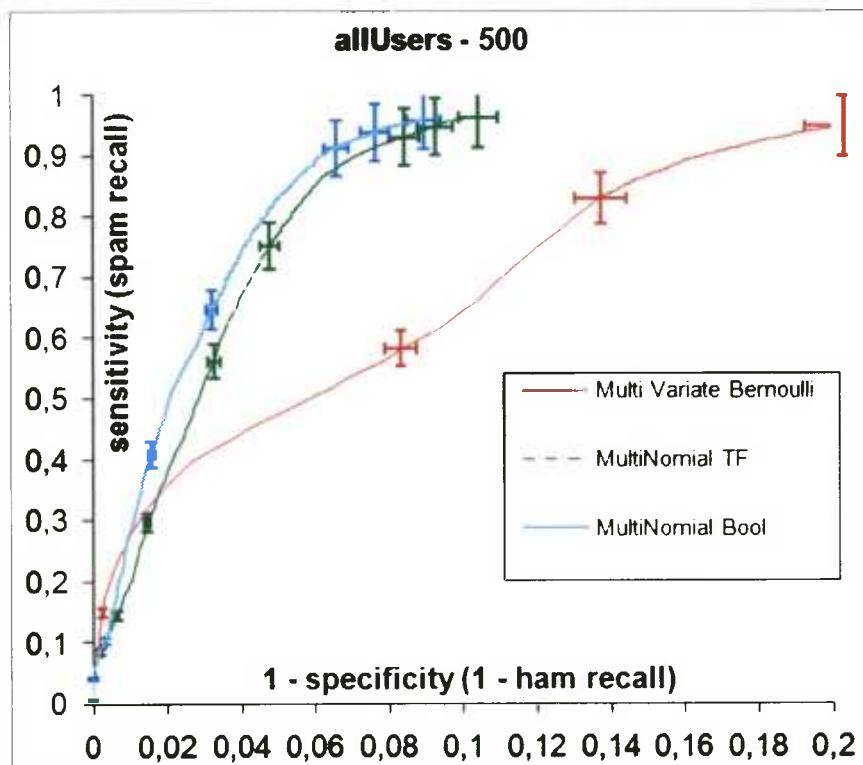
**Enron1 - 3000 - New Results****Enron1 - 3000 Attributes****Enron2 - 3000 - New Results****Enron2 - 3000 Attributes**

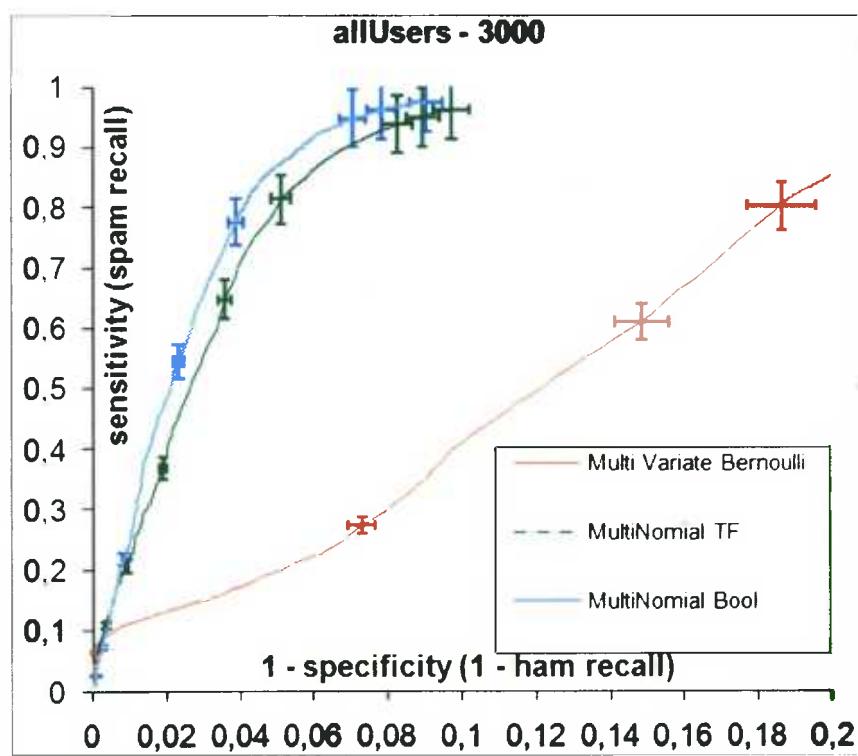
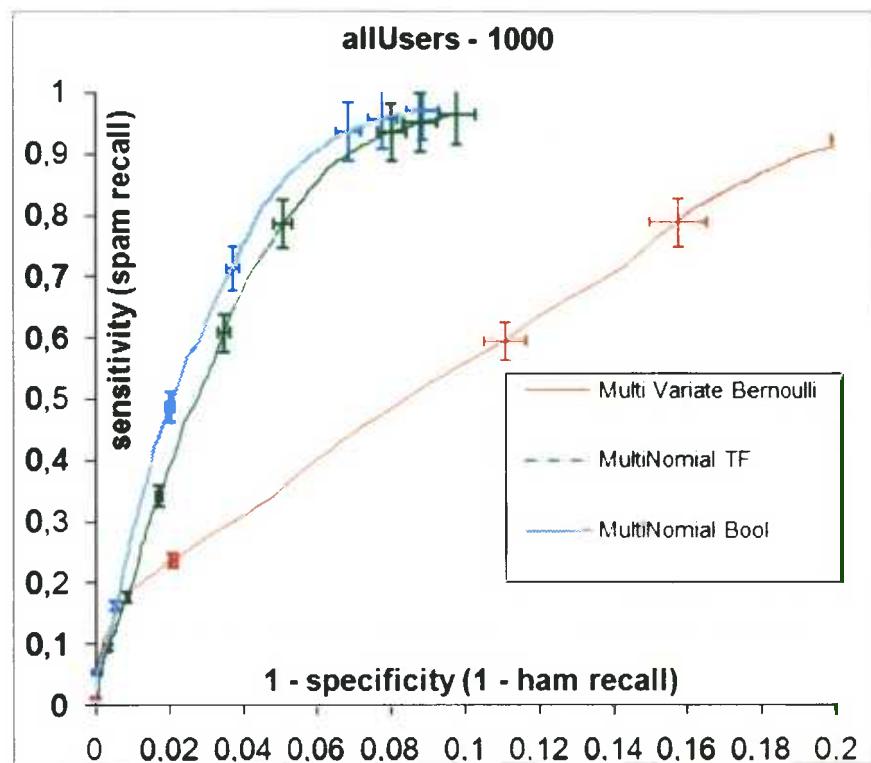




Παρατηρούμε ότι στην πλειοψηφία των υποσυλλογών («χρηστών») της Enron-Spam αποδίδει καλύτερα ο πολυωνυμικός ταξινομητής με δυαδικές ιδιότητες, κάτι που συμφωνεί με τα συμπεράσματα του άρθρου [3]. Ο πολυωνυμικός ταξινομητής με TF ιδιότητες αποδίδει πάντοτε λίγο χειρότερα από τον πολυωνυμικό με δυαδικές ιδιότητες, κάτι που συμφωνεί επίσης με τα συμπεράσματα του άρθρου [3]. Αυτό που μας κάνει εντύπωση είναι πως ο πολυμεταβλητός ταξινομητής Bernoulli σε κάποιες περιπτώσεις (Enron 1 και 2) αποδίδει ιδιαίτερα καλά, σε αντίθεση με τα αποτελέσματα του άρθρου [3]. Ίσως γι' αυτό να ευθύνεται κάποια διαφορά στην υλοποίηση. Πάντως γενικά οι πολυωνυμικές μορφές δείχνουν να έχουν πολύ πιο σταθερές επιδόσεις, κάτι που επίσης συμφωνεί με την εργασία [3].

Παρατηρούμε και πάλι ότι καλύτερα αποτελέσματα επιτυγχάνει ο πολυωνυμικός ταξινομητής με δυαδικές ιδιότητες, ακολουθούμενος από τον πολυωνυμικό ταξινομητή με ιδιότητες TF και τον πολυμεταβλητό ταξινομητή Bernoulli. Παρακάτω παρουσιάζουμε τα αποτελέσματα για τους έξι «χρήστες» της Enron-Spam μαζί, για  $m = 500, 1000, 3000$ .



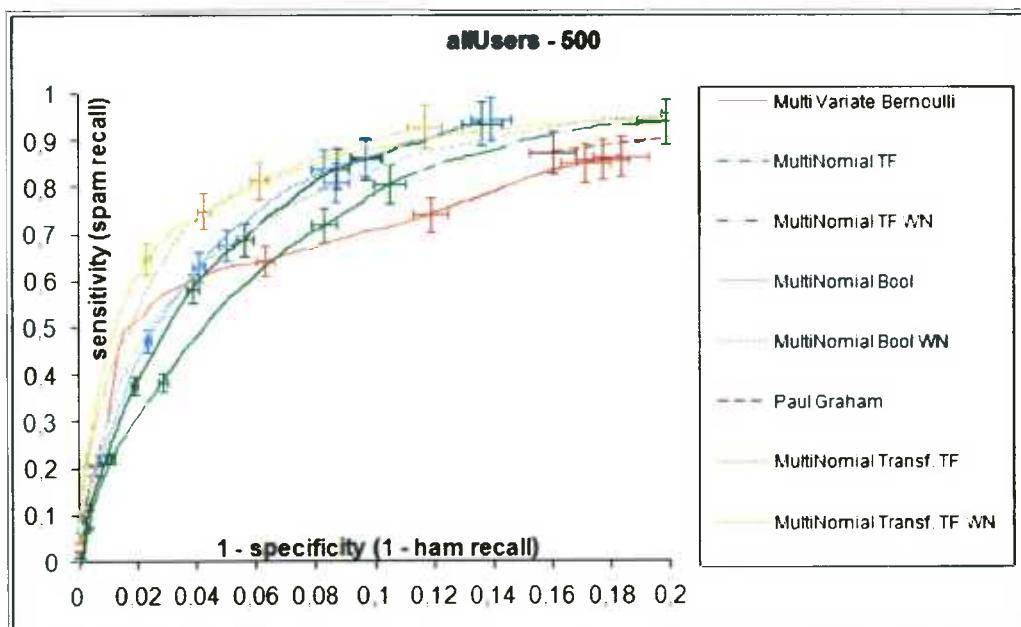


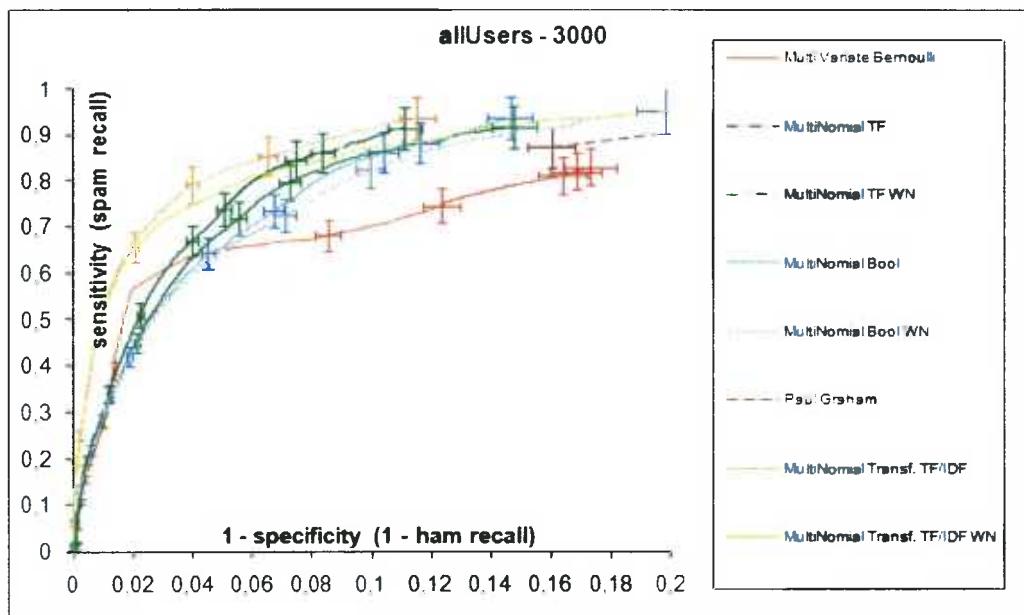
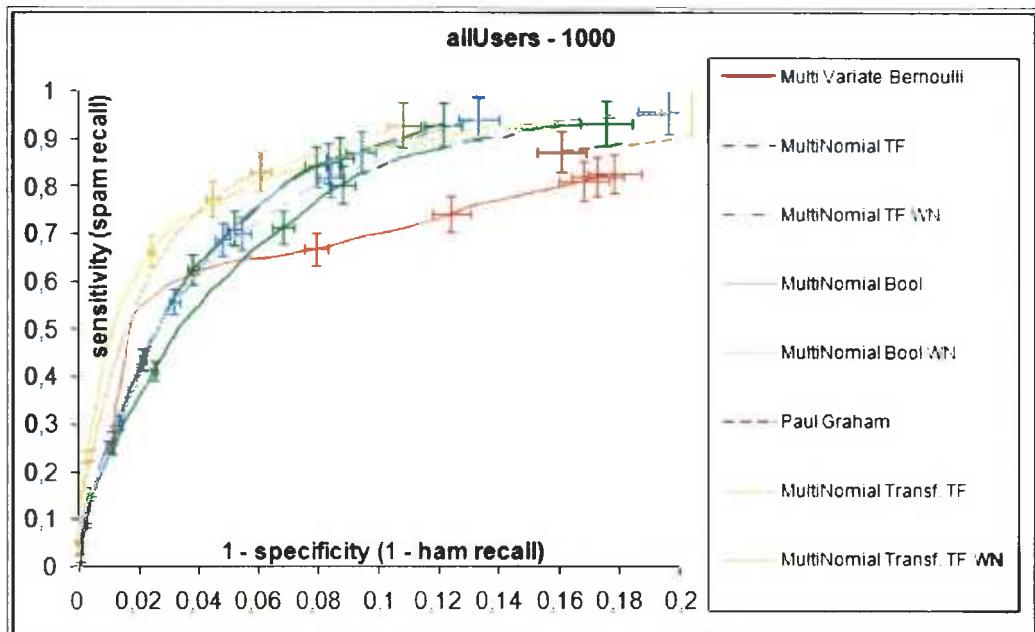
### 3.3 Πειράματα επιλογής της καλύτερης μορφής απλοϊκού ταξινομήτη Bayes

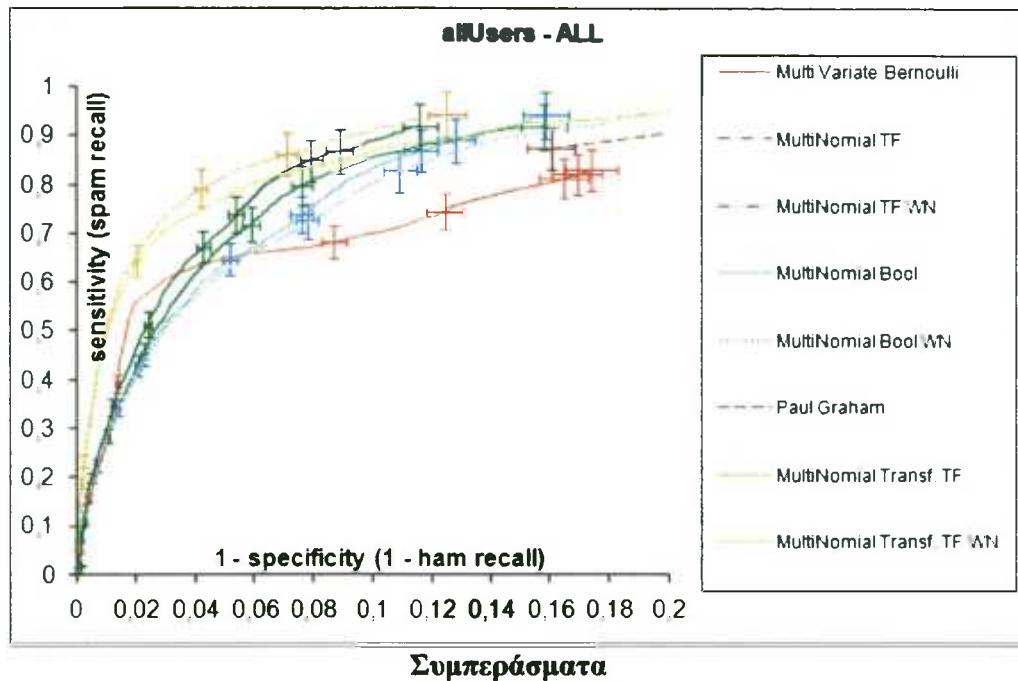
Στη συνέχεια, αφού επιβεβαιώσαμε πως το λογισμικό της εργασίας λειτουργεί ικανοποιητικά, επαναλάβαμε τα ίδια πειράματα με τη συλλογή του Δημόκριτου, που αποτελείται από πραγματικές ροές μηνυμάτων χρηστών. Προσθέσαμε όμως και τον πολυωνυμικό ταξινομητή με μετασχηματισμένες ιδιότητες TF (ενότητα 2.6.2), που δεν τον είχαμε περιλάβει στα πειράματα της προηγούμενης ενότητας επειδή δεν είχε χρησιμοποιηθεί στο άρθρο [3]. Επίσης συμπεριλάβαμε τον αλγόριθμο του Paul Graham (ενότητα 2.6.5).

Σκοπός των πειραμάτων αυτών ήταν να επιλέξουμε την «καλύτερη» μορφή απλοϊκού ταξινομήτη Bayes και την καλύτερη τιμή του  $m$  (αριθμός ιδιοτήτων που κρατάμε μετά την αξιολόγησή τους βάσει πληροφοριακού κέρδους). Στην περίπτωση του αλγορίθμου του Paul Graham, δεν χρησιμοποιείται αξιολόγηση ιδιοτήτων βάσει πληροφοριακού κέρδους, οπότε το  $m$  δεν παίζει ρόλο. Στα πειράματα αυτής της ενότητας, η κάθε «περίοδος» περιείχε τα μηνύματα μίας ημέρας.

Στη συνέχεια παραθέτουμε τα συνολικά διαγράμματα ROC για τους έξι χρήστες της συλλογής του Δημόκριτου μαζί, για  $m = 500, 1000, 3000$  και all.

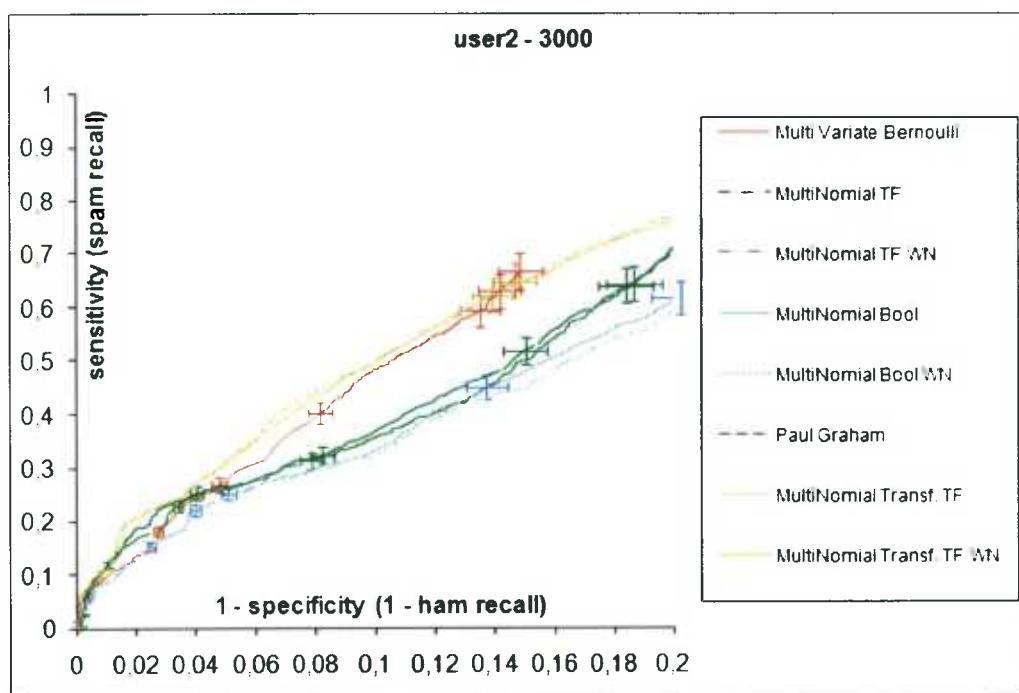
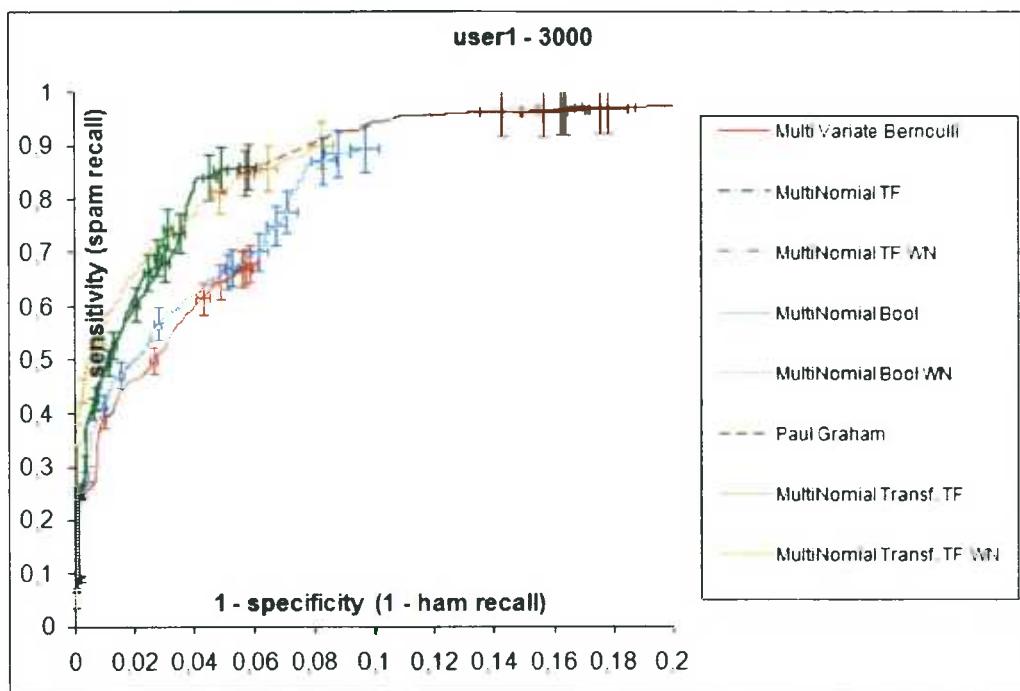


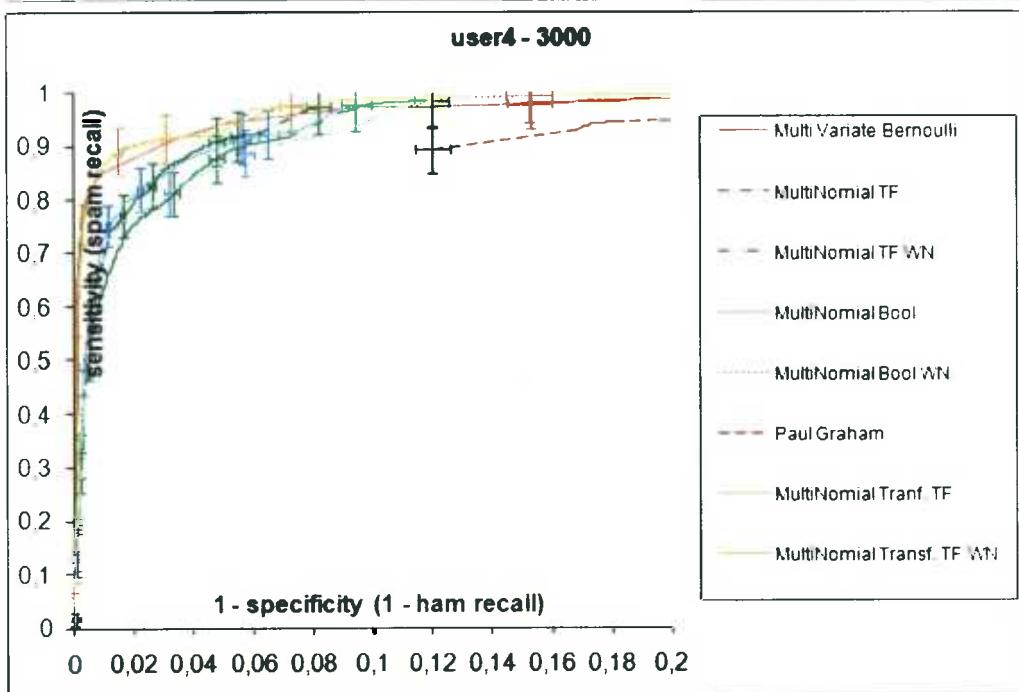
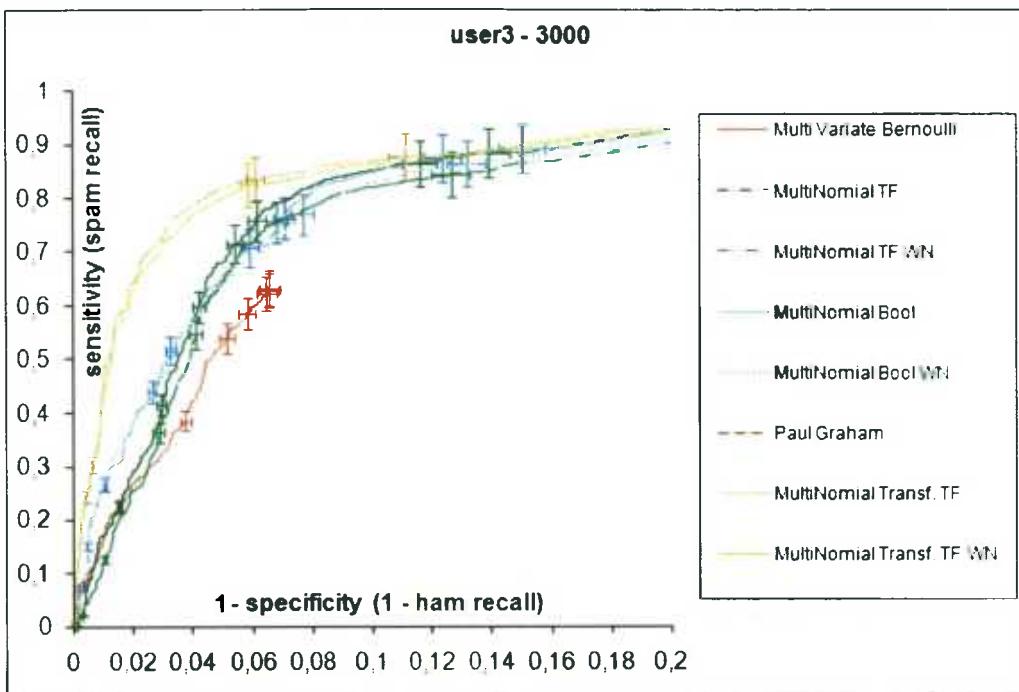


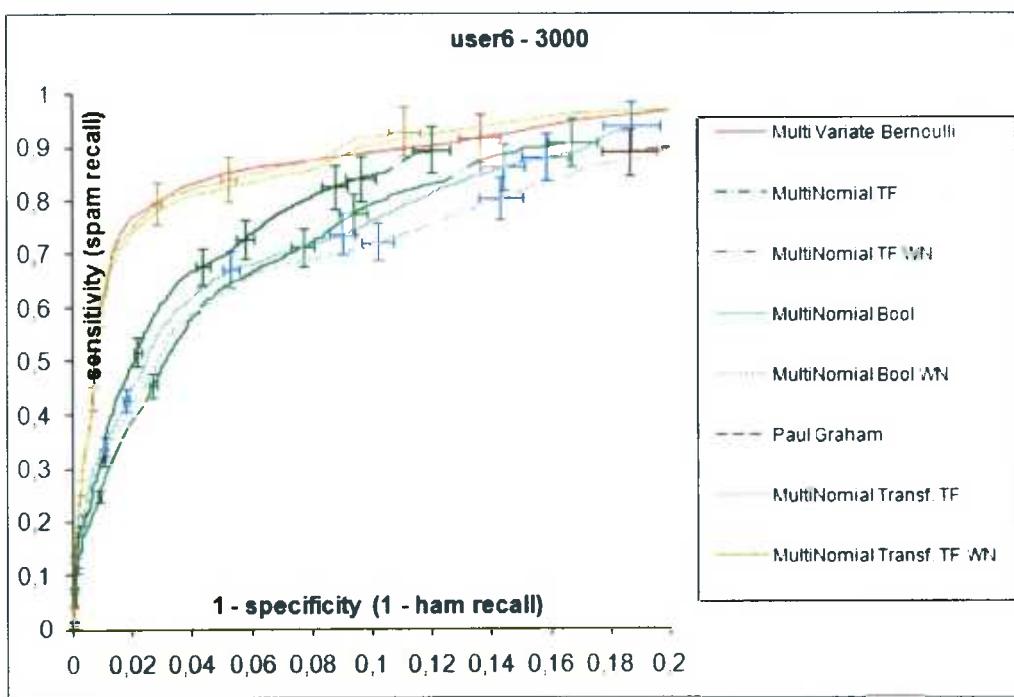
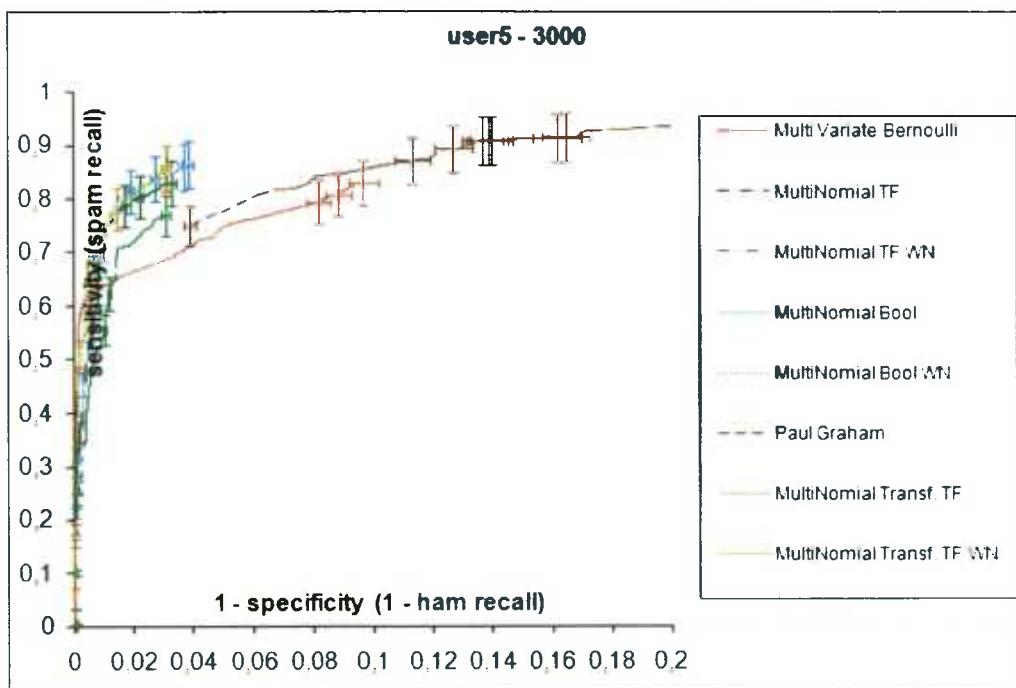


1. Από τα παραπάνω διαγράμματα φαίνεται ξεκάθαρα η υπεροχή του πολυωνυμικού ταξινομητή με μετασχηματισμένες ιδιότητες TF, ο οποίος δεν είχε χρησιμοποιηθεί στα πειράματα της εργασίας [3].. Γι' αυτό και επιλέξαμε αυτή τη μορφή του απλοϊκού ταξινομητή Bayes για την πραγματοποίηση των υπόλοιπων πειραμάτων.
2. Επιβεβαιώνεται το συμπέρασμα της εργασίας [3] ότι η πολυμεταβλητή μορφή Bernoulli του απλοϊκού ταξινομητή Bayes υστερεί έναντι των πολυωνυμικών μορφών.
3. Δεν επιβεβαιώνεται όμως το συμπέρασμα της εργασίας [3] ότι η πολυωνυμική μορφή με δυαδικές ιδιότητες υπερτερεί της πολυωνυμικής μορφής με ιδιότητες TF.
4. Ο αλγόριθμος του Paul Graham δεν καταφέρνει να αποδώσει καλά σε σχέση με τους πολυωνυμικούς ταξινομητές, ενώ εντυπωσιακό είναι το γεγονός πως δεν καταφέρνει να πετύχει ιδιαίτερα υψηλές τιμές ανάκλησης επιθυμητών μηνυμάτων. Αυτό σημαίνει πως δεν καταφέρνει να περιορίσει ικανοποιητικά τα λάθη τύπου 2, που είναι και τα σημαντικότερα.
5. Παρατηρούμε ότι σε όλες τις περιπτώσεις η κανονικοποίηση των πιθανοτήτων  $p(t|c)$  χειροτερεύει τα αποτελέσματα, αντίθετα από τα συμπεράσματα της εργασίας [13].

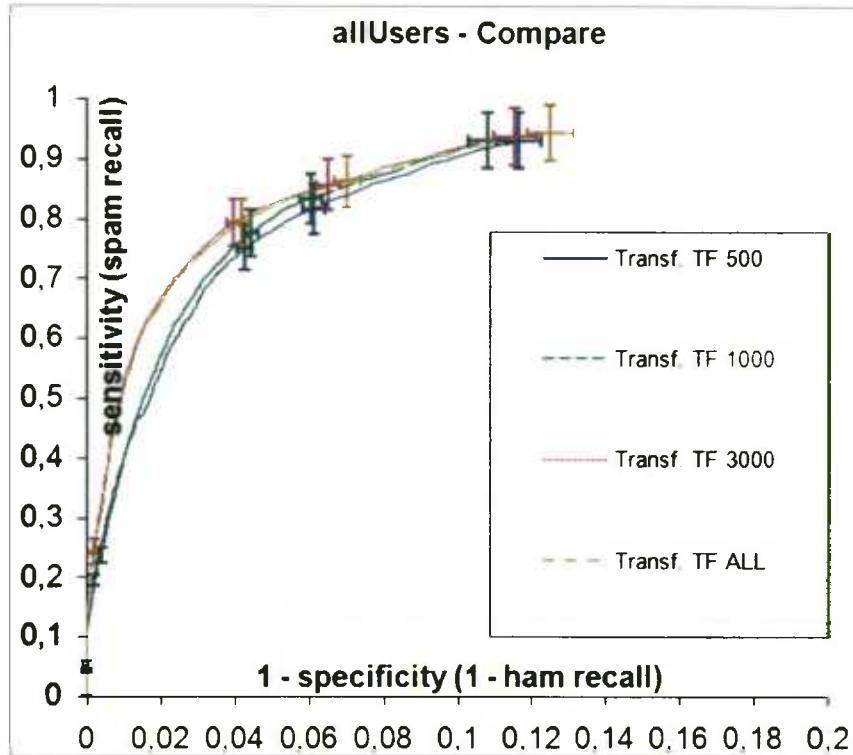
Για λόγους πληρότητας, δείχνουμε στη συνέχεια και τα διαγράμματα ROC για κάθε χρήστη του Δημόκριτου ξεχωριστά, για  $m = 3000$ .







Για να επιλέξουμε την «καλύτερη» τιμή του  $\pi$ , δημιουργήσαμε το ακόλουθο διάγραμμα, στο οποίο φαίνονται τα αποτελέσματα του πολυωνυμικού ταξινομητή με μετασχηματισμένες ιδιότητες TF για όλες τις τιμές του  $\pi$  που δοκιμάσαμε.



Παρατηρούμε ότι για  $m = 3000$  έχουμε καλύτερα αποτελέσματα από ό, τι για  $m = 500$  ή  $1000$ , όπως θα περίμενε κανείς, αν και οι διαφορές δεν είναι στατιστικά σημαντικές. Βλέπουμε, ωστόσο, ότι οι διαφορές μεταξύ  $m = 3000$  και  $m = \text{all}$  είναι ασήμαντες, μάλλον γιατί δεν υπάρχουν πολύ περισσότερες από  $3000$  υποψήφιες ιδιότητες. Επειδή, λοιπόν, τόσο για  $m = 3000$  όσο και για  $m = \text{all}$  καταλήγουμε να έχουμε περίπου τον ίδιο αριθμό ιδιοτήτων, η επιλογή ιδιοτήτων μέσω πληροφοριακού κέρδους απλά επιβραδύνει την επανεκπαίδευση του ταξινομητή, χωρίς να προσφέρει κανένα όφελος. Για το λόγο αυτό, επιλέξαμε  $m = \text{all}$  στα υπόλοιπα πειράματα.

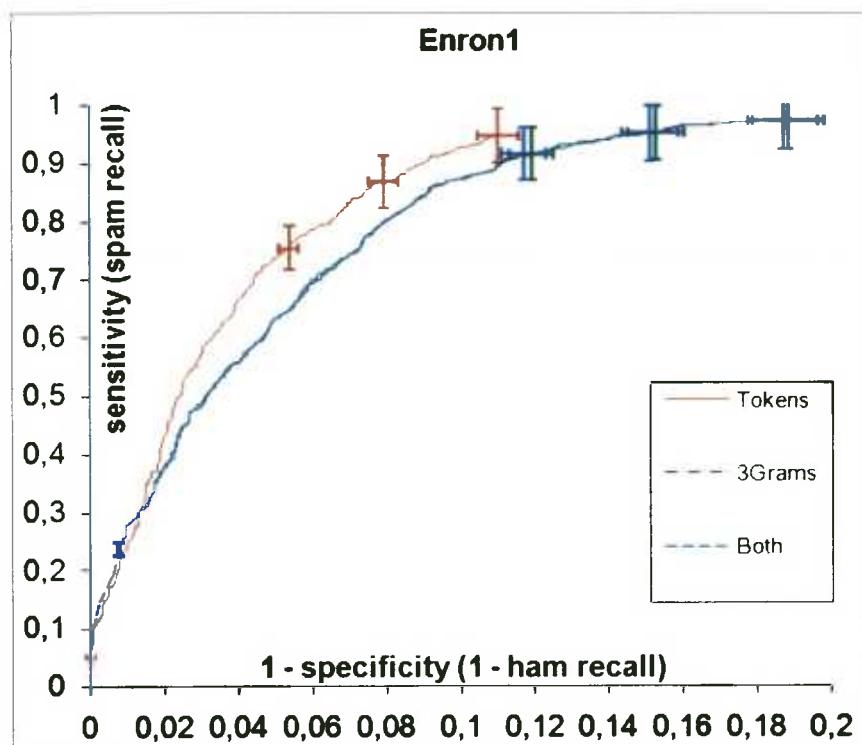
### 3.4 Ιδιότητες που αντιστοιχούν σε $n$ -γράμματα χαρακτήρων

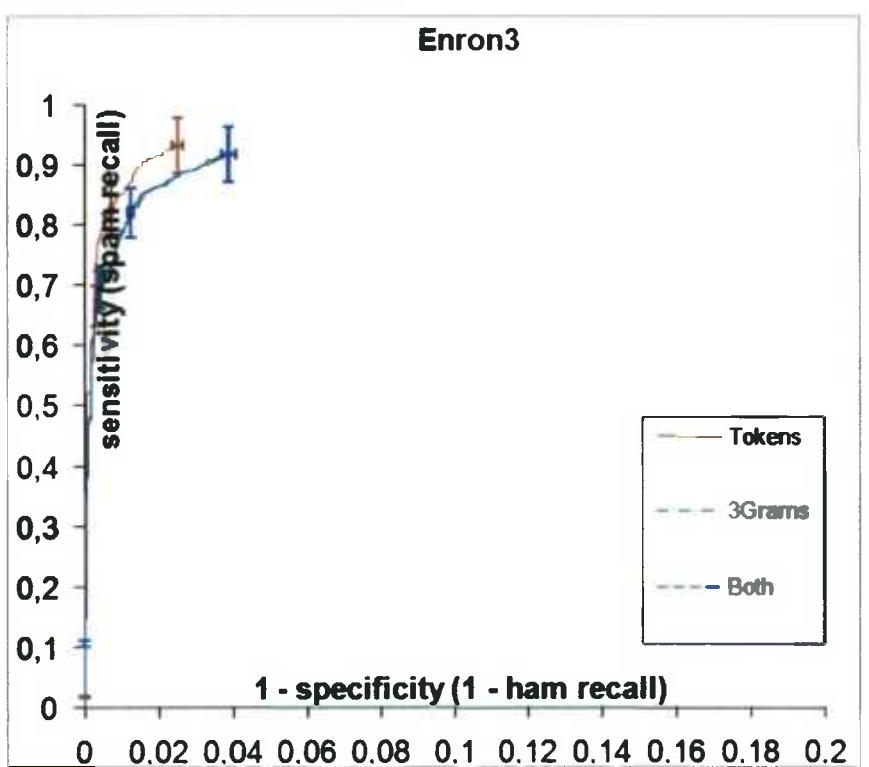
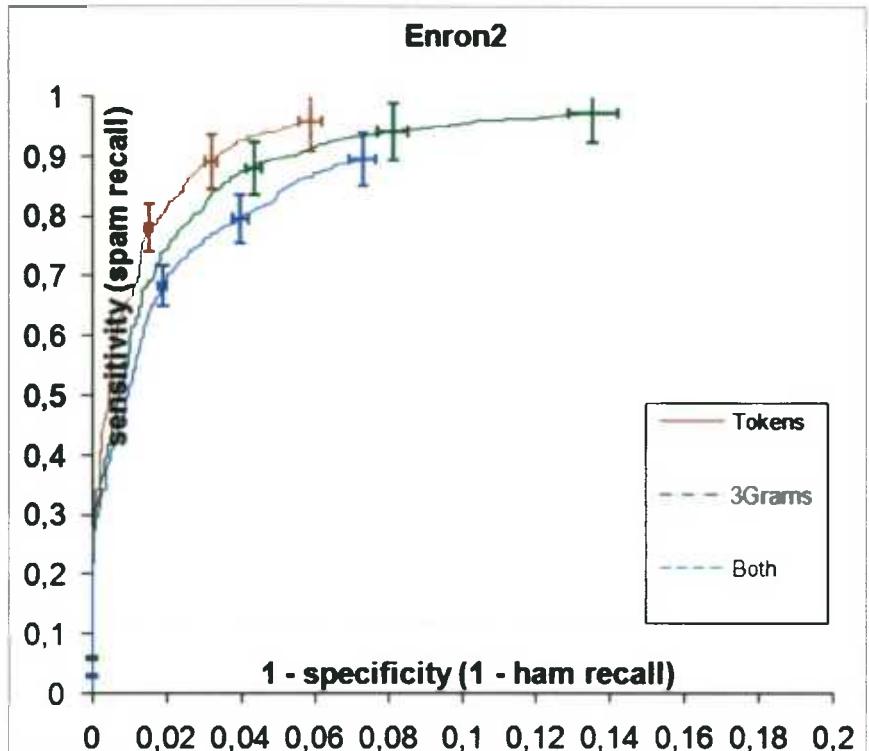
Στόχος αυτού του σταδίου των πειραμάτων ήταν να δούμε μήπως θα έπρεπε να χρησιμοποιήσουμε ιδιότητες που αντιστοιχούν σε  $n$ -γράμματα χαρακτήρων αντί για λεκτικές μονάδες. Ουμίζουμε ότι δεν είναι δυνατόν να πραγματοποιηθούν πειράματα με  $n$ -γράμματα στη συλλογή μηνυμάτων του Δημόκριτου, λόγω της αντικατάστασης των λεκτικών μονάδων από αριθμούς. Επομένως, στη φάση αυτή των πειραμάτων χρησιμοποιήσαμε αναγκαστικά τη συλλογή Enron-Spam. Η μορφή του απλοϊκού ταξινομητή Bayes που χρησιμοποιήσαμε ήταν η πολυωνυμική με μετασχηματισμένες ιδιότητες TF, χωρίς κανονικοποίηση των πιθανοτήτων  $p(t|c)$ , δηλαδή η μορφή που έδωσε τα καλύτερα αποτελέσματα κατά την προηγούμενη φάση. Και σε αυτήν την περίπτωση δεν λάβαμε υπόψιν μας λεκτικές μονάδες ή  $n$ -γράμματα που δεν εμφανίζονταν σε τουλάχιστον 5 μηνύματα εκπαίδευσης. Επίσης δεν εφαρμόσαμε το στάδιο επιλογής των  $m$  ιδιοτήτων με το υψηλότερο πληροφοριακό κέρδος, για τους λόγους που αναλύσαμε στην προηγούμενη ενότητα.

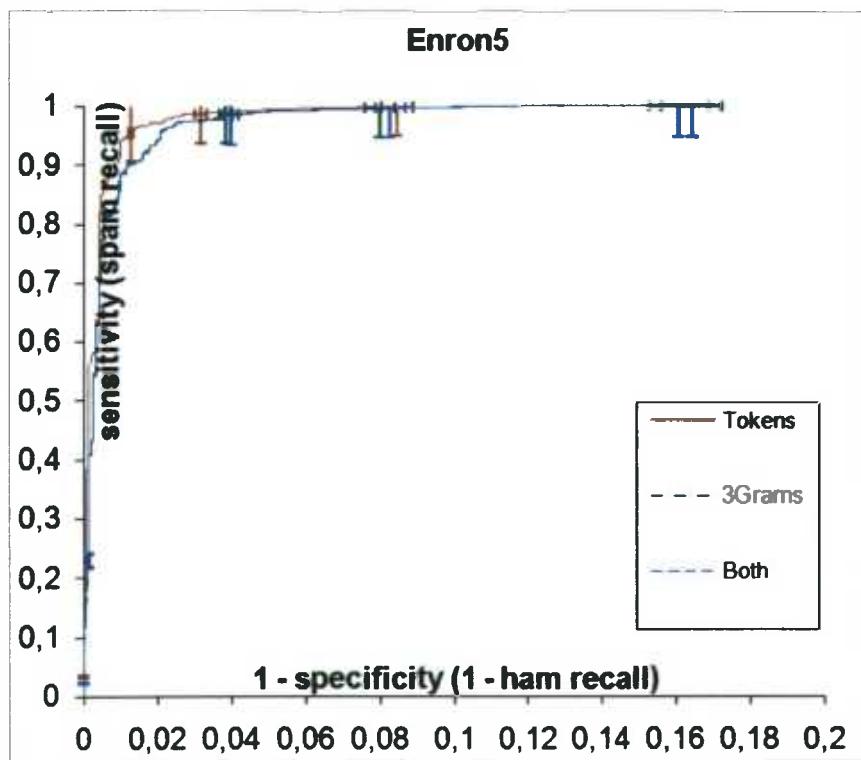
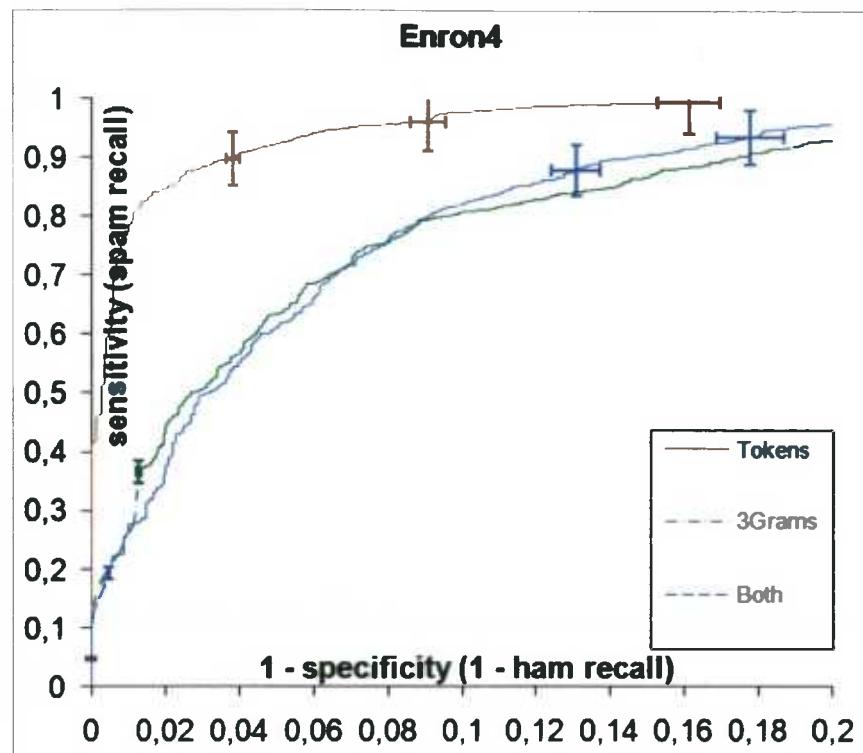
Στα πειράματα αυτά συγκρίναμε τρεις διαφορετικές περιπτώσεις. Στην πρώτη είχαμε ιδιότητες που αντιστοιχούσαν σε λεκτικές μονάδες, όπως στο προηγούμενο στάδιο. Στην δεύτερη

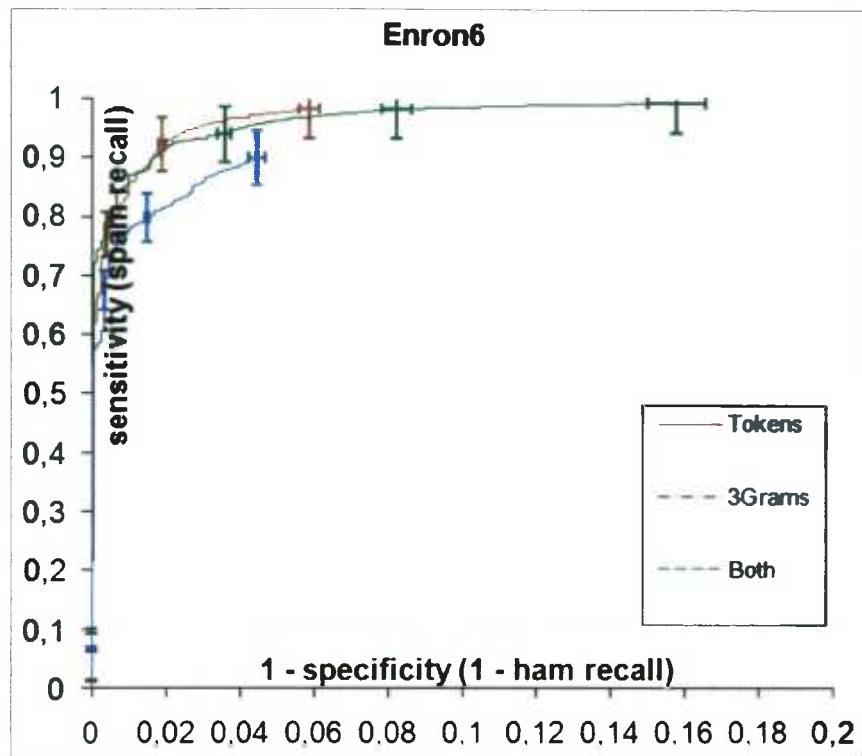
περίπτωση, χρησιμοποιήσαμε ιδιότητες που αντιστοιχούσαν σε τριγράμματα, δηλαδή ακολουθίες από τρεις συνεχόμενους χαρακτήρες. Η τελευταία περίπτωση συνδυάζει και τις δύο μεθόδους, δηλαδή χρησιμοποιήσαμε τόσο ιδιότητες που αντιστοιχούσαν σε λεκτικές μονάδες, όσο και ιδιότητες που αντιστοιχούσαν σε τριγράμματα.

Όπως φαίνεται από τα ακόλουθα διαγράμματα, είναι ξεκάθαρο πως τα τριγράμματα δεν βοηθούν σε καμία περίπτωση. Ακόμα και στην περίπτωση του συνδυασμού των δύο μεθόδων (λεκτικές μονάδες και τριγράμματα μαζί), όπου τα αποτελέσματα σε κάποιες περιπτώσεις ξεπερνούν αυτά των τριγραμμάτων, ποτέ δεν παρατηρούμε καλύτερα αποτελέσματα από αυτά των λεκτικών μονάδων μόνων τους.









### 3.5 Πειράματα με ανταλλαγή φίλτρων και ομαδικό φίλτρο

Στην τελευταία φάση των πειραμάτων διερευνήσαμε αν η ανταλλαγή φίλτρων με άλλους χρήστες βοηθάει στον εντοπισμό ανεπιθύμητων μηνυμάτων. Η μέθοδος αυτή θα συγκριθεί με την περίπτωση, όπου ο κάθε χρήστης έχει μόνο το προσωπικό του φίλτρο, αλλά και με την περίπτωση όπου όλοι οι χρήστες έχουν ένα κοινό φίλτρο, που εκπαιδεύεται στα προηγούμενα μηνύματα όλων των χρηστών.

Τα πειράματα πραγματοποιήθηκαν πάλι με ιδιότητες που αντιστοιχούν σε λεκτικές μονάδες, αγνοώντας λεκτικές μονάδες που δεν εμφανίζονται σε τουλάχιστον πέντε μηνύματα εκπαίδευσης. Δεν χρησιμοποιήθηκε η επιλογή ιδιοτήτων βάσει πληροφοριακού κέρδους. Χρησιμοποιήσαμε τη συλλογή μηνυμάτων του Δημόκριτου και τον πολυωνυμικό απλοϊκό ταξινομητή Bayes με μετασχηματισμένες ιδιότητες TF χωρίς κανονικοποίηση των  $p(t|c)$ .

Τα πειράματα αποτελούνται από 213 περιόδους της μίας ημέρας η κάθε μία. Στο τέλος κάθε μέρας ο κάθε χρήστη επανεκπαιδεύει το φίλτρο του, αφού διορθώσει τα λάθη που έκανε το φίλτρο του στα μηνύματα που έλαβε στη διάρκεια της ημέρας. Το ομαδικό φίλτρο επανεκπαιδεύεται σε όλα τα (διορθωμένα) μηνύματα που έχουν λάβει όλοι οι χρήστες στο παρελθόν. Στην περίπτωση ανταλλαγής φίλτρων, στο τέλος κάθε ημέρας κάθε χρήστης επανεκπαιδεύει πάλι το ατομικό του φίλτρο, αφού διορθώσει τα λάθη που έκανε στα μηνύματα που έλαβε στη διάρκεια της ημέρας, αλλά στη συνέχεια δίνει πρόσβαση σε αυτό σε όλους τους άλλους χρήστες. Συγχρόνως αποκτά πρόσβαση στα φίλτρα των υπολοίπων και έρχεται η ώρα να αποφασίσει σε τι βαθμό θα εμπιστευτεί το καθένα απ' αυτά κατά την κατάταξη των μηνυμάτων που θα λάβει την επόμενη μέρα. Στα πειράματά μας χρησιμοποιήσαμε τη μέθοδο που περιγράφαμε στην ενότητα 2.7 για τον προσδιορισμό του βαθμού εμπιστοσύνης σε κάθε φίλτρο.

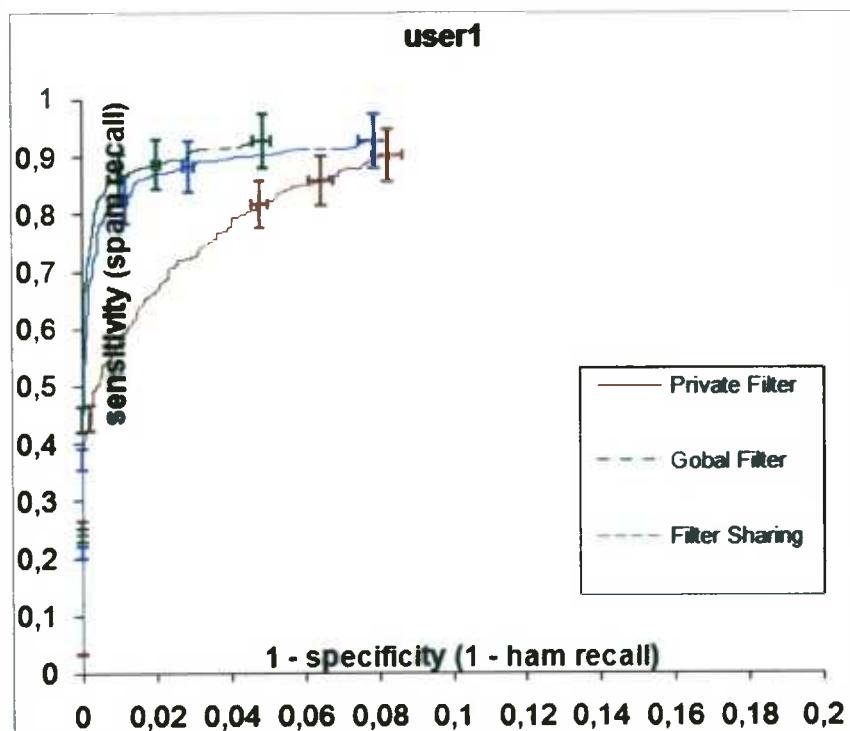
Κατά την κατάταξη ενός νεοεισερχόμενου μηνύματος, χρησιμοποιούμε κάθε φίλτρο, και πολλαπλασιάζουμε τα  $\text{Pham}_i$  και  $\text{Pspam}_i$ , που μας δίνει με το βαθμό εμπιστοσύνης  $w_i$ . Στη συνέχεια θέτουμε το τελικό  $\text{Pham}$  ίσο το άθροισμα όλων των  $\text{Pham}_i \cdot w_i$ , και το τελικό  $\text{Pspam}$  ίσο με το άθροισμα όλων των  $\text{Pspam}_i \cdot w_i$ .

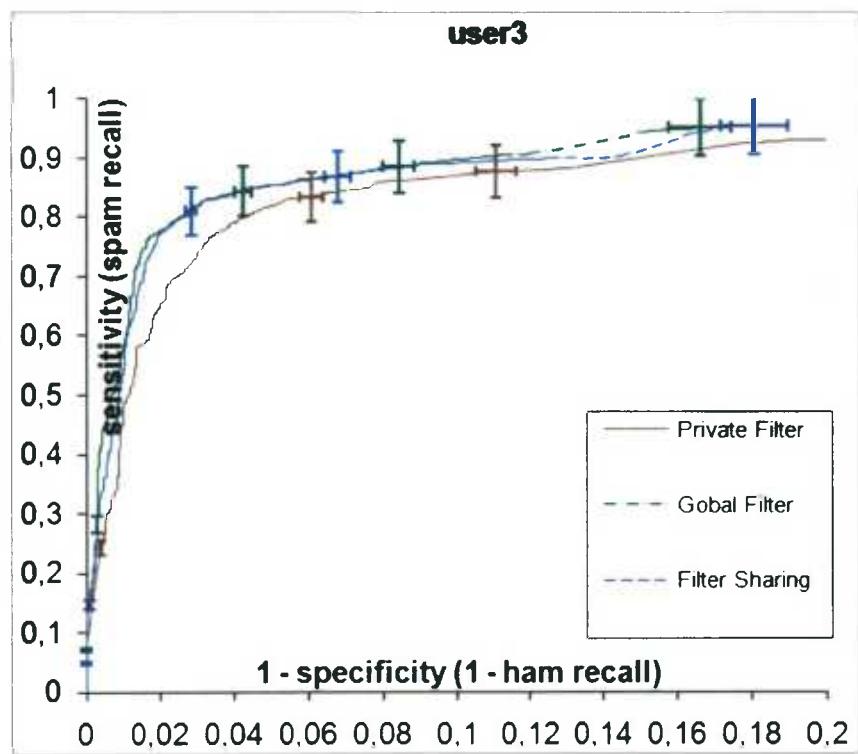
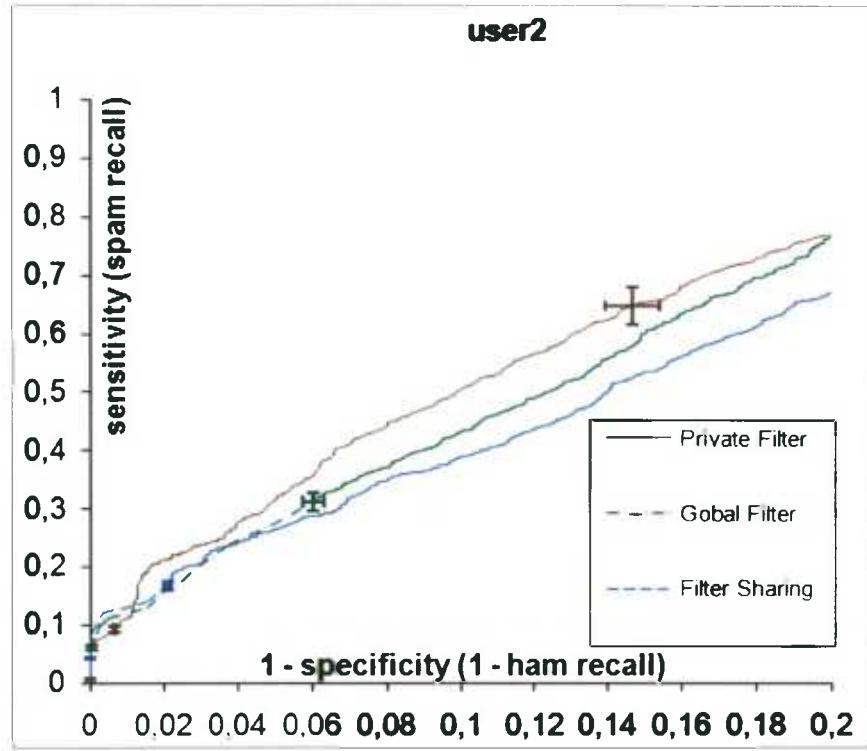
Στη διαδικασία αυτή υπάρχει μια ιδιαιτερότητα που αφορά τον τρόπο προσδιορισμού των  $w_i$ , και η οποία πρέπει να αναφερθεί. Για να μπορέσουμε να υπολογίσουμε τα βάρη αυτά, πρέπει να υπολογίσουμε το σύνολο των προηγουμένων μηνυμάτων του συγκεκριμένου χρήστη που κάθε φίλτρο κατατάσσει σωστά. Για να μπορέσουμε όμως να πάρουμε αυτήν την πληροφορία, πρέπει να χρησιμοποιήσουμε μια συγκεκριμένη τιμή του  $\Delta$ . Θυμίζουμε ότι ένα μήνυμα κατατάσσεται ως ανεπιθύμητο ανν:

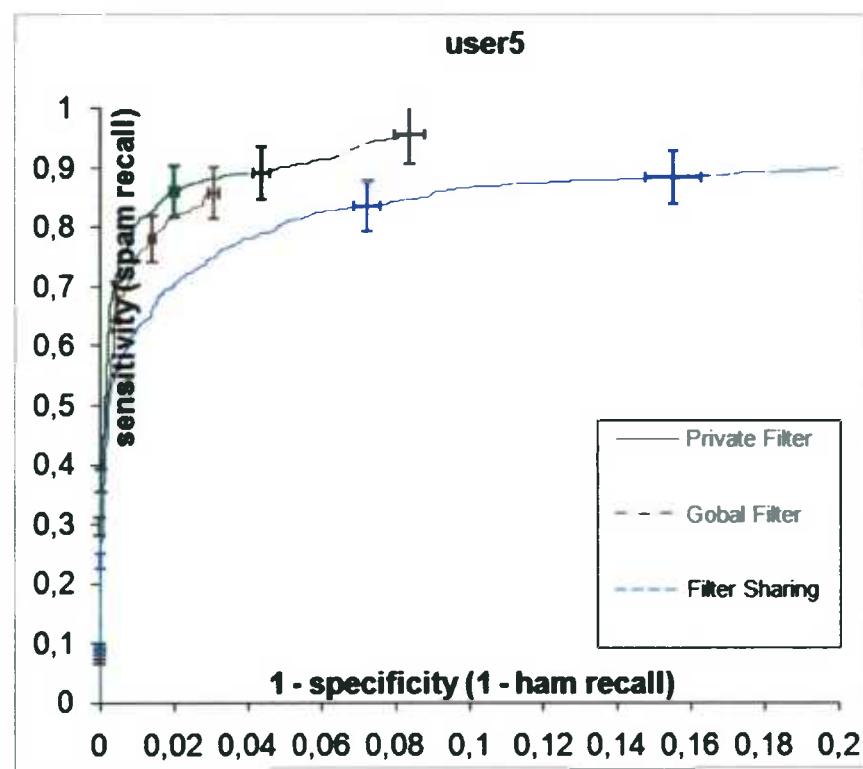
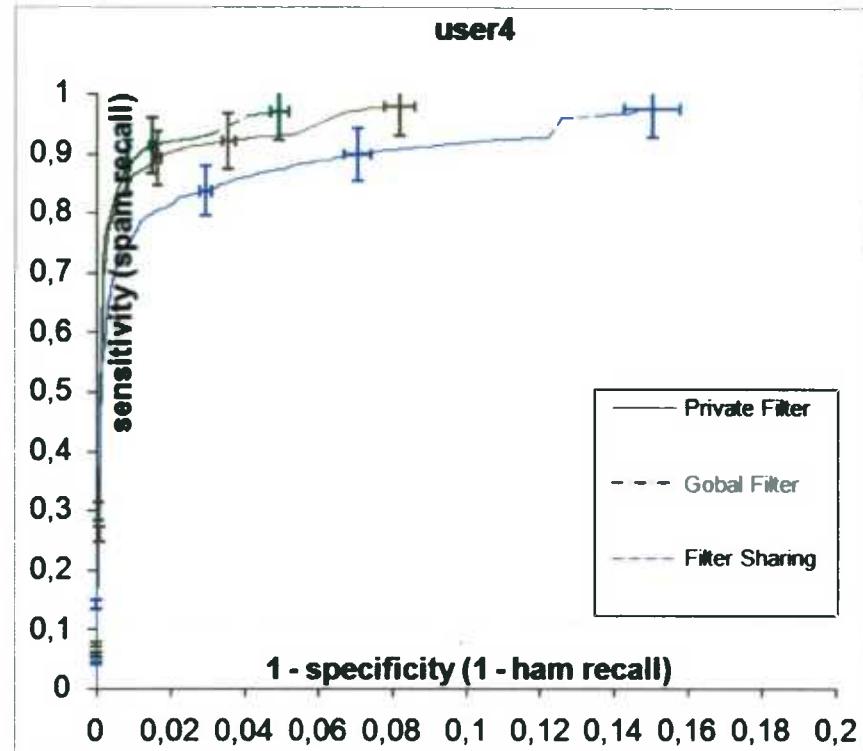
$$[\log(p(c_s)) + \log(p(\vec{x}|c_s))] - [\log(p(c_h)) + \log(p(\vec{x}|c_h))] > \Delta$$

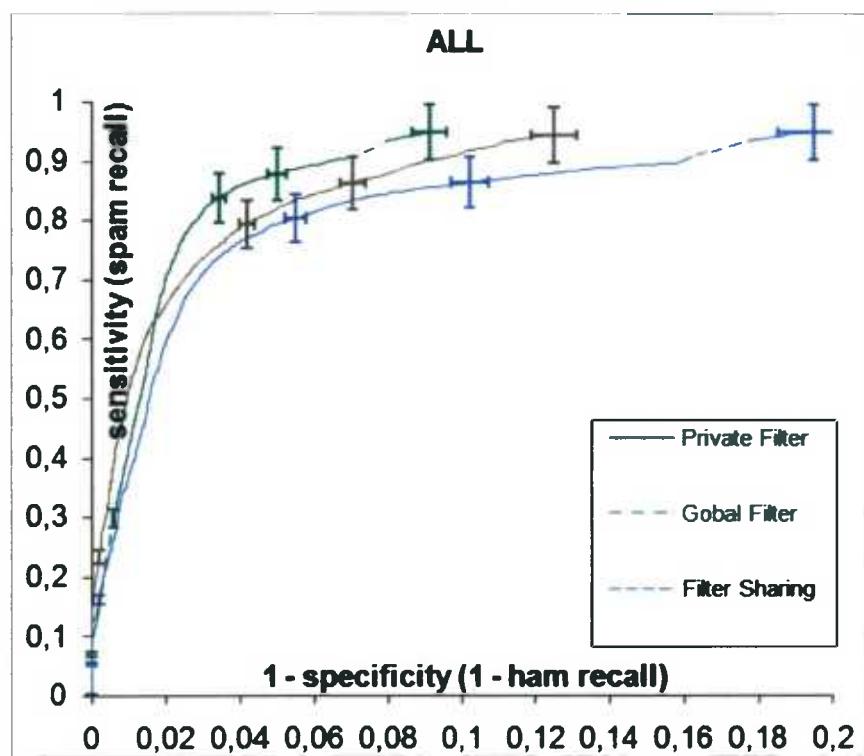
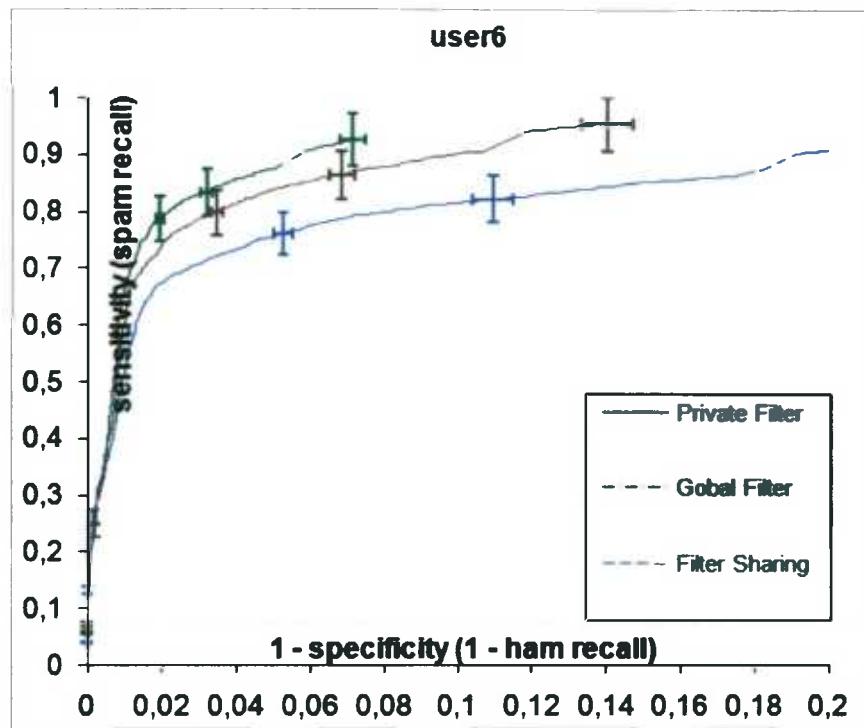
Τα πειράματα έγιναν για  $\Delta = 0$  και πρέπει να γίνει κατανοητό πως για μια άλλη τιμή τα αποτελέσματα και τα συμπεράσματα μπορεί να ήταν διαφορετικά. Μια καλύτερη προσέγγιση θα ήταν τα  $\text{Pham}_i$  και  $\text{Pspam}_i$  των φίλτρων να αποτελούν ιδιότητες ενός άλλου αλγορίθμου μάθησης, που θα παρήγαγε για κάθε χρήστη έναν ταξινομητή ο οποίος θα μάθαινε να κατατάσσει τα εισερχόμενα μηνύματα του χρήστη βάσει των τιμών  $\text{Pham}_i$  και  $\text{Pspam}_i$  των έξι ατομικών φίλτρων. Επιστρέφουμε σε αυτό το σενάριο στο επόμενο κεφάλαιο.

Στη συνέχεια παραθέτουμε τα αποτελέσματα για κάθε χρήστη της συλλογής, αλλά και όλων μαζί αθροιστικά.









## Συμπεράσματα

Σύμφωνα με τα παραπάνω διαγράμματα, η ανταλλαγή φίλτρων δεν δείχνει να υπερτερεί έναντι της χρήσης ατομικών ή ομαδικών φίλτρων σε καμία περίπτωση. Αντίθετα, το ομαδικό φίλτρο δείχνει να δίνει τα καλύτερα αποτελέσματα συνολικά. Αυτό ενισχύει τη θέση πολλών, πως είναι καλύτερα να έχουμε ένα ομαδικό φίλτρο σε έναν κεντρικό εξυπηρετητή, αντί να έχει ο κάθε χρήστης το δικό του ατομικό φίλτρο (συνήθως ως μέρος του προγράμματος ανάγνωσης ταχυδρομείου που χρησιμοποιεί). Από την άλλη, όμως, ένα κοινό φίλτρο δεν θα είχε λογικά το ίδιο καλά αποτελέσματα αν το χρησιμοποιούσαν άτομα με διαφορετικά ενδιαφέροντα και προτιμήσεις. Η συλλογή του Δημόκριτου προέρχεται από χρήστες (ερευνητές πληροφορικής) με παρεμφερή ενδιαφέροντα. Τι θα γινόταν άραγε αν ένας από τους χρήστες ήταν χρηματιστής, ενώ ένας άλλος έφηβος; Μήπως τότε το κοινό φίλτρο θα έδινε χειρότερα αποτελέσματα από τις άλλες δυο μεθόδους;

Η επιλογή  $\Delta = 0$  σε όλα τα φίλτρα, όπως αναφέραμε παραπάνω, ίσως να είναι ένας ακόμα παράγοντας που επέδρασε αρνητικά. Όπως προαναφέραμε, η ανταλλαγή φίλτρων ίσως είχε καλύτερα αποτελέσματα αν ο καθορισμός των βαρών των ατομικών φίλτρων γινόταν μέσω ενός άλλου αλγορίθμου μάθησης, που θα λάμβανε υπόψη του τα Pham<sub>i</sub> και Pspam<sub>i</sub> των ατομικών φίλτρων χωρίς να χρησιμοποιεί ένα συγκεκριμένο  $\Delta$ .

## 4. ΕΠΙΛΟΓΟΣ

### 4.1 Σύνοψη

Στην εργασία αυτή πραγματοποιήσαμε με σειρά πειραμάτων σε δυο συλλογές μηνυμάτων. Η πρώτη συλλογή (Enron-Spam) έχει χρησιμοποιηθεί για πειράματα στο παρελθόν, άρα είναι κατάλληλη για σύγκριση αποτελεσμάτων, ενώ η δεύτερη (συλλογή Δημόκριτου) χρησιμοποιείται για πρώτη φορά, αλλά είναι πιο ρεαλιστική, άρα και καταλληλότερη για εξαγωγή τελικών συμπερασμάτων. Για τη διεξαγωγή των πειραμάτων υλοποιήσαμε μια σειρά φίλτρων βασισμένων σε διάφορες μορφές του απλοϊκού ταξινομητή Bayes, μιας οικογένειας αλγορίθμων που είναι ιδιαίτερα δημοφιλής στα φίλτρα ηλεκτρονικού ταχυδρομείου λόγω της απλότητάς της και των εν γένει γραμμικών απαιτήσεών της σε χρόνο και μνήμη. Η υλοποίηση αυτή έγινε με τέτοιο τρόπο, ώστε να μπορεί να χρησιμοποιηθεί και από τρίτους, για πειραματισμό ή επέκταση.

Στην αρχική φάση των πειραμάτων διερευνήσαμε το κατά πόσο λειτουργεί σωστά η υλοποίησή μας πειραματιζόμενοι με την πρώτη συλλογή και στη συνέχεια βγάλαμε μια σειρά συμπερασμάτων πειραματιζόμενοι με τη δεύτερη:

1. Η μορφή του απλοϊκού ταξινομητή Bayes με τα καλύτερα αποτελέσματα ήταν η πολυωνυμική με ιδιότητες βασισμένες σε τιμές TF στις οποίες εφαρμόζονται επιπλέον μετασχηματισμοί. Πρόκειται για μια μορφή που είχε προταθεί στην εργασία [13], αλλά δεν είχε δοκιμαστεί σε φίλτρα ανεπιθύμητης αλληλογραφίας.
2. Ο αλγόριθμος του Paul Graham, που χρησιμοποιείται σε πολλά φίλτρα ανεπιθύμητης αλληλογραφίας και βασίζεται εν μέρει στην πολυμεταβλητή μορφή του απλοϊκού ταξινομητή Bayes, φαίνεται να έχει αισθητά χειρότερα αποτελέσματα από τις πολυωνυμικές μορφές του απλοϊκού ταξινομητή Bayes.
3. Η διαδικασία επιλογής καλύτερων ιδιοτήτων βάσει πληροφοριακού κέρδους δεν βελτιώνει τα αποτελέσματα του πολυωνυμικού απλοϊκού ταξινομητή Bayes με μετασχηματισμένες ιδιότητες TF, ενώ επιβραδύνει την επανεκπαίδευσή του.
4. Η χρήση ιδιοτήτων που αντιστοιχούν σε π-γράμματα δεν βελτιώνει τα αποτελέσματα του απλοϊκού ταξινομητή Bayes με μετασχηματισμένες ιδιότητες TF.
5. Η ανταλλαγή ατομικών φίλτρων μεταξύ συνεργατών υστερεί έναντι της χρήσης μόνο ατομικών φίλτρων, ενώ καλύτερη όλων είναι η χρήση ενός κοινού ομαδικού φίλτρου.

## 4.2 Μελλοντικές επεκτάσεις

Ο βασικότερος μελλοντικός στόχος είναι να πραγματοποιηθούν και άλλα πειράματα, με εναλλακτικούς τρόπους υλοποίησης της ανταλλαγής φύλτρων. Το πρόβλημα εστιάζεται κυρίως στον τρόπο υπολογισμού της εμπιστοσύνης προς τα διαθέσιμα ατομικά φύλτρα, που μπορεί να βελτιωθεί χρησιμοποιώντας έναν αλγόριθμο μηχανικής μάθησης που θα μαθαίνει πόσο να εμπιστεύεται το κάθε ατομικό φύλτρο.

Σημαντική επίσης μελλοντική επέκταση είναι η ενσωμάτωση του λογισμικού της εργασίας σε ένα πρόγραμμα ανάγνωσης ηλεκτρονικής αλληλογραφίας, όπως το Thunderbird. Ήδη βρίσκεται σε εξέλιξη πτυχιακή εργασία προς αυτή την κατεύθυνση. Η κίνηση αυτή είναι πολύ σημαντική, αφού θα επιτρέψει την πραγματοποίηση πειραμάτων υπό πραγματικές συνθήκες.

Αν διαπιστωθεί ότι με κάποιον εναλλακτικό τρόπο υλοποίησης η ανταλλαγή φύλτρων λειτουργεί καλύτερα, τότε ίσως αξίζει να υλοποιηθεί ένα πραγματικό σύστημα ανταλλαγής φύλτρων. Σε αυτήν την περίπτωση θα πρέπει να λάβουμε και άλλους παράγοντες υπόψιν μας, όπως με ποιους τελικά ανταλλάσσουμε φύλτρα και τι τεχνικές χρησιμοποιούμε για να κρίνουμε την αξιοπιστία τους [10, 11]. Σημαντικό ρόλο επίσης παίζει και με ποιον τρόπο θα πραγματοποιηθεί η ανταλλαγή αυτή, δηλαδή αν θα απαιτείται συγχρονισμός σε συγκεκριμένες ώρες, τι ακριβώς θα στέλνει κάθε χρήστης και αν η ανταλλαγή αυτή θα είναι απαραίτητο να είναι αμφίδρομη.

Ενδιαφέρον επίσης παρουσιάζει και η χρήση ενεργητικής μάθησης στα ατομικά φύλτρα [12], δηλαδή το να μπορεί, για παράδειγμα, το φύλτρο να ρωτάει το χρήστη μόνο για τα μηνύματα που δεν είναι πολύ σίγουρο αν είναι επιθυμητά ή όχι, χωρίς να απαιτείται ο χρήστης να επιθεωρεί περιοδικά το φάκελο ανεπιθύμητων μηνυμάτων του.

## Βιβλιογραφία

- [1] Ι. Ανδρουτσόπουλος, Γ. Παλιούρας και Ε. Μιχελάκης, *Learning to Filter Unsolicited Commercial E-Mail*. Τεχνική αναφορά 2004/2, ΕΚΕΦΕ «Δημόκριτος», 2004.
- [2] Ε. Μιχελάκης, Ι. Ανδρουτσόπουλος, Γ. Παλιούρας, Γ. Σάκκης και Π. Σταματόπουλος, «*Filtron: A Learning-Based Anti-Spam Filter*». Πρακτικά του *1st Conference on Email and Anti-Spam* (CEAS 2004), Mountain View, CA, ΗΠΑ, 2004.
- [3] Β. Μέτσης, Ι. Ανδρουτσόπουλος και Γ. Παλιούρας, «*Spam Filtering with Naive Bayes – Which Naive Bayes?*». Πρακτικά του *3rd Conference on E-mail and Anti-Spam* (CEAS 2006), Mountain View, CA, ΗΠΑ, 2006.
- [4] J. Metzger, M. Schillo και K. Fischer, «*A Multiagent-Based Peer-to-Peer Network in Java for Distributed Spam Filtering*». Πρακτικά του *3rd International/Central and Eastern European Conference on Multi-Agent Systems*, Πράγα, Τσεχία, 2003.
- [5] E. Damiani, S. De Capitani di Vimercati, S. Paraboschi και P. Samarati, «*P2P-Based Collaborative Spam Detection and Filtering*». Πρακτικά του *4th International Conference on Peer-to-Peer Computing* (P2P 2004), σελ. 176–183, Ζυρίχη, Ελβετία, 2004.
- [6] F. Zhou, L. Zhuang, B.Y. Zhao, L. Huang, A.D. Joseph και J. Kubiatowicz, «*Approximate Object Location and Spam Filtering on Peer-to-Peers Systems*». Πρακτικά του *ACM/IFIP/USENIX International Middleware Conference* (Middleware 2003), σελ. 1–20, Rio de Janeiro, Βραζιλία, 2003.
- [7] A. Garg, R. Battiti, R. Casella, *May I Borrow Your Filter?» – Exchanging Filters to Combat Spam in a Community*. Τεχνική αναφορά DIT-05-089, Department of Information and Communication Technology, University of Trento, Ιταλία, 2005.
- [8] Γ. Σάκκης, Ι. Ανδρουτσόπουλος, Γ. Παλιούρας, B. Karakalétsos, K.D. Spyrópoulos και Π. Σταματόπουλος, «*Stacking Classifiers for Anti-Spam Filtering of E-Mail*». Πρακτικά του *6th Conference on Empirical Methods in Natural Language Processing* (EMNLP 2001), Carnegie Mellon University, Pittsburgh, PA, ΗΠΑ, σελ. 44–50, 2001.
- [9] S. Hershkop και S.J. Stolfo, «*Combining Email Models for False Positive Reduction*». Πρακτικά του *11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining* (KDD 2005), σελ. 98–107, Σικάγο, Η.Π.Α., 2005.
- [10] P.O. Boykin και V.P. Roychowdhury, «*Leveraging Social Networks to Fight Spam*». *IEEE Computer*, 38(4):61–68, 2005.
- [11] P.-A. Chirita, J. Diederich και W. Nejdl, «*MailRank: Using Ranking for Spam Detection*». Πρακτικά του *14th ACM International Conference on Information and Knowledge Management*, σελ. 373–380, Βρέμη, Γερμανία, 2005.
- [12] R. Segal, T. Markowitz και W. Arnold, «*Fast Uncertainty Sampling for Labeling Large E-mail Corpora*». Πρακτικά του *3rd Conference on E-mail and Anti-Spam* (CEAS 2006), Mountain View, CA, ΗΠΑ, 2006.

- [13] Jason D. M. Rennie, Lawrence Shih, Jaime Teevan και David R. Karger «Tackling the Poor Assumptions of Naive Bayes Text Classifiers». Πρακτικά του 20ού διεθνούς συνεδρίου μηχανικής μάθησης , Washington DC, 2003.
- [14] Vlado Keselj, Evangelos Milios, Andrew Tuttle, Singer Wang, and Roger Zhang «DaLTREC 2005 Spam Track: Spam Filtering using N-gram-based Techniques».
- [15] K.-M. Schneider. A comparison of event models forNaive Bayes anti-spam e-mail filtering. In 10<sup>th</sup> Conference of the European Chapter of the ACL, pages 307–314, Budapest, Hungary, 2003.
- [16] P. Domingos and M. Pazzani. On the optimality of the simple Bayesian classifier under zero-one loss. Machine Learning, 29(2–3):103 130, 1997.
- [17] Aleksander Kolcz «Local Sparsity Control for Naive Bayes with extreme misclassification costs». Conference on Knowledge Discovery in Data 2005.
- [18] Paul Graham «A plan for spam». [www.paulgraham.com/spam.html](http://www.paulgraham.com/spam.html).
- [19] Paul Graham «Better Bayesian Filtering». [www.paulgraham.com/better.html](http://www.paulgraham.com/better.html)



