ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ
Τμήμα
ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Σύστημα Σύστασης Ταινιών με Αυτόματη Επεξεργασία Κειμένων από το Διαδίκτυο

Κεραμμύδας Παναγιώτης

Διπλωματική εργασία

Ιούλιος 2010

Εξεταστική Επιτροπή:

Αν. Καθηγητής Ποταμιάνος Αλέξανδρος (Επιβλέπων)

Καθηγητής Διγαλάκης Βασίλης

Αν. Καθηγητής Πετράκης Ευριπίδης

ΠΕΡΙΛΗΨΗ

Στις μέρες μας ο συνεχώς αυξανόμενος αριθμός των ανθρώπων οι οποίοι χρησιμοποιούν τις υπηρεσίες που παρέχονται μέσω του Διαδικτύου αλλά και ο τεράστιος όγκος πληροφοριών που προσφέρεται μέσω του Παγκόσμιου Ιστού καθιστούν επιτακτική την ανάγκη να συνδεθεί άμεσα ο χρήστης με τις πληροφορίες του ενδιαφέροντος του. Μία ανάγκη η οποία αποτυπώνεται στο γεγονός πως ιστοσελίδες πολλών επιχειρήσεων ηλεκτρονικού εμπορίου έχουν αναπτύξει εφαρμογές οι οποίες αντλώντας πληροφορίες για τον χρήστη μπορούν να του προτείνουν προσωπικά ένα προϊόν όπως ένα βιβλίο η μία ταινία. Αυτό επιτυγχάνεται από τη συλλογή προηγούμενων επιλογών του χρήστη "όμοιων" προϊόντων, ή του ίδιου προϊόντος από "όμοιους" χρήστες. Τέτοιου είδους εφαρμογές ονομάζονται *συστήματα σύστασης* (*recommendation systems*), τα περισσότερα από τα οποία βασίζονται στη μέθοδο *συνεργατικού φιλτραρίσματος* (*collaborative filtering*).

Με την παρούσα εργασία στοχεύουμε στην *βελτίωση της απόδοσης* ενός συστήματος σύστασης ταινιών το οποίο χρησιμοποιεί λεκτική πληροφορία αυτόματα εξαγόμενη από τον Παγκόσμιο Ιστό. Σκοπός μας ήταν, αφού κατεβάσαμε μέσω μιας μηχανής αναζήτησης πληροφορία για τις ταινίες και τους χρήστες μας, να την προ-επεξεργαστούμε και να εξετάσουμε διάφορους *τρόπους εξαγωγής χαρακτηριστικών λεκτικών γνωρισμάτων*. Αυτά τα γνωρίσματα μας βοήθησαν στην περιγραφή των χρηστών και των ταινιών που τους προτείναμε. Τέλος χρησιμοποιήσαμε κάποιους αλγόριθμους οι οποίοι συνδυάζουν τα λεκτικά χαρακτηριστικά και το συνεργατικό φιλτράρισμα και ενισχύουν την ακρίβεια του συστήματος έτσι ώστε να επιτύχουμε όσο το δυνατόν καλύτερη πρόβλεψη της βαθμολογίας που ένας χρήστης θα βάλει σε μία ταινία και κατ' επέκταση να του την προτείνουμε ή όχι.

Τα αποτελέσματα που πήραμε κατά την εκπόνηση αυτής της εργασίας μας έδειξαν πως μπορούμε να επιτύχουμε μία προσέγγιση του τρόπου με τον οποίο ένας χρήστης βαθμολογεί μία ταινία έχοντας ως εφόδιο κάποια λεκτικά χαρακτηριστικά για την ταινία τα οποία μπορούν να εξαχθούν εύκολα από το Διαδίκτυο. Προφανώς αυτό το εγχείρημα κρίνεται αρκετά φιλόδοξο μιας και η πολυπλοκότητα με την οποία ένας άνθρωπος αποφασίζει την βαθμολογία που θα δώσει σε μία ταινία αποτελεί αντικείμενο μελέτης παγκοσμίως με ασαφή αποτελέσματα μέχρι στιγμής.

Ευχαριστίες

Θα ήθελα να εκφράσω τις ευχαριστίες μου στον επιβλέποντα καθηγητή της διπλωματικής μου εργασίας κ. Αλέξανδρο Ποταμιάνο για τις πολύτιμες οδηγίες του σε αυτήν την ενδιαφέρουσα εργασία που μου ανέθεσε. Επίσης, τους κ.κ. Βασίλη Διγαλάκη και Ευριπίδη Πετράκη για τη συμμετοχή τους στην εξεταστική επιτροπή.

Επιπλέον, θα ήθελα να ευχαριστήσω θερμά, και με μεγάλη εκτίμηση για τον χρόνο που αφιέρωσε, τον Θεοδόση Μοσχόπουλο για την πολύτιμη βοήθεια που μου παρείχε σε όλη τη διάρκεια της εκπόνησης της παρούσας εργασίας. Η καθοδήγησή του ήταν καθοριστική στην προσπάθειά μου. Θα πρέπει εδώ να εκφράσω την ευγνωμοσύνη μου προς όλο το,πάντα πρόθυμο να βοηθήσει, προσωπικό της ομάδας του κ. Ποταμιάνου και κυρίως προς τον Ηλία Ιωσήφ που με την εμπειρία του και τις εύστοχες παρατηρήσεις του έπαιξε καταλυτικό ρόλο στην εργασία μου.

Ο Ορφέας Τσεργούλας ήταν ο άνθρωπος που με εισήγαγε στο αντικείμενο και μου έδωσε χρήσιμες συμβουλές οι οποίες μου φάνηκαν εξαιρετικά χρήσιμες κατά τη διάρκεια της εργασίας μου και τον ευχαριστώ θερμά για αυτό.

Τέλος, θα ήθελα να ευχαριστήσω μέσα από την ψυχή μου την οικογένεια μου, Αργυρώ,Σπύρο και Κατερίνα, την Χαρούλα, καθώς και όλους μου τους φίλους που ο καθένας με τον τρόπο του μου έδειξε όλα αυτά τα χρόνια αμέριστη συμπαράσταση και αγάπη. Χωρίς αυτούς όλα θα ήταν πιο δύσκολα.

Περιεχόμενα

Κεφάλαιο 1

Εισαγωγή

1.1 Εισαγωγή στο θέμα

Στις μέρες μας παρατηρείται συστηματική αύξηση των ατόμων που χρησιμοποιούν το Διαδίκτυο και ,αναλογικά, των προϊόντων τα οποία μπορούν να προμηθευτούν ή να χρησιμοποιήσουν μέσω αυτού. Συνεπώς όλο και περισσότερες εταιρίες επεκτείνουν την δράση τους σε εφαρμογές μέσω του Διαδικτύου. Καθώς ο πελάτης βρίσκεται αντιμέτωπος με χιλιάδες ή ,σε ορισμένες περιπτώσεις, εκατομμύρια διαφορετικά αντικείμενα ένα από τα μεγάλα στοιχήματα για αυτές τις εταιρίες είναι η δυνατότητα να προτείνουν στον πελάτη τους αντικείμενα όσο το δυνατόν πιο κοντά στις πραγματικές του ανάγκες και επιθυμίες έτσι ώστε να αυξήσουν την ανταγωνιστικότητά τους. Αυτός είναι και ο λόγος που η έρευνα, άμεσα συνδεδεμένη με την ανταγωνιστικότητα των εταιριών, έχει επικεντρωθεί στην βελτίωση των συστημάτων σύστασης(από εδώ και στο εξής *recommendation systems*). Συστημάτων ,δηλαδή, τα οποία κάνουν προσωπικές συστάσεις στους χρήστες για τα διάφορα προσφερόμενα αντικείμενα.

Είναι πολύ πιθανόν πολλοί από εμάς να έχουμε έρθει σε επαφή με τέτοιου είδους συστήματα στην προσπάθειά μας να αποκτήσουμε ένα αντικείμενο μέσω του Διαδικτύου,καθώς πολλές εταιρίες τα χρησιμοποιούν. Τέτοιες εταιρίες είναι, για παράδειγμα, η Amazon [1], εταιρία πώλησης βιβλίων, και η CDNOW [2] η οποία είναι το μεγαλύτερο κατάστημα πώλησης CD μέσω Διαδικτύου. Εμείς όμως θα εστιάσουμε στην εταιρία ενοικιάσεων ταινιών NETFLIX [3],

[1] http://www.amazon.com
[2] http://www.cdnow.com
[3] http://www.netflix.com

καθώς είναι άμεσα συσχετισμένη με τον σκοπό αυτής της εργασίας. Η NETFLIX έχει αναπτύξει ένα *recommendation system*, το *Cinematch*, βασικός στόχος του οποίου είναι η πρόβλεψη του κατά πόσον ένας χρήστης θα απολαύσει μία ταινία. Με αυτόν τον τρόπο επιτυγχάνει την σύσταση ταινιών με βάση τα προσωπικά κριτήρια του κάθε χρήστη. Προς βελτίωση του συστήματος αυτού κατα 10% η συγκεκριμένη εταιρία διοργάνωσε έναν παγκόσμιο διαγωνισμό διάρκειας πέντε ετών με έπαθλο ένα εκατομμύριο δολάρια στην ομάδα που θα το κατάφερνε [4].

Τα πρώτα *recommendation systems* χρησιμοποίησαν την μέθοδο του συνεργατικού φιλτραρίσματος (από εδώ και στο εξής *collaborative filtering*). Αυτή η μέθοδος συλλέγει τις προτιμήσεις των χρηστών, οι οποίες αντικατοπτρίζονται σε μία βαθμολόγηση κάποιου προϊόντος, για παρεμφερή αντικείμενα και ομαδοποιεί τους ανθρώπους εκείνους που τυγχάνει να μοιράζονται παρόμοιες προτιμήσεις. Με αυτόν τον τρόπο βασίζει την πρόβλεψη της στην ομοιότητα των προτιμήσεων άλλων χρηστών με αυτές του υπό εξέταση χρήστη. Στην βιβλιογραφία αυτή η μέθοδος ονομάζεται προσωποστραφές συνεργατικό φιλτράρισμα (*user-based or neighborhood-based collaborative filtering*) [7]. Μία διαφορετική προσέγγιση στο collaborative filtering είναι το αντικειμενοστραφές (*item-based collaborative filtering*) με το οποίο οι προβλέψεις βασίζονται στην ομοιότητα μεταξύ αντικειμένων [1].

Εξαιτίας της αυξανόμενης διαθεσιμότητας κειμένων σε ηλεκτρονική μορφή και της ανάγκης ύπαρξης ελαστικότητας στον τρόπο επεξεργασίας τους, οι ερευνητές ανέπτυξαν μεθόδους βασισμένες στο περιεχόμενο των κειμένων(*content-based methods*) ,στις οποίες βάσισαν τις προβλέψεις τους. Η ταξινόμηση κειμένου (*text classifiers*) ανήκει σε αυτές τις μεθόδους και βοηθά στην προσθήκη κειμένων φυσικής γλώσσας σε θεματικές κατηγορίες από ένα προκαθορισμένο σύνολο κατηγοριών.Σε αυτή την εργασία προσπαθούμε να βελτιώσουμε την απόδοση ενός recommendation system με χρήση τέτοιων ταξινομητών κειμένου.

Συγκεκριμμένα, θα κάνουμε χρήση του naive Bayes ταξινομητή και γλωσσικών μοντέλων (*n-gram language models*), αναπτύσσοντας ένα μοντελοποιημένο collaborative filtering με σκοπό την πρόβλεψη της βαθμολογίας που ένας χρήστης θα δώσει σε μία ταινία. Απώτερος στόχος της παρούσας εργασίας είναι να μελετήσουμε κατά πόσον μπορούμε να προσεγγίσουμε τον τρόπο που ένας άνθρωπος βαθμολογεί μία ταινία, έχοντας στη διάθεση μας γλωσσικά χαρακτηριστικά τα οποία εξάγουμε εύκολα από το Διαδίκτυο.

[4] www.netflixprize.com

1.2 Οργάνωση κειμένου

Το υπόλοιπο μέρος της εργασίας οργανώνεται ως εξής:Στο Κεφ. 2 αναλύουμε το συνεργατικό φιλτράρισμα και αναφέρουμε τις ήδη υπάρχουσες μεθόδους πάνω στο αντικείμενο δείχνοντας ποια από αυτές θα επιλέξουμε να υλοποιήσουμε στην εργασία μας. Στο Κεφ. 3 γίνεται μία εισαγωγή στην ταξινόμηση κειμένων (παρ. 3.2) και ειδικότερα στους ταξινομητές naive Bayes (παρ. 3.2.1) και CAN (παρ. 3.2.3) καθώς και μία ανάλυση του τι είναι το γλωσσικό μοντέλο και το back-off weight και πως μας βοηθούν στην προσωπική σύσταση ταινιών στους χρήστες. Ακόμη, θα παρουσιάσουμε κάποιες από τις μεθόδους επιλογής χαρακτηριστικών (*feature selection methods*) καθώς και πώς αυτές μας βοηθούν στην ταξινόμηση κειμένων (παρ. 3.4) και κατ' επέκταση στην πρόβλεψη βαθμολογιών. Στη συνέχεια, στο Κεφ. 4 παρουσιάζουμε την προσέγγιση μας με βάση την οποία υλοποιήσαμε διάφορες μεθόδους επιλογής χαρακτηριστικών και συνδυασμούς αυτών, καθώς επίσης και την διαδικασία ταξινόμησης κειμένων που ακολουθήσαμε έτσι ώστε να προβλέψουμε αποδοτικότερα. Η πειραματική διαδικασία καθώς και αποτελέσματα στα οποία καταλήξαμε παρουσιάζονται στο Κεφ. 5. Τέλος στο Κεφ. 6 παραθέτονται τα συμπεράσματα στα οποία καταλήξαμε καθώς επίσης και χρήσιμες κατευθύνσεις για μελλοντική δουλειά πάνω στο θέμα.

Κεφάλαιο 2

Συνεργατικό φιλτράρισμα

2.1 Εισαγωγή

Φανταστείτε πως επισκέπτεστε ένα κατάστημα ενοικίασης ταινιών με σκοπό να νοικιάσετε μία ταινία για να παρακολουθήσετε. Υποθέτουμε πως υπάρχει μεγάλη αφθονία σε τίτλους ταινιών, παλιών και πρόσφατων, και άρα σας είναι εξαιρετικά δύσκολο να αποφασίσετε ποια ταινία θα επιλέξετε. Η πρώτη και πιο εύκολη λύση σε αυτό το θεωρητικό πρόβλημα είναι να καλέσετε από το κινητό σας τηλέφωνο κάποιον φίλο σας με τον οποίο μοιράζεστε κοινές προτιμήσεις στις ταινίες ρωτώντας τον την γνώμη του για έναν αριθμό από ταινίες που τράβηξαν την προσοχή σας.

Στην καθημερινότητα το να ζητήσεις την γνώμη ενός φίλου σου είναι ένα σύνηθες φαινόμενο. Στον κόσμο της επιστήμης και ειδικότερα στον τομέα της ανάκτησης πληροφορίας αυτή η μέθοδος καλείται συνεργατικό φιλτράρισμα με βάση τον χρήστη (*user-based collaborative filtering*). Εσείς είστε ο χρήστης ενδιαφέροντος και ο φίλος σας του οποίου οι προτιμήσεις στην επιλογή ταινίας μοιάζουν με τις δικές σας είναι ο γείτονας σας. Είναι εύκολα κατανοητό πως η σύσταση ταινίας εξαρτάται από την γνώμη του γείτονά σας.

Επιστρέφοντας στο παράδειγμά μας και υποθέτοντας πως είχατε ξεχάσει το κινητό σας τηλέφωνο στο σπίτι είσαστε αναγκασμένοι να χρησιμοποιήσετε τη δική σας εμπειρία η οποία βασίζεται σε παρόμοιες ταινίες που είχατε δει στο παρελθόν. Σε πλήρη αντιστοιχία αυτή η μέθοδος αναφέρεται ως συνεργατικό φιλτράρισμα με βάση το αντικείμενο (*item-based collaborative filtering*) και η σύσταση εξαρτάται από αντικείμενα (ταινίες εν προκειμένω) παρόμοια με το αντικείμενο ενδιαφέροντος.

Και τα δύο αυτά ήδη συνεργατικού φιλτραρίσματος αποτελούνται από τρία εν γένει βήματα [7] (ανάλογα με το σύστημα αυτά τα βήματα μπορεί να επικαλύπτονται ή να διαφέρουν ως προς τη σειρά εκτέλεσης τους):

1. Υπολογισμός των ομοιοτήτων μεταξύ όλων των ξεχωριστών ζευγαριών χρηστών (στο user-based collaborative filtering) ή αντικειμένων (στο item--based collaborative filtering)

2. Επιλογή των πιο όμοιων με το χρήστη ενδιαφέροντος χρηστών (στο user-based collaborative filtering) ή με το αντικείμενο ενδιαφέροντος αντικειμένων (στο item--based collaborative filtering)

3. Πρόβλεψη της βαθμολογίας που θα δώσει στο αντικείμενο ενδιαφέροντος ο χρήστης και σύσταση του ή όχι.

Στη συνέχεια αυτού του κεφαλαίου αναλύουμε την προηγούμενη δουλειά που έχει γίνει όσον αφορά στο συνεργατικό φιλτράρισμα καθώς επίσης και την προσέγγιση που υιοθετήσαμε εμείς στην παρούσα εργασία.

2.2 Προηγούμενες προσεγγίσεις

Σε αυτή την ενότητα θα παρουσιάσουμε συνοπτικά μερικές πρακτικές που εφαρμόστηκαν πάνω στο συνεργατικό φιλτράρισμα και αναφέρονται στην βιβλιογραφία.

Το σύστημα που υλοποιήθηκε με σκοπό το φιλτράρισμα της ηλεκτρονικής αλληλογραφίας με την ονομασία Tapestry ήταν αυτό μέσω του οποίου οι δημιουργοί του, Goldberg et al. [2], εισήγαγαν για πρώτη φορά τον όρο συνεργατικό φιλτράρισμα. Το συγκεκριμένο σύστημα χρησιμοποιούσε αυτοματοποιημένες μεθόδους συνεργατικού φιλτραρίσματος με αποτέλεσμα οι χρήστες που το χρησιμοποιούσαν να έπρεπε να αναζητούν ένα μήνυμα ακολουθώντας μία συγκεκριμένη γλώσσα επιβολής ερωτημάτων (Tapestry Query Language).

Το πρώτο αυτοματοποιημένο σύστημα συνεργατικού φιλτραρίσματος το οποίο χρησιμοποιούσε αλγόριθμο με βάση τον χρήστη ήταν το GroupLens [3]. Το σύστημα αυτό παρείχε προσωπικές προβλέψεις για τα άρθρα του Usenet. Στην πρώτη του μορφή το GroupLens χρησιμοποιούσε ένα μέτρο ομοιότητας έτσι ώστε να υπολογίζει τις ομοιότητες μεταξύ των χρηστών. Με κατάλληλη χρήση των γειτονικών στον χρήστη ενδιαφέροντος χρηστών πραγματοποιούσε την τελική του πρόβλεψη υπολογίζοντας τον μέσο όρο των βαθμολογήσεων των γειτόνων και προσδίδοντας του κάποιο βάρος.

Ένα παρόμοιο με το GroupLens είναι το Ringo Music Recommender σύστημα [4]. Το σύστημα αυτό χρησιμοποίησε μεθόδους συνεργατικού φιλτραρίσματος με βάση τον χρήστη με σκοπό την προσωπική σύσταση μουσικής σε χρήστες. Αυτό επιτυγχανόταν διαλέγοντας τους περισσότερο συσχετισμένους, με βάση ένα κατάλληλα επιλεγμένο όριο, με τον χρήστη ενδιαφέροντος γείτονες. Έτσι, παρήγαγε προβλέψεις με υπολογισμό του μέσου όρου των προτιμήσεων των γειτονικών χρηστών.

Σε αντίθεση με τους προηγούμενους, οι Sarwar και Karypis [5] ακολούθησαν την προσέγγιση του συνεργατικού φιλτραρίσματος με βάση το αντικείμενο. Οι προβλέψεις που επιχείρησαν υλοποιήθηκαν με χρήση του βεβαρημένου μέσου όρου των βαθμολογήσεων που είχαν δοθεί σε παρόμοια αντικείμενα. Απέδειξαν πως η προσέγγιση με βάση το αντικείμενο μπορεί να φέρει υψηλή ακρίβεια στις προβλέψεις δίνοντας λύση στο πρόβλημα της τεράστιας αύξησης του πλήθους των υπολογισμών που χρειάζονται οι αλγόριθμοι γειτονικών χρηστών καθόσον αυξάνονται οι χρήστες και τα αντικείμενα. Αυτό το πρόβλημα (γνωστό ως scalability) είναι πολύ συχνό στα συστήματα σύστασης με εκατομμύρια χρήστες και χιλιάδες αντικείμενα.

Στη συνέχεια του κεφαλαίου μελετάμε μία διαφορετική προσέγγιση του συνεργατικού φιλτραρίσματος με βάση το αντικείμενο χρησιμοποιώντας το περιεχόμενο, την οποία υιοθετήσαμε κατά την εκπόνηση αυτής της εργασίας.

2.3 Συνεργατικό φιλτράρισμα με μοντελοποίηση

Εν συνεχεία των προηγουμένων μεθόδων συνεργατικού φιλτραρίσματος, εφαρμόστηκαν και κάποιες διαφορετικές τεχνολογίες στα συστήματα σύστασης όπως αλγόριθμοι μηχανικής μάθησης (machine learning algorithms). Ένας από αυτούς είναι τα Bayesian δίκτυα. Τα τελευταία δημιουργούν ένα μοντέλο βασισμένο σε ένα σύνολο εκπαίδευσης. Το μοντέλο αυτό μπορεί να χτιστεί χωρίς χρήση του Διαδικτύου μέσα σε κάποιες ώρες ή μέρες. Επίσης είναι σχετικά μικρού μεγέθους, γρήγορο και αισθητικά το ίδιο ακριβές με τις μεθόδους κοντινότερου γείτονα [6]. Τα δίκτυα αυτά μπορούν να αποδειχθούν πρακτικά για περιβάλλοντα στα οποία η γνώση για τις προτιμήσεις των χρηστών μένει σχετικά αμετάβλητη κατά την πάροδο του χρόνου (π.χ μία ταινία που άρεσε σε έναν χρήστη όταν την πρωτοείδε είναι πολύ πιθανόν να συνεχίσει να του αρέσει για μεγάλο χρονικό διάστημα).

Γίνεται εύκολα κατανοητό πως οι αλγόριθμοι συνεργατικού φιλτραρίσματος που βασίζονται σε μοντελοποίηση παρέχουν σύσταση αντικειμένων μέσω της δημιουργίας ενός μοντέλου για τις προτιμήσεις των χρηστών. Αλγόριθμοι αυτής της κατηγορίας έχουν μία πιθανοθεωρητική προσέγγιση και αναγάγουν την διαδικασία συνεργατικού φιλτραρίσματος στον υπολογισμό της προβλεπόμενης βαθμολογίας που ένας χρήστης θα δώσει με βάση προηγούμενες βαθμολογίες του ίδιου χρήστη σε διαφορετικά αντικείμενα.

Σε αυτή την εργασία και στην προσπάθειά μας να προβλέψουμε την βαθμολογία που ένας χρήστης θα έδινε σε μία άγνωστη σε αυτόν ταινία, δημιουργήσαμε μοντέλα για τις βαθμολογίες τις οποίες ο χρήστης αυτός είχε ήδη δώσει σε ταινίες που είχε δει. Έτσι, με την δημιουργία αυτών των μοντέλων οδηγηθήκαμε στην εκπαίδευση του naive Bayes ταξινομητή και σε επόμενη φάση στην πρόβλεψη της βαθμολογίας του χρήστη σε μία ταινία.

2.4 Περίληψη

Σε αυτό το κεφάλαιο κάναμε μία περιγραφή του συνεργατικού φιλτραρίσματος και των μεθόδων που έχουν εφαρμοστεί στο παρελθόν στα πλαίσια ενός συστήματος σύστασης. Ακόμη παρουσιάσαμε την προσέγγιση μας και εξηγήσαμε γιατί επιλέξαμε την μέθοδο συνεργατικού φιλτραρίσματος με μοντελοποίηση των προτιμήσεων ενός χρήστη. Στο επόμενο κεφάλαιο μελετάμε τον τρόπο με τον οποίο υλοποιούμε αυτή τη μέθοδο μέσω της ταξινόμησης κειμένων και κάποιες πρακτικές βελτίωσης της εκπαίδευσης του ταξινομητή μας.

Κεφάλαιο 3

Ταξινόμηση κειμένων-Επιλογή χαρακτηριστικών

3.1 Εισαγωγή

Σε αυτό το κεφάλαιο μελετάμε μία από τις πιο σημαντικές πτυχές της εξόρυξης δεδομένων, την ταξινόμηση κειμένων η οποία μας βοηθάει στην υλοποίηση του συστήματος μας με μεθόδους συνεργατικού φιλτραρίσματος με μοντελοποίηση.

Η ταξινόμηση κειμένων είναι η προσπάθεια προσθήκης κειμένων φυσικής γλώσσας (τα οποία, για παράδειγμα, είναι διαθέσιμα σε τεράστιες ηλεκτρονικές βιβλιοθήκες, βάσεις δεδομένων ή στο Διαδίκτυο) σε θεματικές ενότητες από ένα προκαθορισμένο σύνολο κατηγοριών. Σε αυτή την προσπάθεια χρησιμοποιούμε διάφορους ταξινομητές κειμένων οι οποίοι πρέπει να εκπαιδευτούν έτσι ώστε να προβλέψουν την κατηγορία στην οποία ανήκει ένα κείμενο. Σε αυτό το κεφάλαιο θα μελετήσουμε κάποιους από τους πιο ευρέως χρησιμοποιούμενους ταξινομητές καθώς και τα μοντέλα στα οποία αυτοί είναι πιστοί.

Η εκπαίδευση των ταξινομητών δεν είναι εύκολη υπόθεση ούτε παρέχει ασφαλή και ακριβή αποτελέσματα (προβλέψεις) όταν έχουμε να κάνουμε με τεράστιο πλήθος χαρακτηριστικών. Για αυτόν τον λόγο έχουν αναπτυχθεί κάποιες μέθοδοι απόδοσης βαρών σε χαρακτηριστικά ανάλογα με την συχνότητα εμφάνισης τους ή με την πληροφορία που μας παρέχουν, τις οποίες αναλύουμε στη συνέχεια. Τέλος, θα μελετήσουμε κάποιες μεθόδους επιλογής χαρακτηριστικών, οι οποίες μας βοηθάνε στο να επιλέξουμε τα χαρακτηριστικά εκείνα που μας παρέχουν την περισσότερη πληροφορία και με τα οποία θα εκπαιδεύσουμε αποδοτικότερα τον ταξινομητή κειμένου.

3.2 Ταξινόμηση Κειμένων

Ταξινόμηση κειμένων είναι το πρόβλημα της ανάθεσης ενός κειμένου D σε μία κατηγορία η οποία ανήκει σε ένα προκαθορισμένο, μεγέθους $|C|$, σύνολο κατηγοριών $C = \{c_1, c_2, ..., c_{|C|}\}$. Αυτό επιτυγχάνεται εάν εκπαιδεύσουμε έναν αλγόριθμο εκμάθησης έτσι ώστε να παράγει μία συνάρτηση ταξινόμησης $F : D \longrightarrow C$. Η εκπαίδευση αυτή γίνεται παρέχοντας ως παραδείγματα στον αλγόριθμο ένα σύνολο από N ήδη ενταγμένα σε κάποια κατηγορία κείμενα. Ο αλγόριθμος εκμάθησης που θα μας απασχολήσει σε αυτή την εργασία είναι ο *Naive Bayes* (βλέπε 3.2.1) τον οποίο αναλύουμε στη συνέχεια. Ένας συνδυασμός του τελευταίου με γλωσσικά μοντέλα (βλέπε 3.2.2) μας δίνει το αλυσιδωτά αυξανόμενο μοντέλο (*Chain Augmented Naive bayes model*) το οποίο εφαρμόζεται στον *CAN text classifier* (βλέπε 3.2.3).

3.2.1 Naive Bayes Ταξινομητής

Ο Naive Bayes ταξινομητής χρησιμοποιεί μία απλή εφαρμογή του κανόνα του *Bayes* [9]:

$$P(C = c|D = d) = \frac{P\,(C = c)\,P\,(D = d|C = c)}{P(D = d)} \tag{3.1}$$

και απλούστερα:

$$P(c|d) = \frac{P\,(c)\,P\,(d|c)}{P(d)} \tag{3.2}$$

όπου με d συμβολίζουμε το κείμενο που έχουμε δώσει ως παράδειγμα, με c την αντίστοιχη κατηγορία που ανήκει το κείμενο αυτό και με D, C τις τιμές του κειμένου και της κατηγορίας αντίστοιχα.

Στην ταξινόμηση κειμένων το κάθε κείμενο αναπαριστάται σαν ένα διάνυσμα N στοιχείων (λέξεων), δηλαδή $d = (\ w_1, w_2, ..., w_N\ )$. Συνεπώς το να υπολογίσουμε την πιθανότητα του

κειμένου δοσμένης της κατηγορίας $P(d|c)$ καθίσταται επουσιώδες δεδομένου ότι ο χώρος των πιθανών κειμένων είναι αχανής. Προς απλούστευση αυτού του υπολογισμού το *naive Bayes* μοντέλο δέχεται τον "αφελή" ισχυρισμό πως, για συγκεκριμένη κατηγορία, όλες οι λέξεις, w_i, είναι ανεξάρτητες μεταξύ τους. Έτσι, η (3.2) εκπίπτει στην εξής:

$$P(c|d) = \frac{P(c)\Pi_{j=1}^{N} P\left(w_j|c\right)}{P(d)} \tag{3.3}$$

Βασιζόμενοι στην (3.3) μπορούμε να αναζητήσουμε την βέλτιστη κατηγορία c^* η οποία είναι αυτή που μεγιστοποιεί την $a-posteriori$ πιθανότητα $P(c|d)$:

$$c^* = \arg\max_{c \in C}\{P(c|d)\} \tag{3.4}$$

Γνωρίζουμε με βάση τον τύπο της φόρμουλας Bayes πως:

$P(c|d) = P(c)P(d|c)$

Σε αυτό το σημείο πρέπει να τονίσουμε πως η ποσότητα $P(d) \approx 1$ δεν παίζει κανέναν ρόλο στον υπολογισμό της πιθανότητας αφού θεωρείται σταθερή ποσότητα και συνεπώς από εδώ και στο εξής θα παραλείπεται.

Σύμφωνα με τα παραπάνω, η (3.4) γράφεται ισοδύναμα:

$$c^* = \arg\max_{c \in C}\{P(c)P(d|c)\} \tag{3.5}$$

και άρα με βάση την (3.3):

$$c^* = \arg\max_{c \in C}\{P(c)\Pi_{j=1}^{N} P\left(w_j|c\right) \tag{3.6}$$

Παρατηρούμε εύκολα πως ο naive Bayes ταξινομητής λαμβάνει υπ'όψιν του μόνο τα N επιλεγμένα χαρακτηριστικά ενώ αγνοεί όλα τα υπόλοιπα, δηλαδή τα εκτός λεξιλογίου χαρακτηριστικά (*OOV attributes*). Αυτό επιδρά αρνητικά στην ακρίβεια της ταξινόμησης μιας

και έχει αποδειχτεί πως ακόμα και τα λιγότερο εμφανιζόμενα χαρακτηριστικά, τα οποία μας προσφέρουν λιγότερη πληροφορία, έχουν προσθετικές επιπτώσεις στην διαδικασία.

3.2.2 Γλωσσικά Μοντέλα

Σκοπός της δημιουργίας γλωσσικών μοντέλων είναι η πρόβλεψη της πιθανότητας μίας αλληλουχίας λέξεων. Ορθότερα, είναι η ανάθεση υψηλής πιθανότητας στις ακολουθίες λέξεων οι οποίες όντως υπάρχουν στο υπό εξέταση κείμενο και αντίστοιχα χαμηλής σε αυτές που δεν υφίστανται. Δοσμένης μίας ακολουθίας λέξεων T = *(* w_1,w_2,...,w_{T} *)* ως συλλογή υπό εξέταση χαρακτηριστικών, ο τρόπος εκτίμησης της ποιότητας και άρα της καταλληλότητας για αξιοποίηση του γλωσσικού μοντέλου μας είναι ο υπολογισμός της πολυπλοκότητας (*Perplexity* (p), 3.7) ή εντροπίας (Entropy (E), 3.8).

$$p = \sqrt[T]{\Pi_{i=1}^{T} \frac{1}{P\left(w_i|w_1...w_{i-1}\right)}} \tag{3.7}$$

$$E = log_2 P \tag{3.8}$$

Ο στόχος της δημιουργίας γλωσσικών μοντέλων είναι η επίτευξη χαμηλής πολυπλοκότητας. Ο απλούστερος και πιο επιτυχής τρόπος για την παραγωγή γλωσσικών μοντέλων είναι το *n*-γραμματικό μοντέλο (*n-gram models*). Αυτό το μοντέλο υπολογίζει την πιθανότητα μίας ακολουθίας λέξεων υποθέτοντας πως οι προηγούμενες *n*-1 λέξεις είναι οι μόνες σχετικές με την πρόβλεψη της $P(w_i|w_1...w_{i-1})$, η οποία με βάση τον κανόνα της αλυσίδας μας βοηθάει να υπολογίσουμε την πιθανότητα οποιασδήποτε ακολουθίας ως εξής:

$$P(w_1 w_2 ... w_{\mathrm{T}}) = \Pi_{i=1}^{T} P(w_i|w_1...w_{i-1}) \tag{3.9}$$

Η παραπάνω εξίσωση με βάση την υπόθεση ανεξαρτησίας μεταξύ των λέξεων ενός κειμένου αλλά και της θέσης αυτών μέσα στο κείμενο μας δίνει:

$$P(w_i|w_1...w_{i-1}) = P(w_i|w_{i-n+1}...w_{i-1}) \tag{3.10}$$

και:

$$P(w_i|w_{i-n+1}...w_{i-1}) = \frac{c(w_{i-n+1}...w_i)}{c(w_{i-n+1}...w_{i-1})} \tag{3.11}$$

Όπου με $c(.)$ αναπαριστούμε τον αριθμό εμφάνισης της συγκεκριμένης ακολουθίας στο υπό εκπαίδευση σύνολο χαρακτηριστικών.

Δυστυχώς, εξαιτίας των περιορισμών που υπάρχουν στους σύγχρονους υπολογιστικούς πόρους, δεν είναι εύκολο να χρησιμοποιήσουμε γραμμικά μοντέλα απεριορίστου μεγέθους αφού κάτι τέτοιο θα οδηγούσε στην ανάγκη υπολογισμού της πιθανότητας W^n γεγονότων, για μέγεθος λεξικού W. Έτσι, περιοριζόμαστε στην παραγωγή λεκτικών ακολουθιών μικρού μεγέθους (στην πράξη χρησιμοποιούμε μέχρι 3-grams). Ακόμη, εξαιτίας της ίδιας της φύσης της γλώσσας και τα χαρακτηριστικά τα οποία αυτή έχει, είναι πολύ πιθανή η αντιμετώπιση λεκτικών ακολουθιών οι οποίες δεν είχαν κάνει την εμφάνισή τους κατά την χρήση του αλγορίθμου εκπαίδευσης. Καθίσταται, λοιπόν, μεγάλης σημασίας όσο και αναπόφευκτη η προσπάθεια εύρεσης κάποιου μηχανισμού ανάθεσης μη μηδενικής πιθανότητας στις πρωτότυπες αυτές ακολουθίες. Συνήθης πρακτική σε αυτή την προσπάθεια είναι η ανάθεση στην ακολουθία ενός, ανάλογου με τον αριθμό εμφάνισής της, βάρους (*back-off weight*) η πιθανότητα του οποίου μπορεί να υπολογιστεί ως εξής:

$$P(w_i|w_{i-n+1}...w_{i-1}) = \begin{cases} \widehat{P}(w_i|w_{i-n+1}...w_{i-1}), \text{εάν}, c(w_{i-n+1}...w_i) > 0 \\ \beta(w_{i-n+1}...w_{i-1})P(w_i|w_{i-n+2}...w_{i-1}), \text{αλλού} \end{cases} \tag{3.12}$$

όπου

$$\widehat{P}(w_i|w_{i-n+1}...w_{i-1}) = \frac{c(w_{i-n+1}...w_i)}{c(w_{i-n+1}...w_{i-1})} \tag{3.13}$$

είναι η μειωμένη πιθανότητα (*discounted probability*) η οποία μπορεί να υπολογιστεί με διάφορες τεχνικές χαλάρωσης όπως η γραμμική(*linear smoothing*), απόλυτη(*absolute smoothing*),

Good-Turing και Witten-Bell [10], ενώ $\beta(w_{i-n+1}...w_{i-1})$ είναι μία σταθερά κανονικοποίησης η οποία υπολογίζεται με την παρακάτω εξίσωση:

$$\beta(w_{i-n+1}...w_{i-1}) = \frac{1 - \Sigma_{x\in(w_{i-n+1}...w_{i-1}x)}\widehat{P}(x|w_{i-n+1}...w_{i-1})}{1 - \Sigma_{x\in(w_{i-n+1}...w_{i-1}x)}\widehat{P}(x|w_{i-n+2}...w_{i-1})} \tag{3.14}$$

3.2.3 Αλυσιδωτά Αυξανόμενος Naive Bayes Ταξινομητής

Η ταξινόμηση κειμένων επαφίεται στην αναγνώριση των χαρακτηριστικών εκείνων τα οποία βοηθούν στην διάκριση κειμένων σε διαφορετικές κατηγορίες. Τέτοια χαρακτηριστικά μπορεί να είναι οι λέξεις που περιλαμβάνονται σε ένα λεξιλόγιο, το μέσο μήκος των λέξεων, τοπικά n-γράμματα ή διάφορες συντακτικές και σημαντικές ιδιότητες. Επίσης τα γλωσσικά μοντέλα επιχειρούν να εντοπίσουν τέτοιες κανονικότητες και ως εκ τούτου να παρέχουν μία διαφορετική διάσταση στην δημιουργία ταξινομητών κειμένων. Ο τρόπος εφαρμογής των γλωσσικών μοντέλων στη ταξινόμηση κειμένων είναι παρόμοιος με αυτόν του naive Bayes μοντέλου. Σε αυτήν την περίπτωση έχουμε:

$$c^* = \arg\max_{c\in C}\{P(c|d)\} \tag{3.15}$$

$$= \arg\max_{c\in C}\{P(d|c)P(c)\} \tag{3.16}$$

όπου για ίσες a-priori πιθανότητες για κάθε κατηγορία ισούται με:

$$c^* = \arg\max_{c\in C}\{P(d|c)\} \tag{3.17}$$

και τελικά με χρήση της υπόθεσης ανεξαρτησίας μεταξύ των όρων ενός κειμένου:

$$c^* = \arg\max_{c\in C}\{\Pi_{i=1}^{T} P_c(w_i|w_{i-n+1}...w_{i-1})\} \tag{3.18}$$

Η βασική αρχή της χρησιμοποίησης των γλωσσικών μοντέλων ως ταξινομητές κειμένων είναι η επιλογή της κατηγορίας εκείνης η οποία κάνει το υπό εξέταση κείμενο *πιο πιθανό* να έχει παραχθεί από το μοντέλο της ίδιας της κατηγορίας (3.18). Συνεπώς κρίνεται αναπόφευκτο το να πραγματοποιηθεί εκπαίδευση ενός ξεχωριστού γλωσσικού μοντέλου για κάθε κατηγορία και η , στη συνέχεια, ταξινόμηση ενός καινούριου κειμένου διαλέγοντας την κατάλληλη κατηγορία με βάση την (3.18).

Σε αντίθεση με τον *naive Bayes* ταξινομητή (παρ. 4.18), η παραπάνω διαδικασία λαμβάνει υπ'όψιν της όλα τα χαρακτηριστικά (λέξεις) κατά την εκπαίδευση ακόμα και αν κάποια από αυτά δεν ανήκουν στο λεξιλόγιο. Αυτή η διαφορά έχει αποδειχτεί πως παίζει σημαντικό ρόλο στην ακρίβεια της ταξινόμησης. Σε έναν *naive Bayes* ταξινομητή τα χαρακτηριστικά θεωρούνται ανεξάρτητα μεταξύ τους για συγκεκριμένη κατηγορία. Ωστόσο, σε μία προσέγγιση με βάση γλωσσικά μοντέλα τα χαρακτηριστικά "μεγαλώνουν" καθώς λαμβάνεται υπ'όψιν η συσχέτιση του Markov μεταξύ παρακείμενων λέξεων. Εξαιτίας αυτού ο ταξινομητής ο οποίος χρησιμοποιεί γλωσσικά μοντέλα αναφέρεται ως *αλυσιδωτά αυξανόμενος naive Bayes ταξινομητής(C A N Bayes classifier)*.

3.3 Απόδοση βαρών σε χαρακτηριστικά

Η απόδοση βαρών σε χαρακτηριστικά (*feature weighting*) κατέχει σημαντικό ρόλο σε πολλούς τομείς της ανάκτησης πληροφορίας(*Information Retrieval (IR)*). Σκοπός των μεθόδων απόδοσης βαρών είναι να εκτιμήσουν πόσο σημαντικό είναι ένα χαρακτηριστικό, ή όρος, σε μία συλλογή κειμένων, έτσι ώστε στα χαρακτηριστικά εκείνα τα οποία χαρακτηρίζουν ένα κείμενο και το διαφοροποιούν από τα υπόλοιπα να αποδοθεί μεγαλύτερο βάρος.

Στην βιβλιογραφία αναφέρεται μεγάλη ποικιλία από μεθόδους απόδοσης βαρών οι οποίες διακρίνονται κυρίως σε δύο κατηγορίες: τις στατιστικές και τις γλωσσικές. Οι μέθοδοι της πρώτης κατηγορίας βασίζονται σε μία στατιστική ανάλυση η οποία αποσπά, από μία συλλογή κειμένων, χαρακτηριστικά με βάση τις συχνότητες εμφάνισης τους ή διάφορα πληροφοριακά θεωρητικά κριτήρια. Από την άλλη πλευρά, οι γλωσσικές μέθοδοι βασίζονται σε συσχετίσεις μεταξύ λέξεων και εξάγουν πληροφορία από παρακείμενες λέξεις.

Σε αυτό το κεφάλαιο θα ασχοληθούμε μόνο με στατιστικές μεθόδους απόδοσης βαρών, οι πιο απλές από τις οποίες αξιοποιούν την συχνότητα εμφάνισης των χαρακτηριστικών ενώ οι πιο σύνθετες λαμβάνουν υπ' όψιν την συχνότητα κειμένων των χαρακτηριστικών.

3.3.1 Συχνότητα Όρου

Η συχνότητα όρου (term frequency) [11] είναι μία προσέγγιση η οποία αποδίδει σε έναν όρο w ενός κειμένου d ένα βάρος το οποίο ισούται με τον αριθμό των εμφανίσεων του w στο κείμενο d. Έτσι το κάθε κείμενο αναπαρίσταται πλέον ως ένας πίνακας κάθε διάσταση του οποίου αντικατοπτρίζει έναν ξεχωριστό όρο του κειμένου και σαν τιμή του την αντίστοιχη συχνότητα εμφάνισης στο κείμενο. Γίνεται εύκολα κατανοητό πως το πλήθος των διαστάσεων του πίνακα είναι ίσο με το πλήθος των ξεχωριστών όρων (συνήθως λέξεων) που εμφανίζονται στο κείμενο.

Η συγκεκριμένη μέθοδος απόδοσης βαρών έχει τα εξής μειονεκτήματα:

- Μεγαλύτερα κείμενα έχουν μεγαλύτερα tf_i βάρη και περιέχουν περισσότερους όρους που εμφανίζονται σπάνια. Αυτοί οι παράγοντες τείνουν να μεγαλώνουν τα βάρη μεγαλύτερων κειμένων γεγονός που είναι αφύσικο [11].
- Όλοι οι όροι θεωρούνται το ίδιο σημαντικοί χωρίς να λαμβάνεται υπ' όψιν πως κάποιοι από αυτούς έχουν ελάχιστη διακριτική ισχύ [11].

3.3.2 Αντίστροφη Συχνότητα Κειμένου

Η αντίστροφη συχνότητα κειμένου (*Inverse Document Frequency (idf)*) είναι μία από τις πιο σημαντικές και ευρέως χρησιμοποιούμενες μεθόδους στην ανάκτηση πληροφορίας. Ορίζεται για έναν όρο w ως εξής [11] :

$$idf_w = log\left(\frac{|D|}{|D_w|}\right) \tag{3.19}$$

Όπου $|D|$ ο συνολικός αριθμός των υπό εξέταση κειμένων, $|D_w|$ ο αριθμός των κειμένων που εμφανίζεται ο w περισσότερες από μία φορές και $|D_w|/|D|$ μία εκτίμηση της πιθανότητας p ένα τυχαίο κείμενο να περιέχει κάποιον όρο.

Η μέθοδος αυτή υπολογίζει την διακριτική ισχύ των χαρακτηριστικών μίας συλλογής κειμένων. Διαισθητικά, θεωρούμε πως ένας όρος που εμφανίζεται σε πολλά κείμενα δεν μας παρέχει αρκετή πληροφορία και άρα θα πρέπει να του δώσουμε μικρότερο βάρος από ότι σε έναν ο οποίος εμφανίζεται σε λίγα κείμενα. Για παράδειγμα, εάν ένας όρος w περιέχεται σε όλα τα κείμενα χάνει την διακριτική του ισχύ αφού:

$|D| = |D_w|$ και άρα: $idf_w = log\left(\frac{|D|}{|D_w|}\right) = 0$.

Σε αυτό το σημείο πρέπει να τονίσουμε πώς η βάση του λογαρίθμου δεν παίζει καμία σημασία και συνεπώς τέτοιες κλιμακώσεις λογαρίθμων είναι βολικές εξαιτίας των αθροιστικών ιδιοτήτων των τελευταίων.

3.3.3 Συχνότητα Όρου - Αντίστροφη Συχνότητα Κειμένου

Συνδυασμός των δύο προηγούμενων και μία από τις παλαιότερες αλλά και πιο αποτελεσματικές μεθόδους απόδοσης βαρών σε όρους με σκοπό την αποδοτικότερη επιλογή χαρακτηριστικών είναι η Συχνότητα Όρου - Αντίστροφη Συχνότητα Κειμένου (γνωστή ως *Term Frequency - Inverse Document Frequency , TFIDF*). Αυτή η μέθοδος δουλεύει υπολογίζοντας την συχνότητα εμφάνισης ενός όρου σε ένα συγκεκριμένο κείμενο και συγκρίνοντας την με την αντίστροφη αναλογία της εμφάνισης του όρου αυτού σε ολόκληρη την υπό εξέταση συλλογή κειμένων. Διαισθητικά, αυτός ο υπολογισμός αποφασίζει πόσο αντιπροσωπευτικός είναι ένας όρος για ένα συγκεκριμένο κείμενο. Έτσι, οι όροι εκείνοι οι οποίοι εμφανίζονται συχνά σε ένα μόνο κείμενο ή σε μια μικρή ομάδα κειμένων, τείνουν να έχουν μεγαλυτερο *tfidf* βάρος από κοινούς σε όλα τα κείμενα όρους(π.χ άρθρα, προθέσεις). Έτσι δεδομένης μίας συλλογής κειμένων D, ενός όρου w και ενός κειμένου το οποίο ανήκει στην D, υπολογίζουμε:

$$tfidf_w = TF_w log\left(\frac{|D|}{|D_w|}\right) \tag{3.20}$$

Όπου TF_w ο αριθμός εμφάνισης του όρου στο συγκεκριμένο κείμενο ,$|D|$ ο συνολικός αριθμός των υπό εξέταση κειμένων, και $|D_w|$ ο αριθμός των κειμένων που εμφανίζεται ο όρος w περισσότερες από μία φορές [15, 16].

Υποθέτοντας πως έχουμε έναν όρο w ο οποίος εμφανίζεται σχεδόν σε όλα τα κείμενα της συλλογής μας, συνεπώς ότι ισχύει $|D| \approx |D_w|$ και άρα $1 < log(|D|/|D_w|) < c$, για κάποια πολύ μικρή σταθερά c, συμπεραίνουμε από την (3.20) πως το $tfidf_w$ θα είναι μεν πολύ μικρό αλλά θα εξακολουθεί να είναι θετικό. Τα παραπάνω μας ωθούν στο να θεωρούμε πως ο συγκεκριμένος όρος παρόλο του ότι είναι ένας μάλλον κοινός όρος σε όλα τα κείμενα εξακολουθεί να έχει σημαντικό ρόλο για τη συλλογή κειμένων μας. Αυτό το παρατηρούμε κυρίως στους όρους εκείνους οι οποίοι είναι υπερβολικά συχνοί σε όλα τα κείμενα όπως τα άρθρα, οι προθέσεις, κ.τ.λ. αν και από μόνοι τους δεν μας παρέχουν κάποια πληροφορία και συνεπώς για αυτόν τον λόγο λαμβάνουν πολύ χαμηλό βάρος.

Σε αντίθεση με τα προηγούμενα, μεγαλύτερο ενδιαφέρον παρουσιάζουν οι όροι οι οποίοι έχουν σχετικά μεγάλο TF_w και μικρό $|D_w|$, και συνεπώς αντίστοιχα μεγάλο $log(|D|/|D_w|)$. Οι όροι, δηλαδή, που ενώ εμφανίζονται με μεγάλη συχνότητα στο συγκεκριμένο κείμενο, τείνουν να μην εμφανίζονται σχεδόν καθόλου σε κανένα από τα υπόλοιπα της συλλογής μας. Από την (3.20) καταλαβαίνουμε πως οι συγκεκριμένοι όροι θα λάβουν μεγάλο βάρος γεγονός που σημαίνει ως πρόκειται για εξαιρετικά σημαντικούς όρους για το υπό εξέταση κείμενο αλλά ταυτόχρονα ασήμαντους για όλη την υπόλοιπη συλλογή. Αυτοί οι όροι λέμε πως έχουν μεγάλη διακριτικήισχύ, είναι εκείνοι δηλαδή που μας βοηθούν περισσότερο στην διάκριση χαρακτηριστικών.

Όλες οι παραπάνω μέθοδοι χρησιμοποιούν και αποδίδουν βάρη σε όλους τους όρους ενός κειμένου. Συνεπώς περιλαμβάνουν αρκετή άσχετη και άρα θορυβώδη πληροφορία. Καθίσταται, λοιπόν, αναγκαία η εύρεση μεθόδων οι οποίες αναγνωρίζουν και κρατούν μόνο τα χαρακτηριστικά εκείνα που μας παρέχουν αρκετή και σχετική πληροφορία. Κάποιες τέτοιες μεθόδους επιλογής χαρακτηριστικών αναλύουμε στη συνέχεια.

3.4 Επιλογή χαρακτηριστικών

Ένα από τα κύρια χαρακτηριστικά, ή/και προβλήματα, της ταξινόμησης κειμένων είναι οι πολύ μεγάλες διαστάσεις του χώρου των χαρακτηριστικών. Ο αρχικός χώρος χαρακτηριστικών αποτελείται από μοναδικούς όρους (λέξεις ή φράσεις) οι οποίοι περιλαμβάνονται σε κείμενα και μπορούν να αριθμούν σε δεκάδες ή και ακόμα εκατοντάδες χιλιάδες ακόμη και για μία συνηθισμένου μεγέθους συλλογή κειμένων. Αυτό το πλήθος χαρακτηριστικών είναι απαγορευτικά μεγάλο για τους περισσότερους αλγορίθμους εκμάθησης. Για παράδειγμα μόνο κάποια νευρωνικά δίκτυα είναι ικανά να διαχειριστούν τέτοιες ποσότητες χαρακτηριστικών.

Σε αντίθεση με τα τελευταία, μοντέλα που υπακούν στον κανόνα του *Bayes* παρουσιάζουν υπολογιστική αναξιοπιστία εκτός και αν υιοθετηθεί μία, πιθανότατα μη αληθής, υπόθεση ανεξαρτησίας μεταξύ των χαρακτηριστικών. Κρίνεται εξαιρετικά επιθυμητή η μείωση του πλήθους των χαρακτηριστικών μας χωρίς, ωστόσο, να θυσιάσουμε την ακρίβεια της ταξινόμησής μας. Τέλος, θα επιθυμούσαμε να επιτύχουμε τον στόχο μας αυτόν με έναν αυτόματο τρόπο χωρίς, για παράδειγμα, να απαιτείται κάποιου είδους χειροκίνητος ορισμός ή κατασκευή των χαρακτηριστικών.

Η αυτοματοποιημένη επιλογή χαρακτηριστικών περιλαμβάνει την απομάκρυνση των όρων

εκείνων που μας παρέχουν ελάχιστη, ή καθόλου, πληροφορία με βάση στατιστικά στοιχεία της συλλογής κειμένων μας, καθώς επίσης και την δημιουργία νέων χαρακτηριστικών τα οποία συνδυάζουν μικρότερου επιπέδου χαρακτηριστικά (για παράδειγμα, λέξεις) σε μεγαλύτερου επιπέδου ορθογώνιες διαστάσεις.

Σε ερευνητικό επίπεδο, για την επιλογή χαρακτηριστικών ο Lewis και ο Ringuette [12] χρησιμοποίησαν κέρδος πληροφορίας (*information gain*) σε *naive Bayes* μοντέλο έτσι ώστε να μειώσουν δραστικά το μέγεθος του λεξιλογίου, ενώ ο Wiener [13, 14] χρησιμοποίησε αμοιβαία πληροφορία (*mutual information*) και ένα χ^2 στατιστικό για να επιλέξει την είσοδο σε ένα νευρωνικό δίκτυο. Στη συνέχεια αναλύουμε κάποιες από τις μεθόδους που μας διευκολύνουν στην επιλογή χαρακτηριστικών.

3.4.1 Αμοιβαία Πληροφορία

Η αμοιβαία πληροφορία (*mutual information*) είναι το κριτήριο εκείνο το οποίο συνήθως χρησιμοποιείται στους τομείς της στατιστικής γλωσσικής μοντελοποίησης που αφορούν σε συσχετίσεις λέξεων ή παρόμοιες εφαρμογές [13]. Έστω ο όρος w, και c η κατηγορία στην οποία ανήκει το κείμενο το οποίο περιέχει τον w, τότε η αμοιβαία πληροφορία του συγκεκριμένου όρου με βάση την κατηγορία υπολογίζεται ως εξής [17]:

$$I(w,c) = log\left(\frac{P(w,c)}{P(w)P(c)}\right) \tag{3.21}$$

και δεδομένου ότι $P(w,c) = P(w|c)P(c)$, ισχύει πως:

$$I(w,c) = log\left(\frac{P(w|c)P(c)}{P(w)P(c)}\right) \tag{3.22}$$

και άρα:

$$I(w,c) = log\left(\frac{P(w|c)}{P(w)}\right) \tag{3.23}$$

με

$$P(w|c) = \frac{f(w,c)}{N(c)} \tag{3.24}$$

$$P(w) = \frac{f(w)}{N(w)} \tag{3.25}$$

και $P(c)$ η $a-priori$ πιθανότητα της κατηγορίας, $f(w,c)$ ο αριθμός εμφάνισης του όρου σε όλα τα κείμενα της κατηγορίας, $N(c)$ ο αριθμός των όρων οι οποίοι περιλαμβάνονται σε όλα τα κείμενα της κατηγορίας c, $f(w)$ ο αριθμός εμφανίσεων του όρου w σε όλα τα κείμενα της συλλογής κειμένων μας και $N(w)$ ο αριθμός των όρων όλων των κειμένων.

Συμπεραίνουμε πως το $I(w,c)$ έχει φυσική τιμή μηδέν εφόσον ο όρος w και η κατηγορία c είναι ανεξάρτητα μεταξύ τους. Όσον αφορά στην κατάλληλη επιλογή των χαρακτηριστικών, επιτυγχάνεται υπολογίζοντας το άθροισμα της αμοιβαίας πληροφορίας του κάθε όρου για την κάθε κατηγορία και επιλέγοντας τα χαρακτηριστικά εκείνα για τα οποία μεγιστοποιείται αυτό το άθροισμα όπως φαίνεται στην 3.26. Το άθροισμα της αμοιβαίας πληροφορίας ενός χαρακτηριστικού σε όλες τις κατηγορίες είναι μεγαλύτερο για τα χαρακτηριστικά εκείνα τα οποία έχουν υψηλή συχνότητα εμφάνισης σε μία κατηγορία ενώ τείνουν να μην εμφανίζονται στις υπόλοιπες.

Με αυτόν τον υπολογισμό έχουμε την δυνατότητα να εξάγουμε τα περισσότερο αντιπροσωπευτικά χαρακτηριστικά γενικώς στο σύνολο των κειμένων μας, ανεξάρτητα από την κατηγορία την οποία αντιπροσωπεύουν.

$$I_{avg}(w) = \sum_{j=1}^{m} \{I(w,c_j)P(c_j)\} \tag{3.26}$$

Μία αδυναμία αυτής της μεθόδου επιλογής χαρακτηριστικών είναι πως τα υπολογιζόμενα βάρη είναι σε μεγάλο βαθμό συσχετισμένα με την συχνότητα εμφάνισης του όρου. Συνεπώς, ενώ σε όρους με ίσες εξαρτημένες πιθανότητες $P(w|c)$ οι λιγότερο εμφανιζόμενοι όροι θα λάβουν μεγαλύτερο βάρος από τους συχνά εμφανιζόμενους, τα βάρη όρων με μεγάλη διαφοροποίηση στη συχνότητα εμφάνισής τους δεν είναι συγκρίσιμα. Παρατηρούμε δηλαδή

απόκλιση στις τιμές των υπολογιζόμενων βαρών σε όρους με διαφορετικές συχνότητες εμφάνισης.

3.4.2 Διασταυρούμενη Εντροπία

Στον τομέα της κατηγοριοποίησης κειμένων και στην προσπάθεια μας για επιλογή των χαρακτηριστικών εκείνων με τα οποία αυτή θα βελτιωθεί, μπορούμε να υπολογίσουμε την διασταυρούμενη εντροπία (*cross entropy*) του κάθε χαρακτηριστικού μας.

Η διασταυρούμενη εντροπία (αναφέρεται στη βιβλιογραφία και ως term saliency, Kullback–Leibler divergence [18], information divergence, information gain, relative entropy) είναι μία μονάδα μέτρησης του ποσού της πληροφορίας που ένα χαρακτηριστικό μας παρέχει για μία κατηγορία και μας βοηθάει στην επιλογή των εξεχόντων χαρακτηριστικών (*salient features*).

Έτσι, για έναν όρο w και για $C = \{ c_1,c_2,...,c_N \}$ να είναι το σύνολο των κατηγοριών στις οποίες θέλουμε να ταξινομήσουμε τα κείμενά μας, η διασταυρούμενη εντροπία ορίζεται ως εξης [19]:

$$S(w) = \sum_{j=1}^{m} P(c_j|w) log \frac{P(c_j|w)}{P(c_j)} \tag{3.27}$$

και σε συνδυασμό με την (3.23) :

$$S(w) = \sum_{j=1}^{m} P(c_j|w) I(w, c_j) \tag{3.28}$$

Όπως εύκολα παρατηρούμε και από την 3.28 αυτή η μονάδα μέτρησης είναι άμεσα συνδεδεμένη και εξαρτώμενη από την αμοιβαία πληροφορία των χαρακτηριστικών.

3.5 Περίληψη

Δεδομένου του τεράστιου πλήθους των χαρακτηριστικών (όροι, λέξεις κ.τ.λ.) που μπορεί να υπάρχουν σε μία συλλογή κειμένων, η ταξινόμηση των κειμένων σε κάποια κατηγορία καθίσταται δύσκολη και αναξιόπιστη. Η διάκριση μέσω απόδοσης βαρών και η κατάλληλη, λοιπόν, επιλογή των χαρακτηριστικών εκείνων τα οποία μας προσφέρουν περισσότερη πληροφορία κρίνεται αναγκαία. Εφαρμόζοντας τις προαναφερθείσες μεθόδους είτε μοναδικά είτε σε συνδυασμό μεταξύ τους, επιτυγχάνουμε την δραστική μείωση του πλήθους των χαρακτηριστικών μας και άρα την ακριβέστερη και αποδοτικότερη ταξινόμηση κειμένων. Στο επόμενο κεφάλαιο θα αναφερθούμε σε ποιες από τις μεθόδους αυτές επιλέξαμε να υλοποιήσουμε καθώς και στον τρόπο με τον οποίο ταξινομήσαμε τα κείμενα μας.

Κεφάλαιο 4

Η προσέγγισή μας

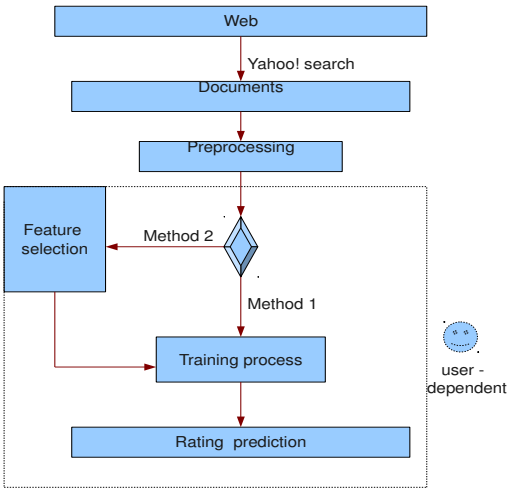
4.1 Εισαγωγή

Στα προηγούμενα κεφάλαια μελετήσαμε την ταξινόμηση κειμένου με χρήση του naive Bayes ταξινομητή, καθώς επίσης και διάφορους τρόπους οι οποίοι μας βοηθούν στο να κάνουμε την ταξινόμηση αυτή ακριβέστερη επιτυγχάνοντας πιο ακριβή πρόβλεψη. Σε αυτό το κεφάλαιο θα περιγράψουμε τη μέθοδο που ακολουθήσαμε για την πρόβλεψη βαθμολογιών βασιζόμενοι σε πληροφορία την οποία αντλήσαμε από το Διαδίκτυο. Στις επόμενες ενότητες, λοιπόν, αναλύουμε τον τρόπο με τον οποίο αντλούμε και επεξεργαζόμαστε την πληροφορία από κείμενα κριτικών ταινιών και την μέθοδο που επιλέγουμε έτσι ώστε να ταξινομήσουμε αυτά τα κείμενα με σκοπό να προβλέψουμε την βαθμολογία που ένας χρήστης θα έδινε σε κάποια ταινία. Η μέθοδος που ακολουθούμε έχει 2 μέρη:

1. Μέρος στο οποίο υλοποιούμε την διαδικασία εκπαίδευσης του συστήματος μας με χρήση όλων των χαρακτηριστικών.
2. Μέρος στο οποίο εκπαιδεύουμε το σύστημά μας με ένα υποσύνολο κατάλληλα επιλεγμένων χαρακτηριστικών.

Ο παραπάνω διαχωρισμός καθώς και όλες οι διαδικασίες οι οποίες ακολουθούνται κατά την εκπόνηση αυτής της εργασίας αναλύονται στη συνέχεια αυτού του κεφαλαίου και μπορούν να αναπαρασταθούν σχηματικά όπως φαίνεται στο Σχήμα 4.1 :

Με τους όρους method 1 και method 2 αναπαριστούμε τα αντίστοιχα μέρη της προσέγγισης μας που παραθέτουμε παραπάνω.



Σχήμα 4.1: Σχηματική αναπαράσταση της προσέγγισης μας

4.2 Άντληση δεδομένων - Προ-επεξεργασία

Σε αυτή την ενότητα περιγράφουμε τον τρόπο με τον οποίο λειτουργεί το σύστημά μας αντλώντας δεδομένα από το Διαδίκτυο έτσι ώστε να εφαρμόσουμε τις μεθόδους ταξινόμησης κειμένων.

Δεδομένης μίας λίστας με τίτλους ταινιών, χρησιμοποιούμε μία μηχανή αναζήτησης μέσω του Διαδικτύου για να ανακτήσουμε τις M πιο ψηλά καταταγμένες ιστοσελίδες για κάθε τίτλο. Η ορθή επιλογή του κριτηρίου αναζήτησης (*query*) είναι εξαιρετικά σημαντική αφού ελαχιστοποιεί την πιθανότητα ανάκτησης μη σχετικής πληροφορίας. Στην προσέγγισή μας επιλέγουμε σαν κριτήριο αναζήτησης το εξής:

query = (review OR reviews OR summary OR comments OR synopsis) AND (movie OR film OR dvd OR cinema) AND ("movie title" AND "year").

Στη συνέχεια και αφού ανακτήσουμε το περιεχόμενο των ιστοσελίδων αφαιρούμε όλα τα *HTML αναγνωριστικά (HTML tags)* κρατώντας, έτσι, μόνο το καθαρό κείμενο από κάθε σελίδα. Επιπροσθέτως, αφαιρούμε όλες τις "συνηθισμένες" λέξεις (*stop words*) όπως a, I, and, my κ.τ.λ. από την λίστα που παρέχεται από το πανεπιστήμιο Cornell αφού πρόκειται για λέξεις με ελάχιστη ή καθόλου πληροφορία.

Η παραπάνω διαδικασία που περιγράψαμε γίνεται για 1000 ταινίες οι οποίες ήταν αυτές με τις περισσότερες βαθμολογίες από τους χρήστες της NETFLIX.

4.3 Διαδικασία εκπαίδευσης

Σε αυτό το στάδιο της εργασίας μας θελήσαμε να εκπαιδεύσουμε τον naive Bayes ταξινομητή κειμένων έτσι ώστε στη συνέχεια να μπορέσουμε να προβλέψουμε την βαθμολογία που θα δώσει ο χρήστης. Πρέπει να τονίσουμε πως η διαδικασία εκπαίδευσης που ακολουθούμε γίνεται **για κάθε χρήστη**, (από ένα σύνολο επιλεγμένων, με βάση τον αριθμό των ταινιών που είχαν δει και βαθμολογήσει, χρηστών), ξεχωριστά. Πρόκειται, λοιπόν για μία $user - dependent$ διαδικασία.

Όπως φαίνεται και στο Σχήμα 4.1 η διαδικασία εκπαίδευσης μπορεί να χωριστεί σε δύο βασικές κατηγορίες:

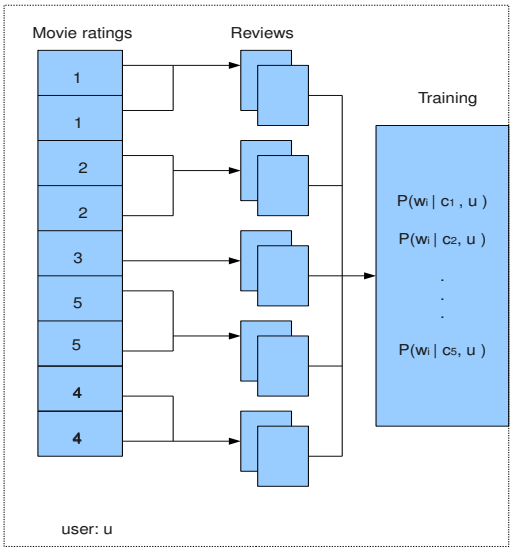
- Εκπαίδευση με χρήση όλων των χαρακτηριστικών (method 1).
- Εκπαίδευση με επιλογή χαρακτηριστικών (method 2).

Στη συνέχεια αναλύουμε αυτές τις δύο περιπτώσεις καθώς και τις ξεχωριστές υποπεριπτώσεις αυτών τις οποίες υλοποιούμε στην παρούσα εργασία.

Η διαδικασία εκπαίδευσης την οποία υλοποιούμε μπορεί να αναπαρασταθεί αναλυτικά στο Σχήμα 4.2, όπου Movie ratings είναι οι βαθμολογίες που ο συγκεκριμένος χρήστης, u, έδωσε σε όλες τις ταινίες που είδε, και Reviews όλα τα κείμενα των κριτικών που κατεβάσαμε για όλες τις ταινίες μίας κατηγορίας.

4.3.1 Εκπαίδευση με όλα τα χαρακτηριστικά

Ο τρόπος εκπαίδευσης του ταξινομητή με χρήση όλων των χαρακτηριστικών συνίσταται στον υπολογισμό για κάθε χρήστη και **για κάθε κατηγορία** (δηλαδή βαθμολογία που ο



Σχήμα 4.2: Σχηματική αναπαράσταση της διαδικασίας εκπαίδευσης για έναν χρήστη

συγκεκριμένος χρήστης έχει δώσει) των πιθανοτήτων όλων των λέξεων των κειμένων δεδομένης της κατηγορίας. Αυτός ο υπολογισμός μπορεί να γίνει είτε χρησιμοποιώντας την συχνότητα εμφάνισης του κάθε όρου, είτε κάποια άλλη μέθοδο απόδοσης βαρών στα χαρακτηριστικά μας και χρησιμοποίηση αυτών.

Αρχικά, χρησιμοποιούμε την συχνότητα εμφάνισης της κάθε λέξης στην συγκεκριμένη κατηγορία. Έτσι, για έναν χρήστη u, για έναν όρο w_i και για σύνολο κατηγοριών $C = \{c_1, c_2, ...c_j\}$, η πιθανότητα του όρου στην κάθε κατηγορία υπολογίζεται ως εξής:

$$P(w_i|c_j, u) = \frac{n_{w_i,c_j} + 1}{N_{c_j} + N} \tag{4.1}$$

όπου n_{w_i,c_j} είναι η συχνότητα εμφάνισης του συγκεκριμένου όρου στην κατηγορία c_j, N_{c_j} ο αριθμός όλων των λέξεων της κατηγορίας c_j, και N ο συνολικός αριθμός των λέξεων του λεξιλογίου μας.

Σε αυτό το σημείο πρέπει να τονίσουμε πως στον υπολογισμό των παραπάνω πιθανοτήτων χρησιμοποιούμε μία μορφή κανονικοποίησης προσθέτοντας στον αριθμητή της (4.1) 1 και στον παρονομαστή τον συνολικό αριθμό των λέξεων όλων των κειμένων. Αυτό γίνεται

προς αποφυγή εμφάνισης μηδενικών πιθανοτήτων σε λέξεις οι οποίες δεν εμφανίζονται καθόλου σε κάποια κατηγορία μιας και στη συνέχεια θα αξιοποιήσουμε τους λογαρίθμους αυτών των πιθανοτήτων.

Στη συνέχεια, υπολογίζουμε εκ νέου τις πιθανότητες της κάθε λέξης δεδομένης της κατηγορίας αυτή τη φορά όμως χρησιμοποιώντας μέθοδο απόδοσης βαρών (βλέπε παρ. 3.3).
Η μέθοδος που επιλέξαμε για να αποδώσουμε βάρος στα χαρακτηριστικά μας ήταν η $tfidf$ (παρ. 3.3.3). Έτσι, υπολογίζουμε για κάθε ταινία m και για κάθε όρο w_i ο οποίος εμφανίζεται σε κάποιο κείμενο από αυτά που αντλήσαμε για την m, τον αριθμό εμφάνισης $tf_{w_i,m}$ όπως επίσης και το df_{w_i} το οποίο είναι το πλήθος των κειμένων στα οποία εμφανίζεται ο w_i τουλάχιστον μία φορά.

Συνεπώς, ορίζοντας τον αριθμό όλων των ταινιών (για κάθε μία από τις ποίες έχουμε μόνο ένα κείμενο το οποίο περιέχει έναν αριθμό από κριτικές) ως $|D|$, το βάρος του κάθε όρου (λέξης) δίδεται από την :

$$\lambda_{w_i,m} = \begin{cases} tf_{w_i,m} \log \frac{|D|}{df_{w_i}}, \text{εάν}, tf_{w_i,m} > 0 \\ 0, \text{αλλού} \end{cases} \tag{4.2}$$

Όπως αναφέρουμε και στο Κεφ. 3, το $\lambda_{w_i,m}$ λαμβάνει μεγάλη τιμή όταν ο w_i εμφανίζεται πολλές φορές σε σχετικά μικρό αριθμό κειμένων ενώ, αντίθετα, λαμβάνει μικρή τιμή εφόσον εμφανίζεται λιγότερες φορές σε ένα κείμενο ή ο αριθμός των κειμένων στα οποία εμφανίζεται είναι πολύ μεγάλος.
Γίνεται εύκολα κατανοητό πως μία ταινία μπορεί να έχει βαθμολογηθεί, ανάλογα με τον χρήστη, με διαφορετικές βαθμολογίες. Έτσι, σε αυτή την προσέγγιση μας, τα βάρη υπολογίζονται ανεξάρτητα από την κατηγορία, ξεχωριστά δηλαδή για κάθε ταινία.

Με βάση τα παραπάνω και σε συνδυασμό με την (4.1), υπολογίζουμε τις πιθανότητες για έναν χρήστη u, για έναν όρο w_i και για σύνολο κατηγοριών $C = \{c_1, c_2, ...c_j\}$ ως εξής:

$$P(w_{i,j}|c_j, u) = \frac{\lambda_{w_{i,m}} + 1}{N_{c_j} + N} \tag{4.3}$$

όπου $\lambda_{w_{i,m}}$ είναι το $tfidf$ βάρος του συγκεκριμένου όρου για την ταινία m η οποία ανήκει στην κατηγορία c_j όπως υπολογίστηκε παραπάνω, N_{c_j} ο αριθμός όλων των λέξεων της κατηγορίας c_j, και N ο συνολικός αριθμός των λέξεων όλου του λεξιλογίου μας. Εφαρμόζουμε την ίδια μορφή κανονικοποίησης των πιθανοτήτων με αυτή της (4.1).

Στη συνέχεια θα περιγράψουμε τον τρόπο εκπαίδευσης του ταξινομητή κειμένων έχοντας κάνει χρήση κάποιων μεθόδων επιλογής χαρακτηριστικών στην προσπάθεια μας να ελαττώσουμε το μέγεθος του λεξιλογίου μας επιτυγχάνοντας ακριβέστερη πρόβλεψη, καθώς επίσης και συνδυασμούς αυτών.

4.3.2 Εκπαίδευση με επιλογή χαρακτηριστικών

Στην προσπάθειά μας να ελαττώσουμε το μέγεθος του λεξιλογίου που χρησιμοποιούμε, το πλήθος δηλαδή των χαρακτηριστικών με τα οποία εκπαιδεύουμε τον ταξινομητή μας, επιστρατεύσαμε μεθόδους επιλογής χαρακτηριστικών (βλέπε παρ. 3.4). Χρησιμοποιούμε τρεις διαφορετικές προσεγγίσεις πάνω στην επιλογή χαρακτηριστικών:

- Επιλογή χαρακτηριστικών με βάση την αμοιβαία πληροφορία.
- Επιλογή χαρακτηριστικών με βάση την διασταυρούμενη εντροπία.
- Επιλογή χαρακτηριστικών με συνδυασμό της διασταυρούμενης εντροπίας και του $tfidf$ βάρους.

Στη συνέχεια της ενότητας αναλύουμε τις τρεις αυτές προσεγγίσεις και τον τρόπο με τον οποίο αυτές μας οδήγησαν στον υπολογισμό των πιθανοτήτων των επιλεγμένων χαρακτηριστικών.

Αρχικά χρησιμοποιήσαμε μία συνηθισμένη μέθοδο επιλογής χαρακτηριστικών η οποία καλείται αμοιβαία πληροφορία.
Η αμοιβαία πληροφορία ενός όρου w_i για μία κατηγορία c_j είναι, όπως αναφέρεται και στην παρ. 3.4.1, το κριτήριο εκείνο που μας δηλώνει την αλληλοσυσχέτιση μεταξύ του w_i και της c_j. Έτσι, για έναν χρήστη u η αμοιβαία πληροφορία του συγκεκριμένου όρου με βάση την κατηγορία υπολογίζεται ως εξής [17]:

$$I(w_i, c_j, u) = log\left(\frac{P(w_i, c_j, u)}{P(w_i, u)P(c_j, u)}\right) \tag{4.4}$$

$$= log\left(\frac{P(w_i|c_j, u)P(c_j, u)}{P(w_i, u)P(c_j, u)}\right) \tag{4.5}$$

άρα:

$$I(w_i, c_j, u) = log\left(\frac{P(w_i|c_j, u)}{P(w_i, u)}\right) \tag{4.6}$$

όπου

$$P(w_i|c_j, u) = \frac{f(w_i, c_j, u)}{N(c_j, u)} \tag{4.7}$$

$$P(w_i, u) = \frac{f(w_i, u)}{N(w, u)} \tag{4.8}$$

και $P(c_j, u)$ η $a - priori$ πιθανότητα της κατηγορίας, $f(w_i, c_j, u)$ ο αριθμός εμφάνισης του όρου σε όλα τα κείμενα της κατηγορίας, $N(c_j, u)$ ο αριθμός των όρων οι οποίοι περιλαμβάνονται σε όλα τα κείμενα της κατηγορίας c_j, $f(w_i, u)$ ο αριθμός εμφανίσεων του όρου w_i σε όλα τα κείμενα της συλλογής κειμένων μας και $N(w, u)$ ο αριθμός των όρων όλων των κειμένων.

Οι παραπάνω εξισώσεις υπολογίζουν πόσο αντιπροσωπευτικός είναι ένας όρος για μία συγκεκριμένη κατηγορία. Θέλοντας, όμως, να υπολογίσουμε την σπουδαιότητα ενός όρου w_i ανεξαρτήτως κατηγορίας έχουμε την εξής εναλλακτική :

$$I_{avg}(w_i, u) = \sum_{j=1}^{m} I(w_i, c_j, u)P(c_j, u) \tag{4.9}$$

Με χρήση των παραπάνω επιτυγχάνουμε να επιλέξουμε τα χαρακτηριστικά εκείνα τα οποία μας παρέχουν την περισσότερη πληροφορία για κάποια κατηγορία, εκείνα δηλαδή για τα οποία το άθροισμα της (4.9) έχει μεγαλύτερη τιμή. Για διαφορετικό πλήθος χαρακτηριστικών που επιλέγουμε, και σε συνδυασμό με την (4.1), υπολογίζουμε τις πιθανότητές τους για κάθε κατηγορία με χρήση της συχνότητας εμφάνισής τους στη συγκεκριμένη κατηγορία.

Στη συνέχεια, και σε άμεση σχέση με τα προηγούμενα, προχωράμε στον υπολογισμό, για κάθε χρήστη u, της διασταυρούμενης εντροπίας (*cross-entropy*) του κάθε χαρακτηριστικού w_i που εμφανίζεται σε κάποιο κείμενο, οποιασδήποτε κατηγορίας c_j. Η διασταυρούμενη εντροπία, όπως αναφέρεται και στην παρ. (3.4.2), είναι μία μονάδα μέτρησης του ποσού της πληροφορίας που ο w_i μας παρέχει για κάποια c_j. Ομοίως με την διαδικασία υπολογισμού της αμοιβαίας πληροφορίας του w_i, μας ενδιαφέρει ο υπολογισμός της σημαντικότητας ενός όρου ανεξαρτήτως κατηγορίας. Συνεπώς, για έναν χρήστη u, ως διασταυρούμενη εντροπία ενός όρου w_i ανεξαρτήτως κατηγορίας c_j, ορίζουμε την ποσότητα [19]:

$$S(w_i, u) = \sum_{j=1}^{m} P(c_j|w_i, u) log \frac{P(c_j|w_i, u)}{P(c_j, u)} \tag{4.10}$$

όμως:

$$P(c_j, u) = \frac{P(w_i|c_j, u)P(c_j, u)}{P(w_i, u)} \tag{4.11}$$

άρα η (4.10) γίνεται:

$$S(w_i, u) = \sum_{j=1}^{m} P(c_j|w_i, u) log \frac{P(w_i|c_j, u)}{P(w_i, u)} \tag{4.12}$$

και σε συνδυασμό με την (4.6) :

$$S(w_i, u) = \sum_{j=1}^{m} P(c_j|w_i, u) I(w_i, c_j, u) \tag{4.13}$$

Επιπροσθέτως, ισχύει πως $\sum_{j=1}^{m} P(c_j|w_i, u) \approx 1$.

Με αυτόν τον υπολογισμό καταφέρνουμε να επιλέξουμε τα χαρακτηριστικά εκείνα τα οποία έχουν το μεγαλύτερο άθροισμα διασταυρούμενης εντροπίας ανεξαρτήτως κατηγορίας και συνεπώς μας παρέχουν περισσότερη πληροφορία για μία κατηγορία. Αυτά τα χαρακτηριστικά καλούνται εξέχοντα χαρακτηριστικά (*salient features*). Έτσι για διαφορετικό πλήθος εξεχόντων χαρακτηριστικών υπολογίζουμε τη συχνότητα εμφάνισής τους σε όλες τις κατηγορίες και σε συνδυασμό με την (4.1) υπολογίζουμε τις πιθανότητές τους για κάθε κατηγορία εκπαιδεύοντας τα σύστημά μας με αυτές.

Στην προσπάθεια μας για αποδοτικότερη εκπαίδευση του ταξινομητή κειμένων μας, κρίναμε απαραίτητο το συνδυασμό της παραπάνω διαδικασίας (διασταυρούμενης εντροπίας) με την μέθοδο απόδοσης βαρών $tfidf$ όπως αυτή υλοποιείται στην (4.2).

Στην τρίτη, λοιπόν, προσέγγισή μας εξάγουμε τα χαρακτηριστικά εκείνα τα οποία από την μία μας προσέφεραν περισσότερη πληροφορία για κάποια κατηγορία αλλά λαμβάνοντας ταυτόχρονα υπ' όψιν και το κατά πόσον τα συγκεκριμένα χαρακτηριστικά είναι αντιπροσωπευτικά και για κάποια ταινία.

Έτσι, για έναν χρήστη u και για έναν όρο w_i και ανεξαρτήτως κατηγορίας c_j ορίζουμε τον παραπάνω συνδυασμό ως εξής:

$$C(w_i, u) = \begin{cases} S(w_i, u)^{\lambda_1} tfidf_{w_i,m}^{\lambda_2}, \text{για}, S(w_i, u) > 0 \\ 0, \text{αλλού} \end{cases} \tag{4.14}$$

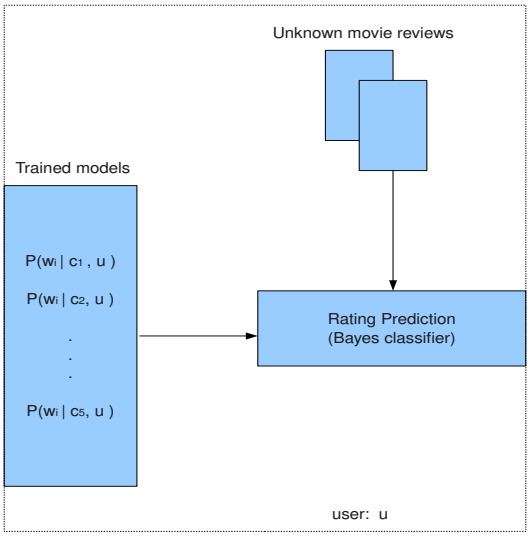
Η επιλογή χαρακτηριστικών σε αυτή την προσέγγιση συνίσταται στην επιλογή εκείνων μόνο των χαρακτηριστικών που έχουν διασταυρούμενη εντροπία μεγαλύτερη του μηδενός, εκείνων δηλαδή που μας παρέχουν πληροφορία για κάποια κατηγορία. Επίσης, χρησιμοποιούμε κάποια βάρη λ_1, λ_2 (με $\lambda_1 + \lambda_2 = 1$) στην διασταυρούμενη εντροπία και στο $tfidf$ αντίστοιχα, έτσι ώστε να επιτύχουμε όσο το δυνατόν καλύτερη επιλογή χαρακτηριστικών.

Και σε αυτή την προσέγγιση ακολουθούμε στη συνέχεια την ίδια διαδικασία για να εκπαιδεύσουμε το σύστημά μας με την διαφορά πως ο υπολογισμός των πιθανοτήτων δεν έγινε με

βάση τις συχνότητες εμφάνισης των χαρακτηριστικών που επιλέξαμε σε όλες τις κατηγορίες αλλά με βάση τα βάρη που υπολογίσαμε παραπάνω και, με χρήση της (4.1), υπολογίζοντας τις πιθανότητές τους για κάθε κατηγορία.

4.4 Πρόβλεψη βαθμολογίας

Σε αυτή την ενότητα θα αναλύσουμε πως τα προηγούμενα βήματα άντλησης δεδομένων από το Διαδίκτυο και εκπαίδευσης μας οδηγούν στην πρόβλεψη της κατηγορίας στην οποία ένα προς εξέταση κείμενο ανήκει. Η διαδικασία της πρόβλεψης κατηγορίας αναπαρίσταται στο Σχήμα 4.3.



Σχήμα 4.3: Σχηματική αναπαράσταση της διαδικασίας πρόβλεψης κατηγορίας για μία άγνωστη ταινία για έναν χρήστη

όπου ως άγνωστη ταινία αναφέρουμε μία ταινία την οποία ο χρήστης δεν έχει βαθμολογήσει και ως $Unknown\ movie\ reviews$ το κείμενο που περιλαμβάνει όλες τις κριτικές της συγκεκριμένης ταινίας της οποίας την βαθμολογία θέλουμε να προβλέψουμε.

Το κριτήριο που μας υποδεικνύει σε ποια κατηγορία c_j ανήκει (δηλαδή τι βαθμολογία θα δώσει ο χρήστης u σε) ένα κείμενο d που εξετάζουμε, είναι η πιθανότητα της κατηγορίας

c_j δεδομένου του κειμένου d, $P(c_j|d)$. Επιλέγουμε, δηλαδή, την κατηγορία c^* που θα προβλέψουμε πως ο χρήστης u θα δώσει στο d να είναι εκείνη η οποία μεγιστοποιεί την $P(c_j|d)$. Έτσι, για τον χρήστη u και για σύνολο κατηγοριών $C = \{c_1, c_2, ..., c_j\}$ έχουμε:

$$c^* = \arg\max_{c_j \in C}\{P(c_j|d, u)\} \tag{4.15}$$

δηλαδή:

$$c^* = \arg\max_{c_j \in C}\{P(d|c_j, u)P(c_j, u)\} \tag{4.16}$$

όπου η $P(c_j, u)$ είναι η $a - priori$ πιθανότητα της κατηγορίας c_j για τον u.
Δεδομένης της αρχής ανεξαρτησίας, σύμφωνα με την οποία κάθε όρος (λέξη) είναι ανεξάρτητη από τους υπόλοιπους όρους του κειμένου και άρα η πιθανότητα του δεν εξαρτάται από την θέση του στο κείμενο (έστω Ν όρων) έχουμε:

$$P(d|c) = \Pi_{i=1}^{\mathrm{N}} P(w_i|c) \tag{4.17}$$

Συνεπώς, η (4.16) εκπίπτει στην εξής:

$$c^* = \arg\max_{c_j \in C}\{P(c_j)\Pi_{i=1}^{N} P\left(w_i|c_j\right)\} \tag{4.18}$$

Ισχύει πως $\sum_{i=1}^{N} P(w_i|c_j, u) \approx 1$ Εφόσον το γινόμενο πιθανοτήτων ισούται με το άθροισμα των λογαρίθμων αυτών, στην εργασία αυτή και στην προσπάθεια πρόβλεψης της κατηγορίας που θα δώσει ο χρήστης σε ένα κείμενο και άρα σε μία ταινία που αυτό αντιπροσωπεύει χρησιμοποιούμε την :

$$c^* = \arg\max_{c_j \in C}\{\lambda_1 \log P(c_j) + \lambda_2 \sum_{i=1}^{N} \log(P\left(w_i|c_j\right))\} \tag{4.19}$$

Γίνεται, λοιπόν, κατανοητό πως η βαθμολογία την οποία προβλέπουμε για μία ταινία για έναν συγκεκριμένο χρήστη, είναι αυτή για την οποία το άθροισμα των λογαρίθμων των πιθανοτήτων των λέξεων που αποτελούν το κείμενο που την αντιπροσωπεύει, όπως αυτές υπολογίστηκαν με τις μεθόδους που περιγράφονται στην ενότητα (4.3), πολλαπλασιασμένων με ένα βάρος λ_2, και της $a-priori$ πιθανότητας της συγκεκριμένης κατηγορίας πολλαπλασιασμένης με ένα βάρος λ_1, μεγιστοποιείται. Στην προσέγγιση μας χρησιμοποιούμε διαφορετικές τιμές για τα βάρη λ_1 και λ_2 λαμβάνοντας υπ' όψιν τον λόγο τους, λ_1/λ_2.

4.5 Περίληψη

Σε αυτό το κεφάλαιο είδαμε αναλυτικά τις προσεγγίσεις που υιοθετούμε έτσι ώστε να προβλέψουμε την βαθμολογία ενός χρήστη σε μία ταινία. Είδαμε πως αντλούμε τα κείμενα των κριτικών των ταινιών που μας ενδιαφέρουν από το Διαδίκτυο καθώς και τον τρόπο που ανακτούμε μόνο την χρήσιμη πληροφορία από αυτά. Στη συνέχεια εκπαιδεύουμε το σύστημα μας, υπολογίζουμε δηλαδή τις πιθανότητες των λέξεων των κειμένων αυτών με βάση την συχνότητα εμφάνισης τους σε κείμενα κάποιας κατηγορίας ενώ, τέλος, επιλέγουμε κάποιες από αυτές έτσι ώστε να κάνουμε την πρόβλεψη μας πιο αποτελεσματική.
Στο Κεφ. 5 θα παρουσιάσουμε τα αποτελέσματα των παραπάνω πρακτικών και θα επιχειρήσουμε μία αξιολόγηση αυτών.

Κεφάλαιο 5

Πειράματα - Αποτελέσματα

5.1 Εισαγωγή

Στα προηγούμενα κεφάλαια μελετήσαμε την διαδικασία με την οποία λειτουργεί το σύστημα μας από την άντληση δεδομένων από το Διαδίκτυο μέχρι και την πρόβλεψη της βαθμολογίας την οποία ένας χρήστης θα δώσει σε μία ταινία. Σε αυτό το κεφάλαιο αξιολογούμε τα μοντέλα μας σε πραγματικά δεδομένα και συγκρίνουμε την απόδοση τους με κάποια βάση σύγκρισης (baseline) αλλά και μεταξύ τους.

Συγκεκριμένα, τα πειράματά μας διεξήχθησαν με χρησιμοποίηση διαφόρων ειδών πληροφορίας:

1. Τίτλους ταινιών.
2. Βαθμολογίες οι οποίες δόθηκαν από συγκεκριμένους χρήστες σε αυτούς τους τίτλους ταινιών.
3. Κείμενα κριτικών για αυτούς τους τίτλους κατεβασμένα από το Διαδίκτυο.

Τα πειραματικά δεδομένα αναφέρονται αναλυτικά στις επόμενες ενότητες.

5.2 Πειραματικά δεδομένα

Σε αυτή την ενότητα περιγράφουμε αναλυτικά τα πειραματικά δεδομένα που χρησιμοποιούμε καθώς επίσης και τον τρόπο με τον οποίο τα οργανώνουμε.

5.2.1 Τίτλοι ταινιών

Η εταιρία ενοικιάσεων ταινιών μέσω του Διαδικτύου NETFLIX μας παρείχε ένα σύνολο δεδομένων με πάνω από 100 εκατομμύρια βαθμολογίες οι οποίες είχαν δοθεί σε 17770 τίτλους ταινιών από 480 χιλιάδες τυχαία επιλεγμένους χρήστες της. Προφανώς η χρησιμοποίηση ολόκληρου αυτού του όγκου δεδομένων θα καθιστούσε την διεξαγωγή των πειραμάτων μας εξαιρετικά χρονοβόρα διαδικασία. Συνεπώς, οι ταινίες που χρησιμοποιούμε στην πειραματική μας διαδικασία είναι ένα υποσύνολο του παρεχόμενου από την εταιρία συνόλου δεδομένων. Η επιλογή των ταινιών που χρησιμοποιήσαμε δεν έγινε τυχαία. Αναλύσαμε τον τεράστιο αυτό όγκο δεδομένων και επιλέξαμε τις 1000 ταινίες με τις περισσότερες βαθμολογήσεις. Ακόμη, δημιουργήσαμε μία λίστα για τις επιλεγμένες ταινίες η οποία περιέχει δύο πεδία πληροφορίας: 1) Τίτλος ταινίας και 2) Χρονολογία παραγωγής της.

5.2.2 Κριτικές ταινιών

Χρησιμοποιώντας την λίστα των ταινιών την οποία δημιουργήσαμε χρησιμοποιήσαμε την μηχανή αναζήτησης στο Διαδίκτυο Yahoo-search API [1] με σκοπό να κατεβάσουμε κριτικές συσχετισμένες με αυτές τις ταινίες. Σαν κριτήριο αναζήτησης όπως αναφέρουμε και στην παρ. 4.2 χρησιμοποιήσαμε το εξής:

query = (review OR reviews OR summary OR comments OR synopsis) AND (movie OR film OR dvd OR cinema) AND ("movie title" AND "year").

Η είσοδος στο κριτήριο αναζήτησης του τίτλου της ταινίας και της χρονολογίας παραγωγής της έγινε με χρήση των παραμέτρων movie title και year.

[1] Web Site of Yahoo Search API http://search.cpan.org/jfriedl/Yahoo-Search- 1.10.13/lib/Yahoo/Search.pm

5.2.2.1 Προ επεξεργασία κριτικών

Για κάθε μία ταινία από αυτές που επιλέξαμε δημιουργήθηκε ένα κριτήριο αναζήτησης το οποίο δόθηκε σαν είσοδος στην μηχανή αναζήτησης. Κατεβάσαμε τις 100 πιο ψηλά καταταγμένες ιστοσελίδες για κάθε τίτλο ως έξοδο της μηχανής αναζήτησης.Για κάθε ένα από τα κατεβασμένα αυτά κείμενα ακολουθήσαμε την εξής διαδικασία:

1. Αφαιρέσαμε τα HTML χαρακτηριστικά (HTML tags)
2. Αφαιρέσαμε τα σημεία στίξης
3. Μετατρέψαμε τα κεφαλαία γράμματα σε πεζά

Στη συνέχεια τα 100 καθαρά κείμενα συνενώθηκαν σε ένα κείμενο για κάθε ταινία από το οποίο αφαιρέσαμε τα επαναλαμβανόμενα τμήματα που πιθανόν να περιείχε αφαιρώντας τις γραμμές με ομοιότητα ίση με 1. Τέλος αφαιρέσαμε όλες τις "συνηθισμένες" λέξεις (*stop words*) όπως a, I, and, my κ.τ.λ. από την λίστα που παρέχεται από το πανεπιστήμιο Cornell αφού πρόκειται για λέξεις με ελάχιστη ή καθόλου παροχή πληροφορίας. Η όλη διαδικασία της προ επεξεργασίας επαναλήφθηκε και για τις 1000 ταινίες.

5.2.3 Επιλογή χρηστών

Δεδομένου του τεράστιου αριθμού από βαθμολογίες που μας παρείχε η NETFLIX και συνεπώς της πολύ μεγάλης απαίτησης χρόνου που θα απαιτούσε η επεξεργασία αυτών, επιλέξαμε να χρησιμοποιήσουμε στην πειραματική διαδικασία μας τις βαθμολογίες 15 τυχαία επιλεγμένων χρηστών. Μοναδικό κριτήριο στην επιλογή τους ήταν να έχουν δει και βαθμολογήσει περισσότερες των μισών από τις 1000 επιλεγμένες ταινίες. Ο κάθε χρήστης από αυτούς που επιλέξαμε, λοιπόν, είχε βαθμολογήσει πάνω από 500 ταινίες.

Δημιουργήσαμε για κάθε χρήστη το προφίλ του το οποίο περιείχε τους τίτλους των ταινιών που είχε δει καθώς επίσης και τις βαθμολογίες που είχε δώσει σε κάθε μία από αυτές. Οι βαθμολογίες λαμβάνουν ακέραιες τιμές μεταξύ του 1 (η ταινία δεν άρεσε καθόλου στον χρήστη) και 5 (ο χρήστης απόλαυσε την ταινία).
Στον πίνακα 5.1 παρουσιάζουμε τον ακριβή αριθμό των βαθμολογιών που έχει δώσει ο κάθε χρήστης.

Παρατηρούμε πως οι τυχαία επιλεγμένοι χρήστες μας δεν έχουν πολλές βαθμολογίες 1 και 4. Αυτό αν και είναι λογικό να συμβαίνει μας δυσκολεύει στην πρόβλεψη αφού δεν έχουμε

χρήστης	κατ.1	κατ.2	κατ.3	κατ.4	κατ.5	Άθροισμα
$user\_1007577$	55	104	324	288	79	840
$user\_12812$	26	58	258	16	145	503
$user\_110938$	0	21	148	0	385	554
$user\_1110156$	27	89	419	17	78	630
$user\_1114324$	20	77	302	30	81	510
$user\_1118103$	11	99	352	7	32	501
$user\_1174530$	43	76	253	22	109	503
$user\_1178846$	14	57	209	27	202	509
$user\_1220185$	21	66	301	30	82	500
$user\_1233297$	18	101	309	21	98	547
$user\_1287892$	37	234	369	37	22	699
$user\_1314869$	12	41	481	3	115	652
$user\_1365840$	0	38	222	0	241	501
$user\_1370564$	59	161	373	70	50	713
$user\_1374197$	7	99	303	19	107	535
Συνολικό άθροισμα βαθμολογιών όλων των χρηστών:						8697

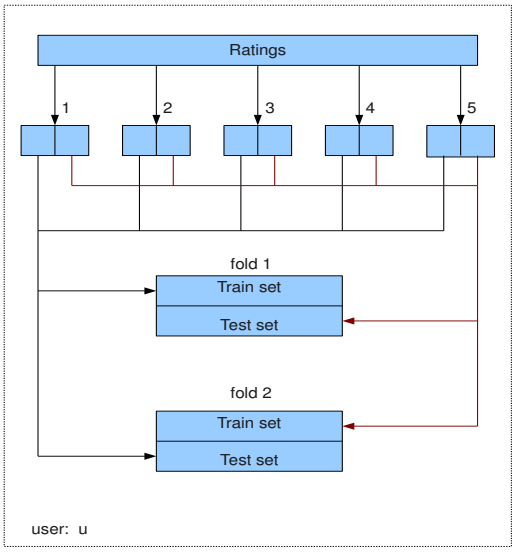
Πίνακας 5.1: Αριθμός βαθμολογιών κάθε χρήστη ανά κατηγορία και συνολικά.

αρκετή πληροφορία για όλες τις βαθμολογίες.

Στη συνέχεια, δημιουργήσαμε για κάθε χρήστη τα σύνολα των κειμένων για εκπαίδευση (train set) και αυτά για αξιολόγηση (test set). Για κάθε διαφορετική βαθμολογία επιλέξαμε τα μισά από τα κείμενα για εκπαίδευση και τα υπόλοιπα για αξιολόγηση. Επιλέξαμε να χρησιμοποιήσουμε την τεχνική αξιολόγησης της απόδοσης του συστήματος μας 2-fold cross validation. Για αυτόν τον σκοπό, λοιπόν, χωρίσαμε το προφίλ του κάθε χρήστη σε δύο διαφορετικά τμήματα (folds). Τα δύο αυτά τμήματα είχαν μεταξύ τους ακριβώς τα αντίθετα σύνολα για εκπαίδευση και αξιολόγηση. Όσα, δηλαδή, κείμενα ανήκαν στο πρώτο τμήμα στο σύνολο εκπαίδευσης στο δεύτερο τμήμα ανήκαν στο σύνολο αξιολόγησης και αντίστροφα. Η διαδικασία αυτή μπορεί να αναπαρασταθεί σχηματικά όπως φαίνεται στο Σχήμα 5.1.

5.2.4 Μετρική αξιολόγησης

Για να αξιολογήσουμε την απόδοση του συστήματος μας χρησιμοποιήσαμε την μετρική του μέσου τετραγωνικού σφάλματος (Mean Square Error, MSE). Το μέσο τετραγωνικό σφάλμα υπολογίζεται από την εξίσωση:



Σχήμα 5.1: Σχηματική αναπαράσταση 2-fold cross validation για έναν χρήστη

$$MSE = \frac{1}{R}\sum_{i \in R}(r_i - \overline{r_i})^2 \tag{5.1}$$

όπου r_i είναι η πραγματική βαθμολογία που έβαλε ο χρήστης στην ταινία, $\overline{r_i}$ η βαθμολογία την οποία προβλέψαμε και R ο συνολικός αριθμός των βαθμολογιών που προβλέψαμε. Είναι προφανές πως μικρότερες τιμές μέσου τετραγωνικού σφάλματος υποδηλώνουν καλύτερα αποτελέσματα. Το εύρος τιμών της συγκεκριμένης μονάδας μέτρησης είναι από 0 (προβλέψαμε ιδανικά όλες τις βαθμολογίες σωστά) μέχρι 16 (κάναμε την χειρότερη δυνατή πρόβλεψη).

5.2.5 Βάση σύγκρισης

Ως μέτρο σύγκρισης στα πειράματα μας χρησιμοποιήσαμε την πρόβλεψη με βάση την μέση τιμή των βαθμολογιών. Αρχικά υπολογίσαμε για κάθε χρήστη και για κάθε τμήμα του προφίλ του τον μέσο όρο των βαθμολογιών που είχε δώσει σε όλες τις ταινίες που είχε δει. Στη συνέχεια, μετρήσαμε το μέσο τετραγωνικό σφάλμα θεωρώντας κάθε φορά σαν βαθμολογία που προβλέψαμε αυτόν τον μέσο όρο. Δηλαδή:

$$MSE_{base} = \frac{1}{R}\sum_{i \in R}(r_i - r_{avg})^2 \tag{5.2}$$

όπου r_{avg} ο μέσος όρος των βαθμολογιών του χρήστη για αυτό το τμήμα του προφίλ του. Η βάση μας για κάθε χρήστη υπολογίστηκε αθροίζοντας το MSE_{base} και των δύο τμημάτων του προφίλ του και διαιρώντας με το δύο. Τέλος, υπολογίσαμε την συνολική βάση σύγκρισης για τα πειράματα μας βρίσκοντας τον μέσο όρο των MSE_{base} όλων των χρηστών. Το αποτέλεσμα αυτού του υπολογισμού και άρα η βάση με την οποία συγκρίνουμε τα αποτελέσματά μας υπολογίστηκε σε Baseline = 1.01.

5.3 Αποτελέσματα

Σε αυτή την ενότητα παρουσιάζουμε τα αποτελέσματα που προκύπτουν από τα διαφορετικά πειράματα που εκτελέσαμε. Επίσης, συγκρίνουμε αυτά τα αποτελέσματα με την βάση σύγκρισης που υπολογίσαμε προηγουμένως καθώς και μεταξύ τους. Παρουσιάζουμε τα αποτελέσματα των πειραμάτων που εκτελέστηκαν σύμφωνα με τις μεθόδους που αναφέρονται στο Κεφ. 4. Όλα τα πειράματα πραγματοποιήθηκαν για τους 15 χρήστες που επιλέξαμε στην παρ. 5.2.3. Ο μέσος όρος του αριθμού των χαρακτηριστικών (λέξεων) με τα οποία εκπαιδεύεται ο ταξινομητής είναι 246541.

5.3.1 Αξιολόγηση μοντέλων

Σε αυτή την ενότητα παρουσιάζουμε τα αποτελέσματα που πήραμε πραγματοποιώντας την πρόβλεψη της κατηγορίας που ένας χρήστης θα δώσει σε μία ταινία χρησιμοποιώντας αρχικά μόνο το $a-priori$ μοντέλο και στην συνέχεια λαμβάνοντας υπ' όψιν μόνο τις πιθανότητες των λέξεων ενός κειμένου. Αυτό το επιτύχαμε χρησιμοποιώντας την εξίσωση:

$$c^* = \arg\max_{c_j \in C}\{\lambda_1 \log P(c_j) + \lambda_2 \sum_{i=1}^{N} \log(P(w_i|c_j))\} \tag{5.3}$$

και θέτοντας αρχικά $\lambda_2 = 0$ και $\lambda_1 = 1$ και στη συνέχεια αντίστροφα.

5.3.1.1 Μοντέλο a-priori

Στον πίνακα 5.2 παρουσιάζουμε το μέσο τετραγωνικό σφάλμα της πρόβλεψης μας με την χρησιμοποίηση μόνο της $a - priori$ πιθανότητας της κατηγορίας.

MSE a-priori model	
$\lambda_1 = 1$,$\lambda_2 = 0$	1.12
Baseline : 1.01	

Πίνακας 5.2: MSE για 15 χρήστες με χρήση μόνο της $a - priori$ πιθανότητας.

5.3.1.2 Μοντέλο χωρίς a-priori

Στη συνέχεια, αντιστρέψαμε τις τιμές των λ_1 και λ_2 έτσι ώστε να αξιολογήσουμε το μοντέλο μας χωρίς να λαμβάνουμε υπ' όψιν τις $a - priori$ πιθανότητες. Αυτό το πείραμα το διεξάγουμε αρχικά για όλα τα χαρακτηριστικά μας (τα οποία αριθμούν κατά μέσο όρο σε 246541), και στη συνέχεια επιλέγοντας εκείνα που έχουν saliency μεγαλύτερο του μηδενός (κατά μέσο όρο 75748 χαρακτηριστικά), τα 10 χιλιάδες με το μεγαλύτερο saliency και τέλος τα 5 χιλιάδες με το μεγαλύτερο saliency. Στους πίνακες 5.3 και 5.5 βλέπουμε για έναν χρήστη ενδεικτικά χαρακτηριστικά με μεγάλο και μικρό saliency αντίστοιχα. Αυτοί οι πίνακες θα μας βοηθήσουν να εξηγήσουμε τα αποτελέσματα που πήραμε από την παραπάνω διαδικασία.

Παρατηρούμε εύκολα πως ενώ τα χαρακτηριστικά με χαμηλό saliency είναι πολύ λογικό να έχουν αυτό το βάρος μιας και είναι λέξεις που δεν προσφέρουν ιδιαίτερη πληροφορία ούτε μπορούν να θεωρηθούν ενδεικτικές για κάποια βαθμολογία, τα αντίστοιχα χαρακτηριστικά με μεγάλο saliency μπορεί να είναι απλά ανορθόγραφες λέξεις ή λέξεις που αν και εμφανίζονται λίγες φορές σε μία και μόνο κατηγορία δεν είναι ενδεικτικές για αυτήν. Αυτό οδηγεί το μοντέλο μας να μην έχει καλή απόδοση στην πρόβλεψη. Τα αποτελέσματα φαίνονται στον πίνακα 5.6. Εάν βάλουμε κάποιο κατώτατο όριο στη συχνότητα εμφάνισης των χαρακτηριστικών (cut-off threshold) έτσι ώστε να απορρίπτονται ανορθόγραφες ή σπάνιες

feature	weigth
realmagic	3.0105
outputthis	3.0105
athiest	3.0105
moonshot	3.0105
theparent	2.9348
chevaliers	2.9348
liscense	2.9348
adrianna	2.8869
titantic	2.8869
cypriot	2.7849
sauron	2.5165
seanconnery	2.4750
mormonism	2.3648
frodo	1.3418
primadonna	1.2976

Πίνακας 5.3: Χαρακτηριστικά με υψηλό saliency χωρίς cut-off threshold.

feature	weigth
coplin	2.4324
indiana	2.4276
tiddlers	2.4284
welton	2.3280
commodus	2.2108
maximus	2.1558
rocky	2.1481
mcfly	2.0983
tombstone	2.0974
archery	1.9179
rasputin	1.8944
keating	1.8936
gandalf	1.3506
frodo	1.3418
goldmember	1.3373

Πίνακας 5.4: Χαρακτηριστικά με υψηλό saliency με cut-off threshold = 5.

im	-0.1138
final	-0.1130
single	-0.1129
key	-0.1124
debut	-0.1123
understand	-0.1123
theaters	-0.1122
usa	-0.1119
year	-0.1117
result	-0.1116
to	-0.1115
working	-0.1114
living	-0.1111
words	-0.1110
digital	-0.1105
body	-0.1097

Πίνακας 5.5: Χαρακτηριστικά με χαμηλό saliency.

λέξεις θα είχαμε καλύτερα αποτελέσματα στην επιλογή χαρακτηριστικών. Στον πίνακα 5.4 βλέπουμε τις λέξεις με υψηλό saliency και cut-off threshold ίσο με 5 όπου παρατηρούμε πως έχουν αφαιρεθεί οι ανορθόγραφες λέξεις και τα χαρακτηριστικά με υψηλό saliency είναι πιο λογικά.

Baseline : 1.01	
χωρίς επιλογή χαρ.	1.16
saliency > 0(75748 χαρ.)	1.33
10k χαρακτηριστικά	1.40
5k χαρακτηριστικά	1.69

Πίνακας 5.6: MSE για 15 χρήστες χωρίς χρήση της $a - priori$ πιθανότητας.

Σε αυτό το σημείο επαναλάβαμε το πείραμα χρησιμοποιώντας το $tfidf$ βάρος κάθε χαρακτηριστικού. Στον πίνακα 5.7 βλέπουμε ενδεικτικά χαρακτηριστικά με μεγάλο $tfidf$ βάρος για μία ταινία.

bullock	443.2459
sandra	211.8814
contestant	137.8036
fbi	118.3165
diamond	107.4507
undercover	72.2951
miss	67.5616
agent	57.2963
beauty	35.0988
terrorist	24.8070
detectable	19.4651
femininity	18.6065
bikini	14.3900

Πίνακας 5.7: Χαρακτηριστικά με υψηλό $tfidf$ βάρος για την ταινία Miss Congeniality.

Το μέσο τετραγωνικό σφάλμα σε αυτή την περίπτωση χρησιμοποιώντας όλα τα χαρακτηριστικά υπολογίστηκε 1.12. Παρατηρούμε πως το MSE αυτό είναι καλύτερο από εκείνο της περίπτωσης που χρησιμοποιούμε μόνο τη συχνότητα εμφάνισης της λέξης (γεγονός φυσιολογικό αφού η μέθοδος απόδοσης βαρών $tfidf$ δίνει υψηλά βάρη σε χαρακτηριστικά τα οποία μας παρέχουν πολλή πληροφορία για κάποια ταινία, όπως ουσιαστικά, ονόματα πρωταγωνιστών, κ.τ.λ.), οπότε επαναλάβαμε τα πειράματα συνδυάζοντας το saliency με το tfidf βάρος.

Χρησιμοποιήσαμε βάρη λ_t και λ_s στο $tfidf$ και το $saliency$ αντίστοιχα για τα οποία ισχύει $\lambda_t + \lambda_s = 1$. Η επιλογή των χαρακτηριστικών σε αυτή τη διαδικασία έγινε για τα χαρακτηριστικά με θετικό saliency. Επαναλάβαμε τα πειράματά μας για διάφορους συνδυασμούς αυτών των βαρών με σκοπό να επιλέξουμε εκείνες τις τιμές οι οποίες μας δίνουν καλύτερα

αποτελέσματα. Τα νούμερα για το μέσο τετραγωνικό σφάλμα της πρόβλεψης που μας έδωσε η παραπάνω διαδικασία φαίνονται στον πίνακα 5.8.

Baseline : 1.01	
	saliency > 0 (75748 χαρ.)
$\lambda_t = 0, \lambda_s = 1$	1.16
$\lambda_t = 0.1, \lambda_s = 0.9$	1.15
$\lambda_t = 0.5, \lambda_s = 0.5$	1.13
$\lambda_t = 0.9, \lambda_s = 0.1$	1.20
$\lambda_t = 1, \lambda_s = 0$	1.27

Πίνακας 5.8: MSE για 15 χρήστες με επιλογή χαρακτηριστικών με βάση το $tfidf^{\lambda_t} * saliency^{\lambda_s}$ χωρίς χρήση της $a-priori$ πιθανότητας.

Παρατηρούμε πως χρησιμοποιώντας μόνο το μοντέλο μας και αγνοώντας τις $a-priori$ πιθανότητες τα καλύτερα αποτελέσματα που παίρνουμε είναι για την περίπτωση που συνδυάζουμε το $tfidf$ με το $saliency$ και επιλέγοντας τα χαρακτηριστικά με θετικό $saliency$. Επειδή όμως το $tfidf$ και το $saliency$ είναι άλλης τάξης μεγέθη, θέτοντας τους εκθέτες $\lambda_t = \lambda_s = 0.5$ κανονικοποιούμε τα μεγέθη παίρνοντας καλύτερα αποτελέσματα. Αυτό συμβαίνει γιατί ναι μεν πλησιάζει η τάξη των δύο μεγεθών αλλά ταυτόχρονα με τον συγκεκριμένο συνδυασμό βαρών δεν ταυτίζεται. Έτσι καταφέρνουμε να δώσουμε μεγάλη πιθανότητα σε χαρακτηριστικά έχουν υψηλό tfidf και είναι και αντιπροσωπευτικά για μία βαθμολογία.

Στη συνέχεια θα δείξουμε τα αποτελέσματα που πήραμε για το τετραγωνικό σφάλμα της πρόβλεψης μας την οποία αυτή τη φορά πραγματοποιήσαμε συνδυάζοντας το (αποδοτικότερο σε πρόβλεψη) $a-priori$ μοντέλο με το δικό μας.

5.3.2 Αξιολόγηση συνδυασμού μοντέλων

Σε αυτή την ενότητα παρουσιάζουμε τα αποτελέσματα που πήραμε συνδυάζοντας τα δύο μοντέλα. Αυτό έγινε βάζοντας διαφορετικές τιμές στα βάρη λ_1 και λ_2 της (5.3) εξετάζοντας τον λόγο τους λ_1/λ_2. Με λ_1 θεωρούμε την προκατάληψη την ποία δίνουμε στην $a-priori$ πιθανότητα. Αυτό έχει σημασία διότι, για παράδειγμα, ένας χρήστης ο οποίος τείνει να βαθμολογεί αυστηρά είναι πιο πιθανό να δώσει χαμηλή βαθμολογία σε κάποια ταινία ενώ

κάποιος άλλος χρήστης που δίνει εν γένει καλές βαθμολογίες είναι πολύ πιθανό να συνεχίσει να βαθμολογεί έτσι.

Αρχικά βλέπουμε στον πίνακα 5.9 τις τιμές του μέσου τετραγωνικού σφάλματος της πρόβλεψης μας για όλους τους διαφορετικούς λόγους λ_1/λ_2, στην περίπτωση χρησιμοποίησης όλων των χαρακτηριστικών μας (κατά μέσο όρο 246541 για κάθε χρήστη).

Baseline : 1.01	
λ_1 / λ_2	χωρίς επιλογή χαρακτηριστικών
0.01	1.15
0.1	1.15
1	1.15
10	1.15
100	1.13
1000	1.09

Πίνακας 5.9: MSE για 15 χρήστες χωρίς επιλογή χαρακτηριστικών.

Στη συνέχεια, δείχνουμε τα αποτελέσματα για την περίπτωση που κάνουμε επιλογή χαρακτηριστικών αξιοποιώντας το $saliency$ του κάθε χαρακτηριστικού και υπολογίζοντας τις πιθανότητες των κειμένων με βάση την συχνότητα εμφάνισης των χαρακτηριστικών αυτών. Ο πίνακας 5.10 δείχνει το MSE για τις περιπτώσεις που επιλέξαμε όλα τα χαρακτηριστικά με θετικό $saliency$ (κατά μέσο όρο 75748 για κάθε χρήστη), στη συνέχεια τα 10 χιλιάδες μεγαλύτερου $saliency$ και τέλος τα αντίστοιχα 5 χιλιάδες χαρακτηριστικά.

Baseline : 1.01			
λ_1 / λ_2	$saliency > 0$ (75748 χαρ.)	10k χαρ.	5k χαρ.
0.01	1.33	1.40	1.68
0.1	1.33	1.37	1.62
1	1.32	1.24	1.32
10	1.28	1.12	1.12
100	1.18	1.11	1.11
1000	1.08	1.12	1.12

Πίνακας 5.10: MSE για 15 χρήστες με επιλογή χαρακτηριστικών με βάση το $saliency$.

Τέλος, στον πίνακα 5.11 βλέπουμε το μέσο τετραγωνικό σφάλμα για διάφορους λόγους βαρών των δύο μοντέλων για την περίπτωση του συνδυασμού του $tfidf$ βάρους με το $saliency$ των χαρακτηριστικών που το τελευταίο είναι μεγαλύτερο του μηδενός. Η διαδικασία αυτή υλοποιήθηκε μόνο για βάρη $\lambda_t = \lambda_s = 0.5$ τα οποία στην προηγούμενη ενότητα μας έδωσαν τα καλύτερα αποτελέσματα.

Baseline : 1.01	
λ_1 / λ_2	$\lambda_t = 0.5, \lambda_s = 0.5$ $(saliency > 0)$
0.01	1.12
0.1	1.13
1	1.13
10	1.12
100	1.10
1000	1.11

Πίνακας 5.11: MSE για 15 χρήστες με επιλογή χαρακτηριστικών με βάση το $tfidf^{\lambda_t} * saliency^{\lambda_s}$.

Παρατηρούμε πως το μοντέλο $a - priori$ όταν πάρει μεγαλύτερο βάρος βελτιώνει τα αποτελέσματα μας. Αυτό είναι φυσιολογικό από την στιγμή που τα μεγέθη $\log P(c_j)$ και $\sum_{i=1}^{N} \log(P(w_i|c_j))$ έχουν μεγάλη διαφορά στην τάξη τους.

Δηλαδή: $|\sum_{i=1}^{N} \log(P(w_i|c_j))| >> |\log P(c_j)|$. και συνεπώς βάζοντας μεγάλο βάρος στην $a - priori$ παίζει όλο και μεγαλύτερο ρόλο στην πρόβλεψη.

5.4 Συμπεράσματα

Σε αυτό το κεφάλαιο δείξαμε τα αποτελέσματα που πήραμε εκτελώντας τα πειράματα μας με τρεις διαφορετικούς τρόπους. Αρχικά, λαμβάνοντας υπ' όψιν μόνο την a-priori πιθανότητα, στη συνέχεια χωρίς να την λαμβάνουμε καθόλου υπ' όψιν και τέλος, συνδυάζοντας τα δύο αυτά μοντέλα αποδίδοντας τους κάποια βάρη. Καταλήγουμε στο συμπέρασμα πως το μοντέλο a- priori δουλεύει καλύτερα από την άλλη υλοποίηση αλλά ο συνδυασμός των δύο μας δίνει σε κάποιες περιπτώσεις λίγο καλύτερα αποτελέσματα. Ακόμη, είδαμε πως εξαιτίας των πολύ κακών χαρακτηριστικών δεν είχαμε κάποια ιδιαίτερη βελτίωση εφαρμόζοντας επιλογή χαρακτηριστικών. Αυτό είναι λογικό επειδή τα δεδομένα μας προέρχονται από το Διαδίκτυο και, φυσικά, περιέχουν άχρηστη πληροφορία. Με κάποια πιο έξυπνη επιλογή χαρακτηριστικών θα μπορούσαμε να επιτύχουμε καλύτερα αποτελέσματα.

Κεφάλαιο 6

Συμπεράσματα - Μελλοντική δουλειά

6.1 Γενικά συμπεράσματα

Σε αυτή την εργασία καταφέραμε να χρησιμοποιήσουμε μεθόδους μηχανικής μάθησης εκπαιδεύοντας τον naive Bayes ταξινομητή και βασιζόμενοι στην υπόθεση ανεξαρτησίας μεταξύ των λέξεων ενός κειμένου οδηγηθήκαμε σε ένα σύστημα σύστασης ταινιών το οποίο συνδυάζει λεκτικά χαρακτηριστικά με βαθμολογίες. Δείξαμε πως λεκτικά χαρακτηριστικά μη δομημένων κειμένων τα οποία είναι αυτόματα προσβάσιμα μέσω του Διαδικτύου μπορούν να χρησιμοποιηθούν έτσι ώστε να μοντελοποιήσουν τον τρόπο με τον οποίο ένας χρήστης βαθμολογεί μία ταινία, εκπαιδεύοντας γλωσσικά μοντέλα για κάθε βαθμολογία από ένα σύνολο ήδη βαθμολογημένων ταινιών για κάποιο συγκεκριμένο χρήστη.

Αρχικά εκπαιδεύσαμε τα μοντέλα μας χωρίς να κάνουμε κάποια επιλογή χαρακτηριστικών. Χρησιμοποιήσαμε δύο διαφορετικές μεθόδους απόδοσης βαρών, την απλή συχνότητα εμφάνισης του κάθε χαρακτηριστικού και το $tfidf$ βάρος του ώστε να δώσουμε υψηλότερο βάρος σε λέξεις σχετικές με κάποια ταινία (ουσιαστικά, ονόματα πρωταγωνιστών κ.α). Είδαμε ότι εκπαιδεύοντας τα μοντέλα μας με το $tfidf$ βάρος έχουμε καλύτερα αποτελέσματα. Είναι λογικό αφού ουσιαστικά και ονόματα ηθοποιών περιέχουν σημαντική πληροφορία για κάποια ταινία.

Επειδή τα δεδομένα που χρησιμοποιήσαμε περιέχουν ένα μεγάλο μέρος από άχρηστη πληροφορία προσπαθήσαμε να εξάγουμε και να επιλέξουμε τις πιο σημαντικές λέξεις, τα χαρακτηριστικά τα οποία διαχωρίζουν καλύτερα τις κατηγορίες (βαθμολογίες) για ένα συγκεκριμένο χρήστη χρησιμοποιώντας το cross-entropy μεταξύ ενός χαρακτηριστικού και κάθε βαθμολογίας. Παρατηρήσαμε ότι η συγκεκριμένη μέθοδος απορρίπτει λέξεις καθημερινές όπως ρήματα, συνδέσμους ή συχνά εμφανιζόμενα επίθετα και κρατά λέξεις που είναι πιο

σχετικές με ταινίες ή πιο σπάνια επίθετα. Δυστυχώς η εφαρμογή της συγκεκριμένης μεθόδου για επιλογή χαρακτηριστικών δεν έδωσε καλύτερα αποτελέσματα καθώς επιλέχτηκαν και πολλές ανορθόγραφες λέξεις ή λέξεις σπάνιες που δεν περιέχουν καμία χρήσιμη πληροφορία. Τελικά, παρατηρήσαμε ότι η εφαρμογή ενός κατώτατου ορίου στην συχνότητα εμφάνισης των λέξεων αφαιρεί τις ανορθόγραφες ή σπάνιες λέξεις και δυνητικά οδηγεί σε καλύτερη επιλογή χαρακτηριστικών.

Επίσης, δοκιμάσαμε να συνδυάσουμε το $tfidf$ βάρος με το $saliency$ για τα πιο salient ($saliency > 0$) χαρακτηριστικά και παρατηρήσαμε πως η χρήση του $tfidf$ βάρους αυξάνει την απόδοση του συστήματος για την συγκεκριμένη επιλογή χαρακτηριστικών.Τέλος, δίνοντας βάρος στη $a - priori$ πιθανότητα παρατηρήσαμε πως η απόδοση του συστήματός μας αυξάνεται. Με όλες τις μεθόδους που εφαρμόσαμε στην εργασία αυτή δεν καταφέραμε να ξεπεράσουμε την βάση σύγκρισης μας. Στη συνέχεια παρουσιάζουμε κάποια μελλοντική δουλειά η οποία θα μπορούσε να επιτύχει ακριβέστερη πρόβλεψη.

6.2 Μελλοντική δουλειά

Η μελλοντική δουλειά μπορεί να αφορά σε διάφορους τομείς της εργασίας μας. Αρχικά, ο τομέας που μπορεί να διαφοροποιηθεί είναι αυτός των δεδομένων μας. Αυτό μπορεί να επιτευχθεί αντλώντας λεκτικά χαρακτηριστικά από συγκεκριμένες πηγές περισσότερο εξειδικευμένες πάνω στο αντικείμενο, ή επιχειρώντας μία εφαρμογή του συστήματος σε μεγαλύτερου όγκου δεδομένα. Αυτό θα εξασφαλίσει την καλύτερη επιλογή χαρακτηριστικών μιας και θα έχουμε περισσότερη και πιο σχετική πληροφορία.

Ένας άλλος τομέας στον οποίο θα μπορούσε να εφαρμοστεί διαφορετική μεθοδολογία είναι αυτός της επεξεργασίας φυσικής γλώσσας. Αυτό περιλαμβάνει, για παράδειγμα, την εξέταση της ρίζας του κάθε λεκτικού χαρακτηριστικού έτσι ώστε το σύστημα να μην αντιλαμβάνεται ως διαφορετικές, λέξεις με κοινή ρίζα (π.χ. computer, computers). Αυτή η ομαδοποίηση των λέξεων ανάλογα με τη ρίζα τους ίσως να οδηγούσε σε καλύτερη απόδοση.

Επιπροσθέτως, θα ήταν ενδιαφέρουσα η χρησιμοποίηση διαφορετικών μεθόδων επιλογής χαρακτηριστικών οι οποίες πιθανόν να ενσωματώνουν κάποιες μεθόδους αποκλεισμού ανορθόγραφων λέξεων ή χαρακτηριστικών χωρίς ουσιαστική πληροφορία (cut-off thresholds). Με αυτόν τον τρόπο οι λέξεις που δεν μας παρέχουν καμία πληροφορία θα αποκλείονταν και πιθανώς να μας βοηθούσαν στην προσπάθεια πρόβλεψης.

Στον τομέα της ταξινόμησης κειμένων θα μπορούσε να χρησιμοποιηθεί κάποιος άλλος ταξινομητής (CAN, SVM, κ.ά). Κάποιοι από αυτούς λαμβάνουν υπ' όψιν τους τις εξαρτήσεις

μεταξύ των λέξεων και ενδεχομένως καθίστανται πιο αποδοτικοί.

Τέλος, θα είχε ενδιαφέρον η αξιολόγηση του συστήματος αυτού σε κάποιο άλλο, διαφορετικό των ταινιών, φάσμα αντικειμένων όπως για παράδειγμα αυτό της μουσικής ή των βιβλίων.

Βιβλιογραφία

[1] J. Wang, Ar. P. de Vries, M. J.T. Reinders *Unifying User-based and Item-based Collaborative Filtering Approaches by Similarity Fusion.* SIGIR '06, Seatle, Washington, USA, pp.501-508, 2006

[2] D. Goldberg, D. Nichols, B.M. Oki and D. Terry *Using Collaborative Filtering to Weave an Information Tapestry.* Communications of the ACM. December, 1992.

[3] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom and J. Riedl. *GroupLens: An Open Architecture for Collaborative Filtering of Netbews.* In Proceedings of CSCW '94, Chapel Hill, NC, 1994.

[4] U. Shardanand and P. Maes. *Social Information Filtering: Algorithms for Automating 'Word of Mouth'.* In Proceedings of CHI '95, Denver, CO, 1995.

[5] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. *Item-Based Collaborative Filtering Recommendation Algorithms.* In Proceedings of the 10th International Conference on World Wide Web 2001, Hong Kong, pp. 285-295. 2001.

[6] J. S. Breese, D. Heckerman, and C. Kadie. *Empirical Analysis of Predictive Algorithms for Collaborative Filtering*. In Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence, pp. 43-52, 1998.

[7] J. L. Herlocker, J. A. Konstan, Al. Borchers, and J. Riedl *An algorithmic Framework for Performing Collaborative Filtering*. In Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Berkley, California, USA, pp.230-237, 1999.

[8] F. Peng and D. Schuurmans *Combining Naive Bayes and n-Gram Language Models for Text Classification.* School of Computer Science,University of Waterloo 200 University Avenue West, Waterloo, Ontario, Canada N2L 3G1

[9] R. Duda and P. Hart. *Pattern Classification and Scene Analysis.* Wiley, NY. 336-337, 1973.

[10] S. Chen and J. Goodman. *An Empirical Study of Smoothing Techniques for Language Modeling*. Technical report, TR-10-98, Harvard University, pp. 338,340, 1998.

[11] C. Manning, P. Raghavan and H. Schutze. *Introduction to Information Retrieval.* (freely available in http://nlp.stanford.edu/IR-book/html/htmledition/irbook.html), 2008.

[12] D.D Lewis and M.Ringuette. *Comparison of two Learning Algorithms for Text Categorization.* In Proceedings of the Third Annual Symposium on Document Analysis and Information Retrieval(SDAIR'94), 1994.

[13] H. Schütze, D.A. Hull and J.O Pedersen. *A Comparison of Classifiers and Document Representations for the Routing Problem.* In 18th Ann Int ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'95), pages 229-237, 1995.

[14] E. Wiener, J.O Pedersen and A.S. Weigend. *A Neural Network Approach to Topic Spotting.* In Proceedings of the Fourth Annual Symposium on Document Analysis and Information Retrieval(SDAIR'95), 1995.

[15] G. Salton and C. Buckley. *Term Weighting Approaches in Automatic Text Retrieval.* Technical Report: TR87-881 , Cornell University, Ithaca, NY, USA, 1987.

[16] J. Ramos. *Using TF-IDF to Determine Word Relevance in Document Queries.* Department of Computer Science, Rutgers University, Piscataway, NJ, 2004.

[17] Y. Yang and J.O. Pedersen. *A Comparative Study on Feature Selection in Text Categorization,* 1996.

[18] S. Kullback. *Information Theory and Statistics.* John Wiley and Sons, NY, 1959.

[19] A. Potamianos, S. Yildirim, C.M. Lee, S. Lee, S. Narayanan *Detecting Politeness and Frustration State of a Child in a Conversational Computer Game.* In Proceedings of InterSpeech, Lisbon, Portugal, pp. 2209-2212, October 2005.

[20] D. Jurafsky, J.H. Martin. *Speech and Language Processing. An Introduction to Natural Processing, Computational Linguistics and Speech Recognition.* Prentice Hall, Englewood Cliffs, New Jersey, 2000.