ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Ανάλυση Διάθεσης με Συνεχή Γλωσσικά Μοντέλα

Διπλωματική Εργασία
του
Κωνσταντίνου Θ. Καράλα

Εξεταστική Επιτροπή:
Καθηγητής Βασίλης Διγαλάκης (ΗΜΜΥ)
Αναπληρωτής Καθηγητής Μιχαήλ Λαγουδάκης (ΗΜΜΥ)
Καθηγητής Μιχάλης Πατεράκης (ΗΜΜΥ)

Χανιά, Σεπτέμβριος 2013

TECHNICAL UNIVERSITY OF CRETE, GREECE
SCHOOL OF ELECTRONIC AND COMPUTER ENGINEERING

Sentiment Analysis Using Continuous-Space Language Models

Diploma Thesis
by
Konstantinos T. Karalas

Thesis Committee:
Professor Vassilis Digalakis (ECE)
Associate Professor Michail Lagoudakis (ECE)
Professor Michael Paterakis (ECE)

Chania, September 2013

Περίληψη

Η εύρεση του συναισθήματος στο γραπτό λόγο, είναι ένας τρόπος για να ανιχνεύσουμε τις στάσεις, τις διαθέσεις και τις απόψεις των ανθρώπων. Το ενδιαφέρον πάνω σε αυτόν τον τομέα έχει παρουσιάσει ραγδαία αύξηση την τελευταία δεκαετία, καθώς η ανάπτυξη των μέσων κοινωνικής δικτύωσης και των ιστολογιών προσέφερε ένα πλήθος διαθέσιμων πηγών για έρευνα. Παράλληλα, προκύπτουν πολλαπλά οφέλη τόσο για τους καταναλωτές, οι οποίοι μπορούν να παρακολουθούν μαζικά προϊόντα, ειδήσεις ή πρόσωπα που τους ενδιαφέρουν, όσο και για τις επιχειρήσεις, στις οποίες δίνεται πλέον η δυνατότητα να αφουγκράζονται ευκολότερα και με μεγαλύτερη ακρίβεια τα παράπονα και τη ζήτηση του καταναλωτικού κοινού.

Σε αυτή τη διπλωματική εργασία, εξετάζεται η κατηγοριοποίηση του συναισθήματος σε θετικό ή αρνητικό χρησιμοποιώντας γλωσσικά μοντέλα που δημιουργήθηκαν από συνεχείς κατανομές και πιο συγκεκριμένα τα Gaussian Mixture Models. Αρχικά μελετήθηκε η επίδραση συναρτήσεων στάθμισης από το πεδίο της Ανάκτησης Πληροφορίας πάνω στα αρχικά δεδομένα χωρίς μείωση διαστάσεων, όπως επίσης και το πως επηρεάζουν την προσέγγισή μας διάφοροι αλγόριθμοι περιστολής. Στη συνέχεια εξετάσαμε δύο τεχνικές για τη μείωση των διαστάσεων των χαρακτηριστικών διανυσμάτων των λέξεων. Η πρώτη εφαρμόζει Singular Value Decomposition στους πίνακες δεδομένων, ενώ η δεύτερη χρησιμοποιεί μεθόδους επιλογής των πιο σημαντικών χαρακτηριστικών διανυσμάτων. Προσπαθούμε να διαπιστώσουμε πως επιδρά το μοντέλο μας πάνω σε μια δημοσίως διαθέσιμη συλλογή από κριτικές ταινιών, σε σχέση με προηγούμενες μεθόδους μηχανικής μάθησης από το διακριτό χώρο. Η μέθοδός μας πετυχαίνει απόδοση μεγαλύτερη του 82%, ένα ποσοστό αρκετά υψηλό για τα συγκεκριμένα δεδομένα.

Λέξεις Κλειδιά: ανάλυση συναισθήματος, κατηγοριοποίηση συναισθήματος, συνεχή γλωσσικά μοντέλα, συνεχής χώρος, μίξη γκαουσιανών μοντέλων, επιλογή χαρακτηριστικών διανυσμάτων.

Abstract

Sentiment analysis allows us to track attitudes, feelings and opinions from the people. The interest for this field has grown up rapidly in the last decade, since the explosion of web platforms has offered plenty of sources for research. In parallel, there are multiple benefits not only for the consumers, who can monitor products, events or even people of their interest, but also for the companies, as it is easier to listen to the grievances and the demands of the society.

In this thesis, it is examined the sentiment classification into positive or negative by using continuous language models and more concretely Gaussian Mixture Models. At the beginning it is studied the effect of weighting schemes adopted from the Information Retrieval field on the initial data without dimensionality reduction, as well as how different stemming algorithms affect our approach. Consequently, we examined two techniques for the dimensionality reduction of the word vectors. The first one applies Singular Value Decomposition on the data matrices, while the second one uses feature selection methods. Our aim is to find out how our model performs on a publicly available collection of movie reviews, in relation to former machine learning approaches from the discrete space. Our method achieves a performance higher than 82%, a rate high enough for the specific data.

Keywords: sentiment analysis, opinion mining, sentiment classification, continuous language models, continuous space, gaussian mixture models, stemming, svd, feature selection.

Περιεχόμενα

Κατάλογος Σχημάτων

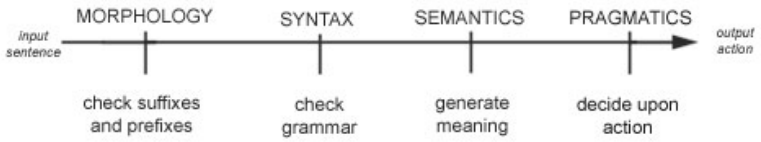
Κατάλογος Πινάκων

Κεφάλαιο 1

Εισαγωγή

Η γλώσσα είναι το κύριο μέσο του ανθρώπου για τη διατύπωση και τη συναλλαγή απόψεων, γνώσεων και πληροφοριών, καθώς και τη μετάδοσή τους από γενιά σε γενιά μέσα στους αιώνες. Στις μέρες μας, λόγω του internet, ο όγκος των γραπτών δεδομένων που μπορεί να έχει πρόσβαση κανείς είναι πλέον τεράστιος. Συνεπώς, είναι επιτακτική η ανάγκη να υπάρξει ένας τρόπος έτσι ώστε να μπορέσουμε να επωφεληθούμε από όλα αυτά τα δεδομένα, εξάγοντας τα στοιχεία εκείνα που μας ενδιαφέρουν και μας αφορούν, χωρίς όμως να χρειαστεί να ξοδέψουμε πολύ από το χρόνο που διαθέτουμε διαβάζοντάς τα ενδελεχώς. Στόχος λοιπόν της επεξεργασίας φυσικής γλώσσας (Natural Language Proccessing - NLP), είναι να καταφέρει ο άνθρωπος με τη βοήθεια των υπολογιστών να επεξεργάζεται με αυτόματο τρόπο προτάσεις μιας φυσικής-καθημερινής γλώσσας, όπως τα αγγλικά ή τα ελληνικά, έναντι μιας συνηθισμένης τεχνητής γλώσσας προγραμματισμού, όπως η C++ ή η Java.

Για να επιτρέψουμε στις μηχανές να καταλάβουν την ανθρώπινη γλώσσα, θα πρέπει να διαιρέσουμε τη διαδικασία σε κάποια βασικά επίπεδα. Αρχικά λοιπόν, πρέπει να εφαρμοστεί η ανάλυση σήματος (Signal Processing), όπου έχουμε ως είσοδο τον ήχο των λέξεων από μία φράση, με σκοπό να μετατραπεί σε κείμενο. Ακολουθεί η συντακτική ανάλυση (Syntactic Analysis), μέσω της οποίας κάθε πρόταση χωρίζεται στα συστατικά της μέρη, επιδιώκοντας την κατανόηση της δομής και της γραμματικής της. Η σημασιολογική ανάλυση (Semantic Analysis) που λαμβάνει χώρα στη συνέχεια, έχει να κάνει με την κατανόηση του νοήματος των λέξεων και των προτάσεων. Τελευταίο και πιο δύσκολο βήμα είναι η πραγματολογία (Pragmatics), η οποία αφορά το πραγματικό νόημα της πρότασης από έναν ομιλητή (ή γράφοντα) σε συνθήκες καθημερινής ανθρώπινης επικοινωνίας, επιδιώκοντας να προσδιορίσει το περιεχόμενο της. Στην παρούσα διπλωματική εργασία θα αγνοήσουμε τελείως το θέμα της ανάλυσης σήματος, μιας και οι υπολογιστές δέχονται στην περίπτωσή μας κατ΄ ευθείαν ως είσοδο το κείμενο, το οποίο προέρχεται από κάποιο αρχείο ή πηγή (είτε από το πληκτρολόγιο γενικότερα).



Σχήμα 1.1: Επίπεδα NLP

Η ανάλυση συναισθήματος (sentiment analysis/opinion mining), αναφέρεται σε μια εφαρμογή της επεξεργασίας φυσικής γλώσσας, μέσω της οποίας μπορούμε να αναγνωρίσουμε αλλά και να εξάγουμε υποκειμενική πληροφορία. Είναι δηλαδή ο κλάδος εκείνος που μας επιτρέπει να προσδιορίσουμε τη στάση του συγγραφέα απέναντι σε ένα συγκεκριμένο ζήτημα, ή τη συνολική πολικότητα ενός εγγράφου. Αν θα θέλαμε να την εντάξουμε ανάμεσα σε κάποια από τα στάδια της ιεραρχίας του NLP έτσι όπως αυτά περιγράφτηκαν παραπάνω, τότε θα υπήρχε ως επιπλέον είσοδος στη σημασιολογική ανάλυση και την πραγματολογία. Μέσω της ανάλυσης των συναισθημάτων λοιπόν, μπορούμε να απαντήσουμε σε χρήσιμα ερωτήματα όπως: «Η συγκεκριμένη κριτική για την ταινία είναι θετική ή αρνητική;», «Τι σκέφτεται το καταναλωτικό κοινό γύρω από ένα συγκεκριμένο προϊόν;», «Το e-mail του χρήστη δείχνει ικανοποίηση ή απογοήτευση;», «Πώς είναι η αυτοπεποίθηση του καταναλωτή;», ή ακόμη και «Πώς βλέπει η κοινωνία ένα συγκεκριμένο (πολιτικό) πρόσωπο ή θέμα;». Είναι επομένως σαφές πως αξίζει από κάθε πλευρά να ασχοληθούμε για την περαιτέρω ανάπτυξη και βελτίωση του συγκεκριμένου τομέα.

1.1 Σκοπός της Διπλωματικής

Μέσα από αυτή τη διπλωματική εργασία, προτείνουμε ένα γλωσσικό μοντέλο συνεχούς χώρου που χρησιμοποιεί συνεχείς κατανομές, με απώτερο σκοπό την ταξινόμηση του συναισθήματος ανάμεσα σε δύο κατηγορίες, θετικό και αρνητικό. Για το λόγο αυτό, υποθέτουμε ότι η συνάρτηση πυκνότητας πιθανότητας που μοντελοποιεί τα χαρακτηριστικά διανύσματα των λέξεων, μπορεί να μοντελοποιηθεί χρησιμοποιώντας ένα Gaussian Mixture Model. Η ιδέα βέβαια για συνεχή γλωσσικά μοντέλα δεν είναι τελείως καινούρια, αφού οι Affify et al. [14] και ο Chen [16], πρότειναν πρώτοι γλωσσικά μοντέλα με Gaussian mixtures για αναγνώριση φωνής. Ωστόσο, απ΄ όσο γνωρίζουμε, ανάλογες τεχνικές συνεχούς χώρου δεν έχουν εφαρμοστεί μέχρι στιγμής στο πεδίο της sentiment analysis. Συνεχίζοντας, χρειαζόμαστε ένα μοντέλο το οποίο να μην είναι ούτε πολύ απλό, για να μπορεί να προσεγγίσει την πραγματική κατανομή, ούτε όμως και τόσο περίπλοκο, για να μπορεί να εκπαιδευτεί και να ελεγχθεί όσο το δυνατόν πιο αποδοτικά. Οπότε, κατά τη διαδικασία της εκπαίδευσης, χρειάστηκε αρκετός πειραματισμός εξετάζοντας διαφορετικές παραμέτρους, έτσι ώστε να καταλήξουμε τελικά σε μια στρατηγική που μας αποφέρει καλά αποτελέσματα.

Δομικό στοιχείο στην όλη διαδικασία, αποτελεί η επιλογή των χαρακτηριστικών διανυσμάτων (features) που αντιπροσωπεύουν τις λέξεις στο συνεχή χώρο, γι΄ αυτό και στο συγκεκριμένο σημείο μελετήθηκαν δύο προσεγγίσεις. Η πρώτη έχει να κάνει με τη μείωση των διαστάσεων των χαρακτηριστικών διανυσμάτων μέσω της τεχνικής SVD, ενώ η δεύτερη με την επιλογή των features που θεωρούμε ότι θα μας βοηθήσουν περισσότερο στη διάκριση του συναισθήματος και κατ΄ επέκταση στην ταξινόμηση. Προέκυψαν ενδιαφέροντα αποτελέσματα τα οποία αναλύουμε στη συνέχεια της εργασίας.

1.2 Συνεισφορά της Διπλωματικής

Αυτή η διπλωματική εργασία αποτυπώνει την έρευνα που έγινε πάνω στα συνεχή γλωσσικά μοντέλα, όσο αφορά την πτυχή της ανάλυσης των συναισθημάτων. Πιο συγκεκριμένα, προσπαθούμε να διαπιστώσουμε τις επιπτώσεις που έχει η χρησιμοποίηση του συνεχή χώρου στη μοντελοποίηση της γλώσσας για τον εν λόγω τομέα, καθώς επίσης και το αν μπορούμε να επωφεληθούμε από τις ιδιότητες που διέπουν το χώρο αυτό, για να πετύχουμε κάποια καλύτερα αποτελέσματα σε σχέση με αυτά που έχουν παρουσιαστεί κατά το παρελθόν και προέρχονται από μοντέλα που βασίζονται στο διακριτό χώρο. Ακόμη, δείχνουμε ότι η τεχνική μείωσης διαστάσεων SVD που συρρικνώνει και τροποποιεί τα διανύσματα των λέξεων δεν αποφέρει πολύ καλά αποτελέσματα (όπως σε άλλους συγγενείς τομείς), εξαιτίας της ιδιαιτερότητας και της «ευαισθησίας» που αναμφίβολα διέπει την κατανόηση του συναισθήματος από έναν υπολογιστή. Η διαδικασία που απέφερε γενικά καλύτερα αποτελέσματα ήταν μέσω feature selection μεθόδων, με την Categorical Probability Proportion Difference να πετυχαίνει τελικά τα υψηλότερα ποσοστά απόδοσης φτάνοντας το 82.7% (accuracy). Το ποσοστό αυτό απέχει αλλά δεν είναι μακριά από αντίστοιχες δουλειές που υπάρχουν στη βιβλιογραφία και χρησιμοποιούν διακριτά μοντέλα. Κάποια καλά αποτελέσματα προέκυψαν και μέσω ορισμένων μετασχηματισμών που δοκιμάστηκαν πάνω στα γνήσια-ακατέργαστα δεδομένα χωρίς μείωση των διαστάσεων, αλλά έτσι απαιτείται πολύ μεγαλύτερος χρόνος εκτέλεσης για την εκπαίδευση σε σχέση με την προηγούμενη προσέγγιση.

1.3 Περίγραμμα της Διπλωματικής

Στο Κεφάλαιο 2 αναφέρουμε κάποιους απαραίτητους ορισμούς σχετικά με το τι αποτελεί συναίσθημα για έναν υπολογιστή και πως αυτός το αντιλαμβάνεται, ενώ στη συνέχεια υπάρχει μια ανασκόπηση της δουλείας που έχει γίνει, συνοδευόμενη πάντα από τις αντίστοιχες αποδόσεις που έχουν επιτευχθεί. Το Κεφάλαιο 3 που ακολουθεί, παραθέτει το λεξικό που χρησιμοποιήθηκε, τους αλγορίθμους stemming που δοκιμάστηκαν για τη συρρίκνωσή του (Lancaster, Porter, WordNet), όπως και τους μετασχηματισμούς που εφαρμόστηκαν πάνω στα αρχικά δεδομένα. Επίσης, παραθέτουμε την ανάλογη θεωρία για την επε-

ξήγηση της μετάβασης των λέξεων από το διακριτό στο συνεχή χώρο, καθώς και τους τρόπους με τους οποίους έγινε η επιλογή των χρήσιμων χαρακτηριστικών διανυσμάτων για την ταξινόμηση. Στο Κεφάλαιο 4, αναλύουμε θεωρητικά τα Gaussian Mixture Models, μαζί με τις δυνατότητές που μας παρέχουν ως προς την επιλογή των παραμέτρων, καθώς και ποιες από αυτές τελικά επιλέχθηκαν και εξετάστηκαν. Μέσα από το Κεφάλαιο 5 επεξηγούνται οι αλγόριθμοι σύμφωνα με τους οποίους ταξινομήθηκαν οι κριτικές, αλλά και οι μετρικές απόδοσης που χρησιμοποιήθηκαν για την εκτίμηση της απόδοσης του μοντέλου. Έχοντας ήδη παραθέσει λοιπόν όλη τη σχετική θεωρία που χρειαζόμαστε για την εκπαίδευση και τον έλεγχο του μοντέλου, στο Κεφάλαιο 6 βρίσκονται τα αποτελέσματα που επιτύχαμε για τα δεδομένα μας. Στην αρχή του ίδιου κεφαλαίου, γίνεται λόγος τόσο για το dataset που χρησιμοποιήσαμε, το οποίο είναι αυτό που δημιούργησαν οι Pang/Lee [5], όσο και για το σημείο εκκίνησής μας (baseline). Τέλος, καταλήγουμε με τα συμπεράσματά μας, συγκρίνουμε τα αποτελέσματα με αυτά προηγούμενων μελετών και προτείνουμε μελλοντική περαιτέρω δουλειά που μπορεί να γίνει στο Κεφάλαιο 7.

Κεφάλαιο 2

Ιστορικό Υπόβαθρο και Σχετική Δουλειά

2.1 Ορισμοί

2.1.1 Τι είναι συναίσθημα;

Η πληροφορία σε ένα κείμενο μπορεί να χωριστεί σε δύο κυρίως τομείς: γεγονότα και απόψεις. Τα γεγονότα επικεντρώνονται στην αντικειμενική μετάδοση των δεδομένων, ενώ οι απόψεις εκφράζουν το συναίσθημα των συντακτών. Σύμφωνα με τον Bing Liu [31, Chapter 28], μία άποψη αποτελεί για τη μηχανή ένα αντικείμενο που απαρτίζεται από τα παρακάτω πέντε διαφορετικά στοιχεία (o_j, f_{jk}, so_{ijkl}, h_i, t_l):

- o_j: το αντικείμενο-στόχος της ερώτησης (μία οντότητα που μπορεί να είναι ένα προϊόν, άνθρωπος, γεγονός, οργανισμός, υπηρεσία ή θέμα)
- f_{jk}: το χαρακτηριστικό του αντικειμένου o_j (attribute)
- so_{ijkl}: η συναισθηματική αξία της γνώμης του συγγραφέα h_i στο χαρακτηριστικό f_{jk} του αντικειμένου o_j τη χρονική στιγμή t_l. Η so_{ijkl} είναι +ve (positive) ή -ve (negative)
- h_i: το άτομο ή ο οργανισμός που εκφράζει τη γνώμη (ο συγγραφέας)
- t_l: η χρονική στιγμή που εκφράστηκε η άποψη

Αυτά τα πέντε στοιχεία πρέπει να προσδιοριστούν από τη μηχανή για να κατανοήσει πλήρως μία γνώμη, με το so_{ijkl} να είναι το στοιχείο εκείνο που απαιτεί sentiment analysis για να εξαχθεί. Η εύρεση και των πέντε αυτών στοιχείων, σημαίνει ότι κάθε είδους ανάλυση είναι δυνατή.

Σκοπός είναι η ταξινόμηση του εγγράφου με βάση το γενικό συναίσθημα που εκφράζει ο συγγραφέας στις κλάσεις positive, negative (ή και περισσότερες).

Το μοντέλο $(o_j, f_{jk}, so_{ijkl}, h_i, t_l)$ υποθέτει ότι κάθε έγγραφο εστιάζει σε ένα και μοναδικό αντικείμενο και περιέχει απόψεις από ένα μόνο συγγραφέα, όπως και συμβαίνει στις κριτικές ταινιών που θα εξετάσουμε.

Υπάρχουν δύο κυρίως μοντέλα από έγγραφα με απόψεις: αυτά που εκφράζουν direct opinion, όπου έχουμε την πεντάδα $(o_j, f_{jk}, so_{ijkl}, h_i, t_l)$ έτσι όπως περιγράφτηκε παραπάνω και αυτά που περιέχουν comparative opinions, τα οποία εκφράζουν μια σχέση ομοιοτήτων και διαφορών μεταξύ δύο ή περισσοτέρων αντικειμένων (συνήθως χρησιμοποιείται ο συγκριτικός ή ο υπερθετικός βαθμός ενός επιθέτου ή επιρρήματος). Για τα τελευταία απαιτείται ξεχωριστή ανάλυση με την οποία δε θα ασχοληθούμε.

2.1.2 Ταξινόμηση Συναισθήματος και Υποκειμενικότητας

Η ταξινόμηση του συναισθήματος (sentiment classification), είναι η περιοχή που έχει ερευνηθεί πιο πολύ. Ο στόχος της είναι να ταξινομηθεί ένα έγγραφο (μία κριτική ταινιών στη συγκεκριμένη περίπτωση) που εκφράζει μια συγκεκριμένη άποψη, σα να εκφράζει μία θετική ή αρνητική γνώμη. Η διαδικασία αυτή, με την οποία ασχοληθήκαμε και εμείς, είναι ευρέως γνωστή και ως ταξινόμηση σε επίπεδο εγγράφου (document-level sentiment classification), καθώς θεωρεί ως κύρια μονάδα πληροφορίας ολόκληρο το κείμενο. Σημαντική λεπτομέρεια είναι πως αυτός ο τρόπος ταξινόμησης θεωρεί γνωστό εκ των προτέρων πως το έγγραφο εκφράζει πάντα μία γνώμη.

Φυσικά όμως, η ταξινόμηση συναισθήματος μπορεί να γίνει μεμονωμένα σε προτάσεις, ή ακόμη και σε φράσεις. Ωστόσο, σε αυτή την περίπτωση δε θεωρείται σίγουρο πως κάθε πρόταση για παράδειγμα, εκφράζει σίγουρα μια γνώμη. Γι΄ αυτό, πρέπει να προηγηθεί η ταξινόμηση της υποκειμενικότητας (subjectivity classification), μέσω της οποίας καθορίζεται εάν η συγκεκριμένη πρόταση εκφράζει άποψη ή όχι (εάν είναι δηλαδή υποκειμενική ή αντικειμενική). Στη συνέχεια, υπάρχει κατηγοριοποίηση μόνο των υποκειμενικών προτάσεων σε θετικές και αρνητικές, πράξη που ονομάζεται ταξινόμηση σε επίπεδο πρότασης (sentence-level sentiment classification). Το δύσκολο σημείο είναι βέβαια πως οι υποκειμενικές προτάσεις δεν εκφράζουν πάντα θετικές ή αρνητικές γνώμες, και οι αντικειμενικές προτάσεις είναι δυνατό να υπαινίσσονται θετική ή αρνητική άποψη (π.χ. I understood the end from the beginning - υπαινίσσεται αρνητική γνώμη).

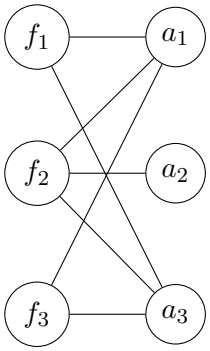
2.2 Τεχνικές Ανάλυσης Συναισθημάτων

Εξετάζουμε το πρόβλημα της ανάλυσης συναισθημάτων από δύο πλευρές: το χαρακτηριστικό γνώρισμα (feature) και τον αλγόριθμο. Ορίζουμε αυτές τις ποσότητες όπως και ο Xu [30] ως εξής:

Ορισμός 1: Sentiment analysis feature είναι μία μετρήσιμη ιδιότητα των κειμένων έτοιμων για ανάλυση συναισθήματος.

Ορισμός 2: Sentiment analysis algorithm είναι μία μέθοδος που βασίζεται πάνω σε sentiment analysis features για να αποφασίσει την πολικότητα των εγγράφων.

Οι αλγόριθμοι χτίζονται με βάση τα features, δηλαδή χρησιμοποιούν ένα ή περισσότερα features για να εξάγουν αποτέλεσμα, ενώ ένα feature μπορεί να χρησιμοποιηθεί από έναν ή περισσότερους αλγορίθμους. Τα παραπάνω αναπαριστώνται συγκεντρωτικά στο σχήμα που ακολουθεί:



Σχήμα 2.1: Σχέση ανάμεσα στα features (f_i) και τον αλγόριθμο (a_j)

Τα features χωρίζονται σε δύο τύπους: sentiment-lexicon features και non-sentiment-lexicon features. Τα πρώτα εξασφαλίζονται από ένα sentiment lexicon, το οποίο βάζει μία ετικέτα σε κάθε όρο του λεξικού με πληροφορία πολικότητας (polarity information). Η πληροφορία αυτή είναι συνήθως ένας αριθμός σε μία κλίμακα, ώστε να μετρηθεί πόσο θετική ή αρνητική είναι η λέξη, με το θετικό σκορ να υποδεικνύει θετικό συναίσθημα και το αρνητικό σκορ αρνητικό συναίσθημα. Από την άλλη, non-sentiment-lexicon feature αποτελεί κάθε feature εκτός από τα sentiment-lexicon features, όπως είναι η παρουσία όρου (term presence), η συχνότητα όρου (term frequency) και η συντακτική πληροφορία (syntax information).

Από πλευράς αλγορίθμων τώρα, υπάρχουν πάλι κατ΄ αναλογία δύο κύριες προσεγγίσεις. Από τη μία βρίσκονται οι sentiment-lexicon-based αλγόριθμοι, που αποτελούν unsupervised sentiment analysis προσεγγίσεις, όπου ανακαλύπτουμε μία καλή εσωτερική αναπαράσταση της εισόδου, και από την άλλη έχουμε τις machine-learning-based approaches, που χρησιμοποιούν κυρίως supervised learning τεχνικές, μέσω των οποίων προβλέπουμε την έξοδο γνωρίζοντας τη φύση των δεδομένων εισόδου (αν είναι δηλαδή θετικά ή αρνητικά).

2.3 Ιστορική Αναδρομή

Η συστηματική μελέτη του τομέα της ανάλυσης συναισθημάτων πάνω σε κείμενα άρχισε περίπου το 2000 και το ενδιαφέρον παραμένει μεγάλο μέχρι και σήμερα. Σχεδόν όλη η δουλειά που έχει παρουσιαστεί, έχει αναπτυχθεί πάνω σε

κριτικές από ταινίες, αλλά υπάρχει και έρευνα πάνω σε άλλες θεματικές περιοχές όπως debates, οικονομικές ειδήσεις, blogs, διάφορα προϊόντα από το Amazon και κοινωνικά δίκτυα (π.χ. twitter). Για την ταξινόμηση του συναισθήματος, όπως προαναφέρθηκε, υπάρχουν δύο κυρίως δρόμοι-επιλογές έρευνας. Οι λεξιλογικές προσεγγίσεις (sentiment-lexicon-based), έχουν ως κύριο μέλημά τους το χτίσιμο ενός επιτυχημένου λεξικού, ενώ για τις προσεγγίσεις που βασίζονται σε machine learning κυρίαρχο ρόλο παίζουν τα features.

Μια λεξιλογική προσέγγιση, χρησιμοποιεί ένα λεξικό, όπου κάθε λέξη έχει αντιστοιχιστεί με μια ενυπάρχουσα θετική ή αρνητική έννοια. Για παράδειγμα, οι λέξεις «adore», «excellent» ή «happy», είναι λέξεις με θετική σημασία και τους αποδίδεται το orientation score +1, ενώ στον αντίποδα έρχονται λέξεις όπως «hate» ή «dislike» με orientation score -1. Εναλλακτικά, μπορεί να χρησιμοποιηθεί μία τιμή στο [0,1]. Έτσι, για κάθε λέξη που συναντάται στο κείμενο, αν βρεθεί στο λεξικό και έχει αρνητική τιμή για παράδειγμα, η τιμή αυτή αφαιρείται από το συνολικό σκορ (total polarity score), υποδεικνύοντας έναν γενικό βαθμό πολικότητας της κριτικής. Αν το τελικό σκορ του κειμένου είναι θετικό, τότε το κείμενο ταξινομείται ως θετικό, αλλιώς το κείμενο θεωρείται αρνητικό. Αν και είναι αρκετά απλοϊκό μοντέλο ως σκέψη, έχει γίνει εκτεταμένη έρευνα πάνω στο συγκεκριμένο θέμα και αρκετές παραλλαγές αυτής της προσέγγισης έχουν προταθεί για καλύτερα αποτελέσματα [1, 2, 4].

Προκύπτουν ωστόσο δύο κυρίως προβλήματα με αυτή τη μεθοδολογία. Το πρώτο είναι ότι αυτή η ανάθεση θετικών ή αρνητικών συναισθημάτων αξιολογεί κάθε λέξη, χωρίς να λαμβάνει υπ' όψιν τις λέξεις που την περιβάλλουν, κάτι που πολλές φορές αλλάζει τη συναισθηματική αξία. Επιπλέον, το λεξικό είναι πολύ περιορισμένο ως προς τον αριθμό τον λέξεων που προσδίδουν θετική ή αρνητική σημασία σε μία έκφραση. Το δεύτερο πρόβλημα έχει να κάνει με το ότι ο κάθε ερευνητής μπορεί να προσδίδει διαφορετικό βαθμό θετικότητας/αρνητικότητας για την ίδια λέξη (ιδιαιτέρως στην περίπτωση των διφορούμενων εκφράσεων), και έτσι δεν υπάρχει μια κοινά αποδεκτή λύση.

Η περισσότερη δουλειά πάντως στον τομέα της ανάλυσης των συναισθημάτων, θα λέγαμε ότι έχει επικεντρωθεί σε supervised machine learning τεχνικές. Με αυτή την προσέγγιση, καθοριστικό ρόλο για την επιτυχία του βαθμού της ταξινόμησης παίζει η επιλογή των features. Η προσπάθεια λοιπόν για την ταξινόμηση της πολικότητας κριτικών από ταινίες με βάση την IMDb χρησιμοποιώντας supervised τεχνικές, ξεκίνησε από τους Pang και Lee [3]. Σκοπός τους ήταν να καθορίσουν εάν η ταξινόμηση του συναισθήματος μπορεί να θεωρηθεί σαν μια ειδική περίπτωση της ταξινόμησης του θέματος (topic-based classification) θεωρώντας δύο θέματα: θετικό και αρνητικό. Έτσι, για την ταξινόμηση χρησιμοποίησαν Naive Bayes (NB), Maximum Entropy (ME) και Support Vector Machines (SVMs), ταξινομητές δηλαδή που είχαν καλή απόδοση στην κατηγοριοποίηση θεμάτων, ενώ ως features χρησιμοποιήθηκαν unigrams, bigrams και βάρη όπως η μέτρηση της συχνότητας, ή η εμφάνιση των λέξεων. Κατέληξαν στο συμπέρασμα ότι η ταξινόμηση του συναισθήματος είναι δυσκολότερη σε σχέση με την ταξινόμηση με βάση το θέμα και ότι χρησι-

μοποιώντας έναν SVM classifier με binary unigram-based features παράγονται τα καλύτερα δυνατά αποτελέσματα (accuracy 82.9%).

Οι ίδιοι, συνέχισαν δύο χρόνια αργότερα με μια ακόμη καινοτομία, σύμφωνα με την οποία γινόταν ανίχνευση και αφαίρεση των αντικειμενικών μερών από τα κείμενα (feature selection) και στη συνέχεια για την ταξινόμηση της πολικότητας χρησιμοποιούνταν τα υπόλοιπα μέρη που είχαν απομείνει. Το αποτέλεσμα ήταν μία σημαντική αύξηση της απόδοσης για τον Naive Bayes classifier, αλλά η βελτίωση σε σχέση με τον SVM classifier ήταν πολύ μικρή. Οι τελευταίες έρευνες τώρα, έχουν επικεντρωθεί στην αναπαράσταση των κειμένων χρησιμοποιώντας πιο σύνθετα features, όπως δομική ή συντακτική πληροφορία (Wilson et. al [9]), έμμεσους συντακτικούς δείκτες (Greene και Resnik [25]), στυλιστική και συντακτική επιλογή features (Abbasi et. al [22]), «annotator rationales» (Zaidan et. al [17]) και άλλα. Εκτός όμως από την εύρεση του συναισθήματος για ένα συγκεκριμένο θέμα και μόνο, υπάρχουν κάποιες μελέτες, όπως αυτή των Blitzer et al. [15], που πραγματεύονται προϊόντα από διαφορετικούς τομείς (adaptation), δείχνοντας ότι η τελική απόδοση που επιτυγχάνεται διαφέρει σημαντικά ανάμεσά τους. Μερικές από τις καλύτερες δουλειές που έχουν δημοσιευτεί, παρουσιάζονται στον Πίνακα 2.1. Συμπερασματικά, η απόδοση (accuracy) της ταξινόμησης, ποικίλει μεταξύ 62% και 91.7%, ανάλογα κυρίως με τα features που χρησιμοποιούνται, και όχι τόσο βάση του αλγορίθμου, που στις περισσότερες περιπτώσεις είναι ένας (ή μία παραλλαγή) εκ των NB, ME και SVM. Το υψηλότερο ποσοστό που έχει επιτευχθεί μέχρι σήμερα (91.7%), προέρχεται από τους Abbasi et al., οι οποίοι συνδυάζουν τόσο information gain όσο και genetic algorithms, σε έναν καινούριο αλγόριθμο που ονόμασαν Entropy Weighted Genetic Algorithm (EWGA).

Τέλος, αξίζει να σημειωθεί, πως σε αντίθεση με όλες τις παραπάνω machine learning τεχνικές που αναφέρθηκαν και απαιτούν να ξέρουμε εκ των προτέρων το «είδος» των δεδομένων εκπαίδευσης (δηλαδή αν πρόκειται για θετική ή αρνητική κριτική), οι Lin και He [26], πρότειναν ένα unsupervised πιθανοτικό μοντέλο που βασίζεται σε Latent Dirichlet Allocation (LDA). Η απόδοση της προσέγγισής τους ήταν τελικώς μόνο κατά λίγο χαμηλότερη σε σχέση με αυτή των Pang and Lee [5], η οποία ήταν πλήρως supervised.

Authors	Classifier	Features	Domain	Accuracy
Turney [2]	Pointwise Mutual Information	bigrams	movies, cars, banks	66-84%
Pang & Lee [3]	NB, ME, SVMs	unigrams, bigrams, feature presence or frequence, negation	IMDb[1]	82.9%
Pang & Lee [5]	NB, SVMs	sentence level subjectivity based on minimum cuts	IMDb	87.2%
Mullen & Collier [7]	Hybrid SVM	Turney values, Osgood values, lemma models	IMDb[1], record reviews	87%
Bai et al. [8]	two-stage Markov Blanket Classifier	dependence among words, minimal vocabulary	IMDb[1]	87.5%
Whitelaw et al. [10]	SMO with linear kernel	appraisal groups	movie reviews	90.2%
Kennedy & Inkpen [13]	SVMs, term counting, combination	term frequencies	IMDb	86.2%
Annett & Kondrak [19]	WordNet and SVM	num of pos/neg adj/adv, min distance from pivot words in WordNet	movie reviews, blog posts	65.4-77.5%
Zhou & Chaovalit [21]	ontology-supported polarity mining	n-grams, words, word senses	movie reviews	72.2%
Abbasi et al. [22]	Genetic Algorithm(GA), Information Gain(IG), GA+IG	stylistic and syntactic features	movie reviews, web forum postings	91.7%

Πίνακας 2.1: Αποτελέσματα με μερικά από τα καλύτερα ποσοστά απόδοσης που έχουν αναφερθεί στη βιβλιογραφία για sentiment classification σε επίπεδο κειμένου

[1] Αφορά προγενέστερη έκδοση του dataset που χρησιμοποιούμε με 700 θετικές και 700 αρνητικές κριτικές.

Κεφάλαιο 3

Μοντελοποίηση Γλώσσας στο Συνεχή Χώρο

3.1 Αντιστοίχιση Λέξεων με Διανύσματα στο Συνεχή Χώρο

Σκοπός της μετάβασης από το διακριτό στο συνεχή χώρο, είναι η προβολή κάθε λέξης σε αυτό το νέο συνεχή χώρο μέσω κάποιου διανύσματος (Vector Space Model.) Πρώτο βασικό σημείο για τη μετάβαση αυτή, είναι η δημιουργία ενός λεξικού με όρους που υπάρχουν στα (training) κείμενα.

Για να καθορίσουμε τις λέξεις που αντιπροσωπεύουν όσο το δυνατόν καλύτερα το περιεχόμενο των κειμένων μας για το σκοπό που μας ενδιαφέρει (εύρεση συναισθήματος), βασιστήκαμε στο πως διαβάζουν στην πραγματικότητα οι άνθρωποι τις κριτικές που τους ενδιαφέρουν. Το κοινό χαρακτηριστικό είναι πως ψάχνουν με μια γρήγορη πρώτη ματιά για συγκεκριμένες λέξεις που δηλώνουν το συναίσθημα του συγγραφέα, έτσι ώστε να καταλάβουν αν τελικά του άρεσε και την προτείνει ή όχι. Λαμβάνοντας λοιπόν υπ΄ όψιν αυτό το χαρακτηριστικό, καθώς και τη μεγάλη ποσότητα των λέξεων που υπάρχει στα δεδομένα, κάτι το οποίο συνεπάγεται και την αύξηση της πολυπλοκότητας καθώς πηγαίνουμε σε έναν μεγαλύτερης διάστασης χώρο, προσπαθήσαμε να αναπαραστήσουμε τελικά μόνο τις λέξεις που εκφράζουν κάποιο θετικό ή αρνητικό συναίσθημα. Έτσι, χρησιμοποιήσαμε ένα λεξικό συναισθημάτων που προτάθηκε αρχικά από τους Hu και Liu [6]. Το λεξικό αυτό, περιέχει γύρω στις 6800 λέξεις, εκ των οποίων περίπου οι 2000 θεωρείται πως εκφράζουν θετικό συναίσθημα ενώ οι υπόλοιπες αρνητικό. Κάποιες χρήσιμες ιδιότητες του συγκεκριμένου λεξικού που συνέβαλαν στο να το επιλέξουμε, είναι ότι περιλαμβάνει λέξεις με συχνά ορθογραφικά λάθη, μορφολογικές παραλλαγές, καθώς και αργκό, χαρακτηριστικά δηλαδή που εξυπηρετούν τη δική μας περίπτωση όπου έχουμε κριτικές από ταινίες.

Στη συνέχεια αυτού του κεφαλαίου, επεξηγούμε τρόπους έτσι ώστε να μειώσουμε περαιτέρω το μέγεθος του λεξικού (συνεπώς και των διαστάσεων), κα-

θώς και ποια είναι τελικά τα χαρακτηριστικά διανύσματα (feature vectors) που επιλέγουμε για να αντιπροσωπεύσουν τις λέξεις μας.

3.2 Stemming and Lemmatization

3.2.1 Γενικά

Stemming είναι μία διαδικασία κατά την οποία οι λέξεις συρρικνώνονται στη ρίζα τους. Κατ΄ αυτόν τον τρόπο σχετικές μεταξύ τους λέξεις αντιστοιχίζονται στην ίδια ρίζα, ακόμη και αν αυτή η ρίζα δεν είναι έγκυρη λεξιλογικά. Αντίστοιχα lemmatization, είναι μια παρόμοια διαδικασία, μόνο που αυτή τη φορά οι λέξεις δε μειώνονται στη ρίζα τους, αλλά στο αντίστοιχο λήμμα τους, πράγμα που σημαίνει ότι ανάγονται σε μια κανονική μορφή της λέξης. Οι παραπάνω ενέργειες έχουν σαν αποτέλεσμα να μειώνεται περαιτέρω το μέγεθος του λεξικού, κάτι που έχει άμεση επίδραση και στα αποτελέσματα. Έχουν αναπτυχθεί διάφοροι αλγόριθμοι για stemming. Στη συνέχεια αναλύουμε αυτούς που είναι περισσότερο δημοφιλείς στον τομέα του συναισθήματος και εξετάστηκαν στην παρούσα διπλωματική εργασία.

3.2.2 Porter Stemmer

Ο Porter stemmer είναι ένας από τους πιο παλιούς και γνωστούς αλγορίθμους για stemming. Δουλεύει με το να βρίσκει ευριστικά τις καταλήξεις των λέξεων και να τις αφαιρεί, με κάποια κανονικοποίηση στα τελειώματα των λέξεων.

Ο stemmer αυτός, συχνά καταστρέφει τις διακρίσεις μεταξύ των συναισθημάτων με το να αντιστοιχίζει, λέξεις με διαφορετικό συναίσθημα στην ίδια ρίζα, κάτι που φαίνεται και στον Πίνακα 3.1. Έτσι, από τις 6786 λέξεις του λεξικού, μετά την εφαρμογή του αλγορίθμου απομένουν οι 4531.

Positive	Negative	Stemmed
indulgence	indulge	indulg
awed	aweful	awe
contentment	contention	content
grateful	grating	grate
impassioned	impasse	impass

Πίνακας 3.1: Porter stemming. 5 τυχαία στιγμιότυπα στα οποία η διάκριση θετικών/αρνητικών λέξεων από το λεξικό καταστρέφεται από τον αλγόριθμο

3.2.3 Lancaster Stemmer

Ο Lancaster stemmer είναι ένας άλλος επίσης ευρέως χρησιμοποιούμενος stemmer. Ωστόσο, για sentiment analysis είναι περισσότερο προβληματικός από τον προηγούμενο, καθώς είναι πιο «επιθετικός» και έτσι υποτιμά ακόμη περισσότερες λέξεις που φέρουν το ίδιο συναίσθημα. Είναι χαρακτηριστικό πως το λεξικό μας αυτή τη φορά μειώθηκε από τις 6786 λέξεις σε μόλις 3789.

Positive	Negative	Stemmed
audible	audacious	aud
comely	commiserate	com
compliment	complicated	comply
famous	famished	fam
flourish	floored	flo
passion	passe	pass
passionate	passe	pass
savings	savage	sav
simplify	simplistic	simpl
suffice	sufferer	suff

Πίνακας 3.2: Lancaster stemming. 10 τυχαία στιγμιότυπα στα οποία η διάκριση θετικών/αρνητικών λέξεων από το λεξικό καταστρέφεται από τον αλγόριθμο

3.2.4 WordNet Lemmatizer

Ο WordNet Lemmatizer χρησιμοποιεί τη βάση δεδομένων WordNet για να ψάχνει τα λήμματα των λέξεων. Το WordNet είναι μία μεγάλη ηλεκτρονική λεξιλογική βάση δεδομένων στα αγγλικά. Ουσιαστικά, ρήματα, επίθετα και επιρρήματα συγκεντρώνονται μαζί σε ομάδες όμοιων νοητικά συνωνύμων (synsets), καθεμία από τις οποίες εκφράζει ένα διακριτό νόημα και έχει μια διαφορετική έννοια.

Το κύριο μειονέκτημα αυτής της μεθόδου που αφορά την ανάλυση συναισθημάτων, είναι ότι σε ορισμένες περιπτώσεις συγχωνεύει το θετικό, το συγκριτικό και τον υπερθετικό βαθμό του επιθέτου. Η συρρίκνωση του λεξικού εδώ ήταν μόνο κατά 343 λέξεις (από 6786 σε 6443).

3.3 Πίνακας «Κειμένου-Λέξης»

Για την αναπαράσταση μιας μη ταξινομημένης συλλογής από λέξεις, χωρίς να λαμβάνεται υπ' όψιν ούτε η γραμματική ούτε η σειρά των λέξεων, χρησιμοποιείται η τεχνική Bag of Words (BoW) model. Σύμφωνα με αυτή, ένα κείμενο αναπαρίσταται σαν ένα διάνυσμα με μήκος ίσο με το μέγεθος του λεξικού που έχουμε επιλέξει, ενώ σε κάθε θέση του διανύσματος υπάρχει ένας αριθμός που

αντιπροσωπεύει τον αριθμό τον φορών που η συγκεκριμένη λέξη εμφανίζεται μέσα στο κείμενο.

Ο πίνακας «κειμένου-λέξης» (document-term matrix) που χρησιμοποιήθηκε σα βασικό δομικό στοιχείο, ουσιαστικά προεκτείνει την προηγούμενη ιδέα για την αναπαράσταση των λέξεων σε έναν πίνακα που εμπεριέχει όλα τα κείμενα. Έτσι, οι γραμμές του πίνακα αντιστοιχούν σε κείμενα που βρίσκονται στα δεδομένα εκπαίδευσης και οι στήλες αναπαριστούν τους όρους, που στη δική μας περίπτωση είναι οι λέξεις που έχουμε πάρει από το λεξικό που καταδεικνύει το συναίσθημα. Αρχικά, κάθε κελί του πίνακα a_{ij} αντιπροσωπεύει τον αριθμό των φορών που ο αντίστοιχος όρος i εμφανίζεται σε ένα συγκεκριμένο κείμενο j. Όπως είναι προφανές, επειδή ο αριθμός των όρων που χρησιμοποιούμε είναι πολύ μικρότερος από τον αριθμό των όρων που υπάρχουν στη συλλογή των κειμένων μας, ο πίνακας αυτός είναι γενικά πολύ μεγάλος και αραιός (sparse).

Ως ένα παράδειγμα που καταδεικνύει τη χρησιμότητα του εν λόγω πίνακα, ας υποθέσουμε ότι έχουμε τα παρακάτω δύο (σύντομα) κείμενα:

- D1 = "I like that movie."
- D2 = "I hate hate that movie."

Χρησιμοποιώντας για λεξικό το {I:1, like:2, hate:3, movie:4}, που σημαίνει ότι δε λαμβάνουμε υπ' όψιν τον όρο «that», ο document-term πίνακας θα έχει ως εξής:

	I	like	hate	movie
D1	1	1	0	1
D2	1	0	2	1

Πίνακας 3.3: Γενική μορφή ενός document-term πίνακα

Χρησιμοποιώντας λοιπόν αυτή την απεικόνιση, είναι ξεκάθαρο ποια κείμενα περιέχουν ποιους όρους και πόσες φορές εμφανίζεται ο κάθε όρος σε αυτά. Στο παράδειγμα παραπάνω, οι τιμές του πίνακα περιέχουν τις συχνότητες των όρων. Υπάρχουν ωστόσο διάφορα άλλα σχήματα-βάρη που μπορούμε να χρησιμοποιήσουμε, τα οποία τροποποιούν την τιμή κάθε κελιού. Ασχοληθήκαμε με αρκετά από αυτά και τα παρουσιάζουμε στη συνέχεια.

Ανάλογα με το τι εξυπηρετεί καλύτερα την εκάστοτε προσέγγιση που ακολουθεί, κατασκευάζουμε είτε έναν document-term matrix ο οποίος έχει στις γραμμές του όλα τα κείμενα που διαθέτουμε (θετικά και αρνητικά), είτε δύο, ανάλογα με την προέλευση των δεδομένων εκπαίδευσης (ο πρώτος έχει στις γραμμές του τις κριτικές που γνωρίζουμε εκ των προτέρων πως έχουν χαρακτηριστεί ως θετικές και ο δεύτερος τις αρνητικές). Οι στήλες του από την άλλη, μπορούν να περιέχουν είτε μόνο μόνο τις θετικές, είτε μόνο τις αρνητικές, είτε το σύνολο των λέξεων που απαντώνται στο Opinion Lexicon. Οι λέξεις

αυτές βρίσκονται για ευκολία σε διάταξη (πρώτα οι θετικές και μετά οι αρνητικές), ενώ είναι δυνατό να έχουν υποστεί επεξεργασία με κάποιον αλγόριθμο που εκτελεί stemming. Μία από τις πιθανές μορφές που μπορεί να πάρει λοιπόν document-term matrix παρουσιάζεται στον Πίνακα 3.4.

Υποθέτουμε τώρα ότι έχουμε την περίπτωση με τον πίνακα που περιέχει τα θετικά δεδομένα εκπαίδευσης και χρησιμοποιούμε όλες τις λέξεις του λεξικού στην αρχική τους μορφή. Τότε, ο πίνακας αυτός περιέχει 42,294 μη μηδενικές τιμές, που σημαίνει ότι μόνο το 0.69% των στοιχείων του είναι μη μηδενικά. Στη συνέχεια, χρησιμοποιώντας τον WordNet lemmatizer, οι μη μηδενικές τιμές του είναι 42,755 και έτσι αυτή τη φορά το 0.74% των στοιχείων είναι μη μηδενικά. Συνεχίζοντας με την εφαρμογή Porter και Lancaster stemmer, έχουμε 58,804 και 87,048 μη μηδενικές τιμές των κελιών του συγκεκριμένου document-term matrix, δηλαδή το ποσοστό των μη μηδενικών στοιχείων ανέρχεται στο 1.44% και τελικά στο 2.55% αντίστοιχα. Παρατηρούμε λοιπόν, πως υπάρχει μια σταδιακή αύξηση στον αριθμό των στοιχείων που είναι μη μηδενικά (ο πίνακας γίνεται ολοένα και λιγότερο αραιός), καθώς χρησιμοποιούμε κάποιας μορφής stemming. Η αύξηση όμως αυτή, θα δείξουμε ότι δε συνοδεύεται από ανάλογη άνοδο των ποσοστών απόδοσης (το αντίθετο μάλιστα), μιας και η συχνότερη εμφάνιση των λέξεων που παρατηρείται, δεν αντιπροσωπεύει πραγματικά χρήσιμη πληροφορία, παρά μονάχα θόρυβο που δυσχεραίνει τη διάκριση του συναισθήματος. Αντίστοιχα συμπεράσματα προκύπτουν και σύμφωνα με τα αρνητικά δεδομένα εκπαίδευσης.

Πίνακας 3.4: Τυπική μορφή ενός document-term matrix πάνω στα training δεδομένα που χρησιμοποιούνται

	Pos_1	Pos_2	$\cdots$	Pos_{2006}	Neg_1	Neg_2	$\cdots$	Neg_{4780}
Doc_1	1	0	$\cdots$	0	3	1	$\cdots$	0
Doc_2	0	0	$\cdots$	0	0	2	$\cdots$	1
Doc_3	0	1	$\cdots$	5	0	0	$\cdots$	0
Doc_4	0	0	$\cdots$	0	2	1	$\cdots$	0
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
Doc_{899}	0	0	$\cdots$	1	1	0	$\cdots$	0
Doc_{900}	0	5	$\cdots$	0	0	4	$\cdots$	2

Pos_i / Neg_j = Θετική / Αρνητική λέξη σύμφωνα με το λεξικό συναισθημάτων

3.3.1 Μελέτη Μετασχηματισμών Πινάκων «Κειμένου-Λέξης»

Δοκιμάσαμε διάφορες συναρτήσεις στάθμισης που υιοθετήθηκαν από το πεδίο της Ανάκτησης Πληροφορίας (Information Retrieval) για να διαπιστώσουμε εάν και κατά πόσο μπορούν να συνεισφέρουν στην απόδοση της κατηγοριοποίησης των γνήσιων δεδομένων τέτοιο μετασχηματισμοί. Η κεντρική ιδέα εδώ είναι

ότι κάποιοι όροι είναι περισσότερο σημαντικοί από κάποιους άλλους. Για να ενισχύσουμε λοιπόν αυτό το διαχωρισμό, πρέπει να δοθεί ένα μεγαλύτερο βάρος στους όρους που θεωρούμε περισσότερο σημαντικούς. Έτσι διαμορφώνουμε ένα νέο «weighted document-term frequency matrix» εφαρμόζοντας διάφορες τοπικές (local), αλλά και καθολικές (global) συναρτήσεις πάνω στον αρχικό πίνακα, που έχουν σαν αποτέλεσμα την τροποποίηση των περιεχομένων του, αλλά αφήνοντας ίδιο τον αριθμό των διαστάσεων. Οι συναρτήσεις στάθμισης, μετασχηματίζουν κάθε κελί $a_{i,j}$ του πίνακα A, ως το γινόμενο του τοπικού όρου $l_{i,j}$, που περιγράφει τη σχετική συχνότητα του όρου μέσα σε ένα κείμενο, και του καθολικού βάρος, g_i, το οποίο περιγράφει τη σχετική συχνότητα του όρου μέσα σε ολόκληρη τη συλλογή των κειμένων. Για όλους τους μετασχηματισμούς που εφαρμόστηκαν, τα feature vectors που επιλέχθηκαν ήταν οι στήλες του διαμορφωμένου document-term πίνακα.

Οι τοπικές συναρτήσεις στάθμισης που εφαρμόστηκαν είναι οι εξής:

- Binary: $l_{i,j} = 1$, εάν ο όρος υπάρχει στο κείμενο, αλλιώς 0

  Έχει αποδειχθεί και από προηγούμενες μελέτες πως στον τομέα της sentiment analysis τα binary features έχουν γενικά καλά αποτελέσματα καθώς τελικά μεγαλύτερο ρόλο παίζει η παρουσία μιας λέξης στο κείμενο, παρά η συχνότητα εμφάνισής της. Η εμφάνιση μιας λέξης παραπάνω από μία φορές λοιπόν, δεν αποτελεί πραγματικά χρήσιμη πληροφορία.

- Term Frequency (TF): $l_{i,j} = tf_{i,j}$, όπου tf_{ij} είναι η συχνότητα του όρου i στο κείμενο j

  Υπάρχει γενικά η αίσθηση πως ενώ παράγεται μία σύνδεση ανάμεσα στον αριθμό των φορών που εμφανίζεται μία λέξη και το αντίστοιχο κείμενο, η σχέση αυτή εξασθενεί καθώς ο αριθμός μεγαλώνει. Φαίνεται πως κάτι τέτοιο ισχύει και εδώ. Επομένως, καλό θα ήταν να γίνει κάποιο βήμα ώστε να μειωθεί αυτή η συσχέτιση, όπως είναι ο log μετασχηματισμός που εξετάζουμε στη συνέχεια.

- Log: $l_{i,j} = log_2(tf_{i,j} + 1)$

  Ουσιαστικά αντικαθιστούμε τις συχνότητες των όρων $tf_{i,j}$ με $log_2(tf_{i,j}+1)$. Έτσι, τα μηδενικά παραμένουν μηδενικά (αφού $log_2(1) = 0$), και όπου υπάρχει 1 παραμένει επίσης (αφού $log_2(2) = 1$), αλλά οι λέξεις με υψηλότερες συχνότητες καταλήγουν με μικρότερο βάρος. Συνεπώς το log βάρος υποβαθμίζει-εξομαλύνει μόνο την ύπαρξη μεγάλου αριθμού εμφανίσεων ενός συγκεκριμένου όρου σε ένα κείμενο, κάτι που όπως φαίνεται έχει αρκετά καλά αποτελέσματα στη δική μας περίπτωση.

- Augnorm: $l_{i,j} = \frac{\frac{tf_{i,j}}{max_i(tf_{i,j})}+1}{2}$

Ακολουθούν οι καθολικές συναρτήσεις στάθμισης που δοκιμάστηκαν στην παρούσα διπλωματική εργασία:

- Normal: $g_i = \frac{1}{\sqrt{\sum_j tf_{i,j}^2}}$

  Έχει τα χειρότερα αποτελέσματα από όλους τους μετασχηματισμούς που δοκιμάστηκαν. Και αυτό, γιατί αν και γενικά (σε άλλους τομείς) βολεύει όλες οι λέξεις να είναι όσο το δυνατόν ισοσταθμισμένες, στη συγκεκριμένη περίπτωση της διάκρισης του συναισθήματος δεν είναι κάτι επιθυμητό, καθώς ορισμένες λέξεις πρέπει να «προεξέχουν» σε σχέση με κάποιες άλλες για να έχουμε καλύτερα αποτελέσματα.

- GfIdf: $g_i = \frac{gf_i}{df_i}$, όπου gf_i είναι ο συνολικός αριθμός των φορών που ο όρος i εμφανίζεται σε ολόκληρη τη συλλογή και df_i είναι ο αριθμός των εγγράφων στα οποία εμφανίζεται ο όρος i

- Idf: $g_i = log_2 \frac{n}{1+df_i}$

  Βασίζεται στον αριθμό των κειμένων στα οποία εμφανίζεται μια συγκεκριμένη λέξη. Είναι ίσως από τις πιο δημοφιλείς συναρτήσεις στάθμισης, καθώς περιορίζει τη σπουδαιότητα των λέξεων με το να κατανέμει λιγότερο βάρος σε λέξεις που απαντώνται συχνότερα στο σώμα του κειμένου και με το να ενισχύει τις λέξεις που απαντώνται σπανιότερα. Το idf λοιπόν, είναι σχεδιασμένο έτσι ώστε να βρίσκει την τοπική ομοιότητα η οποία υποδεικνύεται από τις πιο σπάνιες λέξεις στη συλλογή. Αυτό όμως οδηγεί στο να υποτιμά τη σημασία των λέξεων με συναίσθημα και έτσι τελικά να μην παράγει καλά αποτελέσματα για sentiment analysis.

- Entropy: $g_i = 1 + \sum_j \frac{p_{i,j} log p_{i,j}}{log n}$, όπου $p_{i,j} = \frac{tf_{i,j}}{gf_i}$

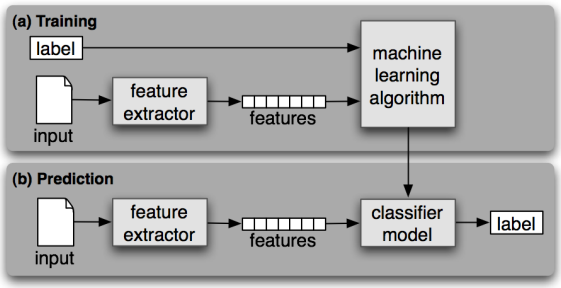
  Η εντροπία, είναι ένας τρόπος να βρούμε πόση (διδακτική) πληροφορία περιέχει κάθε λέξη. Για παράδειγμα, μια κριτική που περιέχει τη λέξη «wonderfull», θεωρούμε ότι περιέχει περισσότερη χρήσιμη πληροφορία (είναι περισσότερο θετική στη συγκεκριμένη περίπτωση) από μία άλλη που περιέχει τη λέξη «then». Επειδή όμως εμείς χρησιμοποιούμε μόνο τις λέξεις του λεξικού στον document-term πίνακα, ουσιαστικά είναι σα να προσδίδουμε ένα διαφορετικό βαθμό πολικότητας σε κάθε μία από αυτές. Μία λέξη με υψηλή εντροπία τείνει να είναι ομοιόμορφα κατανεμημένη μέσα στα κείμενα, ενώ μια λέξη με χαμηλή εντροπία εμφανίζεται σε λίγα από αυτά. Η χρησιμοποίηση αυτού του μετασχηματισμού, δεν απέφερε τα αναμενόμενα ποσοστά απόδοσης.

Σημαντική παρατήρηση για τους μετασχηματισμούς αυτούς, είναι το γεγονός ότι πάντα από κάτω ενυπάρχει η ίδια ιδέα, που περιγράφεται όμως κάθε φορά χρησιμοποιώντας διαφορετικούς όρους. Αυτό σημαίνει, ότι το περιεχόμενο κάθε κελιού δεν είναι παρά ένας σταθμισμένος μέσος όρος που αντιπροσωπεύει τις διαφορετικές εκφάνσεις που μπορεί να πάρει κάθε τιμή του.

3.4 Τεχνικές Μείωσης Διαστάσεων

3.4.1 Γενικά

Επειδή τα διανύσματα των λέξεων είναι πολύ μεγάλα και αραιά, απαιτείται μείωση των διαστάσεων για να επιταχύνουμε τη διαδικασία εκπαίδευσης, αλλά και να μειώσουμε το υπολογιστικό κόστος και τη μνήμη που χρειάζεται για την επεξεργασία των δεδομένων. Για την κατασκευή των χαρακτηριστικών διανυσμάτων για τις λέξεις λοιπόν, χρησιμοποιήθηκαν δύο προσεγγίσεις. Η πρώτη, χρησιμοποιεί την τεχνική Singular Value Decomposition, έτσι ώστε να μειωθούν οι διαστάσεις του κατάλληλου document-term πίνακα και στη συνέχεια εξάγει τα χρήσιμα διανύσματα. Η δεύτερη από την άλλη χρησιμοποιεί μεθόδους feature selection για την επιλογή εκείνων των word vectors που είναι περισσότερο ικανά να συμβάλουν στο διαχωρισμό των κλάσεων. Σε κάθε περίπτωση επικεντρωθήκαμε στα unigrams. Στο παρακάτω σχήμα φαίνεται το σημείο στο οποίο το πρόγραμμα καλείται να εξάγει τα χαρακτηριστικά διανύσματα των λέξεων που θα βοηθήσουν στην ταξινόμηση:



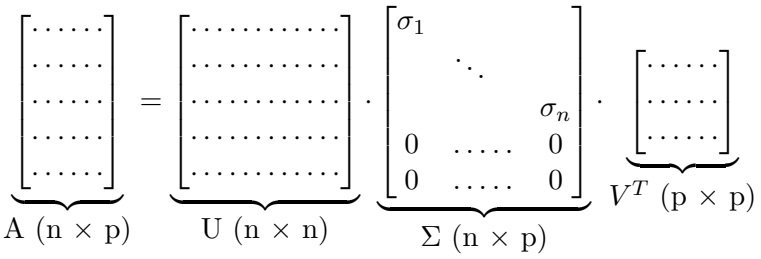
Σχήμα 3.1: Ροή ενός προγράμματος supervised ταξινόμησης

3.4.2 Singular Value Decomposition (SVD)

Το SVD είναι μία μέθοδος που μπορεί να μειώσει τις διαστάσεις ενός πίνακα, με το να αφαιρεί τα πλεονάζοντα δεδομένα που είναι γραμμικά εξαρτημένα από τη σκοπιά της Γραμμικής Άλγεβρας. Έτσι, η τεχνική αυτή παραγοντοποιεί γενικά έναν πίνακα $A \in \mathbb{R}^{m \times n}$ διαστάσεων $n \times p$ σε ένα γινόμενο τριών επιμέρους πινάκων: έναν πίνακα U διαστάσεων $n \times n$, έναν πίνακα Σ διαστάσεων $n \times p$ και έναν πίνακα V^T διαστάσεων $p \times p$. Το θεώρημα λοιπόν έχει ως εξής:

$$A_{n \times p} = U_{n \times n} * \Sigma_{n \times p} * V^T_{p \times p} \tag{3.1}$$

Οι U και V^T είναι ορθογώνιοι πίνακες (δηλαδή ισχύει $U^T \cdot U = I_{n \times n}$ και $V^T \cdot V = I_{p \times p}$), με τις n στήλες του U και τις p στήλες του V (ή τις γραμμές του V^T), να ονομάζονται left και right singular vectors του A και να αποτελούν τα ιδιοδιανύσματα του $A \cdot A^T$ και του $A^T \cdot A$ αντίστοιχα. Ο Σ από την άλλη,



Σχήμα 3.2: Σχηματική αναπαράσταση της τεχνικής SVD

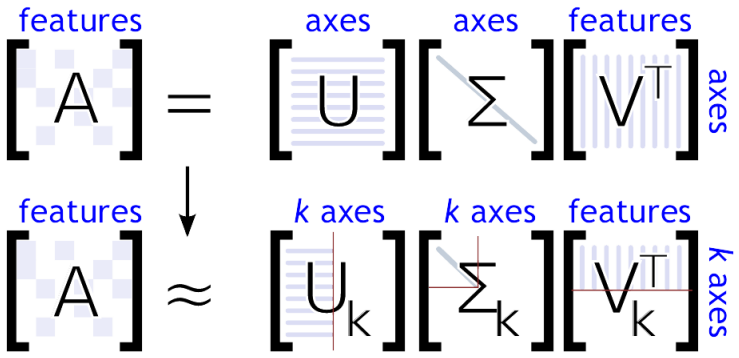
είναι ένας τετραγωνικός διαγώνιος πίνακας με μη αρνητικές πραγματικές τιμές ταξινομημένες σε φθίνουσα διάταξη ($\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$). Οι (διαγώνιες) τιμές του $\sigma_1, \sigma_2, \ldots, \sigma_n$, είναι γνωστές ως singular values του πίνακα A και αποτελούν την τετραγωνική ρίζα των ιδιοτιμών του $A \cdot A^T$ και του $A^T \cdot A$. Για να υπολογίσουμε το SVD πρέπει να βρούμε τις ιδιοτιμές (eigenvalues) και τα ιδιοδιανύσματα (eigenvectors), τα οποία μας καταδεικνύουν ποια features περιέχουν περισσότερη πληροφορία και ποια μπορούν να απαλειφθούν.

Υπάρχει μία πολύ χρήσιμη ιδιότητα που συνοδεύει τον ορισμό του θεωρήματος, σύμφωνα με την οποία, ο αριθμός n των singular values είναι ίσος με την τάξη r του πίνακα A (δηλαδή rank(A) = n). Αυτό, μας δίνει τη δυνατότητα να βρούμε έναν τρόπο για μια προβολή από ένα χώρο μεγαλύτερης διάστασης, σε ένα μικρότερων διαστάσεων.

Η Latent Semantic Indexing (LSI) αφορά ουσιαστικά την εφαρμογή του SVD στην επεξεργασία φυσικής γλώσσας και μας δίνει τη δυνατότητα να μειώσουμε περαιτέρω την τάξη του singular value matrix Σ σε ένα μέγεθος k διαστάσεων όπως φαίνεται και στο Σχήμα 3.3. Επιλέγουμε k = 300, για μια αποδοτική μείωση των διανυσμάτων, ενώ, πριν το SVD, ο term-document πίνακας μετασχηματίζεται με το σχήμα tf-idf για καλύτερα αποτελέσματα. Η λειτουργία του SVD με αυτή την μείωση, έχει σαν αποτέλεσμα να διατηρείται όλη η σημαντική σημασιολογική πληροφορία στο κείμενο, ενώ παράλληλα απαλείφεται ο θόρυβος και άλλες ανεπιθύμητες παρενέργειες του αρχικού χώρου του A. Αυτό το μειωμένο σετ πινάκων συμβολίζεται με μια μετασχηματισμένη φόρμουλα ως εξής:

$$A \approx A_k = U_k * \Sigma_k * V_k^T \tag{3.2}$$

Το SVD εξυπηρετεί όταν υπάρχουν στα δεδομένα μας πλεονάζοντα features, αλλά δε βοηθά τόσο αναφορικά με το sparcity των δεδομένων: δύο features μπορούν να είναι sparse, αλλά να εμπεριέχουν χρήσιμη πληροφορία (relevant) που εξυπηρετεί την ταξινόμηση, οπότε δε μπορούμε να αφαιρέσουμε κανένα από τα δύο. Επιπλέον, η χρησιμοποίηση μιας μικρότερης διάστασης, δε μας εγγυάται πάντα πως θα έχουμε και την καλύτερη δυνατή απόδοση. Κάτι τέτοιο φαίνεται να ισχύει και στην περίπτωσή μας, αφού όπως θα δούμε και



Σχήμα 3.3: Μείωση ενός πίνακα σε k διαστάσεις με χρήση SVD

παρακάτω, η συγκεκριμένη τεχνική είχε κόστος στα αποτελέσματα του συστήματος.

3.4.3 Μέθοδοι Επιλογής Χαρακτηριστικών Διανυσμάτων

Αντί να χρησιμοποιήσουμε όλες τις διαθέσιμες λέξεις του λεξικού, μπορούμε να επιλέξουμε ένα υποσύνολο από αυτές για τον διαχωρισμό των κλάσεων παράγοντας ένα «καλύτερο» μοντέλο. Η τεχνική αυτή ονομάζεται feature selection και βασίζεται στην υπόθεση ότι τα αρχικά δεδομένα περιέχουν αρκετή μη χρήσιμη πληροφορία που δυσχεραίνει την ταξινόμηση. Τα πλεονεκτήματα που προσφέρει είναι πολλά:

(i) Μείωση των διαστάσεων που συνεπάγεται μικρότερο υπολογιστικό κόστος και χρόνο εκτέλεσης

(ii) Μείωση του θορύβου για τη βελτίωση του βαθμού της ταξινόμησης

(iii) Περισσότερο χρήσιμα και ερμηνεύσιμα features που επαρκούν για τον προσδιορισμό του στόχου

(iv) Βοηθά στη γενίκευση με τη μείωση του overfitting

Αυτά τα πλεονεκτήματα είναι ιδιαίτερα σημαντικά στην περίπτωσή μας, όπου βρισκόμαστε σε έναν χώρο μεγάλων διαστάσεων με θόρυβο και οι λέξεις είναι χαμηλής συχνότητας. Επίσης, υπάρχουν πολλά features σε σχέση με τα δείγματα, κάτι που κάνει το μοντέλο πολύπλοκο. Υπάρχουν διάφοροι μέθοδοι και τεχνικές για feature selection. Παρακάτω παρουσιάζουμε αυτές που ασχοληθήκαμε και εφαρμόσαμε στην παρούσα διπλωματική εργασία.

Categorical Proportional Difference (CPD)

Η μέθοδος αυτή εισήχθη από τους Simeon και Hilderman [23] και στοχεύει στο να καθορίσει πόσο συμβάλει ένα συγκεκριμένο unigram στον σωστό καθορισμό μιας κλάσης, ενώ χρησιμοποιήθηκε σα feature selection μέθοδος από

τους Tim O'Keefe και Irena Koprinska [27]. Για να ορίσουμε ευκολότερα τη CPD, θα χρησιμοποιήσουμε έναν τυπικό 2×2 contingency πίνακα, όπως φαίνεται στο 3.5. Σε αυτόν τον πίνακα το γράμμα A υποδηλώνει τον αριθμό των φορών που το feature f (δηλαδή η opinion word) εμφανίζεται στην κατηγορία c, το B είναι ο αριθμός των φορών που η λέξη εμφανίζεται μέσα στο υπόλοιπο corpus (χωρίς την κατηγορία c δηλαδή), το γράμμα C δηλώνει τον αριθμό των φορών όπου έχουμε την κατηγορία c χωρίς το f, το D δηλώνει τον αριθμό των κειμένων που δεν εμφανίζεται ούτε το f ούτε η κατηγορία c, ενώ το N = A + B + C + D.

	c	¬c	Σ
f	A	B	A+B
¬f	C	D	C+D
Σ	A+C	B+D	N

Πίνακας 3.5: Γενική μορφή ενός contingency πίνακα

Η CPD λοιπόν, μετρά το βαθμό στον οποίο συμβάλει μία συγκεκριμένη λέξη στη διαφοροποίηση του θετικού συναισθήματος από το αρνητικό. Οι δυνατές τιμές που μπορεί να πάρει βρίσκονται στο διάστημα (-1,1], όπου τιμές κοντά στο -1 υποδεικνύουν ότι η λέξη απαντάται σχεδόν σε ίσο αριθμό κειμένων και στις δύο κατηγορίες, ενώ 1 έχουμε όταν η λέξη συναντάται σε κείμενα της μιας μόνο κατηγορίας και άρα είναι ενδεικτική του συναισθήματος. Επομένως, η CPD μιας λέξης w στην κατηγορία c ορίζεται ως:

$$CPD(w, c) = \frac{A - B}{A + B} \tag{3.3}$$

Ο υπολογισμός της δεν είναι τίποτε άλλο παρά μόνο ο λόγος της διαφοράς μεταξύ του αριθμού των κειμένων της κατηγορίας στην οποία εμφανίζεται η λέξη w και του αριθμού των κειμένων της άλλης κατηγορίας στα οποία υπάρχει επίσης λέξη w, προς το συνολικό αριθμό κειμένων στα οποία υπάρχει η συγκεκριμένη λέξη. Η CPD για ένα unigram λοιπόν, είναι ο λόγος που σχετίζεται με την κλάση c_i για την οποία επιτυγχάνεται η μεγαλύτερη τιμή. Αυτό σημαίνει:

$$CPD(w) = max_i\{CPD(w, c_i)\} \tag{3.4}$$

Η μέθοδος βρέθηκε πως παρουσιάζει καλύτερα αποτελέσματα, εφόσον έχουμε κάνει αρχικά τη μετατροπή του πίνακα σε binary. Έχοντας υπολογίσει τη CPD για κάθε λέξη, χρησιμοποιούμε ένα είδος filtering για να επιλέξουμε τις καλύτερες (top-ranked) k opinion words.

Categorical Probability Proportion Difference (CPPD)

Για τον ορισμό της CPPD, χρειάζεται πρώτα να ορίσουμε την Probability Proportion Difference (PPD). Η ποσότητα αυτή, μετρά την πιθανότητα ενός

όρου να ανήκει σε μια συγκεκριμένη κλάση. Έτσι, εάν ένας όρος έχει μεγάλη πιθανότητα να ανήκει στην κυρίαρχη κατηγορία/κλάση (δηλαδή θετική ή αρνητική), αυτό υποδεικνύει ότι ο όρος είναι σημαντικός στην ανίχνευση της κατηγορίας της άγνωστης κριτικής. Από την άλλη, εάν ο όρος έχει σχεδόν ίση πιθανότητα να ανήκει και στις δύο κατηγορίες, τότε ο όρος δεν είναι χρήσιμος στο διαχωρισμό των κλάσεων. Η PPD τιμή υπολογίζεται από τη διαφορά των πιθανοτήτων του να ανήκει ο επιθυμητός όρος στη θετική ή στην αρνητική κλάση. Η πιθανότητα της εγγύτητας ενός όρου σε μία κλάση, εξαρτάται και από τον αριθμό των κειμένων στα οποία εμφανίζεται ο όρος, καθώς και από τον αριθμό των μοναδικών όρων που εμφανίζονται σε μία κλάση.

Η CPPD είναι μια αξιόπιστη μέθοδος για feature selection που συνδυάζει τα πλεονεκτήματα και απαλείφει τα μειονεκτήματα των τεχνικών CPD και PPD. Τα πλεονεκτήματα της CPD μεθόδου, είναι ότι μετράει το βαθμό της διαχωριστικής ικανότητας μιας κλάσης για έναν όρο, κάτι που αποτελεί ένα σημαντικό γνώρισμα του κυρίαρχου feature. Επίσης, απαλείφει όρους που απαντώνται ισοπίθανα και στις δύο κλάσεις και άρα δεν παίζουν σημαντικό ρόλο στην ταξινόμηση, ενώ μπορεί να απομακρύνει όρους με υψηλή document frequency, οι οποίοι δεν είναι σημαντικοί όπως τα stop words. Η PPD μέθοδος από την άλλη, απαλείφει τους όρους με χαμηλή document frequency που δεν είναι χρήσιμοι

Algorithm 1: Categorical Probability Proportion Difference Feature Selection Method [36]

Input: Document corpus (D) with labels (C) positive or negative
Output: ProminentFeatureSet
Step 1: Preprocessing
$t \leftarrow ExtractUniqueTerms(D)$
$F \leftarrow TotalUniqueTerms(D)$
$W_p \leftarrow TotalTermsInPositiveClass(D, C)$
$W_n \leftarrow TotalTermsInNegativeClass(D, C)$
Step 2: Main Feature Selection loop
for $t \in \boldsymbol{F}$ **do**
  $N_{tp} = CountPositiveDocumentsInWhichTermAppears(D, t)$
  $N_{tn} = CountNegativeDocumentsInWhichTermAppears(D, t)$
end
for $t \in \boldsymbol{F}$ **do**
  $cpd = \frac{N_{tp} - N_{tn}}{N_{tp} + N_{tn}}$
  $ppd = \frac{N_{tp}}{W_p + F} - \frac{N_{tn}}{W_n + F}$
  if $cpd > T1$ && $ppd > T2$ **then**
    $ProminentFeatureSet \leftarrow SelectTerm(t)$
  end
end

στην ταξινόμηση του συναισθήματος όπως οι σπάνιοι όροι. Μπορεί ακόμη να λαμβάνει υπ' όψιν το μέγεθος των θετικών και των αρνητικών κριτικών, καθώς γενικά κείμενα με θετικό προσανατολισμό έχουν μεγαλύτερο μήκος σε σχέση με κείμενα που ανήκουν στην αρνητική κλάση, οπότε υπάρχει μεγαλύτερη πιθανότητα οι περισσότερες λέξεις που θα επιλέξει μία feature selection μέθοδος να είναι θετικού συναισθήματος. Η μέθοδος CPPD για feature selection, όπως την όρισαν και τη χρησιμοποίησαν οι Agarwal και Mittal [36] περιγράφεται στον Αλγόριθμο 1. Στη δική μας περίπτωση παρήγαγε λίγο καλύτερα αποτελέσματα μια ελαφρά παραλλαγή του συγκεκριμένου αλγορίθμου, όπου δε χρησιμοποιούμε μόνο τις κοινές, αλλά το σύνολο των καλύτερων λέξεων που προκύπτουν από τον υπολογισμό του CPD και του PPD. Όπως και προηγουμένως, προκύπτουν καλύτερα ποσοστά απόδοσης εάν έχουμε μετατρέψει προηγουμένως τους document-term πίνακες σε binary.

Chi-Square (χ^2) Statistic

Υπάρχουν δύο κυρίως τύποι τυχαίων μεταβλητών που σχετίζονται αντίστοιχα με δύο τύπους δεδομένων: αριθμητικές (numerical), που παράγουν δεδομένα σε αριθμητική μορφή και ποιοτικές (categorical), που παράγουν δεδομένα τα οποία ανήκουν σε κατηγορίες (βλ. Πίνακα 3.6).

Το χ^2 χρησιμοποιείται για να διερευνήσει την εξάρτηση μεταξύ δύο μεταβλητών. Έτσι, η συνάρτηση αυτή απομακρύνει τα features που είναι περισσότερο πιθανό να μη σχετίζονται με κάποια κλάση και άρα είναι ασύνδετα με την ταξινόμηση. Για τον υπολογισμό του χ^2, θα χρησιμοποιήσουμε ξανά τα στοιχεία του Πίνακα 3.5, αυτή τη φορά μέσα από την παρακάτω φόρμουλα:

$$\chi^2(w, c) = \frac{N \times (AD - BC)^2}{(A + B) \times (C + D) \times (B + D) \times (A + C)} \tag{3.5}$$

Αφού μετατρέψουμε τον πίνακα σύμφωνα με την tf-idf φόρμουλα, βρίσκουμε το χ^2 statistic για κάθε feature στο corpus και επιλέγουμε τα word vectors των λέξεων που διαθέτουν τις υψηλότερες τιμές από τον πίνακα που περιέχει τις συχνότητες των όρων μέσα στα κείμενα που είναι σχετικά με τις κλάσεις.

Τύπος δεδομένων	Τύπος ερώτησης	Πιθανές απαντήσεις
Categorical	Τι συναίσθημα εκφράζει η κριτική της ταινίας;	Θετικό ή Αρνητικό
Numerical	Διακριτός - Πόσες ταινίες είδες;	Δύο ή Τρεις
Numerical	Συνεχής - Πόσο ψηλός είσαι;	$1.80m$

Πίνακας 3.6: Διάκριση μεταξύ αριθμητικών και ποιοτικών τυχαίων μεταβλητών

Z-Score

Χρησιμοποιώντας και πάλι έναν contingency πίνακα όπως ο 3.5, μπορούμε να ορίσουμε την πιθανότητα της εμφάνισης ενός feature f σε όλο το corpus ως P(f) = (A+B)/N.

Για να ορίσουμε τη διαχωριστική δύναμη του f, θεωρούμε ότι κάθε feature f ακολουθεί διωνυμική κατανομή με μέση τιμή $P(f) \cdot n'$ και διασπορά $P(f) \cdot (1 - P(f)) \cdot n'$. Έτσι, ο κανόνας απόφασης χρησιμοποιεί το τυποποιημένο Z score [34] που συνδέεται με κάθε feature μέσα από την παρακάτω εξίσωση:

$$Zscore(f) = \frac{A - n' \cdot P(f)}{\sqrt{n' \cdot P(f) \cdot (1 - P(f))}} \tag{3.6}$$

Minimum Redundancy Maximum Relevance (mRMR)

Το mRMR είναι μία τεχνική για feature selection που προτάθηκε από τους Peng et al. [11], η οποία προσπαθεί να απαλείψει τα περιττά (redundant) στοιχεία που υπάρχουν στα δεδομένα, κρατώντας ταυτόχρονα τα περισσότερα σχετικά (relevant) με τους στόχους των κλάσεων. Επειδή έχουμε categorical features, το μέτρο που χρησιμοποιείται για να μετρήσουμε την ομοιότητα-εξάρτηση μεταξύ των features είναι η κοινή πληροφορία (Mutual Information). Έχοντας δύο τυχαίες μεταβλητές x και y, η κοινή τους πληροφορία ορίζεται μέσω των συναρτήσεων πυκνότητας πιθανότητας p(x), p(y) και p(x,y) ως εξής:

$$I(x; y) = \int \int p(x, y) log \frac{p(x, y)}{p(x)p(y)} dxdy \tag{3.7}$$

Στόχος είναι να επιλεχθεί ένα υποσύνολο από features που χαρακτηρίζει καλύτερα τη στατιστική ιδιότητα του στόχου της ταξινόμησης, δηλαδή που υπόκεινται στον περιορισμό ότι αυτά τα features είναι αμοιβαία όσο το δυνατόν περισσότερο ανόμοια μεταξύ τους, αλλά οριακά όσο το δυνατό περισσότερο όμοια στη μεταβλητή της ταξινόμησης.

Η συνθήκη για τον ελάχιστο πλεονασμό (redundancy) στα δεδομένα είναι:

$$minW_I,\ W_I = \frac{1}{|S|^2} \sum_{i,j \in S} I(f_i, f_j), \tag{3.8}$$

όπου S είναι το σύνολο των features και I(i,j) είναι η κοινή πληροφορία μεταξύ των features i και j.

Για να μετρήσουμε το επίπεδο της διαχωριστικότητας μεταξύ των features όταν αυτά εκφράζονται με στόχο διαφορετικές κλάσεις, χρησιμοποιείται επίσης κοινή πληροφορία I(c,f_i) μεταξύ των στόχων των κλάσεων (c = c_1, c_2 για την περίπτωσή μας) και του feature f_i. Έτσι, το I(c,f_i) εκτιμά τη σχέση μεταξύ του f_i για το σκοπό της ταξινόμησης.

Η συνθήκη για τη μέγιστη συσχέτιση (relevance) δίνεται μέσα από τον παρακάτω τύπο:

$$maxV_I,\ V_I = \frac{1}{|S|} \sum_{i \in S} I(c, f_i), \tag{3.9}$$

όπου c είναι οι στόχοι των κλάσεων, δηλαδή οι κριτικές που εκφράζουν θετικό ή αρνητικό συναίσθημα.

Το mRMR feature set προκύπτει συνδυάζοντας τις δύο παραπάνω σχέσεις κατά το βέλτιστο δυνατό τρόπο. Υπάρχουν δύο τρόποι για να γίνει αυτό: MID (Mutual Information Difference) και MIQ (Mutual Information Quotient). Ο πρώτος αναπαριστά τη διαφορά της κοινής πληροφορίας:

$$MID = max(V - W) = \max_{i \in \Omega_S}[I(i,h) - \frac{1}{|S|}\sum_{j \in S} I(i,j)], \tag{3.10}$$

ενώ ο δεύτερος το πηλίκο αντίστοιχα:

$$MIQ = max(V/W) = \max_{i \in \Omega_S}[I(i,h)/\frac{1}{|S|}\sum_{j \in S} I(i,j)] \tag{3.11}$$

Για τη δική μας περίπτωση ταξινόμησης, τα αποτελέσματα ήταν ελαφρώς καλύτερα χρησιμοποιώντας τις λέξεις που προέκυψαν από τη διαφορά της κοινής πληροφορίας. Έχοντας λοιπόν τις «καλύτερες» λέξεις που προκύπτουν σύμφωνα με αυτόν τον αλγόριθμο, χρησιμοποιούμε τα αντίστοιχα διανύσματα των λέξεων από τον document-term πίνακα για την εκπαίδευση του μοντέλου.

Delta Tf-Idf

Το Delta Tf-Idf [28] δεν αποτελεί μια τεχνική καθαρά για feature selection, αλλά μια βελτιωμένη προσέγγιση της απλής tf-idf φόρμουλας, εξειδικευμένη για sentiment analysis. Εξερευνούμε λοιπόν τα πλεονεκτήματα της ανάθεσης βαρών από αυτή τη συνάρτηση πάνω στα διανύσματα των λέξεων και επιλέγουμε τα κατάλληλα features που συγκεντρώνουν το υψηλότερο score μετά την εφαρμογή της. Πιο συγκεκριμένα, ξεκινάμε από έναν document-term πίνακα όπου η κάθε λέξη σχετίζεται με τον αριθμό των εμφανίσεών της στο αντίστοιχο κείμενο και ορίζουμε:

(i) $C_{t,d}$ είναι ο αριθμός των φορών που ο όρος t εμφανίζεται μέσα στο κείμενο d

(ii) P_t είναι ο αριθμός των κειμένων που περιέχει τον όρο t στο training set με τις θετικές κριτικές

(iii) $|P|$ είναι ο αριθμός των κειμένων που υπάρχουν στο training set με τις θετικές κριτικές

(iv) N_t είναι ο αριθμός των κειμένων που περιέχει τον όρο t στο training set με τις αρνητικές κριτικές

(v) $|N|$ είναι ο αριθμός των κειμένων που υπάρχουν στο training set με τις αρνητικές κριτικές

(vi) $V_{t,d}$ είναι η τιμή του feature για τον όρο t στο κείμενο d

Επειδή τα training sets είναι ισοσταθμισμένα (δηλαδή υπάρχει ίσος αριθμός θετικών και αρνητικών κριτικών) ισχύει:

$$\begin{aligned} V_{t,d} &= C_{t,d} * log_2\left(\frac{|P|}{P_t}\right) - C_{t,d} * log_2\left(\frac{|N|}{N_t}\right) \\ &= C_{t,d} * log_2\left(\frac{|P|}{P_t}\frac{N_t}{|N|}\right) \\ &= C_{t,d} * log_2\left(\frac{N_t}{P_t}\right) \end{aligned} \tag{3.12}$$

Η τιμή ενός ομοιόμορφα κατανεμημένου feature στο corpus είναι μηδέν. Όσο πιο ανομοιόμορφη είναι η κατανομή, τόσο πιο σημαντική θεωρείται η λέξη. Λέξεις που επικρατούν περισσότερο στο negative training set απ΄ ότι στο positive έχουν θετικό σκορ, ενώ features που είναι επικρατέστερα στο positive training set απ΄ ότι στο negative έχουν αρνητικό σκορ. Βασιζόμενοι σε αυτό το στοιχείο διαχωρίζουμε τα θετικά sentiment features από τα αρνητικά και επιλέγουμε τα περισσότερο αντιπροσωπευτικά (top ranked) από αυτά.

Ο μετασχηματισμός αυτός λοιπόν αναδεικνύει τη σημαντικότητα των λέξεων που δεν είναι ομοιόμορφα κατανεμημένες ανάμεσα στην θετική και την αρνητική κλάση, ενώ περιορίζει τις λέξεις που είναι ομοιόμορφα κατανεμημένες. Επαληθεύουμε πως τα βάρη που αναθέτει στις opinion words, καταφέρνουν να αναπαριστούν τελικά καλύτερα τα κείμενα για ανάλυση συναισθήματος.

Κεφάλαιο 4

Γλωσσικά Πιθανοτικά Μοντέλα

Μέσω της διαδικασίας που περιγράφτηκε στο προηγούμενο κεφάλαιο, καταφέρνουμε τελικά με τον ένα ή με τον άλλο τρόπο, να εξάγουμε τα διανύσματα των λέξεων (feature/word vectors). Σκοπός μας λοιπόν σε αυτό το στάδιο, είναι να εκπαιδεύσουμε το μοντέλο με τα εκάστοτε features, για να εκτιμήσουμε τις παραμέτρους του. Έτσι, θα είμαστε σε θέση να υπολογίσουμε τη log probability, με τη συνεισφορά της οποίας θα κάνουμε την ταξινόμηση, που είναι και το ζητούμενο του προβλήματος.

4.1 Πεπερασμένα Μοντέλα Μίξης

Πολλά από τα δεδομένα που αντιμετωπίζουμε σήμερα, δεν φαίνεται να μπορούν εφαρμοστούν κατάλληλα στα απλά πιθανοτικά μοντέλα λόγω της ετερογένειας. Γι΄ αυτό, χρησιμοποιούμε mixture models, δηλαδή πιθανοτικά μοντέλα που υποθέτουν ότι τα υποκρυπτόμενα δεδομένα ανήκουν σε ένα σύνθετο μείγμα κατανομών (mixture distribution), συχνότερα ίδιου τύπου. Ουσιαστικά δηλαδή, συνδυάζουμε απλά μοντέλα σε ένα πιο σύνθετο. Η συνάρτηση πυκνότητας μιας mixture distribution είναι ο κυρτός συνδυασμός (ένας γραμμικός συνδυασμός όπου όλοι οι συντελεστές-βάρη αθροίζουν στη μονάδα) από άλλες συναρτήσεις πυκνότητας πιθανότητας:

$$p(x) = w_1 p_1(x) + w_2 p_2(x) + ... + w_n p_n(x) \tag{4.1}$$

Οι μεμονωμένες-πεπερασμένες συναρτήσεις πυκνότητας πιθανότητας $p_i(x)$ που συνδυάζονται έτσι ώστε να συνθέσουν τη mixture density $p(x)$, ονομάζονται mixture components, ενώ τα βάρη $w_1, w_2, ..., w_n$ που σχετίζονται με κάθε component ονομάζονται mixture weights ή mixture coefficients.

Ένα mixture model με μεγάλη likelihood πιθανότητα, τείνει να έχει τα παρακάτω θετικά χαρακτηριστικά:

- Οι component distributions έχουν υψηλές κορυφές, κάτι που σημαίνει ότι τα δεδομένα σε μία κλάση έχουν στενή σχέση μεταξύ τους (tight).
- Το mixture model είναι ένα μοντέλο που «καλύπτει» καλά τα δεδομένα και έτσι τα κυρίαρχα πρότυπα (patterns) που υπάρχουν στα δεδομένα καλύπτονται από τις component distributions.
- Υπάρχει ευελιξία ως προς την επιλογή των component distributions.
- Πετυχαίνει μία ξεχωριστή εκτίμηση πυκνότητας για κάθε cluster.
- Είναι ικανό να βρει κάποια «soft» όρια για την κατηγοριοποίηση. Αυτό είναι δυνατόν, επειδή κάθε σημείο στο χώρο ανήκει σε μία κλάση μόνο, έχοντας μία συγκεκριμένη πιθανότητα.

4.2 Gaussian Mixture Models (GMMs)

Τα GMMs είναι ίσως το πιο συχνά χρησιμοποιούμενο παράδειγμα από mixture distributions. Έτσι, ένα GMM είναι μια παραμετρική συνάρτηση πυκνότητας πιθανότητας, που αναπαρίσταται ως ένας σταθμισμένος γραμμικός συνδυασμός (μία μίξη) Γκαουσσιανών κατανομών (components). Υποθέτουμε λοιπόν ότι τα δεδομένα παράγονται από ένα σύνολο K απλών Γκαουσσιανών κατανομών, τότε:

$$p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mu_k, \Sigma_k), \tag{4.2}$$

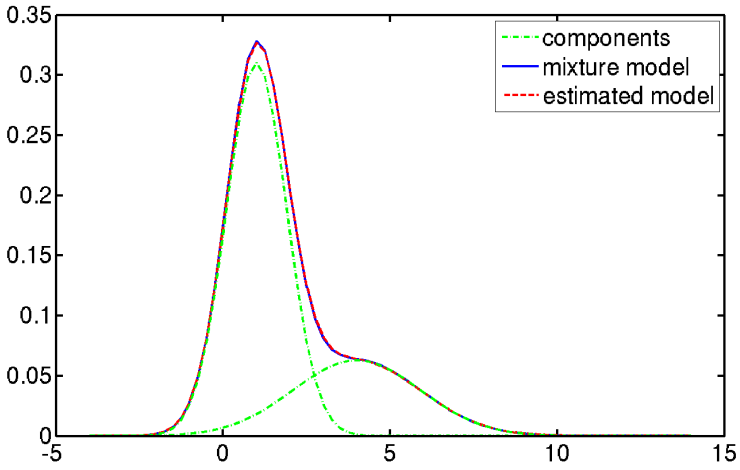
όπου π_k είναι η πιθανότητα ένα δείγμα να «έλκεται» από το k-οστό mixture component, δεδομένου ότι $\sum_{k=1}^{K} \pi_k = 1$ και $\pi_k > 0$. Ένας άλλος τρόπος να σκεφτούμε γύρω από αυτό το μοντέλο, είναι να εισάγουμε μια κρυφή μεταβλητή $z \in \{1, ..., K\}$ για κάθε δείγμα i, η οποία να υποδεικνύει από ποιο mixture component προήλθε. Κατ΄ αυτόν τον τρόπο προκύπτει:

$$p(x) = \sum_{k=1}^{K} p(z = k) p(x|z = k), \tag{4.3}$$

όπου $p(z = k) = \pi_k$ και $p(x|z = k) = \mathcal{N}(\mu_k, \Sigma_k)$.

Έχοντας λοιπόν ένα δείγμα $D = x_1, x_2, ..., x_n$, ο στόχος είναι να εκτιμήσουμε τις παραμέτρους μίξης (π_k, μ_k, Σ_k) για $k = 1, ..., K$. Οι παράμετροι του GMM εκτιμήθηκαν από τα training δεδομένα χρησιμοποιώντας τον αλγόριθμο Expectation Maximization (EM).

Ο EM, είναι ένας γενικός επαναληπτικός αλγόριθμος για εκπαίδευση μοντέλων με κρυφές μεταβλητές (όπως έχουμε και εδώ), που βελτιστοποιεί τη marginal likelihood των δεδομένων. Μεταπηδά μεταξύ δύο καταστάσεων, E και M, οι οποίες είναι υπεύθυνες για την εκτίμηση των κρυφών μεταβλητών δεδομένου του μοντέλου και μετά της εκτίμησης του μοντέλου δεδομένων αυτών των κρυφών μεταβλητών που υπολογίστηκαν αντίστοιχα.



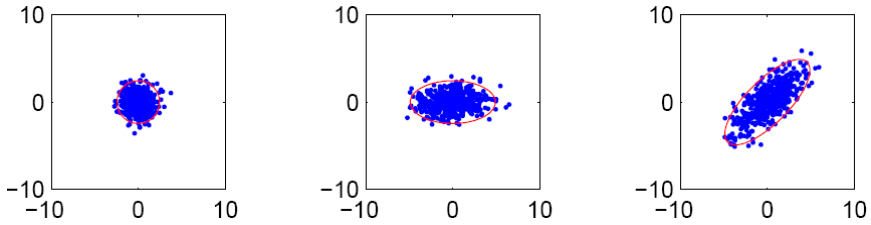
Σχήμα 4.1: Ένα Gaussian Mixture Model με δύο components

Το κύριο μειονέκτημα των GMMs, είναι ότι πρέπει να ξέρουμε εκ των προτέρων τον αριθμό των mixture models που ο αλγόριθμος προσπαθεί να προσαρμόσει (fit) το training dataset. Κάτι τέτοιο συμβαίνει και στη δικιά μας περίπτωση, όπου δε γνωρίζουμε ακριβώς πόσα mixture models πρέπει να χρησιμοποιηθούν και γι΄ αυτό έπρεπε να πειραματιστούμε με έναν διαφορετικό αριθμό μοντέλων, έτσι ώστε να βρούμε τον κατάλληλο αριθμό που θα δουλεύει καλύτερα για το δικό μας πρόβλημα ταξινόμησης. Δοκιμάσαμε λοιπόν να θέσουμε τον αριθμό των components ίσο με 2, 3, 5, 7 ή 10.

Ένα ακόμη σημαντικό χαρακτηριστικό, έχει να κάνει με την κατάλληλη αρχικοποίηση των mixture components. Μια κακή επιλογή των αρχικών παραμέτρων μπορεί να οδηγήσει σε άσχημα αποτελέσματα, καθώς ο EM μπορεί να συγκλίνει κάλλιστα σε ένα τοπικό ελάχιστο. Μία συχνά χρησιμοποιούμενη λύση, την οποία χρησιμοποιήσαμε και εμείς αρχικά τρέχοντας τον αλγόριθμο αρκετές φορές, ήταν η τυχαία αρχικοποίηση των δεδομένων. Μία δεύτερη λύση επίσης, είναι η αρχικοποίηση μέσω K-Means. Ο K-Means όμως, μπορεί να συγκλίνει σε μία υποβέλτιστη λύση, μην παράγοντας τελικά τα καλύτερα δυνατά αποτελέσματα για το mixture model fitting. Ωστόσο, και με τους δύο αυτούς τρόπους που δοκιμάστηκαν, τα ποσοστά της απόδοσης που προέκυψαν διέφεραν λίγο μόνο μεταξύ τους (στα όρια πάντα του στατιστικού σφάλματος), οπότε τελικώς επιλέχθηκε η αρχικοποίηση να ξεκινήσει τυχαία από το μηδέν.

Όσα αφορά τις παραμέτρους διασποράς (variance parameters), υπάρχουν διάφορες επιλογές που μπορούν να εφαρμοστούν για την εκπαίδευση της διασποράς των Gaussian mixtures, τις οποίες χρησιμοποιήσαμε και παραθέτουμε:

(i) Full covariance: ο Σ_k είναι τυχαίος για κάθε κλάση (τα clusters είναι γενικά ελλειψοειδή και υπάρχουν $\mathcal{O}(Km^2)$ παράμετροι)



Σχήμα 4.2: Δείγματα από spherical, diagonal και full covariance Gaussian. Η εικόνα παράχθηκε χρησιμοποιώντας το gaussSampleDemo.

(ii) Tied covariance: ο Σ_k είναι κοινός για όλα τα components (τα clusters είναι όμοια μεταξύ τους ελλειψοειδή)

(iii) Diagonal covariance: ο Σ_k είναι ένας διαγώνιος πίνακας (τα clusters είναι ευθυγραμμισμένα στον άξονα ελλειψοειδή και υπάρχουν $\mathcal{O}(Km)$ παράμετροι)

(iv) Spherical (isotropic) covariance: ο Σ_k είναι ίσος με $\sigma_k^2 I$ (τα clusters έχουν σφαιρικό σχήμα)

Η χρησιμοποίηση ενός GMM με full (unconstrained) covariance matrices, μοντελοποιεί τα δεδομένα σαν ελλειψοειδή σύννεφα «σε γωνία», επιτρέποντας τις συσχετίσεις μεταξύ των feature components. Ωστόσο, απαιτούνται πολλά δεδομένα εκπαίδευσης για την κατάλληλη εκτίμηση των παραμέτρων, ιδιαίτερα εάν βρισκόμαστε σε έναν high-dimensional χώρο, και γι΄ αυτό υπάρχει ο κίνδυνος του overfitting. Συχνά λοιπόν, οι πίνακες συνδιασπορών περιορίζονται σε σφαιρικούς (μοντελοποιούν τα δεδομένα ως σφαιρικό σύννεφο) ή διαγώνιους (μοντελοποιούν τα δεδομένα σαν ελλειψοειδή σύννεφα ευθυγραμμισμένα με τους άξονες των Γκαυσσιανών), όπου δε λαμβάνονται υπ΄ όψιν οι συσχετίσεις μεταξύ των μεταβλητών. Οι σφαιρικοί πίνακες έχουν μία μόνο παράμετρο και την απλούστερη Gaussian pdf, αλλά είναι χρήσιμοι όταν τα δεδομένα που έχουμε είναι αραιά ή το υπολογιστικό κόστος μεγάλο. Οι διαγώνιοι πίνακες από την άλλη, αποτελούν τη συνηθέστερη επιλογή καθώς αποτελούν μία συμβιβαστική λύση ανάμεσα στο κόστος και τη πολυπλοκότητα της εκπαίδευσης. Στη δική μας περίπτωση, ανάλογα με τη μέθοδο που έχει προηγηθεί για τον προσδιορισμό των features, επωφελούμασταν κάθε φορά από διαφορετικό τύπο των Γκαουσσιανών. Στο Σχήμα 4.2, υπάρχει η οπτικοποίηση αυτών των διαφορετικών υποθέσεων.

Κεφάλαιο 5

Ταξινόμηση

5.1 Naive Bayes Model

Ο Naive Bayes ταξινομητής είναι ένα πιθανολογικό μοντέλο βασιζόμενο στο θεώρημα του Bayes, το οποίο (για λόγους πληρότητας) δίνεται από την παρακάτω φόρμουλα:

$$\Pr(A|B) = \frac{\Pr(B|A)Pr(A)}{Pr(B)}, \tag{5.1}$$

όπου $\Pr(A)$ είναι η prior πιθανότητα του γεγονότος A και $\Pr(A|B)$ είναι η υπό συνθήκη πιθανότητα του A δεδομένου του B.

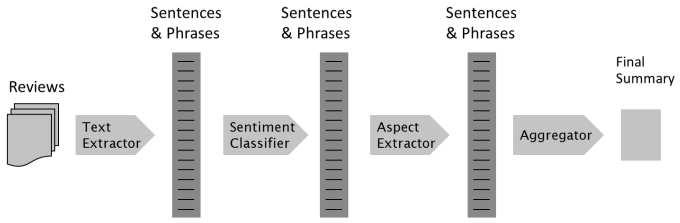
Ο Naive Bayes, όπως και άλλα πιθανολογικά μοντέλα, μεταχειρίζεται τόσο τα attributes όσο και την κατηγοριοποίηση ενός στιγμιότυπου σαν τυχαίες μεταβλητές. Έτσι, αυτό το μοντέλο υποθέτει ότι οι μεταβλητές συνδέονται μεταξύ τους, αλλά μόνο πιθανολογικά. Για παράδειγμα, εάν στο κείμενο υπάρχει η λέξη «disaster», τότε υπάρχει μεγάλη πιθανότητα το γενικό συναίσθημα να είναι αρνητικό. Ωστόσο, υπάρχει πάντα και το ενδεχόμενο (με μικρή πιθανότητα όμως) το συναίσθημα να είναι θετικό («disaster was averted»).

Τα πιθανολογικά μοντέλα έχουν τρία σημεία κλειδιά:

(i) Την εξαρχής πεποίθηση για την κατηγοριοποίηση του κειμένου. Αυτό, αναπαριστάται μέσω της πιθανότητας $Pr(C)$ (prior probability).

(ii) Ένα μοντέλο που μας υποδεικνύει την πιθανότητα της ύπαρξης μιας λέξης $W_1 = w_1, ..., W_n = w_n$ δοσμένης της κλάσης C = c. Αυτό το μοντέλο εκφράζεται σαν την υπό συνθήκη πιθανότητα $Pr(W_1 = w_1, ..., W_n = w_n|C = c)$.

(iii) Χρησιμοποιώντας τη φόρμουλα του Bayes, ανανεώνουμε τη γνώμη μας για την ταξινόμηση του κειμένου δεδομένου των λέξεων που αυτό περιέχει: $Pr(C|W_1, ..., W_n) = \frac{Pr(W_1,...,W_n|C)Pr(C)}{Pr(W_1,...,W_n)}$. Έτσι, η προβλεπόμενη κλάση είναι αυτή που έχει τη μεγαλύτερη υπό συνθήκη πιθανότητα δεδομένου των λέξεων.



Σχήμα 5.1: Γενικό πλάνο ενός Sentiment Summarizer (Εικόνα από [20])

Το πρόβλημα με το παραπάνω πλαίσιο είναι ότι χρειαζόμαστε έναν πολύ μεγάλο αριθμό παραμέτρων, εάν δεν επιβάλλουμε και άλλες συνθήκες. Εάν τα attributes $W_1, ..., W_5$ είναι δυαδικά (έστω η κλάση που εκφράζει θετικό συναίσθημα $\rightarrow 1$, ενώ η κλάση που εκφράζει αρνητικό $\rightarrow 0$), τότε έχουμε $2^5 = 32$ πιθανά στιγμιότυπα. Κάθε στιγμιότυπο $w_1, ..., w_5$ αντιστοιχεί στις δύο παραμέτρους του μοντέλου: μία προσδιορίζει $Pr(W_1 = w_1, ..., W_5 = w_5|C = 0)$ και μία προσδιορίζει $Pr(W_1 = w_1, ..., W_5 = w_5|C = 1)$. Ο αριθμός των παραδειγμάτων που χρειάζεται για να εκπαιδεύσουμε αυτές τις παραμέτρους, αυξάνει τουλάχιστον γραμμικά με τον αριθμό των παραμέτρων.

Επομένως, χρειαζόμαστε υποθέσεις για την απλοποίηση του μοντέλου. Η υπόθεση του Naive Bayes προτείνει ότι όλα τα attributes είναι ανεξάρτητα δοσμένης της κλάσης. Αυτό σημαίνει ότι μπορούμε να γράψουμε την υπό συνθήκη πιθανότητα $Pr(W_1 = w_1, ..., W_5 = w_5|C = 1)$ πλέον ως εξής: $Pr(W_1 = w_1|C = 1) \cdot Pr(W_2 = w_2|C = 1) \cdot ... \cdot Pr(W_5 = w_5|C = 1)$. Χρησιμοποιώντας αυτήν την υπόθεση, ο αριθμός των παραμέτρων γίνεται αυτομάτως γραμμικός με τον αριθμό των μεταβλητών. Για τη μεταβλητή W_i, χρειάζεται να ξέρουμε τις τιμές $Pr(W_i = w_i|C = c)$ όπου $w_i, c \in 0, 1$. Αφού w_i και c μπορούν να πάρουν μόνο δυαδικές τιμές, χρειάζεται να ξέρουμε μόνο 4 παραμέτρους[1]. Συνολικά, αυτό μας δίνει 20 παραμέτρους. Έχοντας λοιπόν μικρότερο αριθμό παραμέτρων, είναι πολύ πιο εύκολο να εκπαιδεύσουμε το μοντέλο.

Ο Naive Bayes αποδεικνύεται έτσι ένας πολύ χρήσιμος αλγόριθμος, ακόμη και σε περιπτώσεις όπου η υπόθεση της ανεξαρτησίας δεν είναι πλήρως αποδεκτή (η κατηγοριοποίηση κειμένων είναι μία από αυτές τις περιπτώσεις).

Για την ταξινόμηση λοιπόν του συναισθήματος ανάμεσα σε μια θετική και μία αρνητική κριτική, χρησιμοποιήσαμε την παρακάτω φόρμουλα:

$$c_{NB} = \underset{c_j \in C}{\operatorname{argmax}} \Pr(c_j) \prod_{i \in positions} Pr(w_i|c_j), \tag{5.2}$$

όπου $Pr(c_j) = \frac{|docs_j|}{|total\ \#\ of documents|}$ και $\hat{Pr}(w|c) = \frac{count(w,c)+1}{count(c)+|V|}$.

Για μαθηματική ευκολία, ο τύπος 5.2 χρησιμοποιήθηκε σε λογαριθμική κλίμακα.

[1] Για την ακρίβεια, αυτό είναι πλεονάζων διότι ισχύει $Pr(W_i = 1|C = c) = 1 - Pr(W_i = 0|C = c)$.

5.2 Boolean Multinomial Naive Bayes

Για τη συγκεκριμένη περίπτωση της ταξινόμησης του συναισθήματος, η ύπαρξη μιας λέξης φαίνεται πως είναι πολύ σημαντικότερος παράγοντας από τη συχνότητα που αυτή απαντάται μέσα στο κείμενο. Κάτι τέτοιο, σημαίνει πως εάν η ύπαρξη της λέξης «fantastic» μέσα στο κείμενο σημαίνει πολλά για εμάς, το γεγονός ότι η συγκεκριμένη λέξη εμφανίζεται στο κείμενο 5 φορές, δεν προσδίδει κάποια επιπλέον χρήσιμη πληροφορία. Γι' αυτό το λόγο, χρησιμοποιείται ευρέως μία παραλλαγή του Naive Bayes αλγορίθμου στην ταξινόμηση κειμένου με βάση το συναίσθημα, που ονομάζεται Binarized (Boolean feature) Multinomial Naive Bayes.

Η εκπαίδευση του Boolean Multinomial Naive Bayes αρχίζει με την εξαγωγή του λεξικού από τα δεδομένα εκπαίδευσης. Στη συνέχεια, για κάθε c_j στο C, υπολογίζουμε τους $Pr(c_j)$ όρους ως εξής:

$$Pr(c_j) \leftarrow \frac{|docs_j|}{|total\ \#\ of documents|} \tag{5.3}$$

όπου $docs_j$, είναι όλα τα κείμενα που ανήκουν στην κλάση c_j. Τέλος, για κάθε λέξη w_k που υπάρχει στο λεξικό, υπολογίζονται οι $Pr(w_k|c_j)$ όροι ως:

$$\Pr(w_k|c_j) \leftarrow \frac{n_k + a}{n + a|Vocabulary|} \tag{5.4}$$

όπου n_k είναι ο αριθμός των εμφανίσεων της λέξης w_k στο $Text_j$, αν $Text_j$ οριστεί να είναι ένα μόνο κείμενο που περιέχει όλα τα $docs_j$.

Κατά την ταξινόμηση ενός test εγγράφου d, ο Boolean Multinomial Naive Bayes αλγόριθμος, απαιτεί πρώτα την αφαίρεση όλων των διπλών λέξεων από το d, ενώ τέλος υπολογίζεται ο Naive Bayes χρησιμοποιώντας την ίδια εξίσωση με προηγουμένως:

$$c_{NB} = \underset{c_j \in C}{\operatorname{argmax}} \Pr(c_j) \prod_{i \in positions} Pr(w_i|c_j) \tag{5.5}$$

5.3 GMM-based Classifier (GMMC)

Για την ταξινόμηση μιας κριτικής έχοντας χρησιμοποιήσει τα GMMs, χρησιμοποιούμε τη log probability του μοντέλου που εκπαιδεύσαμε. Έτσι, ταξινομούμε με βάση τη μεγαλύτερη πιθανότητα που προκύπτει από τα δεδομένα εκπαίδευσης για το εκάστοτε test document στην αντίστοιχη κλάση (θετική ή αρνητική). Στη συνέχεια επεξηγείται η κατασκευή ενός GMMC:

(i) Κατά τη διαδικασία της εκπαίδευσης ορίζουμε ένα GMM για κάθε κλάση. Με άλλα λόγια, χρησιμοποιούμε τα δεδομένα μιας κλάσης για την εκπαίδευση ενός GMM. Αυτό γίνεται διαδοχικά για κάθε κλάση, χωρίς να υπάρχει αλληλεπίδραση μεταξύ των GMM διαφορετικών κλάσεων.

(ii) Κατά τον έλεγχο μιας κριτικής από το test set, χρειάζεται να στείλουμε τα δεδομένα της άγνωστης κλάσης στο GMM κάθε κατηγορίας. Η κλάση που προβλέπεται είναι αυτή που σχετίζεται με το GMM που έχει τη μέγιστη πιθανότητα.

5.4 Μετρικές Απόδοσης

Για την εκτίμηση της απόδοσης του classifier του συστήματος, χρησιμοποιήθηκαν διάφορες μετρικές. Η κυριότερη από αυτές, η οποία έχει χρησιμοποιηθεί στις περισσότερες από τις προηγούμενες δουλειές και τη χρησιμοποιούμε και εμείς για τη σύγκριση των αποτελέσματα μας, είναι το accuracy. Εκτός όμως από αυτό, θεωρήσαμε σκόπιμο να χρησιμοποιήσουμε και δύο άλλες μεθόδους αξιολόγησης που εφαρμόζονται ευρέως στο χώρο της κατηγοριοποίησης κειμένου, γνωστές ως precision και recall (ή sensitivity), οι οποίες συνδυάζονται μάλιστα για να παράγουν το F_1 score (F-score ή F-measure).

Για την ευκολότερη ανάλυση των μετρικών απόδοσης, παρουσιάζουμε αρχικά τον παρακάτω πίνακα:

		Actual class 1	*Actual class* 0
Predicted class	1	True Positive	False Positive
	0	False Negative	True Negative

Πίνακας 5.1: Υπολογισμός Accuracy, Precision και Recall

Για τη δική μας περίπτωση, όπου έχουμε να ταξινομήσουμε κριτικές από ταινίες σε θετικές ή αρνητικές, ο παραπάνω πίνακας μεταφράζεται ως εξής:

- True Positive: η κριτική ταξινομήθηκε από τον αλγόριθμο ως θετική και είναι όντως θετική
- False Positive: η κριτική ταξινομήθηκε από τον αλγόριθμο ως θετική, αν και στην πραγματικότητα είναι αρνητική
- False Negative: η κριτική ταξινομήθηκε από τον αλγόριθμο ως αρνητική, αν και στην πραγματικότητα είναι θετική
- True Negative: η κριτική ταξινομήθηκε από τον αλγόριθμο ως αρνητική και είναι όντως αρνητική

Ξεκινώντας τώρα την επεξήγηση των μέτρων, το accuracy εκφράζει το ποσοστό του συνολικού αριθμού των προβλέψεων της δυαδικής ταξινόμησης που είναι σωστές (τόσο True Positives όσο και True Negatives), υποδεικνύοντας έτσι ουσιαστικά το βαθμό εγγύτητας των αποτελεσμάτων μέτρησης με την αληθινή τιμή. Κατ΄ αυτόν τον τρόπο, ένα accuracy 100%, σημαίνει ότι όλες οι τιμές

που προβλέφθηκαν από τον αλγόριθμο είναι ακριβώς οι ίδιες με τις πραγματικές τιμές. Χρησιμοποιώντας τον Πίνακα 5.1, το accuracy αναπαρίσταται από τον παρακάτω τύπο:

$$\begin{aligned}\text{Accuracy} =& \frac{\text{no. of Correctly Classified Documents}}{\text{Total no. of Documents}} \\ =& \frac{\text{True Positives} + \text{True Negatives}}{\text{True Pos} + \text{False Pos} + \text{False Neg} + \text{True Neg}}\end{aligned} \tag{5.6}$$

Επειδή οι κλάσεις που έχουμε είναι σταθμισμένες (οι θετικές κριτικές του dataset είναι ίσες με τις αρνητικές), το accuracy ενδείκνυται ως μετρική. Αυτός είναι και ο λόγος που χρησιμοποιείται εκτενέστερα για το συγκεκριμένο πεδίο. Ωστόσο, επιλέξαμε να ελέγξουμε την απόδοση του αλγορίθμου και με δύο επιπλέον τρόπους, έτσι ώστε να έχουμε μια πιο ολοκληρωμένη εικόνα. Άλλωστε, υπάρχει και το λεγόμενο παράδοξο του accuracy (accuracy paradox), σύμφωνα με το οποίο μοντέλα πρόβλεψης με συγκεκριμένο ποσοστό accuracy, ίσως να έχουν μεγαλύτερη ισχύ πρόβλεψης από άλλα μοντέλα με υψηλότερο ποσοστό.

Συνεχίζοντας λοιπόν, το precision εκφράζει το ποσοστό των θετικών προβλέψεων που ήταν σωστές, δηλαδή βρίσκει απ΄ όλες τις κριτικές που ταξινομήθηκαν ως θετικές πόσες ήταν τελικά όντως θετικές. Όπως είναι φυσικό, μας ενδιαφέρει αυτή η μετρική να έχει υψηλή τιμή, αφού κάτι τέτοιο σημαίνει πως η πρόβλεψή μας ότι η κριτική της ταινίας εκφράζει θετικό συναίσθημα ήταν ακριβής. Πιο ξεκάθαρα, χρησιμοποιώντας ξανά τον Πίνακα 5.1 είναι:

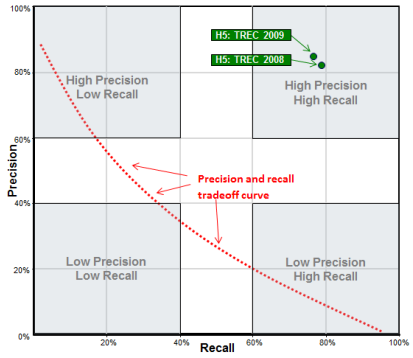
$$\text{Precision} = \frac{\text{True Positives}}{\text{no. of predicted Positives}} = \frac{\text{True Positives}}{\text{True Pos} + \text{False Pos}} \tag{5.7}$$

Το recall από την άλλη, μας δείχνει το ποσοστό των θετικών περιπτώσεων που αναγνωρίστηκαν ως σωστές, δηλαδή από το σύνολο των κριτικών που είναι πραγματικά θετικές, για πόσες από αυτές ο αλγόριθμος προέβλεψε και ταξινόμησε ορθώς πως εκφράζουν θετικό συναίσθημα. Συνεπώς, και αυτό το ποσοστό θα θέλαμε να είναι όσο το δυνατόν υψηλότερο, μιας και καθορίζει το πόσο καλός είναι ο ταξινομητής στην ανίχνευση των θετικών κριτικών. Έτσι, από τον Πίνακα 5.1, αυτή τη φορά θα είναι:

$$\text{Recall} = \frac{\text{True Positives}}{\text{no. of actual Positives}} = \frac{\text{True Positives}}{\text{True Pos} + \text{False Neg}} \tag{5.8}$$

Συνήθως, ένας αλγόριθμος πετυχαίνει να έχει είτε υψηλό precision και χαμηλό recall, είτε το ανάποδο. Αυτό σημαίνει πως μία προσπάθεια να βελτιώσουμε τον έναν παράγοντα, οδηγεί στη μείωση του άλλου και έτσι υπάρχει το λεγόμενο tradeoff μεταξύ precision και recall. Το βέλτιστο αποτέλεσμα, δηλαδή υψηλό precision και υψηλό recall, είναι γενικά δύσκολο να επιτευχθεί. Τα

σημεία της Εικόνας 5.2, που προέρχεται από ένα Text REtrieval Conference (TREC), αναπαριστούν ταυτόχρονα το precision και recall που επιτυγχάνεται από ένα σύστημα ταξινόμησης, τα οποία εκφράζουν και τη γενική περίπτωση. Το tradeoff που περιγράφτηκε παραπάνω αποτυπώνεται με την κόκκινη διακεκομμένη γραμμή που έχει φορά προς τα κάτω. Ο στόχος είναι να πετύχουμε αποτελέσματα που βρίσκονται μέσα στο γκρίζο κουτί που βρίσκεται πάνω δεξιά, έτσι ώστε να είμαστε περισσότερο σίγουροι πως ο classifier δεν «κλέβει», ταξινομεί με ακρίβεια και ο αλγόριθμός μας τα καταφέρνει καλά. Έτσι, θα έχουμε καταφέρει να βρούμε σχεδόν όλες τις σχετικές κριτικές που θέλουμε (High Recall), και να έχουμε απ΄ όλες αυτές τα λιγότερα δυνατά False Positives (High Precision).



Σχήμα 5.2: Σχέση μεταξύ Precision-Recall (Εικόνα από 2008 και 2009 TREC Studies)

Τέλος, το F_1 score, είναι ένα μέτρο απόδοσης που συνδυάζει τις δύο προηγούμενες μετρικές, αξιολογώντας την clustering ποιότητα του αλγορίθμου. Στην ουσία αποτελεί ένα σταθμισμένο μέσο όρο (αρμονικός μέσο) των precision και recall, με την τιμή του να βρίσκεται συνήθως κάπου ανάμεσα στην τιμή που παίρνει το precision και σε αυτή του recall. Η καλύτερη δυνατή τιμή που μπορεί να πάρει είναι το 1 και η χειρότερη το 0. Δίνεται από τον παρακάτω τύπο:

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5.9}$$

Το συνολικό accuracy, precision και recall για κάθε πείραμα, προκύπτει απευθείας από το υπολογισμό του μέσου όρου πάνω σε όλα τα folds (macro-average method - ενδεδειγμένο αφού θέλουμε να δούμε πως επιδρά το σύστημα πάνω στο σύνολο των δεδομένων). Τα ποσοστά πάλι για το F_1 score όπως παρουσιάζονται στο κεφάλαιο που ακολουθεί, είναι αποτέλεσμα της εφαρμογής του τύπου (5.9) χρησιμοποιώντας τις τελικές τιμές από τα precision και recall που έχουν ήδη βρεθεί.

Κεφάλαιο 6

Πειράματα

6.1 Εγκατάσταση

Υλοποιήσαμε ένα πρόγραμμα στην Python 2.7.3, το οποίο υλοποιεί τα πειράματα έτσι όπως παρουσιάστηκαν στο Κεφάλαιο 3 και 4, ενώ εκτιμά την απόδοση του μοντέλου με τις μετρικές από το Κεφάλαιο 5. Πιο συγκεκριμένα, χρησιμοποιήσαμε το NLTK 2.0.4 για να κάνουμε Porter και Lancaster stemming άλλα και WordNet lemmatization. Για την εκτίμηση των παραμέτρων (weights, means, covars) με τον EM αλγόριθμο για τα training δεδομένα, χρησιμοποιήθηκε η βιβλιοθήκη scikit-learn 0.13.1. Επίσης, για την εφαρμογή της μεθόδου LSI χρησιμοποιήθηκε το gensim, ενώ έγινε εκτεταμένη χρήση των βιβλιοθηκών της python, numpy 1.6.1 και scipy 0.9.0, για πράξεις γραμμικής άλγεβρας μεταξύ των πινάκων.

Όλα τα πειράματα έτρεξαν σε ένα laptop με Ubuntu Linux 12.04 και έναν υπολογιστή του εργαστηρίου με λειτουργικό Fedora 18.

6.2 Προεπεξεργασία των Δεδομένων για την Εκπαίδευση του Μοντέλου

Για τη δημιουργία ενός dataset με ομοιόμορφη κατανομή κλάσεων, επιλέξαμε να χωρίσουμε το training set σε 10 ισομεγέθη folds, εκπαιδεύοντας τον αλγόριθμο ισάριθμες φορές. Την πρώτη φορά ο αλγόριθμος εκπαιδεύεται πάνω στα τελευταία 9 folds και επικυρώνεται πάνω στο πρώτο. Τη δεύτερη φορά, ο αλγόριθμος επικυρώνεται πάνω στο δεύτερο fold και εκπαιδεύεται πάνω στα υπόλοιπα 9. Επαναλαμβάνουμε αυτή τη διαδικασία 10 συνολικά φορές, ενώ επικυρώνουμε κάθε fold μία μόνο φορά. Συνεπώς, κάθε κείμενο στο training set χρησιμοποιείται για επικύρωση μία φορά. Αυτή η διαδικασία είναι ευρέως γνωστή ως 10-fold cross validation, και κατ΄ αυτόν τον τρόπο καταφέρνουμε ο αλγόριθμος να μπορεί να γενικευτεί σε περιπτώσεις που δεν έχει ξανασυναντήσει, ενώ τα αποτελέσματα που παίρνουμε είναι αμερόληπτα και άμεσα συγκρίσιμα με τη βιβλιογραφία που υπάρχει.

Για την προετοιμασία των κειμένων (pre-processing), δοκιμάστηκε stemming με τους αλγορίθμους που περιγράφτηκαν παραπάνω στο Κεφάλαιο 3. Επίσης, δεν εφαρμόστηκε κάποιας μορφής standardization (αφαίρεση της μέσης τιμής και διαίρεση με την τυπική απόκλιση για να αποκτήσει η τυχαία μεταβλητή τη μορφή τυπικής κανονικής κατανομής $\mathcal{N}(0,1)$) ή normalization (διαίρεση ενός διανύσματος με τη νόρμα του) πάνω στα αρχικά δεδομένα εισόδου.

6.3 Δεδομένα και Σημείο Εκκίνησης

Όπως έχει αναφερθεί και παραπάνω, χρησιμοποιήθηκε το polarity dataset version 2.0[1], που προτάθηκε από τους Pang και Lee το 2004 [5]. Η συλλογή απαρτίζεται από 2000 κριτικές ταινιών γραμμένες από 312 συγγραφείς, όπου κάθε κριτική σχετίζεται με μια δυαδική ταμπέλα συναισθήματος (έχουμε δηλαδή 1000 θετικά και 1000 αρνητικά αξιολογημένες κριτικές). Οι κριτικές πάρθηκαν από την επίσημη βάση δεδομένων της Internet Movie Database (IMDb). Το μέσο μήκος ενός κειμένου είναι 3,893 χαρακτήρες (755 λέξεις), με το ελάχιστο να είναι μόλις 91 και το μέγιστο να φτάνει τους 14,957 χαρακτήρες. Οι διακριτές λέξεις που υπάρχουν είναι συνολικά 48,205.

Η ταξινόμηση των κριτικών από ταινίες παρουσιάζει ιδιαίτερο ενδιαφέρον, αφενός γιατί είναι μια χρήσιμη υπηρεσία που ενδιαφέρει πολύ κόσμο (κάτι που υποδεικνύει η μεγάλη επισκεψιμότητα του site της IMDb), και αφετέρου γιατί είναι προφανώς ένα πολύ δυσκολότερο πεδίο σε σχέση με την ταξινόμηση κριτικών από άλλα προϊόντα, όπως βρήκε ο Turney [2]. Άλλωστε, η προτίμηση ή όχι μιας ταινίας, είναι ίσως το πιο χαρακτηριστικό παράδειγμα που εκφράζει τις διαφορές στις προτιμήσεις και τον χαρακτήρα των ανθρώπων. Επίσης, είναι θετικό το γεγονός ότι αυτός που γράφει την κριτική, είναι ο ίδιος άνθρωπος ο οποίος αναθέτει μία θετική ή αρνητική αξιολόγηση και κατ' αυτόν τον τρόπο έχουμε μια στενή σχέση ανάμεσα στη ταξινόμηση και το κείμενο. Πέρα από αυτά όμως, το συγκεκριμένο dataset, έχει χρησιμοποιηθεί στις περισσότερες δουλειές που έχουν να κάνουν με sentiment analysis. Γι' αυτό το προτιμήσαμε και εμείς, σε μία προσπάθεια για άμεση σύγκριση με τα διακριτά μοντέλα που έχουν χρησιμοποιηθεί μέχρι σήμερα σε αντίστοιχες προηγούμενες μελέτες.

Ως αφετηρία για τα πειράματά μας (baseline), θεωρήσαμε μια υλοποίηση που κάναμε στο διακριτό χώρο, όπου τα δεδομένα χωρίστηκαν με 10-fold cross validation, το λεξικό που χρησιμοποιήθηκε ήταν οι λέξεις του corpus χωρίς stemming και ως classifier εφαρμόστηκε ο Naive Bayes με Laplace smoothing (a.k.a. add-one smoothing). Για να βρούμε την απαραίτητη δεσμευμένη πιθανότητα για τον Naive Bayes σύμφωνα με τη Σχέση 5.2, βρίσκουμε αρχικά πόσες φορές κάθε λέξη του test document υπάρχει στα positive/negative training δεδομένα και διαιρούμε με το συνολικό αριθμό των θετικών/αρνητικών λέξεων. Στον παρονομαστή προσθέτουμε επίσης και το μέγεθος του λεξικού. Η αντίστοιχη prior πιθανότητα για κάθε κλάση βρίσκεται υπολογίζοντας τον α-

[1] http://www.cs.cornell.edu/people/pabo/movie-review-data/

ριθμό των θετικών/αρνητικών κειμένων, προς το συνολικό αριθμό των κριτικών που υπάρχουν στα δεδομένα. Ταξινομούμε στην πιο πιθανή κλάση σύμφωνα με την μεγαλύτερη πιθανότητα που προέκυψε από το άθροισμα (αφού δουλεύουμε σε λογαριθμική κλίμακα) της prior και της δεσμευμένης πιθανότητας. Με αυτές τις συνθήκες παρήχθησαν τα ακόλουθα αποτελέσματα:

Naive Bayes	*Accuracy (%)*	F_1 *score (%)*
Without Filtering Stopwords	**0.817**	**0.813**
Filtering Stopwords	0.811	0.808

Πίνακας 6.1: Απόδοση που λήφθηκε ως baseline

Παρατηρούμε ότι η απόδοση με ή χωρίς τη χρήση των stopwords είναι πολύ κοντά. Ωστόσο, το αξιοσημείωτο είναι πως ενώ η αφαίρεση των stopwords σε άλλους τομείς μπορεί να γίνει άφοβα, εδώ μειώνει (έστω και ελάχιστα) την απόδοση. Αυτό αποδίδεται στο γεγονός ότι ορισμένα από αυτά, όπως η άρνηση (π.χ. «not»), καταδεικνύουν συναίσθημα και έτσι μπορούν να βοηθήσουν στην ταξινόμηση. Συμπεραίνουμε ακόμη μια φορά λοιπόν, πως η ανάλυση συναισθημάτων αποτελεί μία ξεχωριστή περίπτωση ταξινόμησης, όπου ακόμη και αυτές οι συχνά χρησιμοποιούμενες λέξεις μέσα στα κείμενα μπορούν να παίξουν κάποιο ρόλο. Τα stopwords που χρησιμοποιήθηκαν ήταν συνολικά 571 λέξεις, που αντιστοιχούσαν κυρίως σε άρθρα και συνδέσμους. Παρόμοια ερμηνεία έχει και η χρήση των σημείων στίξης, καθώς και αυτά είναι ενδεικτικά του συναισθήματος (π.χ. «!»), γι΄ αυτό και δεν πρέπει τελικά να αφαιρούνται.

Ακολουθούν τα αποτελέσματα της ταξινόμησης με βάση τον Boolean Multinomial Naive Bayes αυτήν τη φορά:

Boolean Multinomial NB	Accuracy (%)	F_1 score (%)
Without Filtering Stopwords	0.820	0.807
Filtering Stopwords	**0.828**	**0.825**

Πίνακας 6.2: Απόδοση χρησιμοποιώντας Boolean Multinomial Naive Bayes

Όπως αναμενόταν σύμφωνα με τη θεωρία που παραθέσαμε στην Ενότητα 5.2, η παραλλαγή αυτή του Naive Bayes classifier οδηγεί σε (ελαφρώς) καλύτερα αποτελέσματα σε σχέση με το baseline. Σε αντίθεση όμως με προηγουμένως, παρατηρούμε μία αντιστροφή των ρόλων, αφού εδώ η καλύτερη απόδοση επιτυγχάνεται φιλτράροντας τις λέξεις του Opinion Lexicon με τη λίστα των stopwords που χρησιμοποιείται. Αυτό συμβαίνει, γιατί η μεμονωμένη εμφάνιση μιας stopword στο test document, δε μπορεί να προσφέρει κάποια επιπλέον χρήσιμη πληροφορία όπως πριν, όπου φάνηκε ότι ο συνολικός αριθμός των stopwords συνεισφέρει στην ταξινόμηση.

6.4 Παρουσίαση Αποτελεσμάτων και Σχολιασμός

Ξεκινάμε την παρουσίαση των αποτελεσμάτων με τους μετασχηματισμούς των document-term πινάκων που παρουσιάστηκαν στο Κεφάλαιο 3, για να διαπιστώσουμε εάν μπορούν να προσθέσουν βάρος στους πραγματικά σημαντικούς όρους. Μιας και οι διαστάσεις των πινάκων παραμένουν αμετάβλητες, εφαρμόζουμε αρχικά stemming με έναν από τους αλγορίθμους που αναφέρθηκαν (Lancaster, Porter, ή WordNet), έτσι ώστε μειωθούν τα διανύσματα των λέξεων και ο χρόνος εκτέλεσης.

Τα αποτελέσματα παρουσιάζονται ανά ομάδες, σύμφωνα με το βαθμό του stemming και τους πίνακες συνδιασποράς που χρησιμοποιήθηκαν κάθε φορά.

Ακολουθούν τα αποτελέσματα της χρησιμοποίησης διαγώνιων πινάκων συνδιακύμανσης, πρώτα για Lancaster και στη συνέχεια για Porter και WordNet stemming:

	Accuracy (%)			F_1 score (%)		
Weighting	2 comp	3 comp	5 comp	2 comp	3 comp	5 comp
Binary	0.681	0.680	0.682	0.652	0.652	0.652
Tf	0.684	0.684	**0.685**	0.663	0.664	0.666
Log	0.683	0.683	0.682	0.660	0.660	0.659
Augnorm	0.671	0.672	0.667	**0.739**	0.738	0.733
GfIdf	0.652	0.651	0.654	0.565	0.563	0.567
Idf	0.526	0.531	0.531	0.177	0.199	0.196
Entropy	0.547	0.543	0.667	0.686	0.686	0.683

Πίνακας 6.3: Accuracy και F_1 score για διαγώνιους πίνακες συνδιασποράς χρησιμοποιώντας local και global συναρτήσεις για Lancaster stemming [2]

	Accuracy (%)			F_1 score (%)		
Weighting	2 comp	3 comp	5 comp	2 comp	3 comp	5 comp
Binary	0.691	0.692	0.692	0.662	0.663	0.664
Tf	0.688	0.689	0.690	0.669	0.670	0.672
Log	0.720	0.721	0.720	0.685	0.686	0.685
Augnorm	0.760	0.763	**0.765**	0.765	0.769	**0.771**
GfIdf	0.632	0.631	0.631	0.523	0.522	0.522
Idf	0.542	0.542	0.542	0.214	0.212	0.212
Entropy	0.717	0.707	0.709	0.750	0.747	0.748

Πίνακας 6.4: Accuracy και F_1 score για διαγώνιους πίνακες συνδιασποράς χρησιμοποιώντας local και global συναρτήσεις για Porter stemming [2]

[2] Ο Normal μετασχηματισμός ταξινομούσε στην τύχη, γι' αυτό και δεν παρουσιάζεται στο συγκεκριμένο πίνακα

	Accuracy (%)			F_1 score (%)		
Weighting	2 comp	3 comp	5 comp	2 comp	3 comp	5 comp
Binary	0.707	0.706	0.705	0.683	0.682	0.682
Tf	0.701	0.701	0.702	0.684	0.685	0.686
Log	0.730	0.730	0.731	0.694	0.694	0.695
Augnorm	0.681	0.681	0.683	0.571	0.571	0.576
Normal	0.553	0.553	0.551	0.365	0.387	0.354
GfIdf	0.635	0.634	0.633	0.520	0.519	0.517
Idf	0.537	0.537	0.537	0.186	0.186	0.186
Entropy	0.749	**0.749**	0.748	0.759	**0.789**	0.761

Πίνακας 6.5: Accuracy και F_1 score για διαγώνιους πίνακες συνδιασποράς χρησιμοποιώντας local και global συναρτήσεις για WordNet lemmatizer

Σαν μια γενική παρατήρηση, θα λέγαμε ότι η απόδοση δε φαίνεται να επηρεάζεται πολύ από τον αριθμό των components, καθώς οι διαφορές είναι αμελητέες (πάντα μέσα στα όρια του στατιστικού σφάλματος). Επίσης, στους περισσότερους από τους μετασχηματισμούς που δοκιμάστηκαν, παρατηρείται μια βαθμιαία αύξηση των ποσοστών που έχει να κάνει με τον αλγόριθμο του stemming που εφαρμόζεται. Πιο συγκεκριμένα, η αύξηση αυτή μεταφράζεται από τις ολοένα και περισσότερες λέξεις που χρησιμοποιούνται ως λεξικό. Εξαίρεση βέβαια αποτελούν οι μετασχηματισμοί Augnorm, Idf και GfIdf, καθώς πετυχαίνουν το καλύτερο accuracy για Porter οι δύο πρώτοι και Lancaster stemming ο τρίτος αντίστοιχα. Πάντως, και οι τρεις αυτοί μετασχηματισμοί έχουν χειρότερες αποδόσεις συγκριτικά με τους υπόλοιπους.

Συνεχίζουμε παρουσιάζοντας τα αποτελέσματα των πειραμάτων από τη χρήση full πινάκων συνδιακύμανσης. Λόγω του αυξημένου χρόνου εκτέλεσης αυτή τη φορά, υποθέτουμε πως τα διανύσματα των λέξεων μπορούν να μοντελοποιηθούν σε ικανοποιητικό βαθμό από ένα GMM με τρία components.

Weighting	Accuracy (%)	F_1 score (%)
Binary	0.755	0.738
Tf	0.738	0.752
Log	**0.772**	**0.770**
Augnorm	0.686	0.741
Normal	0.600	0.467
GfIdf	0.708	0.713
Idf	0.614	0.580
Entropy	0.705	0.726

Πίνακας 6.6: Accuracy και F_1 score για full πίνακες συνδιασποράς χρησιμοποιώντας local και global συναρτήσεις για Lancaster stemmer με 3 components

Weighting	Accuracy (%)	F_1 score (%)
Binary	0.784	0.770
Tf	0.751	0.756
Log	**0.789**	**0.783**
Augnorm	0.762	0.773
Normal	0.592	0.464
GfIdf	0.710	0.705
Idf	0.622	0.573
Entropy	0.708	0.716

Πίνακας 6.7: Accuracy και F_1 score για full πίνακες συνδιασποράς χρησιμοποιώντας local και global συναρτήσεις για Porter stemmer με 3 components

Weighting	Accuracy (%)	F_1 score (%)
Binary	**0.804**	**0.788**
Tf	0.767	0.760
Log	0.796	0.782
Augnorm	0.726	0.672
Normal	0.615	0.515
GfIdf	0.711	0.690
Idf	0.647	0.677
Entropy	0.739	0.732

Πίνακας 6.8: Accuracy και F_1 score για full πίνακες συνδιασποράς χρησιμοποιώντας local και global συναρτήσεις για WordNet με 3 components

Σύμφωνα με τους Πίνακες 6.6, 6.7 και 6.8, συμπεραίνουμε ότι τα GMMs με full covariance matrices πετυχαίνουν καλύτερες αποδόσεις σε σχέση με αυτές που παρατηρήθηκαν για διαγώνιους πίνακες. Έτσι, για όλους τους μετασχηματισμούς που πειραματιστήκαμε (εκτός από την εντροπία), τα ποσοστά είναι βελτιωμένα. Επίσης, και εδώ τα αποτελέσματα γίνονται σταδιακά καλύτερα, ακολουθώντας το βαθμό του stemming (με εξαίρεση πάλι τον Augnorm και τον Normal μετασχηματισμό, οι οποίοι έχουν καλύτερη απόδοση χρησιμοποιώντας Porter και Lancaster stemmer αντίστοιχα). Ο αλγόριθμος του stemming, παίζει το σημαντικότερο ρόλο και στο χρόνο εκτέλεσης του προγράμματος, για την ολοκλήρωση του οποίου απαιτούνται περίπου 7.5h με τον Lancaster, 12h με τον Porter και 31h με τον WordNet αλγόριθμο για τον binary μετασχηματισμό.

Για την ολοκλήρωση της εικόνας της κατάστασης, δε μένει παρά να παραθέσουμε και τα αποτελέσματα που προκύπτουν χωρίς κάποιον αλγόριθμο που να μειώνει το μέγεθος του λεξικού αυτή τη φορά. Παρακάτω παρουσιάζονται λοιπόν σε αντίστοιχους πίνακες με προηγουμένως, τα αποτελέσματα των μετρικών απόδοσης χρησιμοποιώντας όλες τις λέξεις (και τις 6786) του Opinion Lexicon ξεκινώντας από διαγώνιους πίνακες συνδιακύμανσης:

	Accuracy (%)			F_1 score (%)		
Weighting	2 comp	3 comp	5 comp	2 comp	3 comp	5 comp
Binary	0.707	0.707	0.707	0.678	0.677	0.678
Tf	0.712	0.713	0.713	0.692	0.692	0.692
Log	0.711	0.711	0.711	0.686	0.692	0.687
Augnorm	0.692	0.692	0.689	0.593	0.594	0.587
Normal	0.548	0.549	0.545	0.343	0.341	0.338
GfIdf	0.632	0.632	0.631	0.511	0.515	0.513
Idf	0.537	0.538	0.537	0.172	0.177	0.173
Entropy	0.751	**0.785**	0.747	**0.761**	0.7593	0.7589

Πίνακας 6.9: Accuracy και F_1 score για διαγώνιους πίνακες συνδιασποράς χρησιμοποιώντας local και global συναρτήσεις για όλες τις λέξεις του Opinion Lexicon

	Accuracy (%)	F_1 score (%)
Binary	**0.805**	0.782
Tf	0.772	0.766
Log	0.800	**0.788**
Augnorm	0.757	0.723
Normal	0.607	0.489
GfIdf	0.722	0.702
Idf	0.661	0.693
Entropy	0.751	0.742

Πίνακας 6.10: Accuracy και F_1 score για full πίνακες συνδιασποράς χρησιμοποιώντας local και global συναρτήσεις με 3 components για όλες τις λέξεις του Opinion Lexicon

Στον Πίνακα 6.10, φαίνεται η καλύτερη συνολικά απόδοση που επιτύχαμε μέχρι στιγμής φτάνοντας το 80.5% για accuracy και το 78.2% για F_1 score, δαπανώντας όμως περίπου 35h. Το ποσοστό αυτό, είναι αποτέλεσμα του δυαδικού μετασχηματισμού χρησιμοποιώντας όλες τις λέξεις του λεξικού, κάτι που συμφωνεί με τους Pang και Lee [3], οι οποίοι είχαν και αυτοί το καλύτερο accuracy με binary features. Υψηλό ποσοστό της τάξης του 80%, επιτυγχάνεται επίσης και με τη λογαριθμική μετατροπή.

Τα επόμενα πειράματα αφορούν τα αποτελέσματα από τη μείωση των διανυσμάτων χρησιμοποιώντας SVD. Να σημειωθεί, πως σε αυτήν την περίπτωση δε χρησιμοποιήθηκε καμία μορφή stemming, που σημαίνει ότι όλες οι λέξεις του λεξικού παρέμειναν ως είχαν στην αρχική τους μορφή. Ο Πίνακας 6.11 λοιπόν απεικονίζει το accuracy και το F_1 score για full, tied, diagonal και spherical πίνακες συνδιακύμανσης με αυτήν την προσέγγιση, ανάλογα με τον αριθμό των components.

Όπως παρατηρούμε, τα αποτελέσματα δεν είναι όσο καλά αναμένονταν. Αυ-

τό οφείλεται στο γεγονός ότι τελικά, με τη μείωση των διαστάσεων, δεν καταφέραμε να απαλείψουμε μόνο θόρυβο που ήταν το ζητούμενο, αλλά χάσαμε πολλά από τα χρήσιμα διαχωριστικά χαρακτηριστικά του συναισθήματος. Το καλύτερο accuracy φτάνει το 77%, με τον αντίστοιχο χρόνο εκτέλεσης να είναι τώρα 15min περίπου.

	Accuracy (%)		F_1 score (%)	
Cov Type	3 comp	5 comp	3 comp	5 comp
Full	0.683	0.696	0.720	0.710
Tied	0.766	0.763	0.763	0.759
Diagonal	0.702	0.703	0.754	0.758
Spherical	0.767	0.752	0.793	0.779
Cov Type	7 comp	10 comp	7 comp	10 comp
Full	0.703	0.705	0.710	0.714
Tied	0.769	**0.770**	0.767	**0.767**
Diagonal	0.718	0.713	0.760	0.754
Spherical	0.737	0.728	0.769	0.760

Πίνακας 6.11: Accuracy και F_1 score (%) χρησιμοποιώντας SVD για full, tied, diagonal και spherical πίνακες συνδιασποράς - 300 factors

Για την ανάλυση των αποτελεσμάτων με τις μεθόδους feature selection που δοκιμάστηκαν, παραθέτουμε αρχικά για κάθε μέθοδο τις λέξεις συναισθήματος που επιλέγονται από τον κάθε αλγόριθμο ως οι πιο αντιπροσωπευτικές για το διαχωρισμό των κλάσεων και στη συνέχεια τα αντίστοιχα ποσοστά της απόδοσης που πετυχαίνει για τον καταλληλότερο αριθμό features που βρέθηκε. Ο αριθμός των features επιλέχθηκε μετά από πειραματισμό και κυμαίνεται μεταξύ 50 και 350.

Positive Feats	Negative Feats
like	plot
good	bad
well	funny
best	hard
great	unfortunately
work	problem
love	worst
right	wrong
better	boring
enough	dead

Πίνακας 6.12: Z score top features

Positive Feats	Negative Feats
best	bad
great	worst
perfect	plot
outstanding	stupid
wonderful	boring
well	ridiculous
wonderfully	waste
perfectly	awful
memorable	wasted
excellent	lame

Πίνακας 6.13: Delta Tf-Idf top features

Αναλύοντας τη feature list παραπάνω για καθεμία από τις δύο κατηγορίες, βλέπουμε ότι όροι όπως «work», «enough» ή «plot», δε φέρουν από μόνες τους κάποιο έντονο συναίσθημα. Ωστόσο, υπάρχουν στο λεξικό που χρησι-

# comp.	Accuracy	F_1 score
3	0.712	0.742
5	0.709	0.711
7	0.712	0.713
10	**0.714**	**0.718**

Πίνακας 6.14: Accuracy και F_1 score για Z score - 300 features, spherical cov matrices

# comp.	Accuracy	F_1 score
3	**0.805**	**0.801**
5	0.804	0.799
7	0.804	0.800
10	0.803	0.800

Πίνακας 6.15: Accuracy και F_1 score για Delta Tf-Idf - 300 features, spherical cov matrices

μοποιούμε, οπότε αφήνουμε το μοντέλο να συμπεριλάβει αυτούς τους όρους. Άλλωστε, αν η συχνότητα αυτών των όρων είναι υψηλή και στις δύο κατηγορίες, τα scores τους δε θα επηρεάσουν σε μεγάλο βαθμό το accuracy. Η παρουσία λέξεων που δε φαίνονται με μια πρώτη ματιά να είναι πολύ αντιπροσωπευτικές του συναισθήματος, οφείλεται κατά τη γνώμη μας σε δύο λόγους. Ο πρώτος λόγος έχει να κάνει με το γεγονός ότι ορισμένες από αυτές αναπαριστούν μέρος φρασεολογικών εκφράσεων που χρησιμοποιούνται πολύ συχνά σε μια συγκεκριμένη κατηγορία. Δεύτερον, αφού χρησιμοποιούμε τον υπολογισμό στατιστικών scores κατά την εκπαίδευση του μοντέλου πάνω στις δύο κλάσεις, μπορεί να τύχει στοιχεία σε μια κατηγορία να λάβουν υψηλότερα scores, ακόμη και αν το ποσοστό όλων των συχνοτήτων που κατανέμεται πάνω στις δύο κατηγορίες είναι το ίδιο.

Το Z Score δε τα καταφέρνει πολύ καλά με την ταξινόμηση του συναισθήματος, αφού το accuracy που προκύπτει μόλις που ξεπερνά το 71%. Από την άλλη πάντως, φαίνεται ξεκάθαρα η «δύναμη» του μετασχηματισμού Delta Tf-Idf, ο οποίος είναι ειδικευμένος για sentiment analysis πετυχαίνοντας μία απόδοση πάνω από 80% με μόνο 300 features σε χρόνο περίπου 10min. Η απόδοση αυτή ήταν η υψηλότερη που πετύχαμε λίγο πριν χρησιμοποιώντας όλες τις λέξεις του λεξικού και έχοντας φυσικά πολλαπλάσιο υπολογιστικό κόστος.

Παρακάτω φαίνονται τα αποτελέσματα βάση των feature selection μεθόδων χ^2 και mRMR:

Top Feats
worst
wasted
waste
stupid
ridiculous
lame
boring
bad
awful
outstanding

Πίνακας 6.16: χ^2 top features

Top Feats
bad
worst
boring
outstanding
ridiculous
wasted
waste
stupid
awful
wonderful

Πίνακας 6.17: mRMR top features

# comp.	Accuracy	F_1 score
3	0.767	0.748
5	**0.773**	**0.764**
7	0.771	0.761
10	0.765	0.756

Πίνακας 6.18: Accuracy και F_1 score για χ^2 - 50 features, spherical cov matrices

# comp.	Accuracy	F_1 score
3	**0.790**	**0.786**
5	0.789	0.786
7	0.787	0.786
10	0.786	0.784

Πίνακας 6.19: Accuracy και F_1 score για mRMR - 50 features, spherical cov matrices

Το mRMR αποδεικνύεται και στην περίπτωση μας, μία ισχυρή μέθοδος επιλογής των features που περικλείουν την περισσότερο χρήσιμη πληροφορία, αφού καταφέρνει με μόλις 50 features να διακρίνει τόσο καλά το συναίσθημα, ώστε να φτάσει σε μια απόδοση του 79%. Το χ^2 πάλι, έχει χαμηλότερη απόδοση, με το τελικό accuracy να είναι γύρω στο 77%.

Ακολουθούν τα αποτελέσματα από τις δύο τελευταίες μεθόδους feature selection που εφαρμόστηκαν:

Positive Feats	Negative Feats
perfect	bad
best	worst
great	boring
wonderful	stupid
perfectly	plot
strong	ridiculous
memorable	waste
effective	awful
hilarious	unfortunately
brilliant	wasted

Πίνακας 6.20: CPD top features

Positive Feats	Negative Feats
perfect	waste
best	plot
strong	boring
works	bad
memorable	unfortunately
effective	stupid
brilliant	awful
excellent	ridiculous
hilarious	worse
great	worst

Πίνακας 6.21: CPPD top features

# comp.	Accuracy	F_1 score
3	**0.805**	**0.797**
5	0.792	0.788
7	0.795	0.792
10	0.788	0.784

Πίνακας 6.22: Accuracy και F_1 score για CPD - 200 features, spherical cov matrices

# comp.	Accuracy	F_1 score
3	0.822	0.821
5	0.823	0.822
7	**0.827**	**0.826**
10	0.824	0.825
Baseline	0.817	
P. & L. [5]	0.862[3]	
W. et al. [10]	0.902	
An. & K. [19]	0.775	

Πίνακας 6.23: Accuracy και F_1 score για CPPD - 350 features, spherical covariance matrices

Η CPPD λοιπόν, φάνηκε πως είναι και στην πράξη μια καλύτερη μέθοδος για sentiment analysis από την προγενέστερη CPD, που πετυχαίνει όμως και αυτή ένα υψηλό ποσοστό απόδοσης. Στο κάτω μέρος του τελευταίου πίνακα που περιέχει τα καλύτερα αποτελέσματα που επιτεύχθηκαν σε αυτή τη διπλωματική εργασία χρησιμοποιώντας τη μέθοδο CPPD, τα οποία συνδυάζονται μάλιστα με το μικρότερο χρόνο εκτέλεσης ($\approx$ 5min), σημειώνουμε και τις αντίστοιχες αποδόσεις από το baseline μας στο διακριτό χώρο, καθώς και από προγενέστερες δουλειές που χρησιμοποιούν το ίδιο dataset. Η πρώτη από αυτές, αποτελεί τη δουλειά ορόσημο για το συγκεκριμένο τομέα από τους Pang/Lee, η δεύτερη πετυχαίνει σύμφωνα με αυτά που γνωρίζουμε το καλύτερο accuracy με βάση το δεδομένο dataset χρησιμοποιώντας appraisal taxonomies, ενώ η τελευταία είναι μία μελέτη που βασίζεται στον SVM αλγόριθμο ταξινόμησης και τη λεξιλογική βάση δεδομένων του WordNet.

[3]Το συγκεκριμένο ποσοστό αφορά την ταξινόμηση με Naive Bayes.

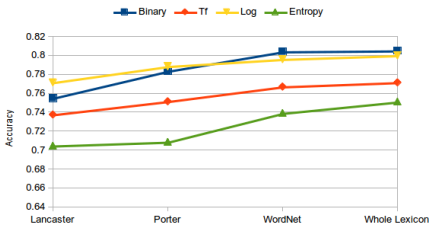
Κεφάλαιο 7

Συμπεράσματα και Προοπτικές

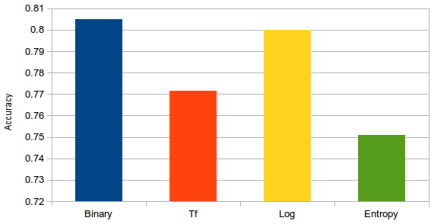
7.1 Ανασκόπηση Διπλωματικής Εργασίας

Σε αυτή τη διπλωματική εργασία παρουσιάσαμε ένα supervised μοντέλο συνεχούς χώρου που εφαρμόστηκε για να συμπεράνει την πολικότητα (θετική ή αρνητική) από κριτικές ταινιών. Επειδή οι αλγόριθμοι με τους οποίους προσπαθήσαμε να μειώσουμε το μέγεθος του λεξικού διαμορφώνουν έμμεσα το μέγεθος των feature vectors , διαδραματίζουν πρωτεύοντα ρόλο στο χρόνο που απαιτείται για την εκπαίδευση του μοντέλου. Επίσης, σαν ένα γενικό συμπέρασμα μπορούμε να πούμε ότι ο Porter και ο Lancaster stemmer καταστρέφουν πολλές από τις διαφορές των συναισθημάτων και έτσι αρκετές από τις λέξεις που ανήκουν στην μία κατηγορία (π.χ. θετική), καταλήγουν στην άλλη (δηλαδή αρνητική). Αντιθέτως, ο WordNet lemmatizer, δεν «πάσχει» από αυτό το πρόβλημα τόσο έντονα, αλλά από την άλλη δεν κάνει και αρκετές συρρικνώσεις έτσι ώστε να αξίζει τελικά από άποψη πόρων να τον χρησιμοποιήσουμε. Ο αλγόριθμος του stemming που εφαρμόζεται λοιπόν, έχει άμεση επίπτωση στα αποτελέσματα των μετρικών απόδοσης, κάτι που οφείλεται στο πλήθος των λέξεων που συγχωνεύει στην ίδια ρίζα. Το χαρακτηριστικό αυτό αποτυπώνεται μέσα από όλους τους μετασχηματισμούς που εξετάστηκαν (με εξαίρεση τον Augnnorm που δρα ιδιόρρυθμα), παρατηρώντας μία σταθερά ανοδική πορεία των ποσοστών απόδοσης: τα αποτελέσματα βελτιώνονται καθώς οδηγούμαστε από τον Lancaster, που είναι ο πιο «βίαιος» αφαιρώντας κοντά στις 3000 λέξεις του λεξικού, στον Porter που αφαιρεί 2255, έως τον πιο «μετριοπαθή» απ΄ όλους WordNet, που αφαιρεί μόλις 343 λέξεις. Ακολούθως, τα αποτελέσματα του τελευταίου, σε σχέση με αυτά που προκύπτουν από τη χρησιμοποίηση του ακατέργαστου Opinion Lexicon, δε διαφέρουν πολύ. Τα παραπάνω λεγόμενα για το accuracy της ταξινόμησης ανάλογα με τον αλγόριθμο του stemming, αποτυπώνονται στο Σχήμα 7.1α΄.

Η χρήση μετασχηματισμών που επεξεργάζονται τα γνήσια δεδομένα χωρίς



(α′) Accuracy ανάλογα με το βαθμό του stemming



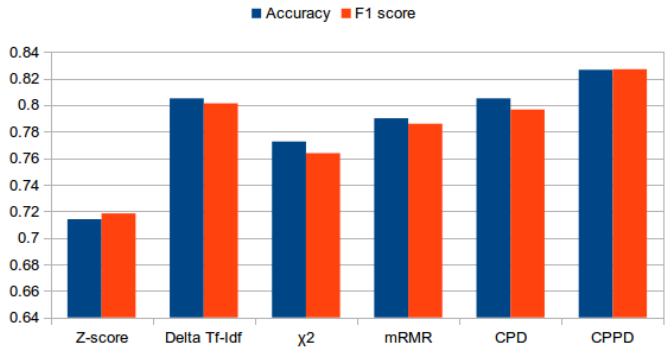
(β′) Οι 4 καλύτερες επιδόσεις accuracy που επιτεύχθηκαν

Σχήμα 7.1: Ποσοστά accuracy που προκύπτουν από τον binary, tf, log και entropy μετασχηματισμό

να μειώνονται οι διαστάσεις με κάποιον τρόπο, παρήγαγε ικανοποιητικά αποτελέσματα από πλευράς μετρικών απόδοσης για μερικούς από αυτούς. Σύμφωνα με τα πειράματά μας, την καλύτερη απόδοση είχε ο δυαδικός μετασχηματισμός, κάτι που συμφωνεί και με τους Pang και Lee [3]. Ωστόσο, η απαίτηση για πολύ μεγάλο ποσοστό μνήμης και αυξημένο χρόνο εκτέλεσης για την αποθήκευση και την επεξεργασία των δεδομένων, δεν έκανε αυτή τη λύση πολύ ελκυστική, γι′ αυτό και κατευθυνθήκαμε σε τεχνικές που μειώνουν τα διανύσματα που χρησιμοποιούμε.

Εντύπωση προκάλεσε το γεγονός ότι χρησιμοποιώντας SVD για τη μείωση των διαστάσεων και την εξαγωγή των features, δεν παρήχθησαν καλά αποτελέσματα, συγκρινόμενα με τα αυτά που έχει η ίδια προσέγγιση πάνω σε άλλους συγγενείς τομείς (topic spotting), επιβεβαιώνοντας κατ′ αυτόν τον τρόπο για άλλη μια φορά την ιδιαιτερότητα του συγκεκριμένου τομέα. Τα αρχικά διανύσματα των λέξεων περιείχαν προφανώς μεγάλο ποσοστό χρήσιμης πληροφορίας, απ′ όπου υπήρχε σημαντική απώλεια καθώς οι διαστάσεις μειωνόντουσαν. Θα πρέπει λοιπόν να είμαστε ιδιαίτερα προσεχτικοί σχετικά με την προσπάθεια μείωσης του θορύβου στον τομέα της sentiment analysis, καθώς λέξεις ή ακόμη και σημεία στίξης που σε άλλες περιοχές θεωρούνται ασήμαντη πληροφορία και μπορούν να απαλειφθούν, εδώ βοηθούν στη διάκριση των συναισθημάτων και στην καλύτερη δυνατή ταξινόμηση. Στραφήκαμε λοιπόν σε μία δεύτερη προσέγγιση που χρησιμοποιείται ευρύτατα στο πεδίο αυτό, η οποία χρησιμοποιεί μεθόδους feature selection για την επιλογή των πιο αντιπροσωπευτικών λέξεων (word vectors στην περίπτωσή μας) από κάθε κατηγορία. Αυτή ήταν και η λύση που παρήγαγε τα καλύτερα αποτελέσματα συνδυάζοντας και το μικρότερο κόστος. Αξίζει να σημειωθεί, πως τον καθοριστικότερο ρόλο για την απόδοση του μοντέλου έπαιξε η επιλογή των πινάκων συνδιασποράς. Ανάλογα με την τεχνική λοιπόν που ακολουθούσαμε κάθε φορά, επωφελούμασταν από διαφορετικό τύπο πινάκων (άλλοτε full ή tied και άλλοτε spherical), με τους διαγώνιους πίνακες συνδιακύμανσης να αποτελούν πάντα μία μέση λύση.

Μπορούμε τελικά να επωφεληθούμε από τις ιδιότητες του συνεχή χώρου, έτσι ώστε να προβλέψουμε με ακρίβεια το συναίσθημα των κριτικών από ταινίες; Η απάντηση είναι ναι σε έναν ικανοποιητικό βαθμό, αν και υστερούμε ακόμη σε σχέση με τις καλύτερες δουλειές που έχουν δημοσιευτεί στο διακριτό χώρο. Το καλύτερο ποσοστό απόδοσης που πετύχαμε, ήταν αυτό που προερχόταν από τον τη μέθοδο CPPD για feature selection και έφτασε το 82.7%, ενώ συνοδευόταν από ένα επίσης υψηλό ποσοστό F_1 score πάνω από 82%. Στο Σχήμα 7.2, αποτυπώνονται τα ποσοστά των μετρικών απόδοσης accuracy και F_1 score που επιτύχαμε χρησιμοποιώντας τις μεθόδους feature selection που μελετήθηκαν.



Σχήμα 7.2: Accuracy και F_1 score σύμφωνα με τις τεχνικές feature selection που εφαρμόστηκαν

7.2 Προεκτάσεις για Μελλοντική Έρευνα

Όπως σε κάθε ερευνητική προσπάθεια, έτσι και στη δικιά μας, δεν υπάρχει τέλος. Υπάρχουν πάντα διάφορες ιδέες και αρκετές παράμετροι που μπορούν να εφαρμοστούν και να δοκιμαστούν αντίστοιχα, με την ελπίδα ότι θα επιφέρουν κάποια βελτίωση στα αποτελέσματα. Η επέκταση της διπλωματικής λοιπόν μπορεί να γίνει σε διάφορες κατευθύνσεις.

Με βάση το υπάρχων dataset αρχικά, θα θέλαμε να πειραματιστούμε με την ταξινόμηση των κριτικών σε μια κλίμακα τριών έως πέντε αστεριών αυτή τη φορά, σε αντίθεση με το δυαδικό σχήμα ταξινόμησης που δοκιμάστηκε μέχρι στιγμής. Επίσης, μπορεί να γίνει περαιτέρω μελέτη όσο αφορά την εφαρμογή άλλων ταξινομητών (π.χ. SVMs ή Maximum Entropy), αλλά και διαφορετικών τεχνικών μείωσης διαστάσεων και περισσοτέρων μεθόδων feature selection, έτσι ώστε να δούμε τι επιπτώσεις θα έχουν τελικά στην απόδοση του συστήματος.

Πέρα από αυτά όμως, θα μπορούσαμε να ελέγξουμε την απόδοση χρησιμοποιώντας κάποιο άλλο dataset που να περιέχει μεγαλύτερο αριθμό ταινιών προς

ταξινόμηση, έτσι ώστε να έχουμε μία καλύτερη εικόνα του μοντέλου. Τέλος, μία μεγαλύτερη πρόκληση που θέτουμε, είναι να επεκτείνουμε τη δουλειά μας ως ένα σύστημα πρόβλεψης (recommender system). Στόχος του συστήματος αυτού, θα είναι να προβλέπει/προτείνει ταινίες που μπορεί να αρέσουν σε ένα συγκεκριμένο άτομο, με βάση ορισμένα κοινά χαρακτηριστικά (όπως το είδος της ταινίας ή τους ηθοποιούς που συμμετέχουν), τα οποία εμφανίζονται στις ταινίες που έχει ήδη δει, ακολουθώντας τους Sato, Anse και Tabe [18].

Βιβλιογραφία

[1] Vasileios Hatzivassiloglou and Janyce M. Wiebe, *Effects of Adjective Orientation and Gradability on Sentence Subjectivity*, Proceedings of the 18th conference on Computational linguistics - Volume 1, 2000.

[2] Peter D. Turney, *Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews*, Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, 2002.

[3] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan, *Thumbs up? Sentiment Classification using Machine Learning Techniques*, Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10, 2002.

[4] Peter D. Turney and Michael L. Littman, *Measuring Praise and Criticism: Inference of Semantic Orientation from Association*, ACM Trans. Inf. Syst., 2003.

[5] Bo Pang and Lillian Lee, *A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts*, Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, 2004.

[6] Minqing Hu and Bing Liu, *Mining and summarizing customer reviews*, Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, 2004.

[7] Tony Mullen and Nigel Collier, *Sentiment Analysis using Support Vector Machines with Diverse Information Sources*, Proceedings of Conference on Empirical Methods in Natural Language Processing, 2004.

[8] Xue Bai, Rema Padman, and Edoardo Airoldi, *On Learning Parsimonious Models for Extracting Consumer Opinions*, Proceedings of the 38th HICSS, 2005.

[9] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann, *Recognizing contextual polarity in phrase-level sentiment analysis*, Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing, 2005.

[10] Casey Whitelaw, Navendu Garg, and Shlomo Argamon, *Using Appraisal Groups for Sentiment Analysis*, Proceedings of the 14th ACM international conference on Information and knowledge management, 2005.

[11] Hanchuan Peng, Fuhui Long, and Chris Ding, *Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005.

[12] Chris Ding and Hanchuan Peng, *Minimum redundancy feature selection from microarray gene expression data*, Journal of Bioinformatics and Computational Biology, 2005.

[13] Alistair Kennedy and Diana Inkpen, *Sentiment Classification of Movie Reviews Using Contextual Valence Shifters*, Computational Intelligence, 2006.

[14] Mohamed Afify, Olivier Siohan, and Ruhi Sarikaya, *Gaussian Mixture Language Models for Speech Recognition*, IEEE International Conference on Acoustics, Speech, and Signal Processing, 2007.

[15] John Blitzer, Mark Dredze, and Fernando Pereira, *Biographies, Bollywood, Boomboxes and Blenders: Domain Adaptation for Sentiment Classification*, Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, 2007.

[16] Wei Chen, *Building Language Model On Continuous Space Using Gaussian Mixture Models*, 2007.

[17] Omar Zaidan, Jason Eisner, and Christine D. Piatko, *Using "Annotator Rationales" to Improve Machine Learning for Text Categorization*, NAACL-HLT, 2007.

[18] Noriaki Sato, Michiko Anse, and Tsutomu Tabe, *A method for constructing a movie-selection support system based on Kansei engineering*, Proceedings of the 2007 conference on Human interface: Part I, 2007.

[19] Michelle Annett and Grzegorz Kondrak, *A Comparison of Sentiment Analysis Techniques: Polarizing Movie Blogs*, Proceedings of the Canadian Society for computational studies of intelligence, 21st conference on Advances in artificial intelligence, 2008.

[20] Sasha Blair-Goldensohn, Tyler Neylon, Kerry Hannan, George A. Reis, Ryan Mcdonald, and Jeff Reynar, *Building a Sentiment Summarizer for Local Service Reviews*, In NLP in the Information Explosion Era, 2008.

[21] Lina Zhou and Pimwadee Chaovalit, *Ontology-Supported Polarity Mining*, J. Am. Soc. Inf. Sci. Technol., 2008.

[22] Ahmed Abbasi, Hsinchun Chen, and Arab Salem, *Sentiment Analysis in Multiple Languages: Feature Selection for Opinion Classification in Web Forums*, ACM Transactions on Information Systems, 2008.

[23] M. Simeon and R. Hilderman, *Categorical Proportional Difference: A Feature Selection Method for Text Categorization*, Seventh Australasian Data Mining Conference, 2008.

[24] Pablo Daniel Azar, *Sentiment Analysis in Financial News*, 2009.

[25] Stephan Greene, and Philip Resnik, *More than words: syntactic packaging and implicit sentiment*, Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, 2009.

[26] Chenghua Lin and Yulan He, *Joint sentiment/topic model for sentiment analysis*, Proceedings of the 18th ACM conference on Information and knowledge management, 2009.

[27] Tim O'Keefe and Irena Koprinska, *Feature Selection and Weighting in Sentiment Analysis*, Proceedings of 14th Australasian Document Computing Symposium, 2009.

[28] Justin Martineau and Tim Finin *Delta TFIDF: An Improved Feature Space for Sentiment Analysis*, ICWSM, 2009.

[29] Georgios Paltoglou and Mike Thelwall, *A study of Information Retrieval weighting schemes for sentiment analysis*, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, 2010.

[30] Zhe Xu, *A Sentiment Analysis Model Integrating Multiple Algorithms and Diverse Features*, 2010.

[31] Bing Liu, *Handbook of Natural Language Processing, Second Edition*, ISBN 978-1420085921, 2010.

[32] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts, *Learning Word Vectors for Sentiment Analysis*, Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1, 2011.

[33] Yelena Mejova and Padmini Srinivasan, *Exploring Feature Definition and Selection for Sentiment Analysis*, Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media, 2011.

[34] Olena Kummer and Jacques Savoy, *Feature Selection in Sentiment Analysis*, Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analys, 2012.

[35] Panagiotis Symeonidis, Ivaylo Kehayov, and Yannis Manolopoulos, *Text Classification by Aggregation of SVD Eigenvectors*, Proceedings of the 16th East European conference on Advances in Databases and Information Systems, 2012.

[36] Basant Agarwal and Namita Mittal, *Categorical Probability Proportion Difference(CPPD): A Feature Selection Method for Sentiment Classification*, Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, 2013.