ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ
ΥΠΟΛΟΓΙΣΤΩΝ

Ανάλυση Συναισθήματος από Κείμενο με Τεχνικές Μηχανικής Μάθησης και Χρήση Λεξικού

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Εμμανουήλ Παπαδάκης

Επιβλέπων : Στέφανος Κόλλιας

Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούνιος 2016

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ
ΥΠΟΛΟΓΙΣΤΩΝ

Ανάλυση Συναισθήματος από Κείμενο με Τεχνικές Μηχανικής Μάθησης και Χρήση Λεξικού

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Εμμανουήλ Παπαδάκης

Επιβλέπων : Στέφανος Κόλλιας

Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 22η Ιουνίου 2016.

.....	.....	.....
Στέφανος Κόλλιας	Ανδρέας-Γεώργιος Σταφυλοπάτης	Γεώργιος Στάμου
Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.	Επίκουρος Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούνιος 2016

.....

Εμμανουήλ Παπαδάκης
Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Περίληψη

Η ανάπτυξη του διαδικτύου τα τελευταία χρόνια και η ανταλλαγή τεραστίων ποσοτήτων πληροφορίας μεταξύ των χρηστών σε όλο τον κόσμο καθιστά επιτακτική την μελέτη και ανάλυση αλγορίθμων που συμπεραίνουν αυτοματοποιημένα τα συναισθήματα, τις επιθυμίες και τις πεποιθήσεις των ανθρώπων με βάση το κείμενο. Το πρόβλημα αυτό μελετάται από το πεδίο της ανάλυσης συναισθήματος, το οποίο αναπτύσσεται ραγδαία λόγω του έντονου ενδιαφέροντος της επιστημονικής και βιομηχανικής κοινότητας.

Στην παρούσα διπλωματική εξετάζεται το πρόβλημα της ταξινόμησης κριτικών ταινιών με βάση την πολικότητα της άποψης σε θετικές ή αρνητικές. Το σύνολο δεδομένων από κριτικές ταινιών που χρησιμοποιήθηκε είναι αυτό που εισηγήθηκε από τους Pang και Lee και χρησιμοποιείται έκτοτε ευρέως. Για την αντιμετώπιση του προβλήματος εξετάσαμε τη χρήση συναισθηματικού λεξικού και συγκεκριμένα του SenticNet, ένα συναισθηματικό λεξικό 30000 εννοιών της αγγλικής γλώσσας δίνοντας προσοχή στα φαινόμενα της άρνησης και της αντίθεσης. Εξετάσαμε επίσης τη χρήση αλγορίθμων παραδοσιακής επιβλεπόμενης μηχανικής μάθησης, όπως ο Naive Bayes, ο Maximum Entropy, οι Μηχανές Διανυσμάτων Υποστήριξης (SVMs) και τα Τεχνητά Νευρωνικά Δίκτυα αλλά και αλγορίθμων βαθιάς μηχανικής μάθησης, όπως είναι τα Συνελικτικά Νευρωνικά Δίκτυα (ΣΝΔ). Στον αλγόριθμο Naive Bayes, πειραματιστήκαμε με την χρήση και των δύο βασικών εκδοχών του που χρησιμοποιούνται στην ταξινόμηση κειμένου, Multinomial Naive Bayes και Bernoulli Naive Bayes. Στην υλοποίηση με SVMs πειραματιστήκαμε με τον πυρήνα και σαν πυρήνες χρησιμοποιήθηκαν ο γραμμικός και ο rbf γκαουσιανός. Στην υλοποίηση με τεχνητά νευρωνικά δίκτυα επικεντρωθήκαμε σε αρχιτεκτονικές τριών επιπέδων και πειραματιστήκαμε με τον αριθμό των κρυφών νευρώνων. Σαν χαρακτηριστικά για τους αλγορίθμους μηχανικής μάθησης (πλην των ΣΝΔ που μαθαίνουν μόνα τους τα χαρακτηριστικά κάτι που αποτελεί πλεονέκτημά τους) χρησιμοποιήσαμε βασικά την Bag-of-Concepts αναπαράσταση του κειμένου και σαν έννοιες χρησιμοποιήσαμε ένα υποσύνολο των καταχωρήσεων του SenticNet. Στο τελικό στάδιο της εργασίας, επιχειρήσαμε να συνδυάσουμε τους επιμέρους ταξινομητές για να επωφεληθούμε από το συνδυασμό της γνώσης. Ο συνδυασμός αυτός καλείται συνολική μάθηση και πειραματιστήκαμε και με τους δύο κανόνες πραγμάτωσής της: τον κανόνα της πλειοψηφίας και τον κανόνα της σταθμισμένης ψηφοφορίας. Για την μελέτη της αποτελεσματικότητας των διάφορων μοντέλων μάθησης χρησιμοποιήσαμε κυρίως την μετρική της συνολικής ακρίβειας ή ορθότητας.

Συμπεράναμε από την εργασία μας ότι ο ταξινομητής μας με βάση το λεξικό δίνει μέτρια αποτελέσματα κάτι που οφείλεται κυρίως στην απλότητα της ανάλυσής μας με την εξέταση λίγων γλωσσολογικών κανόνων. Ο αλγόριθμος Naive Bayes, παρά την απλότητά του, δίνει ικανοποιητικά αποτελέσματα ταξινόμησης κειμένου, εμφανώς ανώτερα από τον βασισμένο σε λεξικό ταξινομητή και σε πολλές περιπτώσεις ανώτερα από αυτά που πετυχαίνουν οι πολυπλοκότεροι αλγόριθμοι του ταξινομητή μέγιστης εντροπίας, των μηχανών διανυσμάτων υποστήριξης και των νευρωνικών δικτύων. Τα ΣΝΔ βέβαια πέτυχαν αρκετά καλύτερα αποτελέσματα από τον αλγόριθμο Naive Bayes, κατά ένα ποσοστό κοντά στο 10%, αλλά είχαν πολύ μεγαλύτερη πολυπλοκότητα υλοποίησης που αντιστοιχούσε σε πολύ μεγαλύτερο χρόνο εκπαίδευσης. Τέλος, ο συνδυασμός των επιμέρους ταξινομητών για την ενίσχυση της απόδοσης δεν βελτίωσε σημαντικά τα αποτελέσματα ταξινόμησης και αυτό οφείλεται στο ότι οι ταξινομητές έπαιρναν συσχετισμένες αποφάσεις κάνοντας παρόμοια λάθη.

Λέξεις κλειδιά: Ανάλυση συναισθήματος, συναισθηματικό λεξικό, μηχανική μάθηση, βαθιά μάθηση, συνελικτικά νευρωνικά δίκτυα, συνολική μάθηση

Abstract

The development of the Internet, in recent years and the interchange of huge quantities of information among the users all over the word renders the study and analysis of algorithms which automatically deduce people's sentiments, desires and beliefs based on text, necessary. This is what the field of sentiment analysis faces, and this field is being greatly developing due to the great interest of scientific and industrial community.

In this thesis, we dealt with the problem of classifying movie reviews to positive or negative ones based on the polarity of opinion expressed. The dataset of movie reviews that we used is that one which was introduced by Pang and Lee and has been widely used since then. For facing the problem, we examined the use of a sentiment lexicon called SenticNet which is a sentiment lexicon of 30,000 english concepts, putting emphasis on the phenomena of negation and opposition. We also examined the use of supervised machine learning algorithms, such as Naive Bayes, Maximum Entropy, Support Vector Machines (SVMs) and Artificial Neural Networks but also of deep learning algorithms, such as Convolutional Neural Networks (CNN). Concerning Naive Bayes, we experimented with the use of both basic versions that are used in text classification, Multinomial Naive Bayes and Bernoulli Naive Bayes. Concerning SVMs, we experimented with the use of kernels and selected the linear and rbf gaussian kernels. When it comes to neural networks, we focused on 3-layer architectures and experimented with the number of hidden neurons. As features for the machine leaning algorithms (except CNNs which learn the features on their own, fact that's their advantage) we basically used the Bag-of-Concepts representation of text and as concepts we used a subset of registrations of SenticNet. In the final step of our thesis, we tried to combine the individual classifiers so as to take advantage of knowledge combination. This combination is called ensemble learning and we experimented with both rules of its implementation: the rule of majority voting and the rule of weighted voting. For studying the effectiveness of the various models, we mainly used the metric of total accuracy or correctness.

We concluded from our thesis that our lexicon-based classifier gives mediocre results mainly because of the simplicity of our analysis as we included few linguistic rules. The Naive Bayes algorithm, despite its simplicity, yields satisfying results in classifying text, obviously superior to those obtained from the lexicon-based classifier and in many cases superior to those obtained from more complicated algorithms, such as maximum entropy, support vector machines and neural networks. For sure, CNNs accomplished much better results than Naive Bayes, at a rate close to 10%, but they had a much more complicated implementation which corresponded to a much longer training time. Finally, the combination of individual classifiers to boost the performance didn't improve significantly the classification results and this is due to the fact that the classifiers were taking associated decisions and making similar mistakes.

Keywords: Sentiment analysis, sentiment lexicon, machine learning, deep learning, convolutional neural networks, ensemble learning

Ευχαριστίες

Αρχικά, θα ήθελα να ευχαριστήσω τον καθηγητή κύριο Στέφανο Κόλλια για την εμπιστοσύνη που μου έδειξε δίνοντάς μου τη δυνατότητα να εκπονήσω αυτή τη διπλωματική εργασία σε ένα τόσο ενδιαφέρον θέμα. Η καθοδήγηση, οι συζητήσεις που είχαμε και οι συμβουλές του ήταν ιδιαίτερα σημαντικές κατά την εκτέλεση της εργασίας.

Θα ήθελα επίσης να ευχαριστήσω τον Πάνο Γεωργαντά για την παραχώρηση των υπολογιστικών πόρων του εργαστηρίου για την εκτέλεση των διάφορων πειραμάτων.

Τέλος, θα ήθελα να ευχαριστήσω τους φίλους μου από τη σχολή για τις συζητήσεις και τον χρόνο που περάσαμε μαζί, και κυρίως την οικογένειά μου, που είναι πάντα δίπλα μου στηρίζοντας με σε κάθε μου βήμα.

Περιεχόμενα

Κεφάλαιο 1 Εισαγωγή 1
1.1 Ανάλυση Συναισθήματος από Κείμενο: Ορισμός του Προβλήματος 1
1.2 Οργάνωση του κειμένου 2
Κεφάλαιο 2 Μηχανική Μάθηση 3
2.1 Ορισμός Μηχανικής Μάθησης και Βασικές Έννοιες 3
2.2 Ανάλυση συναισθήματος – Σύνδεση με Μηχανική Μάθηση 7
2.3 Ο αλγόριθμος Naive Bayes 9
2.4 Ο αλγόριθμος Maximum Entropy 12
2.5 Μηχανές Διανυσμάτων Υποστήριξης 14
2.5.1 Γραμμικά Διαχωρίσιμα Προβλήματα 15
2.5.2 Μη Γραμμικά Διαχωρίσιμα Προβλήματα – Μεταβλητές Χαλαρότητας 17
2.5.3 Μη Γραμμικά Διαχωρίσιμα Προβλήματα – Συναρτήσεις Πυρήνα 19
2.6 Τεχνητά Νευρωνικά Δίκτυα 22
2.6.1 Perceptron 22
2.6.2 Νευρωνικά Δίκτυα Πολλών Επιπέδων 24
2.6.3 Εκπαίδευση MLP – Αλγόριθμος Backpropagation 26
Κεφάλαιο 3 Συνελικτικά Νευρωνικά Δίκτυα 29
3.1 Αρχιτεκτονική ΣΝΔ 30
3.1.1 Επίπεδο Συνέλιξης 31
3.1.2 Επίπεδο Συγκέντρωσης 34
3.1.3 Πλήρως Συνδεδεμένο Επίπεδο 35
3.2 Κανονικοποίηση και Εκπαίδευση ΣΝΔ 36
3.2.1 Υπερπαράμετροι ΣΝΔ 36
3.2.2 Κανονικοποίηση ΣΝΔ 36
3.2.3 Εκπαίδευση ΣΝΔ 38
3.3 Χρήση ΣΝΔ στην Επεξεργασία Φυσικής Γλώσσας 38
3.3.1 Αλγόριθμοι Glove και Word2vec 40
3.3.2 Εφαρμογές ΣΝΔ σε Προβλήματα NLP 44
Κεφάλαιο 4 Ανάλυση Συναισθήματος Βασισμένη σε Λεξικό 47
4.1 Ανάλυση Βασισμένης σε Λεξικό Προσέγγισης 47
4.2 Συναισθηματικά Λεξικά 50
4.3 Ενδιαφέρουσες Ερευνητικές Προσπάθειες 54

Κεφάλαιο 5 Υλοποίηση 55
5.1 Δεδομένα 55
5.2 Προεπεξεργασία Δεδομένων 56
5.3 Ανάλυση με Λεξικό 59
5.4 Ανάλυση με Μηχανική Μάθηση 62
5.4.1 Υλοποίηση Naive Bayes, Maximum Entropy, SVM 62
5.4.2 Υλοποίηση MLP 63
5.5 Ανάλυση με Συνελικτικά Νευρωνικά Δίκτυα 64
5.6 Συνδυασμός Τεχνικών 66
Κεφάλαιο 6 Πειραματικά Αποτελέσματα 69
6.1 Εφαρμογή Λεξικού 69
6.2 Εφαρμογή Αλγορίθμων Μηχανικής Μάθησης 70
6.2.1 Εφαρμογή Naive Bayes, Maximum Entropy, SVM 70
6.2.2 Εφαρμογή MLP 71
6.3 Εφαρμογή ΣΝΔ 72
6.4 Συνδυασμός Τεχνικών 73
6.5 Σύνοψη της Εργασίας 75
Βιβλιογραφία 77

Κατάλογος Σχημάτων

Σχήμα 1: Βέλτιστο υπερεπίπεδο για γραμμικά διαχωρίσιμο πρόβλημα 17
Σχήμα 2: Βέλτιστο υπερεπίπεδο για μη γραμμικά διαχωρίσιμο πρόβλημα 18
Σχήμα 3: RBF kernel 20
Σχήμα 4: RBF kernel με γ=1 20
Σχήμα 5: RBF kernel με γ=10 21
Σχήμα 6: RBF kernel με γ=100 21
Σχήμα 7: RBF kernel με γ=1000 22
Σχήμα 8: Το πρόβλημα της XOR είναι μη γραμμικά διαχωρίσιμο και μπορεί να επιλυθεί από MLP 24
Σχήμα 9: MLP 3 επιπέδων 25
Σχήμα 10**:** Αριστερά: Ένα 3-layer MLP, Δεξιά: Ένα ΣΝΔ 30
Σχήμα 11: Παράδειγμα αρχιτεκτονικής ΣΝΔ 31
Σχήμα 12: Συνέλιξη με ένα 3x3 φίλτρο 32
Σχήμα 13: Οργάνωση Συνελικτικού Επιπέδου σε 3 διαστάσεις 33
Σχήμα 14: Max Pooling στα ΣΝΔ 35
Σχήμα 15: Μορφή ΣΝΔ για ταξινόμηση προτάσεων 39
Σχήμα 16: Το δίκτυο που χρησιμοποιείται από τον αλγόριθμο Word2vec 41
Σχήμα 17: Το δίκτυο που χρησιμοποιείται από τον αλγόριθμο Word2vec και το μοντέλο CBOW 43
Σχήμα 18: Αρχιτεκτονική ΣΝΔ με 2 κανάλια με στόχο την ταξινόμηση προτάσεων, Y.Kim (2014) 45
Σχήμα 19: Διάγραμμα ροής του αλγορίθμου προεπεξεργασίας των δεδομένων 56

Κατάλογος Πινάκων

Πίνακας 1: Ένα απόσπασμα του MPQA subjectivity lexicon 51
Πίνακας 2: Ένα απόσπασμα του SentiWordNet 51
Πίνακας 3: Ένα απόσπασμα του Harvard General Inquirer 52
Πίνακας 4: Ένα απόσπασμα του LIWC 52
Πίνακας 5: Ένα απόσπασμα του WordNet Affect 53
Πίνακας 6: Ένα απόσπασμα του SenticNet 53
Πίνακας 7: Οι αρχιτεκτονικές του 3-layer MLP που εξετάσαμε 64
Πίνακας 8: Οι συνδυασμοί περιττού πλήθους ταξινομητών που εξετάσαμε για την εφαρμογή του κανόνα της πλειοψηφίας 66
Πίνακας 9: Αποτελέσματα από τη βασισμένη σε λεξικό προσέγγιση 69
Πίνακας 10**:** Αποτελέσματα από την εφαρμογή αλγορίθμων μηχανικής μάθησης 70
Πίνακας 11: Ποσοστά ορθότητας από την εφαρμογή του Multinomial Naive Bayes 71
Πίνακας 12: Ποσοστά ορθότητας από την εφαρμογή διαφόρων αρχιτεκτονικών 3-layer MLP 72
Πίνακας 13: Ποσοστό ορθότητας από την εφαρμογή ΣΝΔ 72
Πίνακας 14: Ποσοστά ορθότητας με εφαρμογή του κανόνα της πλειοψηφίας 73
Πίνακας 15: Ποσοστά ορθότητας με εφαρμογή του κανόνα της σταθμισμένης ψηφοφορίας 74

Κεφάλαιο 1

Εισαγωγή

1.1 Ανάλυση Συναισθήματος από Κείμενο: Ορισμός του Προβλήματος

Η ανάλυση συναισθήματος από κείμενο (sentiment analysis) είναι μια εφαρμογή του data mining που στοχεύει κυρίως στην εξαγωγή της άποψης από κείμενο και στην ταξινόμησή της ως αρνητική ή θετική. Η εργασία αυτή ονομάζεται και ανίχνευση πολικότητας (polarity detection) ή ανάλυση άποψης από κείμενο και δεν θα πρέπει να συγχέεται με την κατηγοριοποίηση με βάση την συναισθηματική κατάσταση του συγγραφέα κατά τη συγγραφή (πχ. χαρά, λύπη, θυμός) η οποία επίσης ανήκει στον τομέα του sentiment analysis. Η χρησιμότητα της εφαρμογής αυτής υπαγορεύεται από την ανάπτυξη του διαδικτύου και την ανταλλαγή τεραστίων ποσοτήτων πληροφορίας μεταξύ των χρηστών σε όλο τον κόσμο. Καθημερινά, αναπαράγονται κριτικές και απόψεις για διάφορα πολιτικά, αθλητικά ή άλλα γεγονότα, προϊόντα, ταινίες κτλ. με αποτέλεσμα ο όγκος της πληροφορίας που αναπτύσσεται να είναι αδύνατο να επεξεργαστεί μόνο από τον άνθρωπο χωρίς τη βοήθεια του υπολογιστή. Έτσι γίνεται εύκολα αντιληπτό γιατί η επιστημονική αλλά και η βιομηχανική κοινότητα έχει δείξει έντονο ενδιαφέρον στον τομέα αυτό.

Στην διπλωματική αυτή εργασία εξετάζουμε τη χρήση λεξικού, αλγορίθμων επιβλεπόμενης μηχανικής μάθησης αλλά και συνδυασμού αυτών για την αυτόματη ταξινόμηση κριτικών ταινιών σε δύο κατηγορίες: θετική και αρνητική. Το dataset που χρησιμοποιήσαμε περιέχει 10662 μικρές σε έκταση κριτικές ταινιών (σε μέγεθος μιας πρότασης σχεδόν όλες) που χρησιμοποιήθηκαν σε πειράματα από τους Pang και Lee. Οι κριτικές αυτές είναι γραμμένες στην αγγλική γλώσσα με εξαίρεση κάποιες λίγες προτάσεις που είναι γραμμένες στα ισπανικά.

Μια τέτοια εφαρμογή που ταξινομεί μια κριτική ταινίας σε θετική ή αρνητική είναι ενδιαφέρουσα και χρήσιμη αφού μπορεί να συνδυάσει κριτικές από πολλούς διαφορετικούς χρήστες και να προτείνει σε τρίτους μια ταινία που έχει συγκεντρώσει πολλές θετικές κριτικές ή να αποφύγει την πρόταση μίας ταινίας η οποία ενδέχεται να μην είναι καλή λόγω των πολλών αρνητικών κριτικών που έχει λάβει. Μπορεί ακόμη να δημιουργεί μία λίστα με τις πιο δημοφιλείς ταινίες του κοινού για μια συγκεκριμένη χρονική περίοδο πχ. ένα έτος.

Η ανάλυση συναισθήματος με βάση το λεξικό προϋποθέτει ένα συναισθηματικό λεξικό το οποίο θα περιέχει λέξεις ή φράσεις καθώς και το αντίστοιχο σκορ, μια βαθμολογία η οποία εκφράζει το πόσο η έννοια μιας λέξης συνδέεται με το αντίστοιχο συναίσθημα. Για ταξινόμηση σε 2 κατηγορίες, positive και negative, όπως η περίπτωση μας, συνήθως οι παραπάνω βαθμολογίες ανήκουν στο διάστημα [-1,+1] με το -1 να δηλώνει απόλυτα αρνητική λέξη και αντίστοιχα το +1 να δηλώνει απόλυτα θετική. Τέτοια σκορ χρησιμοποιεί και το SenticNet, ένα συναισθηματικό λεξικό 30000 concepts (απλές λέξεις ή και συνδυασμοί λέξεων-φράσεις) της Αγγλικής γλώσσας, το οποίο χρησιμοποιήσαμε.

Όσον αφορά την εφαρμογή της μηχανικής μάθησης στο πρόβλημά μας, οι αλγόριθμοι που εξετάστηκαν και χρησιμοποιήθηκαν είναι οι:

- Naive Bayes με διάφορες παραλλαγές
- Ταξινομητής Μέγιστης Εντροπίας (Maximum Entropy Classifier)
- Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machines) με και χωρίς πυρήνα
- Τεχνητά Νευρωνικά Δίκτυα
- Συνελικτικά Νευρωνικά Δίκτυα (ΣΝΔ)

1.2 Οργάνωση του κειμένου

Το κείμενο οργανώνεται σε 5 κεφάλαια ως εξής:

Στο κεφάλαιο 2 παρουσιάζεται η μία από τις τρεις βασικές προσεγγίσεις του προβλήματος του sentiment analysis, η βασισμένη στη μηχανική μάθηση προσέγγιση. Εκεί εξηγούμε τις βασικές έννοιες της μηχανικής μάθησης, τα προβλήματα που αυτή καλείται να αντιμετωπίσει και τους αλγορίθμους που χρησιμοποιεί . Αναφέρουμε επίσης το βασικό μαθηματικό υπόβαθρο πίσω από την εφαρμογή των αλγορίθμων αλλά και τα μέτρα επίδοσης που χρησιμοποιούνται για την αξιολόγησή τους.

Στο κεφάλαιο 3 περιγράφουμε τα Συνελικτικά Νευρωνικά Δίκτυα, μια ειδική κατηγορία νευρωνικών δικτύων που ανήκουν στις τεχνικές βαθιάς μηχανικής μάθησης (deep learning). Εκεί εξηγούμε πως πέρα από την περιοχή της όρασης υπολογιστών και την ταξινόμηση εικόνων, βρίσκουν εφαρμογές και σε προβλήματα επεξεργασίας φυσικής γλώσσας, όπως αυτό του sentiment analysis, με σημαντικά μάλιστα αποτελέσματα.

Στο κεφάλαιο 4 παρουσιάζεται η δεύτερη βασική προσέγγιση του προβλήματος του sentiment analysis, η προσέγγιση με βάση το λεξικό και εξηγούνται κάποιες από τις αιτίες που η μέθοδος αυτή συνήθως δεν δίνει ικανοποιητικά αποτελέσματα. Αναφέρονται επίσης κάποια διαθέσιμα συναισθηματικά λεξικά και ενδιαφέρουσες ερευνητικές προσπάθειες.

Στο κεφάλαιο 5 εξηγούνται κάποιες λεπτομέρειες της υλοποίησης μας. Αναφερόμαστε στα στάδια προεπεξεργασίας και εξαγωγής χαρακτηριστικών και πώς τα χαρακτηριστικά αυτά χρησιμοποιούνται από το λεξικό, τους αλγορίθμους μηχανικής μάθησης αλλά και τον συνδυασμό αυτών για την εξαγωγή της πολικότητας του συναισθήματος που εκφράζεται στο κείμενο.

Τέλος, στο κεφάλαιο 6 παρουσιάζονται τα αποτελέσματα της εφαρμογής των παραπάνω μεθόδων και καταγράφουμε τα συμπεράσματα της εργασίας μας.

Κεφάλαιο 2

Μηχανική Μάθηση

Οι βασικές προσεγγίσεις επίλυσης του προβλήματος της ανάλυσης συναισθήματος κειμένου είναι:

- Η βασισμένη σε μηχανική μάθηση προσέγγιση (machine learning – based approach)
- Η βασισμένη σε λεξικό προσέγγιση (lexicon-based approach)
- Η βασισμένη σε υβριδικές μεθόδους προσέγγιση (hybrid approach), η οποία συνδυάζει τις προηγούμενες δύο μεθόδους.

Στο κεφάλαιο αυτό θα αναλύσουμε τη βασισμένη σε μηχανική μάθηση προσέγγιση. Πιο συγκεκριμένα, θα αναλύσουμε τον όρο μηχανική μάθηση, θα περιγράψουμε τα προβλήματα που επιλύει και θα εξηγήσουμε τους αλγορίθμους που χρησιμοποιούνται ευρέως μεταξύ των άλλων και στο πρόβλημα του sentiment analysis.

2.1 Ορισμός Μηχανικής Μάθησης και Βασικές Έννοιες

Με τον όρο μηχανική μάθηση (Machine Learning) εννοούμε τη χρήση δεδομένων από έναν αλγόριθμο ο οποίος εκτελείται σε μια υπολογιστική μηχανή έτσι ώστε να βελτιώνεται σταδιακά κατά την εκτέλεση μιας λειτουργίας. Οι λειτουργίες αυτές είναι αντίστοιχες αυτών που ανήκουν στην ανθρώπινη νοημοσύνη και μπορεί να είναι:

- Η μηχανική κατανόηση της γλώσσας και η παραγωγή ομιλίας (natural language processing/understanding)
- Η μηχανική αναγνώριση προτύπων (pattern recognition)
- Η ανάπτυξη στρατηγικής σε διάφορες καταστάσεις (πχ. παιχνίδια) κ.ά.

Η μηχανική μάθηση αποτελεί βασικό συστατικό της Τεχνητής Νοημοσύνης και χρησιμοποιεί στοιχεία από τη στατιστική, τη θεωρία πληροφορίας και τη γνωσιακή επιστήμη.

Εν γένει, ένα πρόβλημα μάθησης θεωρεί ένα σύνολο n δειγμάτων από δεδομένα

$$D = \{\boldsymbol{x_1}, \boldsymbol{x_2}, \dots, \boldsymbol{x_n}\}$$

και προσπαθεί να μάθει ιδιότητες άγνωστων δεδομένων. Το σύνολο αυτό ονομάζεται σύνολο εκπαίδευσης (training set) και όπως δηλώνει το όνομά του χρησιμοποιείται για την εκπαίδευση

του συστήματος machine learning. Το κάθε δείγμα $\boldsymbol{x_i}$ ονομάζεται χαρακτηριστικό και μπορεί να είναι βαθμωτό (single feature) ή διάνυσμα (feature vector).

Οι τρεις βασικοί τύποι μηχανικής μάθησης είναι:

1. Η μάθηση με επίβλεψη (supervised learning). Σε αυτή τη κατηγορία μηχανικής μάθησης κάθε δείγμα-χαρακτηριστικό $\boldsymbol{x_i}$ συνοδεύεται από μία επιπρόσθετη ιδιότητα που ονομάζεται ετικέτα-στόχος και είναι η μεταβλητή που το σύστημα machine learning καλείται να προβλέψει. Το πρόβλημα προς επίλυση μπορεί να είναι:
    - Ταξινόμηση (Classification). Τα δείγματα ανήκουν σε 2 ή περισσότερες κατηγορίες ή κλάσεις και εμείς θέλουμε το σύστημά μας να μάθει από τα επισημειωμένα (labeled) δεδομένα του συνόλου εκπαίδευσης ώστε να προβλέπει κατά το δυνατόν σωστά την κλάση άλλων άγνωστων προς αυτό δεδομένων. Το πρόβλημα αυτό εντάσσεται στην κατηγορία της αναγνώρισης προτύπων [1], δηλαδή του επιστημονικού πεδίου που σκοπό έχει την κατάταξη των αντικειμένων – προτύπων σε κλάσεις. Η αναγνώριση προτύπων βρίσκει πολλές εφαρμογές όπως:
        - Στην όραση υπολογιστών (computer vision): αναγνώριση προσώπων, οπτική αναγνώριση χαρακτήρων (Optical Character Recognition – OCR)
        - Στην ακουστική: αναγνώριση ομιλίας, αναγνώριση μουσικής κ.ά.
        - Στην ιατρική/βιολογία: διάγνωση ασθενειών όπως καρκίνου με τη βοήθεια υπολογιστή , επεξεργασία ηλεκτροκαρδιογραφήματος (ECG), αναγνώριση γονιδίων κ.ά.
        - Στην επεξεργασία φυσικής γλώσσας (Natural Language Processing – NLP): αυτόματη συνόψιση κειμένου, μηχανική μετάφραση, εξαγωγή ονοματικών οντοτήτων (Named Entity Recognition), συντακτική ανάλυση – κατασκευή συντακτικού δέντρου πρότασης, φιλτράρισμα ανεπιθύμητης αλληλογραφίας (spam filtering), ανάλυση συναισθήματος (sentiment analysis) κ.ά.
    - Παλινδρόμηση (Regression). Στο πρόβλημα αυτό η μηχανή καλείται να εκτιμήσει την τιμή εξόδου που αντιστοιχεί σε ένα πρότυπο εισόδου. Η τιμή εξόδου αναζητάται μέσα από ένα συνεχές σύνολο τιμών, για παράδειγμα το σύνολο των πραγματικών αριθμών, $\mathbb{R}$. Τέτοια προβλήματα μπορεί να είναι η εκτίμηση της θερμοκρασίας με βάση την υγρασία, το υψόμετρο και την πίεση του αέρα, ή η πρόβλεψη του μήκους του σολομού σαν συνάρτηση της ηλικίας και του βάρους του.
2. Η μάθηση χωρίς επίβλεψη (unsupervised learning). Σε αυτή τη κατηγορία μηχανικής μάθησης τα δεδομένα εκπαίδευσης δεν συνοδεύονται από τις αντίστοιχες τιμές στόχου. Ο σκοπός σε τέτοια προβλήματα μπορεί να είναι η συσταδοποίηση (clustering) , δηλαδή η ανακάλυψη ομοιοτήτων μεταξύ των προτύπων εισόδου, ή ο καθορισμός της κατανομής των δεδομένων στον χώρο εισόδου, πρόβλημα γνωστό ως εκτίμηση πυκνότητας, ή η συμπίεση δεδομένων στην οποία όγκος δεδομένων μεγάλων διαστάσεων αντικαθίσταται από δεδομένα μικρότερης διάστασης.
3. Ενισχυτική μάθηση (reinforcement learning). Σε αυτή τη κατηγορία μηχανικής μάθησης

το σύστημα μαθαίνει την επιθυμητή συμπεριφορά μέσω συνεχούς αλληλεπίδρασης με το περιβάλλον και την ύπαρξη ενός κριτή που τιμωρεί ή επιβραβεύει. Έτσι, επιλέγει σε κάθε κατάσταση τη συμπεριφορά αυτή που θα οδηγήσει στο μεγαλύτερο δυνατό κέρδος-ανταμοιβή με βάση αυτά που έχει μάθει. Η ενισχυτική μάθηση βρίσκει εφαρμογές στην ανάπτυξη στρατηγικής σε παιχνίδια πχ. σκάκι, στον ρομποτικό έλεγχο και την αλληλεπίδραση με ανθρώπους κ.ά.

Δύο βασικές έννοιες της μάθησης είναι η εκπαίδευση και η ανάκληση. Με τον όρο εκπαίδευση εννοούμε την παρουσίαση πολλών παραδειγμάτων-προτύπων (στοιχείων του συνόλου εκπαίδευσης) στο σύστημα (με ή χωρίς στόχους, ανάλογα με τον τύπο μάθησης) με σκοπό την ρύθμιση των παραμέτρων του ώστε αυτό να βελτιώνεται στην λειτουργία αναγνώρισης ή σε όποια άλλη λειτουργία τάχθηκε. Με τον όρο ανάκληση εννοούμε την εισαγωγή ενός ή περισσότερων προτύπων με στόχο την εξαγωγή της απόκρισης του συστήματος χωρίς εκπαίδευση.

Αφού εκπαιδεύσουμε ένα σύστημα μάθησης μένει να εξετάσουμε την επίδοση του συστήματος όσον αφορά την λειτουργία στην οποία τάχθηκε. Στην περίπτωση ενός συστήματος αναγνώρισης χρησιμοποιούμε συνήθως το μετρικό της συνολικής ακρίβειας (accuracy) όπου παρουσιάζουμε πολλά άγνωστα παραδείγματα στο σύστημά μας (άγνωστα με την έννοια ότι δεν έχουν παρουσιαστεί σε αυτό κατά τη φάση της εκπαίδευσης) και εξετάζουμε την ικανότητα γενίκευσης, δηλαδή πόσα από αυτά ταξινομούνται σωστά. Τα παραδείγματα αυτά λέμε ότι ανήκουν στο σύνολο ελέγχου (test set) , έστω το σύνολο *T*, οπότε η συνολική ακρίβεια ορίζεται ως εξής:

$$accuracy = \frac{\sigma\omega\sigma\tau\acute{\alpha}\ \tau\alpha\xi\iota\nu o\mu\acute{\eta}\mu\varepsilon\nu\alpha\ \pi\rho\acute{o}\tau\upsilon\pi\alpha\ \pi o\upsilon\ \alpha\nu\acute{\eta}\kappa o\upsilon\nu\ \sigma\tau o\ T}{\pi\lambda\eta\theta\iota\kappa\acute{o}\varsigma\ \alpha\rho\iota\theta\mu\acute{o}\varsigma\ \sigma\upsilon\nu\acute{o}\lambda o\upsilon\ T}$$

Αν μας ενδιαφέρει το πόσο καλά μαθαίνει το σύστημά μας την κάθε κατηγορία θα πρέπει να χρησιμοποιήσουμε διαφορετικές μετρικές απόδοσης, οι οποίες ορίζονται για κάθε κατηγορία και ονομάζονται ακρίβεια (precision) και ανάκληση (recall). Για τον ορισμό αυτών ας θεωρήσουμε την απλή περίπτωση της δυαδικής ταξινόμησης σε μία από δύο κατηγορίες, έστω positive και negative. Ονομάζουμε:

- True Positives (TP): ο αριθμός των δειγμάτων που ταξινομήθηκαν σωστά στην κατηγορία positive
- False Positives (FP): ο αριθμός των δειγμάτων που ταξινομήθηκαν λανθασμένα στην κατηγορία positive
- True Negatives (TN): ο αριθμός των δειγμάτων που ταξινομήθηκαν σωστά στην κατηγορία negative
- False Negatives (FN): ο αριθμός των δειγμάτων που ταξινομήθηκαν λανθασμένα στην κατηγορία negative

Η ακρίβεια τότε για μία κατηγορία πχ. την θετική, ορίζεται ως:

$$precision(pos) = \frac{TP}{TP + FP}$$

και εκφράζει το ποσοστό των σωστών ταξινομήσεων στην κατηγορία αυτή.

Η ανάκληση για μία κατηγορία πχ. τη θετική, ορίζεται ως:

$$recall(pos) = \frac{TP}{TP + FN}$$

και εκφράζει το ποσοστό των δειγμάτων που ταξινομήθηκαν σωστά μεταξύ όλων των δειγμάτων που ανήκουν στη κατηγορία αυτή.

Οι 2 αυτές μετρικές βρίσκονται σε διένεξη μεταξύ τους: η αύξηση του ενός συνεπάγεται τη μείωση του άλλου και αντίστροφα. Ο συνδυασμός τους σε μία μόνο τιμή ονομάζεται *F - measure* (ή F_1 *- measure*) και προκύπτει ως ο αρμονικός μέσος των δύο:

$$F = 2\frac{precision \cdot recall}{precision + recall}$$

Η γενικότερη μορφή του παραπάνω μέτρου ονομάζεται F_β *– measure* (για μη αρνητικές πραγματικές τιμές του β) και ορίζεται ως:

$$F_\beta = (1 + \beta^2)\frac{precision \cdot recall}{\beta^2 \cdot precision + recall}$$

Τα άλλα δύο συχνότερα χρησιμοποιούμενα *F - measures* είναι το F_2 *– measure* που δίνει μεγαλύτερη έμφαση στην ανάκληση και το $F_{0.5}$ *- measure* που δίνει μεγαλύτερη έμφαση στην ακρίβεια.

Τελικά, εξαρτάται από την εφαρμογή σε ποια μετρική θα δοθεί έμφαση για τον έλεγχο της απόδοσης του συστήματος αναγνώρισης. Γενικά, όταν το σύνολο ελέγχου είναι ισορροπημένο ως το πλήθος των δειγμάτων που ανήκουν σε κάθε κατηγορία ώστε τα αποτελέσματα της συνολικής ακρίβειας να μην είναι παραπλανητικά και τα λάθη ταξινόμησης είναι εξίσου σημαντικά ανεξάρτητα την κατηγορία που αφορούν, τότε χρησιμοποιείται η συνολική ακρίβεια (accuracy) σαν μέτρο απόδοσης. Στα επόμενα, όταν θα λέμε ακρίβεια θα εννοούμε το μέγεθος του accuracy.

Στη παρούσα διπλωματική εργασία θα ασχοληθούμε με αλγορίθμους επιβλεπόμενης μηχανικής μάθησης και συγκεκριμένα με τους:

- Naive Bayes, που αναλύεται στην ενότητα 2.3
- Maximum Entropy Classifier (ή αλλιώς Logistic Regression), που αναλύεται στην ενότητα 2.4
- SVMs, που αναλύονται στην ενότητα 2.5
- Τεχνητά Νευρωνικά Δίκτυα , που αναλύεται στην ενότητα 2.6
- Συνελικτικά Νευρωνικά Δίκτυα, που περιγράφονται στο κεφάλαιο 3

2.2 Ανάλυση συναισθήματος – Σύνδεση με Μηχανική Μάθηση

Όπως αναφέραμε και προηγουμένως το πρόβλημα της ανάλυσης συναισθήματος, δηλαδή η κατηγοριοποίηση της άποψης του συγγραφέα σχετικά με ένα θέμα που ενδιαφέρει (πχ. ένα δημοψήφισμα ή μια κριτική ταινίας) ως θετικής ή αρνητικής μπορεί να αντιμετωπιστεί με τεχνικές machine learning και έχουν γίνει αρκετές τέτοιες προσπάθειες. Το 2002, ο Turney [2], υλοποίησε ένα σύστημα αυτόματης ταξινόμησης κριτικών ταινιών και προϊόντων με μια μέθοδο unsupervised learning που βασίζεται στη σημειακή αμοιβαία πληροφορία (δείκτης Pointwise Mutual Information – PMI) μεταξύ μια φράσης που περιέχει επίθετο ή επίρρημα και των λέξεων “excellent” και “poor”. Την ίδια χρονιά, οι Pang, Lee και Vaithyanathan [3], εξέτασαν την εφαρμογή αλγορίθμων επιβλεπόμενης μηχανικής μάθησης (Naive Bayes, Maximum Entropy Classification και SVMs) χρησιμοποιώντας διάφορα χαρακτηριστικά για την ταξινόμηση στο πρόβλημα του sentiment analysis για κριτικές ταινιών. Η δυαδική (binary) ταξινόμηση ενός κειμένου σε μια από 2 κατηγορίες (θετική ή αρνητική) αν και είναι η πιο συχνή προσέγγιση δεν είναι η μοναδική. Έχουν γίνει και εργασίες πάνω σε πολυεπίπεδη συναισθηματική κατάταξη, όπως στο [4], όπου οι Pang και Lee εξέτασαν την ταξινόμηση κριτικών ταινιών σε μία από 5 κατηγορίες, που εκφράζουν βαθμολογίες από ένα έως πέντε αστέρια. Ακολούθησαν και άλλες παρόμοιες εργασίες στο χώρο του sentiment analysis, όπου χρησιμοποιούνταν ολοένα και περισσότερα πολύπλοκα μοντέλα μηχανικής μάθησης (πχ Hidden Markov Models – HMMs [5], ή Conditional Random Fields – CRFs [6], μοντέλα που λαμβάνουν υπόψη τη σειρά των λέξεων στο κείμενο) και διαφορετικά σύνολα χαρακτηριστικών, ώσπου τα τελευταία χρόνια οι έρευνες προσανατολίζονται στη χρήση μοντέλων deep learning (όπως συνελικτικά νευρωνικά δίκτυα ή επαναλαμβανόμενα (recurrent) νευρωνικά δίκτυα), που επιδεικνύουν αξιοσημείωτα αποτελέσματα.

Ένα ζήτημα που προκύπτει όταν υλοποιεί κάποιος ένα σύστημα μηχανικής αναγνώρισης συναισθήματος από κείμενο είναι η θεώρηση ή η αγνόηση της ουδέτερης κλάσης. Οι περισσότεροι άνθρωποι που ασχολούνται με το πρόβλημα αυτό τείνουν να αγνοούν την ουδέτερη κλάση επικεντρώνοντας μόνο στην θετική και την αρνητική κλάση. Παρ’ όλα αυτά, δεν περιέχουν όλες οι προτάσεις συναίσθημα. Η εκπαίδευση του ταξινομητή στην ανίχνευση μόνο αυτών των δύο κλάσεων έχει ως αποτέλεσμα αρκετές ουδέτερες λέξεις να ταξινομούνται είτε ως θετικές είτε ως αρνητικές οδηγώντας σε υπερπροσαρμογή του μοντέλου στο σύνολο εκπαίδευσης, φαινόμενο

γνωστό και ως overfitting. Έρευνες που έχουν γίνει, όπως στο [7], έχουν δείξει ότι οι SVM και Maximum Entropy ταξινομητές μπορούν να βελτιώσουν τις προβλέψεις τους και την συνολική ακρίβεια συμπεριλαμβάνοντας την ουδέτερη κλάση.

Για την εφαρμογή των αλγορίθμων μηχανικής μάθησης απαιτείται η αναπαράσταση του κειμένου από ένα διάνυσμα χαρακτηριστικών $\boldsymbol{x}$.

Η πιο απλή αναπαράσταση είναι η αναπαράσταση του κειμένου ως σύνολο από λέξεις (Bag of Word representation – BoW). Σύμφωνα με αυτή, το κείμενο αντιμετωπίζεται σαν ένα σύνολο ανεξάρτητων μεταξύ τους λέξεων με τη σειρά τους να αγνοείται. Επειδή όμως αγνοείται η σειρά των λέξεων, το φαινόμενο της άρνησης συνήθως δεν αντιμετωπίζεται σωστά, μιας και δεν παρουσιάζεται στον αλγόριθμο η σωστή εμβέλεια της άρνησης (πχ της λέξης not). Αυτό φαίνεται και από το κάτωθι παράδειγμα:

- That's not true, I'm a fan of this movie.
  $\rightarrow (BoW) : \{that, 's, not, true, , , I, 'm, a, fan, of, this, movie, .\}$
- That's true, I'm not a fan of this movie
  $\rightarrow (BoW) : \{that, 's, true, , , I, 'm, not, a, fan, of, this, movie, .\}$

όπου οι δύο προτάσεις έχουν την ίδια αναπαράσταση αν και το νόημα είναι τελείως διαφορετικό. Παρά την απλότητα της αναπαράστασης, το μοντέλο BoW χρησιμοποιείται συχνά σε προβλήματα text classification , ιδιαίτερα spam filtering, και πετυχαίνει ικανοποιητικά αποτελέσματα.

Η BoW αναπαράσταση θεωρεί απλές λέξεις όπως εμφανίζονται στο κείμενο, δηλαδή unigrams. Σαν χαρακτηριστικά μπορούμε να επιλέξουμε bigrams (δύο διαδοχικές λέξεις) ή και *n*-grams (*n* διαδοχικές λέξεις) γενικότερα. Η διαίσθηση πίσω από την επιλογή αυτή είναι ότι η απόδοση θα ενισχυθεί καθώς ο ταξινομητής θα μπορεί πιο εύκολα να συμπεράνει τη σωστή κλάση από το συνδυασμό 2 ή περισσότερων διαδοχικών λέξεων. Η επιλογή του *n* εξαρτάται από την εφαρμογή. Συνήθως αρκούν τα bigrams ή trigrams για την ενίσχυση της απόδοσης. Επιλέγοντας μεγαλύτερο *n* η απόδοση του ταξινομητή μπορεί να μειωθεί. Σαν μειονέκτημα της χρησιμοποίησης *n*-grams μπορούμε να αναφέρουμε την αύξηση του συνόλου των χαρακτηριστικών (προκύπτουν feature spaces μεγαλύτερων διαστάσεων) η οποία όμως μπορεί να προτιμηθεί όταν βελτιώνει τα αποτελέσματα.

Τα παραπάνω, επιλογή unigrams ή n-grams, συνδυάζονται συνήθως με κάποιες τεχνικές preprocessing όπως:

- Αφαίρεση stopwords, δηλαδή λέξεων που δεν συνεισφέρουν συναίσθημα στο context, όπως πχ. στα αγγλικά οι συχνά χρησιμοποιούμενες λέξεις and,the,for,if,I κτλ., τα κύρια ονόματα και κάποιες λέξεις που δηλώνουν χρόνο όπως Monday,yesterday,tomorrow κτλ.
- Αφαίρεση των σημείων στίξης, καθώς αυτά από μόνα τους δε δηλώνουν κάτι για το συναίσθημα του κειμένου
- Λημματοποίηση (lemmatization) ή αποκοπή (stemming) λέξεων. Η ίδια λέξη μπορεί να έχει διαφορετικές μορφές ανάλογα με τη γραμματική της χρήση (ρήμα, επίθετο, ουσιαστικό κτλ.). Η ιδέα της κανονικοποίησης μέσω της χρήσης λήμματος ή προθέματος

(stem) είναι να μειώσει τις λέξεις στη μικρότερή τους μορφή (λήμμα ή πρόθεμα αντίστοιχα) θεωρώντας ότι η γραμματική είναι ένα άσχετο χαρακτηριστικό για τον ταξινομητή μας. Για παράδειγμα, όλες οι παρακάτω λέξεις:

- Running
- Runner
- Runs
- Ran
- Runners

αντιστοιχούν στο λήμμα 'run' όσον αφορά τον ταξινομητή. Με τον τρόπο αυτό και μειώνεται ο χώρος των χαρακτηριστικών (πολλές λέξεις απεικονίζονται σε μία) και μειώνεται ο κίνδυνος να συναντήσει ο ταξινομητής νέες λέξεις στις οποίες δεν έχει εκπαιδευτεί , οδηγώντας σε ενίσχυση της απόδοσης.

Τέλος να σημειώσουμε ότι κατά την κατασκευή των χαρακτηριστικών μπορούμε να δηλώσουμε τον αριθμό εμφανίσεων ενός όρου ή ενός n-gram στο κείμενο αντί να δηλώσουμε απλά την ύπαρξή του ή όχι. Στο πρόβλημα του sentiment analysis όμως ο αριθμός των εμφανίσεων μίας λέξης στο κείμενο δεν κάνει μεγάλη διαφορά. Συνήθως οι δυαδικοποιημένες εκδοχές (εμφανίσεις κατωφλιωμένες στη μονάδα) των αλγορίθμων αποδίδουν καλύτερα από εκείνους που χρησιμοποιούν τον αριθμό των εμφανίσεων.

2.3 Ο αλγόριθμος Naive Bayes

Ο αλγόριθμος Naive Bayes ανήκει στους αλγορίθμους επιβλεπόμενης μηχανικής μάθησης. Πρόκειται για ένα απλό πιθανοτικό ταξινομητή που βασίζεται όπως υποδηλώνει το όνομά του στο θεώρημα του Bayes και στην αφελή (naive) υπόθεση της υπό συνθήκη ανεξαρτησίας μεταξύ των χαρακτηριστικών $x_i, x_j, i \neq j$, δεδομένης της κλάσης c_k .

Ο αλγόριθμος Naive Bayes είναι από τις βασικές τεχνικές ταξινόμησης κειμένου και παρά την απλότητά του και τις υποθέσεις ανεξαρτησίας που κάνει, αποδίδει καλά σε πολλά προβλήματα. Η καλή απόδοση συνδυάζεται μάλιστα με χαλαρές απαιτήσεις ως προς τη CPU και τη μνήμη, ενώ και ο χρόνος εκπαίδευσης είναι σημαντικά μικρότερος σε σχέση με άλλες μεθόδους. Όμως, ο Naive Bayes είναι ένας κακός εκτιμητής, καθώς συχνά υπερεκτιμά τις πιθανότητες εξόδου.

Στο γενικότερο πρόβλημα της αναγνώρισης προτύπων, όπως αναφέραμε και προηγουμένως, καλούμαστε να επιλέξουμε μία κλάση c_j στην οποία θεωρούμε ότι ανήκει ένα πρότυπο με βάση το διάνυσμα χαρακτηριστικών του, έστω $\boldsymbol{x}$. Η επιλογή μας γίνεται μέσα από N πιθανές κλάσεις $c_1, c_2, \dots, c_N$. Αν ορίσουμε τη δεσμευμένη πιθανότητα: $P(c_j|\boldsymbol{x}), j = 1, \dots, N$, ως την πιθανότητα το πρότυπο $\boldsymbol{x}$ να ανήκει στην κλάση c_j , γνωστή και ως εκ των υστέρων πιθανότητα (a posteriori probability), τότε η διαίσθησή μας λέει να επιλέξουμε για το $\boldsymbol{x}$, την κλάση που μεγιστοποιεί την παραπάνω a posteriori πιθανότητα , έστω την κλάση k. Δηλαδή θεωρούμε τον ακόλουθο κανόνα απόφασης:

Το πρότυπο $\boldsymbol{x}$ αντιστοιχίζεται στην κλάση c_k, όπου:

$$k = arg \ _j \max P(c_j|\boldsymbol{x}), \qquad j = 1, \dots, N$$

Αυτός ακριβώς είναι ο κανόνας απόφασης στον ταξινομητή Naive Bayes, και γι' αυτό ονομάζεται και Maximum A Posteriori (MAP) ταξινομητής.

Η πιθανότητα $P(c_j|\boldsymbol{x})$ εφαρμόζοντας το θεώρημα του Bayes υπολογίζεται ως εξής:

$$P(c_j|\boldsymbol{x}) = \frac{P(c_j, \boldsymbol{x})}{P(\boldsymbol{x})} = \frac{P(\boldsymbol{x}|c_j)P(c_j)}{P(\boldsymbol{x})}$$

Όπου:

- $P(c_j)$ είναι η πρότερη πιθανότητα (prior probability) της κλάσης j
- $P(\boldsymbol{x}|c_j)$ είναι η πιθανότητα του χαρακτηριστικού $\boldsymbol{x}$ δεδομένης της κλάσης c_j (class conditional probability density function)

Παρατηρούμε πως η πιθανότητα $P(x)$ δεν χρειάζεται να υπολογιστεί διότι στην εφαρμογή του κανόνα Naive Bayes εμφανίζεται ως σταθερή ποσότητα, ανεξάρτητη του j και δεν επηρεάζει τη μεγιστοποίηση.

Στο σημείο αυτό έρχεται να εφαρμοστεί και η υπόθεση της ανεξαρτησίας μεταξύ των χαρακτηριστικών δεδομένης της κλάσης c_j, οπότε η πιθανότητα $P(\boldsymbol{x}|c_j)$ υπολογίζεται ως το γινόμενο των επιμέρους $P(x_i|c_j)$. Δηλαδή:

$$P(\boldsymbol{x}|c_j) = \prod_{i=1}^{n} P(x_i|c_j)$$

Για την απόφαση του ταξινομητή λοιπόν, αρκεί ο υπολογισμός των πιθανοτήτων $P(c_j)$ και $P(x_i|c_j)$. Οι πιθανότητες αυτές εκτιμώνται κάνοντας χρήση της εκτίμησης μέγιστης πιθανοφάνειας (Maximum Likelihood Estimation – MLE) πάνω στο training set. Σύμφωνα με τη MLE, οι παράμετροι ενός στατιστικού μοντέλου επιλέγονται έτσι ώστε να συμφωνούν με τα δεδομένα που έχουμε στη διάθεσή μας. Έτσι, η πιθανότητα $P(c_j)$ υπολογίζεται ως το ποσοστό των προτύπων στο training set που ανήκουν στη κλάση c_j και η πιθανότητα $P(\boldsymbol{x}|c_j)$ υπολογίζεται από τις επιμέρους πιθανότητες $P(x_i|c_j)$ οι οποίες εκτιμώνται ίσες με τις αντίστοιχες συχνότητες των χαρακτηριστικών στο ίδιο training set.

Υπάρχουν διάφορες παραλλαγές του αλγορίθμου Naive Bayes. Η διαφορά τους έγκειται μόνο στην υπόθεση που κάνουν σχετικά με την κατανομή $P(x_i|c_j)$. Κάποιες εκδόσεις Naive Bayes είναι:

- Gaussian Naive Bayes. Εδώ γίνεται η υπόθεση ότι η κατανομή $P(x_i|c_j)$ είναι συνεχής και μάλιστα Gaussian. Δηλαδή:

$$P(x_i|c_j) = \frac{1}{\sqrt{2\pi\sigma_j^2}} \exp(-\frac{(x_i - \mu_j)^2}{2\sigma_j^2})$$

όπου η μ*έση* τιμή μ_j και η τυπική απόκλιση σ_j υπολογίζονται μέσω εκτίμησης μέγιστης πιθανοφάνειας. Ο αλγόριθμος αυτός συνήθως δεν βρίσκει εφαρμογή σε tasks επεξεργασίας φυσικής γλώσσας, όπως αυτό του sentiment analysis και δεν θα μας απασχολήσει στη συνέχεια.

- Multinomial Naive Bayes. Η έκδοση αυτή υλοποιεί τον αλγόριθμο Naive Bayes για πολυωνυμικά κατανεμημένα δεδομένα και είναι μία από τις δύο κλασικές παραλλαγές του αλγορίθμου Naive Bayes που χρησιμοποιούνται στην ταξινόμηση κειμένου. Εδώ τα δεδομένα αναπαριστώνται συνήθως ως μετρήσεις απλών λέξεων ή n-grams ανάλογα με τη θεώρηση . Η κατανομή παραμετροποιείται από τα διανύσματα $\boldsymbol{\theta}_{c_j} = (\theta_{c_j1}, \dots, \theta_{c_jn})^T$, όπου ο αριθμός n των χαρακτηριστικών για την ταξινόμηση κειμένου ισούται με το μέγεθος του λεξιλογίου και θ_{c_j1} είναι η πιθανότητα $P(x_i|c_j)$ του χαρακτηριστικού-token i να εμφανιστεί σε ένα δείγμα της κλάσης c_j. Τα στοιχεία του διανύσματος $\boldsymbol{\theta}_{c_j}$ υπολογίζονται μέσω μίας εξομαλυμένης εκδοχής MLE ως εξής:

$$\theta_{c_ji} = \frac{N_{c_ji} + a}{N_{c_j} + an}$$

όπου N_{c_ji} είναι ο αριθμός των φορών που το χαρακτηριστικό i εμφανίζεται στα δείγματα της κλάσης c_j στο training set D και N_{c_j} είναι το συνολικό πλήθος των χαρακτηριστικών για τη κλάση c_j. Η παράμετρος ομαλοποίησης a εισάγεται για την αντιμετώπιση χαρακτηριστικών που δεν εμφανίζεται καθόλου στο σύνολο εκπαίδευσης και εμποδίζει τη διάδοση μηδενικών πιθανοτήτων στους υπολογισμούς. Η παραπάνω τεχνική αν $\alpha = 1$ ονομάζεται εξομάλυνση Laplace (Laplace smoothing ή add-one smoothing), αλλιώς αν $\alpha < 1$ ονομάζεται εξομάλυνση Lidstone (Lidstone smoothing). Αν αντί να μετράμε όλες τις εμφανίσεις μιας λέξης ή ενός n-gram στο κείμενο, τις μετράμε μόνο μία φορά, τότε προκύπτει η δυαδικοποιημένη (binarized) εκδοχή του Multinomial Naive Bayes που ονομάζεται και Boolean Multinomial Naive Bayes.
- Bernoulli Naive Bayes. Πρόκειται για τη δεύτερη κλασική παραλλαγή του αλγορίθμου Naive Bayes που χρησιμοποιείται στην ταξινόμηση κειμένου. Εδώ κάθε όρος του λεξιλογίου ισούται με 1 εάν εμφανίζεται στο κείμενο αλλιώς με 0. Η διαφορά του από τον Boolean Naive Bayes είναι ότι λαμβάνει υπόψη τους όρους που δεν εμφανίζονται στο κείμενο. Ενώ στο μοντέλο Boolean Multinomial οι όροι που δεν εμφανίζονται αγνοούνται τελείως, στο μοντέλο Bernoulli οι όροι αυτοί παραγοποιούνται όταν υπολογίζονται οι δεσμευμένες πιθανότητες και άρα η απουσία των όρων συνυπολογίζεται. Η πιθανότητα $P(x_i|c_j)$ υπολογίζεται ως:

$$P(x_i|c_j) = P(i|c_j)x_i + (1 - P(i|c_j))(1 - x_i)$$

Συνήθως ο Multinomial Naive Bayes χρησιμοποιείται όταν οι πολλαπλές εμφανίσεις των λέξεων είναι σημαντικές στο πρόβλημα ταξινόμησης. Ένα τέτοιο παράδειγμα είναι όταν προσπαθούμε να κάνουμε ταξινόμηση με βάση το θέμα (topic classification). Ο δυαδικοποιημένος Multinomial Naive Bayes χρησιμοποιείται όταν οι συχνότητες των λέξεων δεν παίζουν σημαντικό ρόλο στην ταξινόμησή μας. Ένα τέτοιο παράδειγμα είναι η ανάλυση συναισθήματος, όπου δεν ενδιαφέρει τόσο το πόσες φορές αναφέρει κάποιος τη λέξη "bad" αλλά περισσότερο το γεγονός ότι απλά την αναφέρει. Τέλος, ο Bernoulli Naive Bayes μπορεί να χρησιμοποιηθεί όταν στο πρόβλημά μας η απουσία κάποιας συγκεκριμένης λέξης παίζει ρόλο. Για παράδειγμα, ο Bernoulli Naive Bayes χρησιμοποιείται συνήθως στην ανίχνευση spam ή στην ανίχνευση περιεχομένου για ανηλίκους με πολύ καλά αποτελέσματα.

2.4 Ο αλγόριθμος Maximum Entropy

Ο αλγόριθμος μέγιστης εντροπίας (Maximum Entropy) που ονομάζεται και αλγόριθμος λογιστικής παλινδρόμησης (Logistic Regression) υλοποιεί παρά την ονομασία του ένα γραμμικό μοντέλο με σκοπό την ταξινόμηση και όχι την παλινδρόμηση. Πρόκειται για έναν πιθανοτικό ταξινομητή του οποίου οι πιθανότητες εξόδου μοντελοποιούνται κάνοντας χρήση μιας λογιστικής συνάρτησης, δηλαδή συνάρτησης της μορφής:

$$f(x) = \frac{L}{1 + e^{-k(x-x_0)}}$$

Η γενίκευση της λογιστικής συνάρτησης στην περίπτωση πολλών εισόδων ονομάζεται softmax function και ορίζεται παρακάτω.

Όπως υποδηλώνει το όνομά του, ο Max Entropy βασίζεται στην αρχή της μέγιστης εντροπίας, σύμφωνα με την οποία μεταξύ όλων των μοντέλων που ταιριάζουν με τα δεδομένα επιλέγεται εκείνο που δεν κάνει καμία άλλη υπόθεση πέρα των περιορισμών που επιβάλλονται από το training set και συνεπώς η κατανομή είναι όσο το δυνατόν ομοιόμορφη. Ο ταξινομητής Max Entropy χρησιμοποιείται σε πολλά προβλήματα ταξινόμησης κειμένου, όπως ανίχνευση γλώσσας, ταξινόμηση με βάση το θέμα του κειμένου, sentiment analysis και άλλα.

Ανόμοια με τον επίσης πιθανοτικό ταξινομητή Naive Bayes, που αναπτύχθηκε στην προηγούμενη ενότητα, ο Max Entropy δεν υποθέτει ότι τα χαρακτηριστικά είναι υπό συνθήκη ανεξάρτητα μεταξύ τους. Το γεγονός ότι ο αλγόριθμος μέγιστης εντροπίας δεν κάνει κάποια υπόθεση ανεξαρτησίας μεταξύ των χαρακτηριστικών τον καθιστά ιδιαίτερα αποδοτικό σε προβλήματα ταξινόμησης κειμένου, όπου τα χαρακτηριστικά-λέξεις δεν είναι προφανώς ανεξάρτητα μεταξύ τους. Το μειονέκτημά του σε σχέση με τον Naive Bayes είναι ο μεγαλύτερος χρόνος εκπαίδευσης λόγω του προβλήματος βελτιστοποίησης που πρέπει να επιλυθεί προκειμένου να προσδιοριστούν οι παράμετροί του.

Ας θεωρήσουμε την κλασική BoW αναπαράσταση και έστω $\{w_1, w_2, \ldots, w_n\}$ οι λέξεις του λεξιλογίου. Ο στόχος είναι να κατασκευάσουμε ένα στοχαστικό μοντέλο που θα δέχεται στην είσοδο ένα κείμενο $\boldsymbol{x}$ και θα το αντιστοιχεί σε μία κατηγορία c_j (θετική ή αρνητική για το

πρόβλημά μας). Αρχικά από το training set που έχουμε στη διάθεσή μας, υπολογίζουμε με MLE την εμπειρική πιθανότητα το τυχαίο κείμενο $\boldsymbol{x}$ να ανήκει στην κατηγορία c:

$$\tilde{P}(\boldsymbol{x}, c) = \frac{\alpha\rho\iota\theta\mu\acute{o}\varsigma\ \varphi o\rho\acute{\omega}\nu\ \pi o\upsilon\ \tau o\ \delta\epsilon\acute{\iota}\gamma\mu\alpha\ (x, c)\ \epsilon\mu\varphi\alpha\nu\acute{\iota}\zeta\epsilon\tau\alpha\iota\ \sigma\tau o\ training\ set}{\mu\acute{\epsilon}\gamma\epsilon\theta o\varsigma\ training\ set}$$

Ορίζουμε στη συνέχεια την παρακάτω Boolean συνάρτηση:

$$f_i(\boldsymbol{x}, c) = \begin{cases} 1, & \epsilon\acute{\alpha}\nu\ c = c_j\ \kappa\alpha\iota\ \tau o\ \boldsymbol{x}\ \pi\epsilon\rho\iota\acute{\epsilon}\chi\epsilon\iota\ \tau\eta\ \lambda\acute{\epsilon}\xi\eta\ w_i \\ 0, & \alpha\lambda\lambda\iota\acute{\omega}\varsigma \end{cases}$$

η οποία στην βιβλιογραφία του Max Entropy ονομάζεται χαρακτηριστικό.
Ορίζουμε και τις ακόλουθες δύο προσδοκίες χαρακτηριστικών (feature expectations):

- Αναμενόμενη τιμή χαρακτηριστικού ως προς την εμπειρική κατανομή $\tilde{P}(\boldsymbol{x}, c)$:

$$E(f_i) = \sum_{\boldsymbol{x},c} \tilde{P}(\boldsymbol{x}, c) f_i(\boldsymbol{x}, c) \quad (1)$$

- Αναμενόμενη τιμή χαρακτηριστικού ως προς το μοντέλο $P(c|\boldsymbol{x})$:

$$E(f_i) = \sum_{\boldsymbol{x},c} \tilde{P}(\boldsymbol{x}) P(c|\boldsymbol{x}) f_i(\boldsymbol{x}, c) \quad (2)$$

όπου $\tilde{P}(\boldsymbol{x})$ είναι η εμπειρική κατανομή του x στο training set και συνήθως:

$$\tilde{P}(\boldsymbol{x}) = \frac{1}{\mu\acute{\epsilon}\gamma\epsilon\theta o\varsigma\ training\ set}$$

Επιβάλλοντας τον περιορισμό η αναμενόμενη τιμή να είναι ίση με την εμπειρική τιμή, έχουμε από τις εξισώσεις (1) και (2):

$$\sum_{\boldsymbol{x},c} \tilde{P}(\boldsymbol{x}) P(c|\boldsymbol{x}) f_i(\boldsymbol{x}, c) = \sum_{\boldsymbol{x},c} \tilde{P}(\boldsymbol{x}, c) f_i(\boldsymbol{x}, c) \quad (3)$$

Η εξίσωση (3) καλείται περιορισμός και έχουμε έναν περιορισμό για κάθε χαρακτηριστικό f_i.

Οι παραπάνω περιορισμοί μπορούν να ικανοποιηθούν από άπειρα στοχαστικά μοντέλα. Κάνοντας χρήση της αρχής της μέγιστης εντροπίας, ο αλγόριθμος επιλέγει το μοντέλο που είναι κατά το δυνατόν ομοιόμορφο. Δηλαδή επιλέγει το μοντέλο P^* :

$$P^* = arg\ _{P\epsilon C} max(-\sum_{x,c} \tilde{P}(\boldsymbol{x}) P(c|\boldsymbol{x}) log P(c|\boldsymbol{x}))$$

σύμφωνα με τους περιορισμούς C:

- $P(c|\boldsymbol{x}) \geq 0\ για\ κάθε\ \boldsymbol{x}, c$
- $\sum_c P(c|\boldsymbol{x}) = 1\ για\ κάθε\ \boldsymbol{x}$
- $\sum_{x,c} \tilde{P}(\boldsymbol{x}) P(c|\boldsymbol{x}) f_i(\boldsymbol{x}, c) = \sum_{x,c} \tilde{P}(\boldsymbol{x}, c) f_i(\boldsymbol{x}, c)\ , i = 1, \dots, n$

Το παραπάνω πρόβλημα μετατρέπεται στο δυικό πρόβλημα χωρίς περιορισμούς κάνοντας χρήση των πολλαπλασιαστών Lagrange $\lambda_1, \dots, \lambda_n$. Η εκτίμηση των παραμέτρων λ_i απαιτεί τη χρησιμοποίηση ενός επαναληπτικού αλγορίθμου κλιμάκωσης, όπως ο GIS (Generalized Iterative Scaling) ή ο IIS (Improved Iterative Scaling). Αποδεικνύεται ότι αφού βρούμε τους πολλαπλασιαστές Lagrange, η εκ των υστέρων πιθανότητα το κείμενο $\boldsymbol{x}$ να ανήκει στην κατηγορία c_j δίνεται από την συνάρτηση softmax και είναι:

$$P(c_j|x) = \frac{\exp(\sum_i \lambda_i f_i(\boldsymbol{x}, c_j))}{\sum_c \exp(\sum_i \lambda_i f_i(\boldsymbol{x}, c))}$$

Η παράμετρος λ_i δηλώνει το βάρος του χαρακτηριστικού i στην επιλογή της κλάσης c_j. Μεγάλη θετική τιμή δηλώνει ότι η λέξη i πιθανότατα σχετίζεται με την κλάση c_j, ενώ μεγάλη αρνητική τιμή σημαίνει ότι η λέξη i πιθανότατα δεν σχετίζεται με την κλάση c_j.

2.5 Μηχανές Διανυσμάτων Υποστήριξης

Οι μηχανές διανυσμάτων υποστήριξης (support vector machines ή SVMs απλούστερα) είναι ένα ιδιαίτερα δημοφιλές σύνολο μεθόδων επιβλεπόμενης μάθησης. Τα πλεονεκτήματά τους είναι:

- Τα SVMs είναι αποτελεσματικά σε χώρους πολλών διαστάσεων ακόμη και όταν ο αριθμός των χαρακτηριστικών είναι μεγαλύτερος του αριθμού των δειγμάτων αποφεύγοντας το overfitting
- Χρησιμοποιούν ένα υποσύνολο των παραδειγμάτων εκπαίδευσης για την κατασκευή του συνόρου απόφασης και άρα δεν απαιτούν μεγάλη μνήμη
- Λύνουν γραμμικά και μη γραμμικά προβλήματα διαχωρισμού επιτρέποντας στον χρήστη να χρησιμοποιήσει κάποιες προκαθορισμένες ή και τις δικές του συναρτήσεις πυρήνα (θα μιλήσουμε για αυτές παρακάτω)

Το κύριο μειονέκτημά τους είναι ότι δεν υπολογίζουν απευθείας πιθανότητες όπως οι προηγούμενοι δύο αλγόριθμοι.

Έστω ότι έχουμε ένα πρόβλημα ταξινόμησης δύο κλάσεων, όπως αυτό του sentiment analysis με το οποίο ασχολούμαστε. Οι 2 κλάσεις συμβολίζονται ως C_0 (έστω η negative) και C_1 (έστω η positive) και οι αντίστοιχες ετικέτες σημειώνονται με -1 ή 1 . Το σύνολο εκπαίδευσης αποτελείται από πολλά (έστω N) επισημειωμένα παραδείγματα $(\boldsymbol{x_i}, d_i)$ που ανήκουν σε μία από τις δύο κλάσεις.

2.5.1 Γραμμικά Διαχωρίσιμα Προβλήματα

Έστω ότι το πρόβλημά μας είναι γραμμικά διαχωρίσιμο, δηλαδή όλα τα παραδείγματα που ανήκουν στην κλάση C_0 διαχωρίζονται από ένα υπερεπίπεδο από τα παραδείγματα που ανήκουν στην κλάση C_1. Η εξίσωση μιας τέτοιας επιφάνειας απόφασης είναι:

$$\boldsymbol{w}^T\boldsymbol{x} + b = 0$$

όπου $\boldsymbol{x}$ είναι ένα διάνυσμα εισόδου, $\boldsymbol{w}$ ένα προσαρμόσιμο διάνυσμα βαρών και b είναι μία πόλωση.

Για ένα δεδομένο διάνυσμα βαρών $\boldsymbol{w}$ και πόλωση b, ο διαχωρισμός μεταξύ του υπερεπιπέδου που ορίζει η παραπάνω εξίσωση και του πλησιέστερου σημείου δεδομένων αποκαλείται περιθώριο διαχωρισμού και συμβολίζεται ως ρ. Ο στόχος μιας μηχανής διανυσμάτων υποστήριξης είναι να βρει το συγκεκριμένο υπερεπίπεδο για το οποίο το περιθώριο διαχωρισμού, ρ, μεγιστοποιείται. Υπό αυτή τη συνθήκη, η επιφάνεια απόφασης αναφέρεται ως βέλτιστο υπερεπίπεδο. Αυτό συνάδει και με τη διαίσθησή μας ότι όσο μεγαλύτερο είναι το περιθώριο διαχωρισμού τόσο μικρότερο θα είναι το σφάλμα γενίκευσης. Στο σχήμα 1 απεικονίζεται η γεωμετρική κατασκευή ενός βέλτιστου υπερεπιπέδου για ένα δισδιάστατο χώρο εισόδου.

Αν επιλέξουμε τη πόλωση b έτσι ώστε το περιθώριο διαχωρισμού να είναι ίσο και για τις δύο κλάσεις και κλιμακώσουμε το διάνυσμα βαρών $\boldsymbol{w}$ έτσι ώστε:

$$\boldsymbol{w}^T\boldsymbol{x}_i + b \geq 1\,, \text{εάν } d_i = 1$$

$$\boldsymbol{w}^T\boldsymbol{x}_i + b \leq -1\,, \text{εάν } d_i = -1$$

τότε αποδεικνύεται ότι το περιθώριο διαχωρισμού ρ μεταξύ των 2 κλάσεων είναι ίσο με:

$$\rho = \frac{2}{\|\boldsymbol{w}\|}$$

Έτσι προκύπτει το ακόλουθο πρόβλημα βελτιστοποίησης (P_1) με περιορισμούς:

$$\min_{\boldsymbol{w}} J(\boldsymbol{w}) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w}$$

$$d_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1\,, i = 1, \dots, N$$

Διανύσματα υποστήριξης (support vectors) καλούνται όλα τα πρότυπα $\boldsymbol{x}_k$ για τα οποία:

$$d_k(\boldsymbol{w}^T\boldsymbol{x}_k + b) = 1$$

Από το (P_1) το διάνυσμα βαρών προκύπτει ίσο με: $\boldsymbol{w} = \sum_{i=1}^{N} a_i d_i \boldsymbol{x}_i$ όπου a_i είναι οι πολλαπλασιαστές Lagrange που εισάγονται από τους περιορισμούς. Από τις συνθήκες Karush-Kuhn-Tucker (KKT) μόνο εκείνοι οι πολλαπλασιαστές Lagrange που αντιστοιχούν στα διανύσματα υποστήριξης λαμβάνουν μη μηδενικές τιμές και άρα το βέλτιστο $\boldsymbol{w}$ είναι ένας γραμμικός συνδυασμός των διανυσμάτων υποστήριξης (δ.υ.) και μόνο.

Το (P_1) μπορεί να αποδειχτεί ότι είναι ισοδύναμο με το λεγόμενο δυικό πρόβλημα (P_2) το οποίο εκφράζεται συναρτήσει των πολλαπλασιαστών Lagrange $\alpha_1, \dots, \alpha_N$ και είναι:

$$\min_{\boldsymbol{a}} Q(\boldsymbol{a}) = \frac{1}{2}\boldsymbol{a}^T Q \boldsymbol{a} - \boldsymbol{e}^T \boldsymbol{a}$$

$$\sum_{i=1}^{N} a_i d_i = 0,\ a_i \geq 0,\ \ i = 1, \dots, N$$

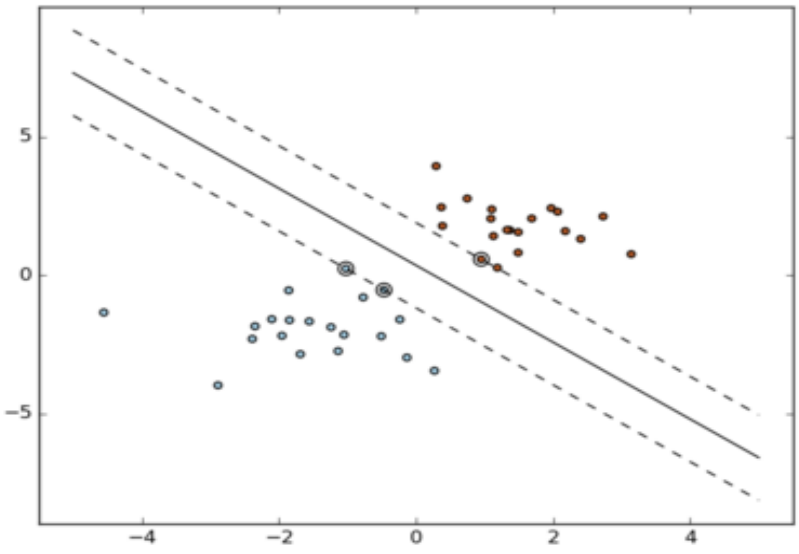
Όπου :

- e είναι ένα διάνυσμα μήκους N με όλα τα στοιχεία του ίσα με 1
- Q ένας nxn θετικά ημιορισμένος πίνακας με στοιχεία $Q_{ij} = d_i d_j \boldsymbol{x}_i^T \boldsymbol{x}_j$

Από την επίλυση του (P_2) προκύπτουν οι τιμές για το βέλτιστο υπερεπίπεδο:

$$\boldsymbol{w}_o = \sum_{i=1}^{N_s} a_i d_i \boldsymbol{x_i}\,, N_s = \pi\lambda\acute{\eta}\theta o\varsigma\ \tau\omega\nu\ \delta.\upsilon.$$

$$b_o = 1 - \boldsymbol{w}_o^T \boldsymbol{x}, \acute{o}\pi o\upsilon\ \boldsymbol{x}\ \acute{\varepsilon}\nu\alpha\ \delta.\upsilon.\ \mu\varepsilon\ \acute{\varepsilon}\xi o\delta o = 1$$



Σχήμα 1: Βέλτιστο υπερεπίπεδο για γραμμικά διαχωρίσιμο πρόβλημα

2.5.2 Μη Γραμμικά Διαχωρίσιμα Προβλήματα – Μεταβλητές Χαλαρότητας

Αν το πρόβλημα δεν είναι γραμμικά διαχωρίσιμο, τότε εισάγουμε για κάθε πρότυπο μια μεταβλητή χαλαρότητας $\xi_i \geq 0$ έτσι ώστε:

$$d_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) \geq 1 - \xi_i \quad (4)$$

Αν $\xi_i > 1$, το πρότυπο $\boldsymbol{x}_i$ έχει ταξινομηθεί λάθος, ενώ αν $0 < \xi_i \leq 1$, το πρότυπο ταξινομείται σωστά αλλά εμπίπτει μέσα στην περιοχή διαχωρισμού.

Προσθέτουμε στην αρχική συνάρτηση κόστους $J(w)$ ένα κόστος ανάλογο του όρου $\sum_{i=1}^{N} \xi_i$, ο οποίος όρος είναι ένα άνω φράγμα στον αριθμό των λάθος ταξινομήσεων . Έτσι το (P_1) μετατρέπεται στο (P_3):

$$\min_{\boldsymbol{w},\boldsymbol{\xi}} J(\boldsymbol{w}, \boldsymbol{\xi}) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{i=1}^{N} \xi_i$$

$$d_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) \geq 1 - \xi_i, \xi_i \geq 0, i = 1, \dots, N$$

Η παράμετρος C δίνεται από το χρήστη και ανάλογα με την τιμή της δίνουμε περισσότερη έμφαση στη σωστή ταξινόμηση των προτύπων ή στη μεγιστοποίηση του περιθωρίου: Όταν η C λαμβάνει

μεγάλες τιμές, ο χρήστης έχει μεγάλη εμπιστοσύνη στην ποιότητα του συνόλου εκπαίδευσης και ο αλγόριθμος προσπαθεί να ταξινομήσει όλα τα πρότυπα σωστά. Όταν η C λαμβάνει μικρές τιμές, το σύνολο εκπαίδευσης θεωρείται θορυβώδες και ο αλγόριθμος δίνει μεγαλύτερη έμφαση στην ομαλοποίηση της επιφάνειας απόφασης.

Αντίστοιχα με το (P2) προκύπτει και εδώ το δυικό πρόβλημα (P4):

$$\min_{\boldsymbol{a}} Q(\boldsymbol{a}) = \frac{1}{2}\boldsymbol{a}^T Q\boldsymbol{a} - \boldsymbol{e}^T\boldsymbol{a}$$

$$\sum_{i=1}^{N} a_i d_i = 0,\ 0 \le a_i \le C,\ \ i = 1, \dots, N$$

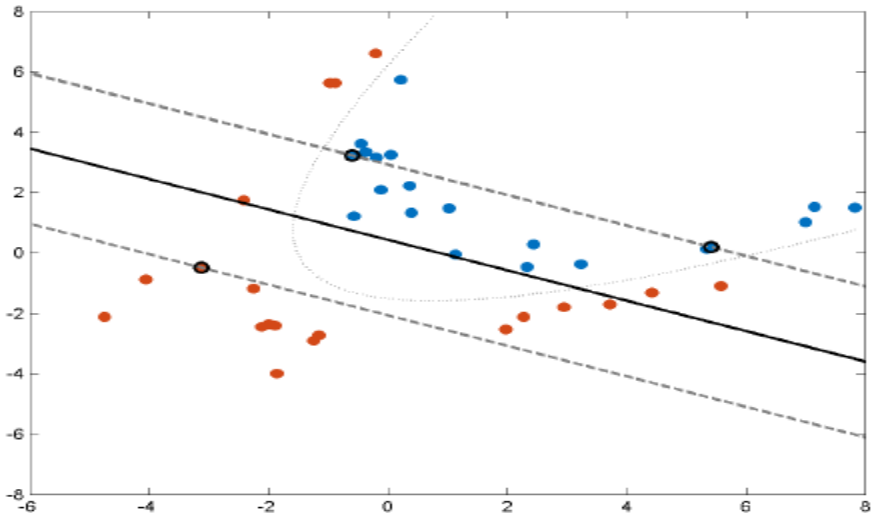
και το βέλτιστο υπερεπίπεδο είναι:

$$\boldsymbol{w}_o = \sum_{i=1}^{N} a_i d_i \boldsymbol{x_i}$$

$$b_o = 1 - \boldsymbol{w}_o^T\boldsymbol{x}, \text{όπου } \boldsymbol{x} \text{ ένα } \delta.\upsilon. \text{με έξοδο} = 1$$

Τα διανύσματα υποστήριξης ορίζονται με τον ίδιο τρόπο όπως πριν, ως τα διανύσματα $\boldsymbol{x}_i$ για τα οποία η ανισότητα (4) ισχύει ως ισότητα ακόμη και αν $\xi_i > 0$. Για τους αντίστοιχους πολλαπλασιαστές Lagrange ισχύει $a_i > 0$.

Στο σχήμα 2 φαίνεται η κατασκευή του βέλτιστου υπερεπιπέδου για ένα μη γραμμικά διαχωρίσιμο πρόβλημα.



Σχήμα 2: Βέλτιστο υπερεπίπεδο για μη γραμμικά διαχωρίσιμο πρόβλημα

2.5.3 Μη Γραμμικά Διαχωρίσιμα Προβλήματα – Συναρτήσεις Πυρήνα

Αν το πρόβλημα δεν είναι γραμμικά διαχωρίσιμο, τότε μπορούμε να το αναγάγουμε σε μεγαλύτερη (ενδεχομένως και άπειρη) διάσταση, κάνοντας χρήση του μετασχηματισμού

$$\boldsymbol{x} \rightarrow \Phi(\boldsymbol{x})$$

Στο νέο χώρο μεγαλύτερης διάστασης, τα παραδείγματα εκπαίδευσης γίνονται «αραιά» και το πρόβλημα μετατρέπεται σε γραμμικά διαχωρίσιμο. Ο υπολογισμός του $\Phi(\boldsymbol{x})$ είναι εξαιρετικά πολύπλοκος όσο αυξάνεται ο αριθμός των διαστάσεων. Ευτυχώς όμως δεν χρειάζεται να υπολογίσουμε το $\Phi(\boldsymbol{x})$ αλλά το εσωτερικό γινόμενο $K(\boldsymbol{x}, \boldsymbol{y}) = \Phi(\boldsymbol{x})^T \Phi(\boldsymbol{y})$ που είναι απλά ένας αριθμός. Η συνάρτηση καλείται πυρήνας (kernel) και συνήθως είναι μία από τις επόμενες:

- Γκαουσιανή Rbf: $K(\boldsymbol{x}, \boldsymbol{y}) = \exp(-\gamma \|\boldsymbol{x} - \boldsymbol{y}\|^2)$
- Πολυωνυμική: $K(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{x}^T \boldsymbol{y} + r)^d$
- Σιγμοειδής: $K(\boldsymbol{x}, \boldsymbol{y}) = \tanh(\gamma \boldsymbol{x}^T \boldsymbol{y} + r)$

Ειδικότερα για την rbf συνάρτηση πυρήνα, η παράμετρος γ ορίζει πόσο μακριά φτάνει η επιρροή ενός παραδείγματος εκπαίδευσης με τις χαμηλές τιμές να σημαίνουν «μακριά» και τις μεγάλες «κοντά». Δηλαδή, η παράμετρος γ μπορεί να θεωρηθεί ως το αντίστροφο της ακτίνας επιρροής των δειγμάτων που επιλέγονται από το μοντέλο ως διανύσματα υποστήριξης.

Τα προβλήματα βελτιστοποίησης, πρωτεύον και δυικό, ορίζονται ακριβώς αντίστοιχα με τα (P_1) και (P_2) ή (P_3) και (P_4), και στη γενικότερη περίπτωση όπου ο χρήστης εισάγει την παράμετρο ομαλοποίησης C, η συνάρτηση απόφασης είναι:

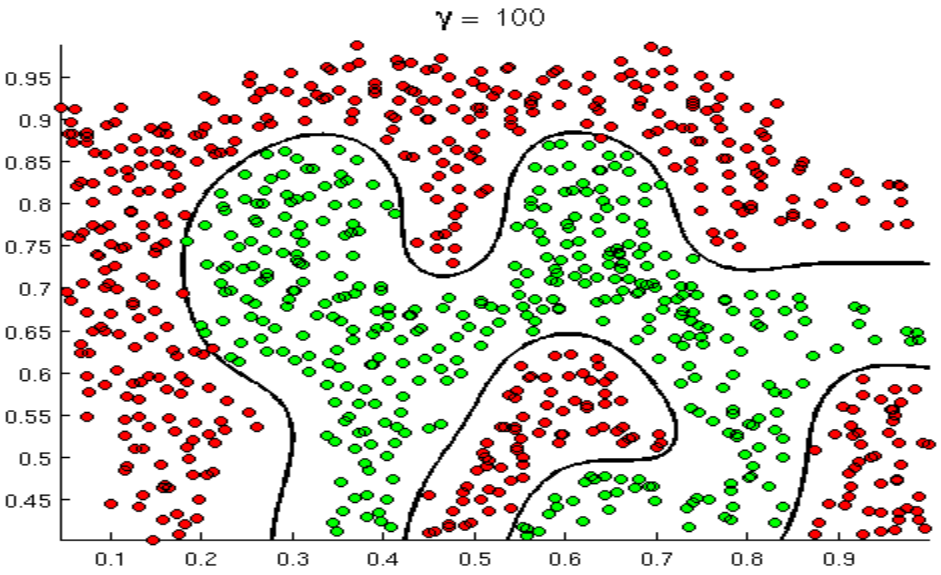
$$sgn(\sum_{i=1}^{N} a_i d_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b_o)$$

όπου τα a_i προκύπτουν από την επίλυση του (P_4) (με τον πίνακα Q να έχει τώρα στοιχεία $Q_{ij} = d_i d_j K(\boldsymbol{x_i}, \boldsymbol{x_j})$) και η πόλωση b_o είναι ανεξάρτητη της συνάρτησης Φ και δίνεται από τη σχέση:

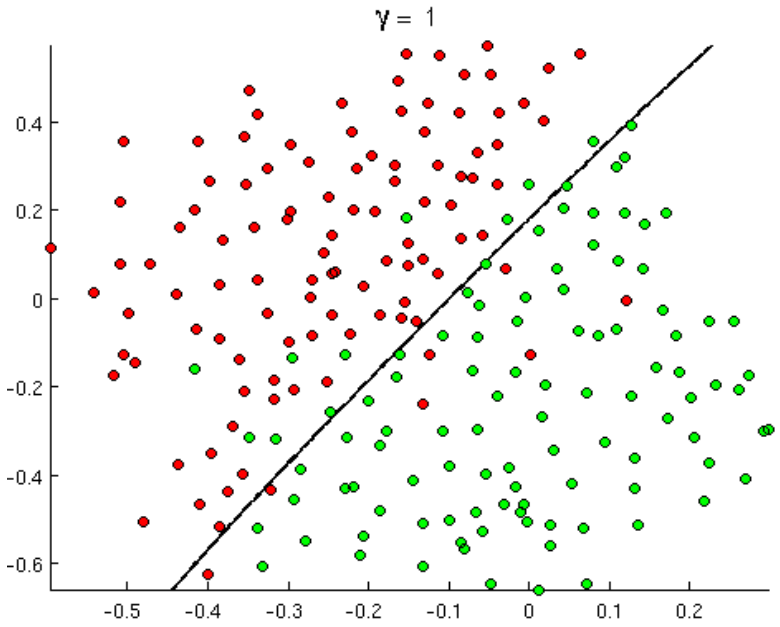
$$b_o = 1 - w_o^T \Phi(\boldsymbol{x}) = 1 - \sum_{i=1}^{N} a_i d_i \Phi(\boldsymbol{x_i})^T \Phi(\boldsymbol{x}) = 1 - \sum_{i=1}^{N} a_i d_i K(\boldsymbol{x}_i, \boldsymbol{x})$$

όπου $\Phi(\boldsymbol{x})$ ένα δ.υ. με έξοδο = 1.

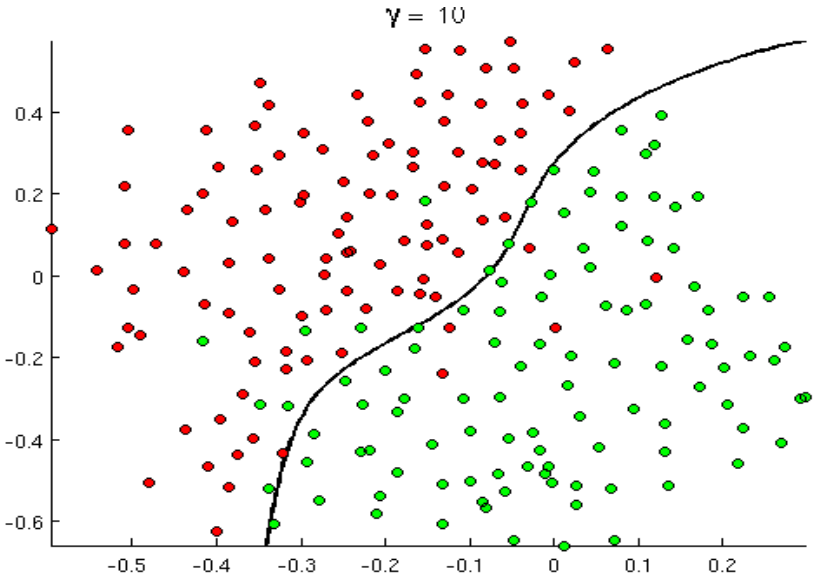
Στο σχήμα 3 φαίνεται η κατασκευή του decision boundary κάνοντας χρήση του rbf πυρήνα, ενώ στα σχήμα 4,5,6 και 7 φαίνεται η επιρροή της παραμέτρου γ του rbf πυρήνα στην κατασκευή του decision boundary οδηγώντας τελικά σε overfitting .



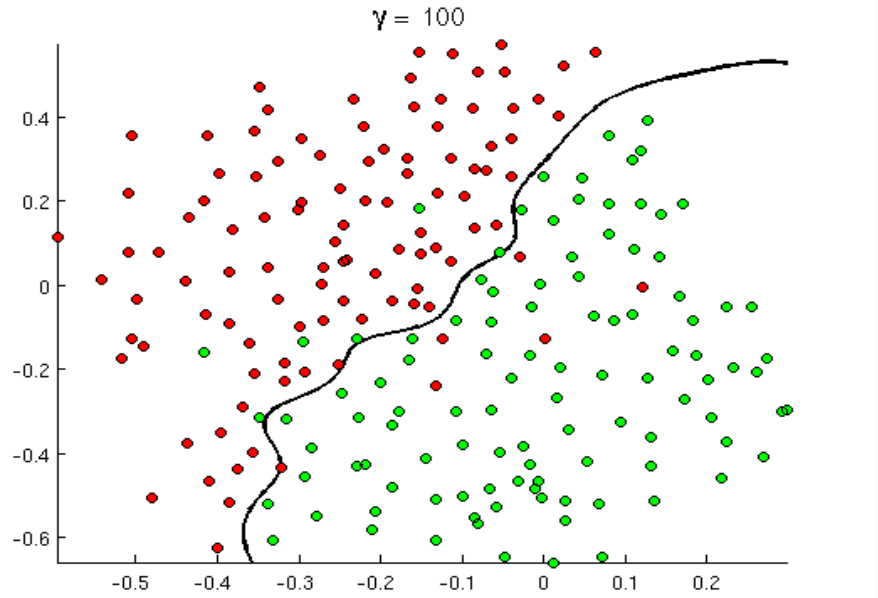
Σχήμα 3: RBF kernel



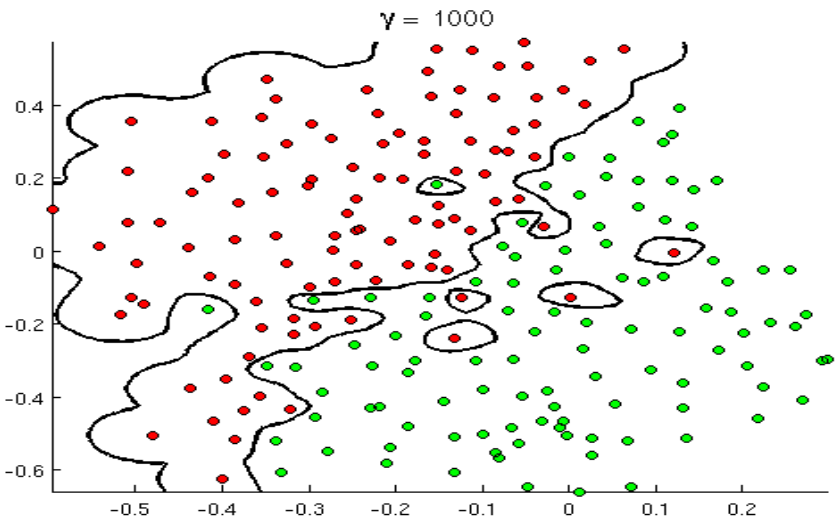
Σχήμα 4: RBF kernel με $\gamma = 1$



Σχήμα 5: RBF kernel με $\gamma = 10$



Σχήμα 6: RBF kernel με $\gamma = 100$



Σχήμα 7: RBF kernel με $\gamma = 1000$

2.6 Τεχνητά Νευρωνικά Δίκτυα

Τα τεχνητά νευρωνικά δίκτυα [8] (artificial neural networks), ή απλούστερα νευρωνικά δίκτυα, εμπνευσμένα από τη λειτουργία των νευρώνων και του εγκεφάλου, χρησιμοποιούνται ευρέως σε πολλά προβλήματα επιβλεπόμενης μηχανικής μάθησης και επιτυγχάνουν αρκετά καλά αποτελέσματα. Τα νευρωνικά δίκτυα, αν και ξεκίνησαν ως προσπάθεια μοντελοποίησης της συμπεριφοράς του ανθρώπινου εγκεφάλου, απέκτησαν τα τελευταία χρόνια διαφορετική πορεία εξέλιξης από αυτή της νευροβιολογίας, χωρίς όμως να παύουν να υφίστανται κάποιες αναλογίες. Αυτό το μοντέλο μηχανικής μάθησης εκπαιδεύεται (ελαχιστοποιώντας μια συνάρτηση κόστους όπως θα δούμε) με σκοπό τη ρύθμιση των εσωτερικών του παραμέτρων που καλούνται συναπτικά βάρη. Η βασική δομική του μονάδα είναι ο τεχνητός νευρώνας, η απλούστερη μορφή του οποίου είναι το perceptron του Rosenblatt.

2.6.1 Perceptron

Ο νευρώνας perceptron του Rosenblatt είναι το απλούστερο νευρωνικό δίκτυο και αποτελείται από ένα μόνο νευρώνα. Σύμφωνα με το μοντέλο αυτό, ο νευρώνας:

- Δέχεται m σήματα εισόδου, $x_1, x_2, \dots, x_m$
- Δέχεται μία σταθερή είσοδο $x_0 = 1$ που αντιστοιχεί στη πόλωση b
- Υπολογίζει τον γραμμικό συνδυασμό $v = \boldsymbol{w}^T \boldsymbol{x}$, όπου $\boldsymbol{w} = (b, w_1, w_2, \dots, w_m)^T$ είναι το επαυξημένο διάνυσμα βαρών, b η εξωτερικά εφαρμοζόμενη πόλωση και $w_1, w_2, \dots, w_m$ τα συναπτικά βάρη του perceptron που αντιστοιχούν στις εισόδους $x_1, x_2, \dots, x_m$

- Περνά το γραμμικό συνδυασμό v μέσα από έναν απότομο περιοριστή φ, που ονομάζεται συνάρτηση ενεργοποίησης (ή συνάρτηση μεταφοράς) και είναι είτε η μοναδιαία βηματική (unit step function με τιμές 0/1) είτε η συνάρτηση προσήμου (sign function με τιμές -1/1) και παράγει τελικά το σήμα εξόδου y

Οι δυνατότητες ωστόσο του perceptron είναι περιορισμένες. Μπορεί να επιλύει μόνο γραμμικά διαχωρίσιμα προβλήματα κατασκευάζοντας ένα υπερεπίπεδο $\boldsymbol{w}^T\boldsymbol{x} + b = 0$ του οποίου οι παράμετροι $\boldsymbol{w}$ και b βρίσκονται από το κανόνα εκπαίδευσης του perceptron που περιγράφεται παρακάτω. Έτσι, ενώ μπορεί να υλοποιεί κάποιες λογικές συναρτήσεις, όπως τις συναρτήσεις AND, OR και NOT, δεν μπορεί να υλοποιήσει την συνάρτηση XOR, αφού όπως φαίνεται και στο σχήμα 8 το πρόβλημα αυτό είναι μη γραμμικά διαχωρίσιμο.

Κανόνας Εκπαίδευσης Perceptron

Δεδομένα: N πρότυπα εισόδου $x_1, \dots, x_N$ μαζί με τα αντίστοιχα διανύσματα επιθυμητών αποκρίσεων $d_1, \dots, d_N$

1. Αρχικοποιούμε το επαυξημένο διάνυσμα βαρών $\boldsymbol{w}(0) = 0$
2. Εισάγουμε τα πρότυπα με τη σειρά (η κυκλική παρουσίαση όλων των προτύπων συνιστά μία εποχή). Για κάθε πρότυπο:

2a. Υπολογίζουμε την απόκριση του perceptron ως

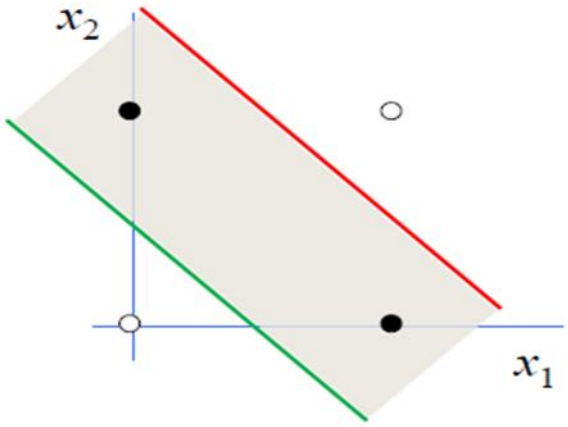
$$y(n) = sgn(\boldsymbol{w}^{\boldsymbol{T}}(n)\boldsymbol{x}(n))$$

2b. Ενημερώνουμε το διάνυσμα βαρών του perceptron σύμφωνα με τον κανόνα

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) + n\big(d(n) - y(n)\big)\boldsymbol{x}(n)$$

Ο αλγόριθμος τερματίζεται όταν ταξινομούνται σωστά όλα τα πρότυπα. Σε μη γραμμικά διαχωρίσιμα προβλήματα, ο αλγόριθμος δεν τερματίζεται ποτέ.

Η παράμετρος n ονομάζεται ρυθμός μάθησης. Μεγάλη τιμή του n μπορεί να οδηγήσει σε γρηγορότερη σύγκλιση αλλά και σε ταλάντωση γύρω από τις βέλτιστες τιμές βαρών. Από την άλλη, μικρή τιμή του n, έχει ως αποτέλεσμα πιο αργή σύγκλιση.



Σχήμα 8: Το πρόβλημα της XOR είναι μη γραμμικά διαχωρίσιμο και μπορεί να επιλυθεί από MLP

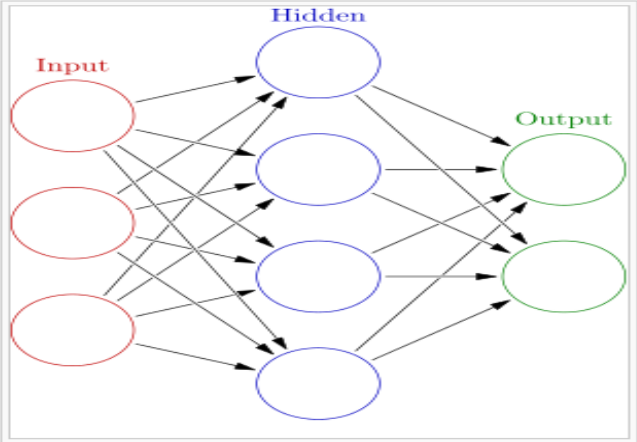
2.6.2 Νευρωνικά Δίκτυα Πολλών Επιπέδων

Προβλήματα που δεν είναι γραμμικά διαχωρίσιμα μπορούν να επιλυθούν από τεχνητά νευρωνικά δίκτυα πολλών επιπέδων, τα οποία καταχρηστικά ονομάζονται και πολυστρωματικά perceptrons (multilayer perceptrons – MLP). Ένα δίκτυο MLP αποτελείται από:

- Το επίπεδο εισόδου, το οποίο απλά στέλνει τα σήματα εισόδου σε όλους τους νευρώνες του κρυφού επιπέδου
- Ένα ή περισσότερα κρυφά επίπεδα (hidden layers) μη γραμμικών νευρώνων
- Το επίπεδο εξόδου, το οποίο αποτελείται από γραμμικούς ή μη γραμμικούς νευρώνες. Η επιλογή αυτή εξαρτάται συνήθως από την εκάστοτε εφαρμογή. Αν έχουμε πρόβλημα προσέγγισης συνάρτησης επιλέγονται συνήθως γραμμικοί νευρώνες, ενώ αν έχουμε πρόβλημα ταξινόμησης επιλέγονται στην έξοδο μη γραμμικοί νευρώνες.

Σύμφωνα με το θεώρημα του καθολικού προσεγγιστή (universal approximator) ένα κρυφό επίπεδο μη γραμμικών νευρώνων αρκεί για την προσέγγιση οποιασδήποτε συνεχούς συνάρτησης ή αντιστοιχίας εισόδου-εξόδου. Γι’ αυτό συνήθως εξετάζονται νευρωνικά δίκτυα 3 επιπέδων, με ένα δηλαδή κρυφό επίπεδο και δεν μελετώνται πιο βαθιές αρχιτεκτονικές. Όσο μάλιστα πιο πολύπλοκη είναι η συνάρτηση που θέλουμε να προσεγγίσουμε τόσο περισσότερους κρυφούς νευρώνες χρειαζόμαστε.

Η αρχιτεκτονική ενός δικτύου 3 επιπέδων με 3 εισόδους και 2 εξόδους φαίνεται στο σχήμα 9.



Σχήμα 9: MLP 3 επιπέδων

Η συνάρτηση ενεργοποίησης ενός τυχαίου νευρώνα του δικτύου είναι απαραίτητο να είναι διαφορίσιμη, ώστε να μπορεί να γίνει η εκπαίδευση με τον κανόνα του gradient descent και συνήθως είναι μία από τις ακόλουθες:

- Λογιστική Συνάρτηση. Αυτή η μορφή σιγμοειδούς μη γραμμικότητας, στη γενική μορφή της, ορίζεται ως:

$$\varphi(v) = \frac{1}{1 + \exp(-av)}$$

όπου α μια θετική παράμετρος.

- Συνάρτηση υπερβολικής εφαπτομένης. Μια άλλη ευρέως χρησιμοποιούμενη μορφή σιγμοειδούς μη γραμμικότητας είναι η συνάρτηση υπερβολικής εφαπτομένης, η οποία στη γενική της μορφή ορίζεται ως:

$$\varphi(v) = atanh(bv) = a\frac{\exp(bv) - \exp(-bv)}{\exp(bv) + \exp(-bv)}$$

όπου α και b θετικές σταθερές.

- Γραμμική Συνάρτηση που ορίζεται ως:

$$\varphi(v) = av$$

όπου α μια θετική σταθερά. Συνήθως επιλέγεται για τους νευρώνες εξόδου ενός δικτύου με στόχο την παλινδρόμηση.

- Softmax Συνάρτηση. Συνήθως επιλέγεται για τους νευρώνες εξόδου ενός δικτύου με στόχο την ταξινόμηση, αφού απεικονίζει το διάνυσμα εισόδου σε διάνυσμα εξόδου με στοιχεία στο διάστημα [0,1] και με ερμηνεία πιθανότητας. Για τον i νευρώνα εξόδου και για επίπεδο εξόδου με k νευρώνες (όσοι και οι κατηγορίες) ορίζεται ως:

$$\varphi(v_i) = \frac{e^{v_i}}{\sum_{j=1}^{k} e^{v_j}}$$

2.6.3 Εκπαίδευση MLP – Αλγόριθμος Backpropagation

Αφού ορίσουμε την αρχιτεκτονική του δικτύου, δηλαδή επιλέξουμε τον αριθμό των επιπέδων, τον αριθμό των νευρώνων ανά επίπεδο και τις συναρτήσεις ενεργοποίησης σε κάθε επίπεδο, καλούμαστε να το εκπαιδεύσουμε έχοντας στη διάθεσή μας ένα επισημειωμένο σύνολο παραδειγμάτων εκπαίδευσης της μορφής:

$$\{(\boldsymbol{x_1}, \boldsymbol{d}_1), (\boldsymbol{x_2}, \boldsymbol{d}_2), \dots, (\boldsymbol{x_N}, \boldsymbol{d}_N)\}$$

Για ένα πρόβλημα ταξινόμησης με k κατηγορίες οι επιθυμητές έξοδοι $\boldsymbol{d_i}$ κωδικοποιούνται ως one-hot vectors διάστασης k με την μονάδα να δηλώνει την αντίστοιχη κατηγορία. Έτσι το νευρωνικό θα έχει k νευρώνες εξόδου, ενώ οι νευρώνες εισόδου είναι φυσικά ίσοι με τη διάσταση του feature vector $\boldsymbol{x}$. Για το πλήθος των κρυφών νευρώνων δεν υπάρχει κάποιος συγκεκριμένος κανόνας και συνήθως γίνεται κάποιος πειραματισμός.

Η εκπαίδευση του νευρωνικού δικτύου έχει ως στόχο την κατάλληλη επιλογή των συναπτικών βαρών και πολώσεων για όλους τους νευρώνες ώστε να παράγονται οι σωστές έξοδοι για κάθε είσοδο. Η επιλογή αυτή γίνεται με βάση την ελαχιστοποίηση μιας κατάλληλης συνάρτησης κόστους J και η διόρθωση των βαρών κατευθύνεται από το επίπεδο εξόδου προς το επίπεδο εισόδου με έναν κανόνα που ονομάζεται backpropagation (οπισθοδιάδοση σφάλματος). Ο αλγόριθμος διεξάγεται σε 2 φάσεις:

- Φάση εμπρόσθιας διάδοσης (forward propagation): Το διάνυσμα εισόδου $\boldsymbol{x}$ εφαρμόζεται στο επίπεδο εισόδου και αφού γίνουν οι υπολογισμοί σε κάθε επίπεδο, παράγεται τελικά το διάνυσμα εξόδου $\boldsymbol{y} = (y_1, \dots, y_k)^T$
- Φάση ανάστροφης διάδοσης (backward propagation): Αφού υπολογιστεί το διάνυσμα εξόδου $\boldsymbol{y}$ που αντιστοιχεί στο διάνυσμα εισόδου $\boldsymbol{x}$ μπορεί να συγκριθεί με το στόχο $\boldsymbol{d}$

και να υπολογιστεί ένα κριτήριο μέτρησης του σφάλματος που κάνει το δίκτυο για το συγκεκριμένο πρότυπο. Τέτοια κριτήρια μπορεί να είναι το στιγμιαίο τετραγωνικό σφάλμα (για γραμμικό ή μη γραμμικό-σιγμοειδή επίπεδο εξόδου)

$$J(\boldsymbol{y}, \boldsymbol{d}) = \frac{1}{2}\|\boldsymbol{d} - \boldsymbol{y}\|^2$$

ή το στιγμαίο κόστος διεντροπίας (cross-entropy error, για softmax επίπεδο εξόδου)

$$J(\boldsymbol{y}, \boldsymbol{d}) = -\sum_{i=1}^{k} d_i ln y_i = -ln\, y_m$$

όπου m η κατηγορία του προτύπου εισόδου $\boldsymbol{x}$.
Οι παραπάνω συναρτήσεις J εξαρτώνται από την έξοδο $\boldsymbol{y}$ του δικτύου και άρα και από τα βάρη σε κάθε επίπεδο. Άρα, αν υπολογίσουμε τις μερικές παραγώγους του κριτηρίου J, πρώτα ως προς τα βάρη του επιπέδου εξόδου και μετά χρησιμοποιώντας τον κανόνα της αλυσίδας και ως προς τα βάρη των προηγούμενων επιπέδων, έχουμε τις ευαισθησίες του J ως προς όλες τις προσαρμόσιμες παραμέτρους του δικτύου. Στη συνέχεια εφαρμόζουμε τον κανόνα της κατάβασης δυναμικού/κλίσης (gradient descent) για την ανανέωση των παραμέτρων, δηλαδή:

$$\boldsymbol{w} = \boldsymbol{w} - n\nabla J(\boldsymbol{w}),$$

(όπου $\boldsymbol{w}$ το διάνυσμα βαρών ενός νευρώνα και n μια μικρή θετική παράμετρος γνωστή ως ρυθμός μάθησης) και συνεχίζουμε με τη φάση εμπρόσθιας διάδοσης και το επόμενο πρότυπο εισόδου.

Η κυκλική παρουσίαση όλων των παραδειγμάτων $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ στο δίκτυο κατά την εκπαίδευση ονομάζεται εποχή. Η εκπαίδευση μπορεί να τερματιστεί με διάφορα κριτήρια, όπως η επιλογή κατωφλίου για την ελαχιστοποίηση του μέσου τετραγωνικού σφάλματος ή του μέσου κόστους διεντροπίας ή απλά ο μέγιστος αριθμός επαναλήψεων. Ακόμη μπορούμε για την αποφυγή της υπερπροσαρμογής (overfitting) του δικτύου μας στα δεδομένα εκπαίδευσης, φαινόμενο που σχετίζεται με την κακή ικανότητα γενίκευσης του μοντέλου μας, να χρησιμοποιήσουμε κάποια από τα δείγματα εκπαίδευσης , που λέμε ότι συνιστούν το σύνολο επικύρωσης (validation set) για την περιοδική εξέταση της απόδοσης σε άγνωστα δεδομένα στο τέλος κάθε εποχής. Η εκπαίδευση τότε μπορεί να τερματίζεται όταν η απόδοση πάνω στο validation set αδυνατεί να πέσει για κάποιο συνεχόμενο αριθμό φορών (Early Stopping).

Υπάρχουν 2 είδη μάθησης ανάλογα με τη συνάρτηση που ελαχιστοποιείται και το πότε γίνονται οι ανανεώσεις των βαρών. Παραπάνω περιγράψαμε την on-line μάθηση (on-line learning), όπου οι διορθώσεις των βαρών εκτελούνται σε βάση παράδειγμα προς παράδειγμα και συνεπώς η συνάρτηση κόστους προς ελαχιστοποίηση είναι το στιγμιαίο τετραγωνικό σφάλμα ή το στιγμιαίο

κόστος διεντροπίας. Υπάρχει όμως και η μαζική μάθηση (batch learning), όπου οι προσαρμογές στα συναπτικά βάρη του perceptron πολλών επιπέδων εκτελούνται μετά την παρουσίαση του συνόλου των N παραδειγμάτων εκπαίδευσης και συνεπώς η συνάρτηση κόστους προς ελαχιστοποίηση είναι το μέσο τετραγωνικό σφάλμα ή το μέσο κόστος διεντροπίας.

Η on-line μάθηση είναι απλή στην υλοποίηση και συγκλίνει πιο γρήγορα. Από την άλλη, η μαζική μάθηση έχει το πλεονέκτημα ότι μπορεί να παραλληλοποιηθεί, αλλά απαιτεί περισσότερο χώρο αποθήκευσης για την ανανέωση των βαρών και μεγαλύτερο αριθμό εποχών εκπαίδευσης.

Κεφάλαιο 3

Συνελικτικά Νευρωνικά Δίκτυα

Στη μηχανική μάθηση, τα συνελικτικά νευρωνικά δίκτυα (Convolutional Neural Networks – CNNs) είναι ένας τύπος ευθείας τροφοδότησης (feed-forward) τεχνητών νευρωνικών δικτύων στο οποίο το pattern σύνδεσης μεταξύ των νευρώνων είναι εμπνευσμένο από την οργάνωση του οπτικού φλοιού των ζώων, όπου οι νευρώνες είναι οργανωμένοι κατά τέτοιο τρόπο ώστε να αποκρίνονται σε επικαλυπτόμενες περιοχές του οπτικού πεδίου.

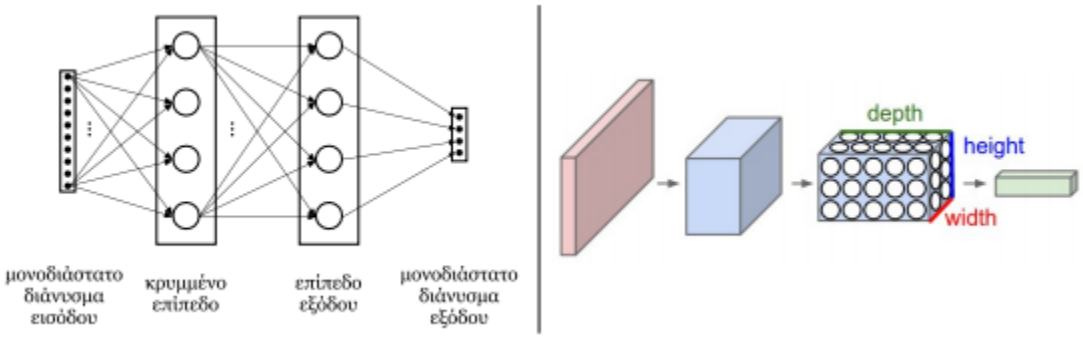
Τα ΣΝΔ ανήκουν στην κατηγορία των αλγορίθμων βαθιάς μάθησης (deep learning). Η βαθιά μάθηση είναι ένας ευρύτερος κλάδος της μηχανικής μάθησης με μεθόδους που βασίζονται στη μάθηση αναπαραστάσεων των δεδομένων. Μία παρατήρηση (πχ. μία εικόνα) μπορεί να αναπαρασταθεί με πολλούς τρόπους, όπως ένα διάνυσμα με τιμές έντασης φωτεινότητας ανά pixel ή με ένα πιο αφηρημένο (υψηλότερου επιπέδου) τρόπο ως ένα σύνολο ακμών, περιοχών συγκεκριμένου σχήματος κ.τ.λ. Κάποιες αναπαραστάσεις είναι καλύτερες από κάποιες άλλες στην απλοποίηση του έργου μάθησης (πχ. στην αναγνώριση προσώπου ή στην αναγνώριση έκφρασης προσώπου) από τα παραδείγματα. Το πλεονέκτημα του deep learning έναντι του machine learning είναι ότι αντικαθιστά τους handcrafted κανόνες εξαγωγής χαρακτηριστικών με αποδοτικούς αλγορίθμους ιεραρχικής μάθησης και εξαγωγής χαρακτηριστικών εφαρμόζοντας πολλαπλά επίπεδα μη γραμμικών μετασχηματισμών όπως θα δούμε στη συνέχεια.

Τα συνελικτικά νευρωνικά δίκτυα εφαρμόζονται σε μία πληθώρα εφαρμογών της όρασης υπολογιστών (computer vision). Από προβλήματα χαμηλού επιπέδου, όπως η αποθορυβοποίηση (denoising), η όξυνση (sharpening) και η εύρεση ακμών (edge detection) σε εικόνες, μέχρι και σε προβλήματα υψηλότερου επιπέδου, όπως η ταξινόμηση (image classification), η αναγνώριση προσώπων ή αντικειμένων (face/object recognition) και η εκτίμηση βάθους (depth estimation), τα συνελικτικά νευρωνικά δίκτυα βρίσκουν συνεχώς νέες εφαρμογές τα τελευταία χρόνια με σπουδαία αποτελέσματα. Αν και τα ΣΝΔ είναι άρρηκτα συνδεδεμένα με την όραση υπολογιστών και τις εφαρμογές της, τον τελευταίο καιρό έχει γίνει μία προσπάθεια εφαρμογής τους σε προβλήματα επεξεργασίας φυσικής γλώσσας (NLP) και σε κάποιες περιπτώσεις τα αποτελέσματα είναι ενδιαφέροντα.

Στο παρόν κεφάλαιο θα περιγράψουμε την αρχιτεκτονική των συνελικτικών νευρωνικών δικτύων και πώς αυτά μπορούν να εφαρμοστούν σε ένα πρόβλημα επεξεργασίας φυσικής γλώσσας, όπως είναι αυτό του sentiment analysis.

3.1 Αρχιτεκτονική ΣΝΔ

Στην παραδοσιακή μηχανική μάθηση, ένα νευρωνικό δίκτυο δέχεται στην είσοδό του ένα διάνυσμα χαρακτηριστικών $\boldsymbol{x}$ σταθερής διάστασης. Απεναντίας, στο deep learning, ένα ΣΝΔ δέχεται στην είσοδό του μία εικόνα, η οποία μπορεί να είναι grayscale και άρα οργανωμένη σε ένα διδιάστατο πίνακα, ή RGB και άρα οργανωμένη σε ένα τριδιάστατο πίνακα. Η διαφορά αυτή των δύο φαίνεται στο σχήμα 10.



Σχήμα 10: Αριστερά: Ένα 3-layer MLP, Δεξιά: Ένα ΣΝΔ

Μία άλλη διαφορά είναι ότι στο MLP η βασική πράξη είναι το εσωτερικό γινόμενο, ενώ στα ΣΝΔ η βασική πράξη είναι η συνέλιξη, όπως υποδηλώνει και το όνομά τους.

Ένα ΣΝΔ στη γενική περίπτωση της ταξινόμησης, αποτελείται από κάποια μη γραμμικά επίπεδα εξαγωγής χαρακτηριστικών και έναν ταξινομητή (συνήθως επίπεδο softmax) στην έξοδο που δέχεται τα χαρακτηριστικά αυτά και εκτελεί την ταξινόμηση. Τα βασικά επίπεδα των ΣΝΔ είναι:

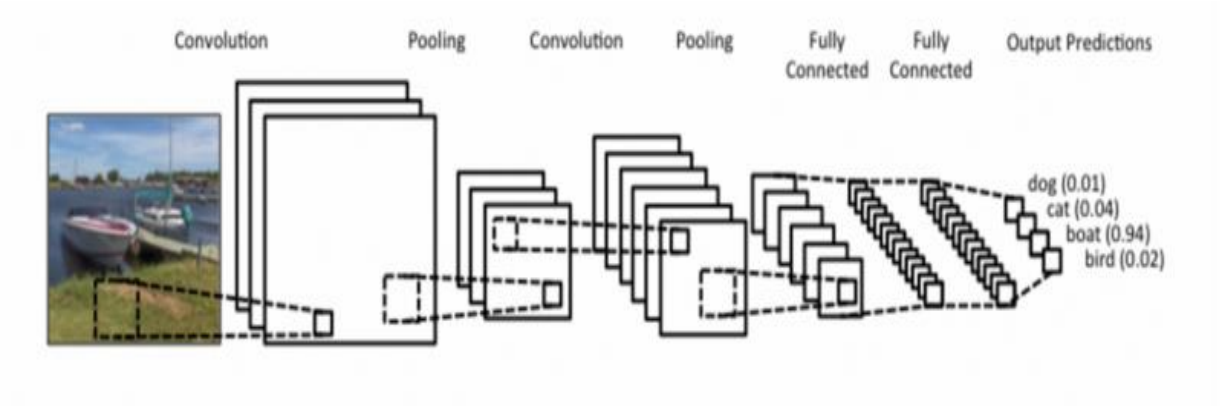
- Επίπεδο Συνέλιξης (Convolutional Layer). Το επίπεδο αυτό υλοποιεί τη πράξη της συνέλιξης μεταξύ της εισόδου και πολλών, εκατοντάδων ή και χιλιάδων, «κυλιόμενων» παραθύρων που ονομάζονται φίλτρα. Η προκύπτουσα έξοδος μεταξύ της εισόδου και ενός φίλτρου προστίθεται σε μία σταθερά-πόλωση b και περνά από μια μη γραμμική συνάρτηση ενεργοποίησης $\varphi()$. Το τελικό αποτέλεσμα αυτής της εφαρμογής ονομάζεται χάρτης χαρακτηριστικών (feature map) και προφανώς προκύπτουν τόσοι χάρτες χαρακτηριστικών όσα και τα φίλτρα. Η συνάρτηση $\varphi()$ είναι συνήθως μία από τις ακόλουθες:
    - Rectified linear , $\varphi(x) = \max(0, x)$
    - Συνάρτηση υπερβολικής εφαπτομένης, $\varphi(x) = \tanh(x)$
    - Λογιστική Συνάρτηση, $\varphi(x) = \frac{1}{1+e^{-x}}$

  και συχνά προτιμάται η πρώτη, επειδή καταλήγει σε αρκετά γρηγορότερη εκπαίδευση του νευρωνικού και το σφάλμα γενίκευσης παραμένει παρόμοιο.
- Επίπεδο συγκέντρωσης (Pooling Layer). Το επίπεδο αυτό τοποθετείται συνήθως μετά το συνελικτικό επίπεδο και έχει σαν στόχο την ελάττωση των προσαρμόσιμων παραμέτρων

του δικτύου και την αποφυγή της υπερεκπαίδευσης ή υπερπροσαρμογής (overfitting).

- Πλήρως συνδεδεμένο επίπεδο (Fully connected layer). Τελικά, μετά από αρκετά συνελικτικά και pooling layers, τοποθετείται ένα ή περισσότερα πλήρως συνδεδεμένα επίπεδα νευρώνων για την τελική κρίση, όπως ακριβώς στα κλασικά νευρωνικά δίκτυα. Πρόκειται δηλαδή για νευρώνες οργανωμένους σε μία διάσταση που έχουν πλήρεις συνδέσεις με όλες τις ενεργοποιήσεις του προηγούμενου επιπέδου.

Ένα παράδειγμα ΣΝΔ που λειτουργεί ως ταξινομητής εικόνων σε κατηγορίες ανάλογα με το θεματικό τους περιεχόμενο φαίνεται στο σχήμα 11.



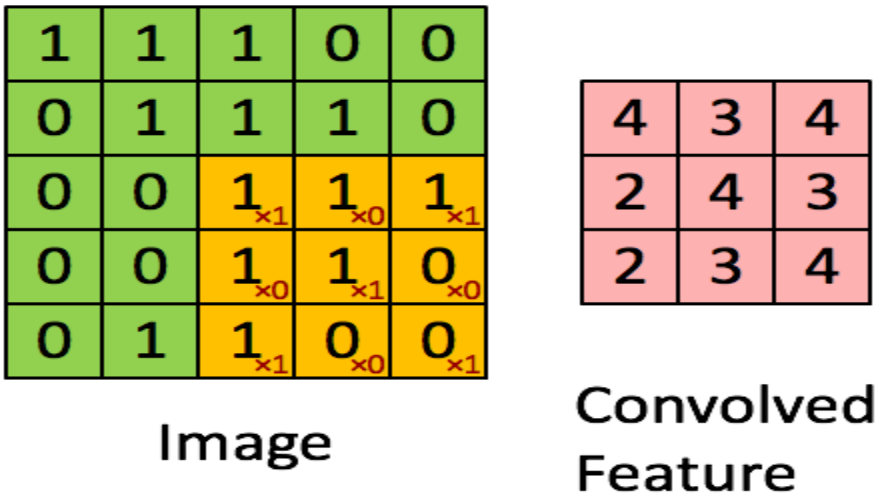
Σχήμα 11: Παράδειγμα αρχιτεκτονικής ΣΝΔ

Από τα παραπάνω επίπεδα, μόνο τα συνελικτικά και το πλήρως συνδεδεμένο επίπεδο εισάγουν προσαρμόσιμες παραμέτρους, βάρη οι βέλτιστες τιμές των οποίων αναζητούνται κατά την εκπαίδευση ελαχιστοποιώντας μία κατάλληλη συνάρτηση κόστους, που μπορεί να είναι το κόστος διεντροπίας (cross-entropy loss) για προβλήματα ταξινόμησης ή το τετραγωνικό σφάλμα για προβλήματα παλινδρόμησης.

3.1.1 Επίπεδο Συνέλιξης

Η συνέλιξη είναι μία βασική μαθηματική πράξη που εφαρμόζεται σε δύο συνεχείς συναρτήσεις ή διακριτά σήματα με εφαρμογές σε τομείς όπως στην στατιστική, στις πιθανότητες, στην όραση υπολογιστών, στην επεξεργασία σημάτων κτλ.

Ειδικά στην περίπτωση της όρασης υπολογιστών και της επεξεργασίας εικόνων, μπορούμε να σκεφτούμε την πράξη της συνέλιξης ως την ολίσθηση ενός κυλιόμενου παραθύρου που ονομάζεται πυρήνας (kernel) ή φίλτρο πάνω σε μία εικόνα, όπως απεικονίζει το σχήμα 12.



Σχήμα 12: Συνέλιξη με ένα 3x3 φίλτρο

Αν φανταστούμε πως ο πίνακας στα αριστερά αναπαριστά μία ασπρόμαυρη εικόνα, τότε ο πίνακας στα δεξιά προκύπτει αν ολισθήσουμε το συμμετρικό $3x3$ φίλτρο με τιμές που σημειώνονται με κόκκινο, από το πάνω αριστερό pixel της εικόνας μέχρι το κάτω δεξιά, και σε κάθε θέση πάρουμε το σημείο προς σημείο γινόμενο της αντίστοιχης υποπεριοχής της εικόνας με το φίλτρο και προσθέσουμε τα επιμέρους γινόμενα. Αν το φίλτρο δεν ήταν συμμετρικό, θα χρειαζόταν προηγουμένως να γίνει κατοπτρισμός ως προς το κεντρικό του pixel, πριν ακολουθήσει η προαναφερθείσα διαδικασία.

Μαθηματικά, το αποτέλεσμα της συνέλιξης μεταξύ της εικόνας x και του φίλτρου h δίνεται από την εξίσωση

$$y(i,j) = (x * h)(i,j) = \sum_{m,n} x(m,n)h(i-m, j-n)$$

Η πράξη της συνέλιξης χρησιμοποιείται ευρέως στην επεξεργασία των εικόνων. Για παράδειγμα, η συνέλιξη μιας εικόνας με το φίλτρο h_1

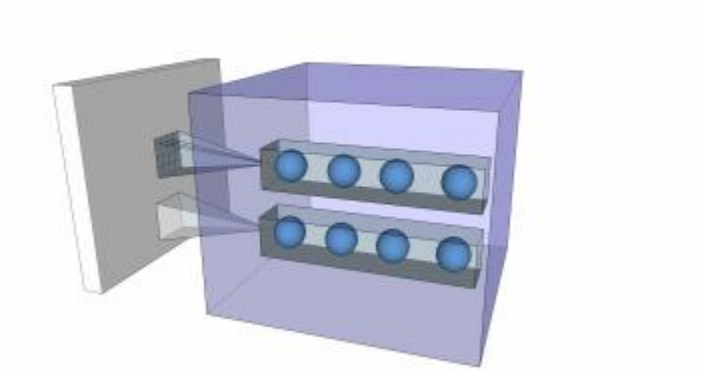
$$h_1 = \frac{1}{9}\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

προκαλεί το θόλωμα της εικόνας, ενώ η συνέλιξή της με το φίλτρο h_2

$$h_2 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

προκαλεί την ανίχνευση των ακμών της.

Αν αντιστοιχίσουμε το φίλτρο που ολισθαίνει πάνω στην εικόνα με νευρώνες στις αντίστοιχες θέσεις, τότε προκύπτει ένα στρώμα νευρώνων που ο καθένας έχει ένα συγκεκριμένο, περιορισμένο οπτικό πεδίο της εικόνας εύρους μερικών μόνο pixel και αγνοεί εντελώς την υπόλοιπη εικόνα. Όπως αναφέραμε σε κάθε συνελικτικό επίπεδο μπορούν να αντιστοιχούν ίσως και χιλιάδες, τοπικά φίλτρα. Επομένως, το συνελικτικό επίπεδο οργανώνεται σε 3 διαστάσεις, ως μια τριδιάσταση διάταξη νευρώνων της μορφής $W * H * D$, όπου η $3^{η}$ διάσταση αντιστοιχεί στο πλήθος των φίλτρων και όλοι οι D νευρώνες που βρίσκονται στην «ίδια θέση (w, h)» έχουν το ίδιο τοπικό πεδίο (βλ. σχήμα 13).



Σχήμα 13: Οργάνωση Συνελικτικού Επιπέδου σε 3 διαστάσεις

Έτσι, κάθε νευρώνας υπολογίζει ένα εσωτερικό γινόμενο (άθροισμα γινομένων) μεταξύ του τμήματος της εικόνας x που «βλέπει» και ενός διανύσματος βαρών που είναι κοινό για όλους τους νευρώνες του ίδιου επιπέδου και το περνά μέσα από μια μη γραμμικότητα (πχ. τη συνάρτηση ReLU).

Το γεγονός ότι όλοι οι νευρώνες του ίδιου επιπέδου έχουν κοινά βάρη είναι γνωστό ως διαμοιρασμός βαρών ή weight sharing και πέραν του ότι μειώνει τον αριθμό των ελεύθερων παραμέτρων βασίζεται στην εύλογη υπόθεση ότι ένα χαρακτηριστικό που είναι σημαντικό για το έργο μας, είναι σημαντικό σε οποιαδήποτε θέση και αν εμφανίζεται. Επίσης, ο διαμοιρασμός βαρών επιτρέπει κατά το ευθύ πέρασμα, ένα slice του συνελικτικού επιπέδου να υπολογιστεί ως η συνέλιξη της εισόδου με τα βάρη των νευρώνων (από εκεί προκύπτει και η ονομασία του επιπέδου) πριν εφαρμοστεί σε κάθε pixel η συνάρτηση μη γραμμικότητας που προαναφέραμε.

Οι υπερπαράμετροι που σχετίζονται με το επίπεδο της συνέλιξης είναι:

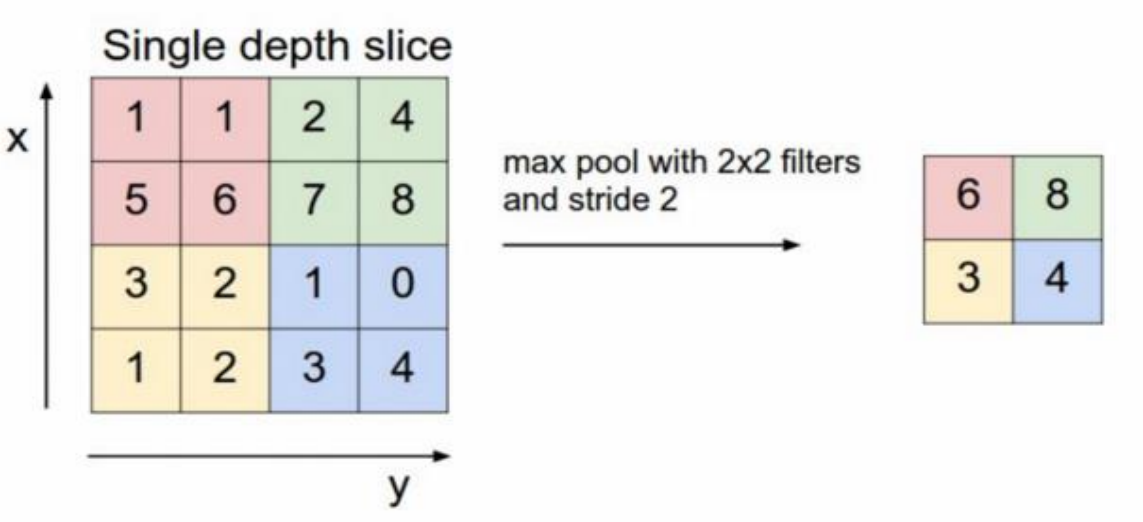
- Ο αριθμός των φίλτρων D: Ο αριθμός αυτός αντιστοιχεί στον αριθμό των features maps που κατασκευάζονται από το επίπεδο της συνέλιξης. Καθένα τέτοιο feature map θα ανιχνεύει και ένα διαφορετικό χαρακτηριστικό της εισόδου, πχ. ανίχνευση ακμών σε διαφορετικούς προσανατολισμούς.
- Το μέγεθος του φίλτρου f: Το μέγεθος του φίλτρου f καθορίζει το οπτικό πεδίο των νευρώνων και εξαρτάται κυρίως από το χρησιμοποιούμενο dataset. Τα βέλτιστα αποτελέσματα για μικρές εικόνες (πχ $28x28$ pixels) συνήθως χρησιμοποιούν στο πρώτο επίπεδο φίλτρα $5x5$, ενώ για μεγαλύτερες εικόνες (εκατοντάδων pixels σε κάθε διάσταση), συνήθως χρησιμοποιούνται στο πρώτο επίπεδο μεγαλύτερο φίλτρα, όπως $12x12$ ή $15x15$.
- Το βήμα του φιλτραρίσματος (stride size) S: Ο αριθμός αυτός ορίζει το πόσο μετατοπίζουμε το φίλτρο κάθε φορά. Στο παράδειγμα προηγουμένως $S = 1$, που είναι και η συνήθης τιμή. Μεγαλύτερο μήκος βήματος οδηγεί σε έξοδο μικρότερου μεγέθους και χρησιμοποιείται λιγότερο συχνά.
- Η μορφή της συνέλιξης, κανονική ή εκτεταμένη: Η συνέλιξη που περιγράψαμε προηγουμένως γραφικά ήταν κανονική και η προκύπτουσα εικόνα εξόδου ήταν μικρότερη από την αρχική. Μπορούμε όμως αν θέλουμε η φιλτραρισμένη εικόνα να έχει τις ίδιες διαστάσεις με την αρχική, να επεκτείνουμε πρώτα την αρχική εικόνα συμμετρικά με κάποια τεχνική (πχ. zero-padding ή επανάληψη των τιμών pixels του συνόρου) και μετά να εφαρμόσουμε συνέλιξη, που τότε ονομάζεται εκτεταμένη.

3.1.2 Επίπεδο Συγκέντρωσης

Στα ΣΝΔ, τα επίπεδα συγκέντρωσης τυπικά εφαρμόζονται μετά τα συνελικτικά επίπεδα και υποδειγματοληπτούν την είσοδό τους. Συνήθως χωρίζουν την εικόνα εισόδου σε ορθογώνιες, μη επικαλυπτόμενες υποπεριοχές και για κάθε τέτοια υποπεριοχή επιλέγουν την μέγιστη τιμή (max pooling), όπως για παράδειγμα στο σχήμα 14. Η διαίσθηση είναι ότι αφού έχει εντοπιστεί ένα χαρακτηριστικό, η ακριβής του θέση δεν είναι τόσο σημαντική, όσο η προσεγγιστικά σχετική του θέση ως προς τα άλλα χαρακτηριστικά. Η λειτουργία του επιπέδου συγκέντρωσης είναι να μειώνει προοδευτικά το μέγεθος της αναπαράστασης μειώνοντας έτσι τον αριθμό των προσαρμόσιμων παραμέτρων και των υπολογισμών και ελέγχοντας τελικά το overfitting. Επίσης η έξοδος του επιπέδου αυτού δεν εξαρτάται από μικρές μετατοπίσεις ή περιστροφές (shifting/rotation invariance), καθώς ο τελεστής max θα επιλέξει σε τέτοιες περιπτώσεις την ίδια τιμή.

Το επίπεδο συγκέντρωσης εφαρμόζεται ανεξάρτητα σε κάθε slice του συνελικτικού επιπέδου αλλάζοντας τελικά το μέγεθός του. Η πιο συνήθης μορφή είναι ένα επίπεδο συγκέντρωσης με φίλτρα μεγέθους $2x2$ ($F = 2$) και μήκος βήματος $S = 2$, το οποίο υποδειγματοληπτεί κάθε slice του συνελικτικού επιπέδου κατά 2 και ως προς τις 2 διαστάσεις, απορρίπτοντας το 75% των ενεργοποιήσεων. Κάθε πράξη max σε αυτήν την περίπτωση θα έπαιρνε το μέγιστο 4 τιμών, όπως στο σχήμα 14. Η 3η διάσταση του βάθους μένει ακριβώς η ίδια.

Εκτός από συγκέντρωση μεγίστου, το επίπεδο συγκέντρωσης μπορεί να εφαρμόζει και άλλες συναρτήσεις, όπως συγκέντρωση μέσου όρου (average pooling) ή ακόμη και συγκέντρωση L2 νόρμας (L2-norm pooling). Η συγκέντρωση μέσου όρου χρησιμοποιούνταν συχνά κατά το παρελθόν αλλά πρόσφατα έχει εκτοπιστεί από τη συγκέντρωση μεγίστου, η οποία φαίνεται να λειτουργεί καλύτερα στην πράξη.



Σχήμα 14: Max Pooling στα ΣΝΔ

Οι υπερπαράμετροι που σχετίζονται με το επίπεδο συγκέντρωσης είναι:

- το μέγεθος F της ορθογώνιας FxF περιοχής και
- το μήκος βήματος S.

Συνήθως $S = F$, ώστε να μην υπάρχει επικάλυψη μεταξύ των ορθογωνίων.

Σε εφαρμογές NLP που θα δούμε παρακάτω, παίρνουμε τυπικά το μέγιστο από ολόκληρο το χάρτη χαρακτηριστικών που αντιστοιχεί σε ένα φίλτρο.

3.1.3 Πλήρως Συνδεδεμένο Επίπεδο

Τα επίπεδα συνέλιξης και συγκέντρωσης που αναπτύχθηκαν προηγουμένως χρησιμοποιούνται για την εξαγωγή χαρακτηριστικών και την υποδειγματοληψία του όγκου των χαρακτηριστικών αντίστοιχα. Το πλήρως συνδεδεμένο επίπεδο νευρώνων τοποθετείται μετά από αυτά ενδεχομένως και σε αλληλουχία σχηματίζοντας στην περίπτωση αυτή ένα πολυστρωματικό perceptron , MLP, και στοχεύει στην υλοποίηση της κύριας λειτουργίας του νευρωνικού, συνήθως ταξινόμησης. Η

είσοδος είναι ένα διάνυσμα χαρακτηριστικών x συγκεκριμένης διάστασης και η έξοδος πχ. στην ταξινόμηση είναι τόσοι νευρώνες όσοι και οι κλάσεις του προβλήματος. Το πλήρως συνδεδεμένο επίπεδο εισάγει για την εκπαίδευση κατά τα γνωστά ένα πίνακα βαρών και ένα διάνυσμα πολώσεων.

3.2 Κανονικοποίηση και Εκπαίδευση ΣΝΔ

3.2.1 Υπερπαράμετροι ΣΝΔ

Οι υπερπαράμετροι ενός ΣΝΔ περιλαμβάνουν, εκτός από αυτές που αναφέραμε προηγουμένως, το ρυθμό μάθησης και τον τρόπο μεταβολής του, τον μέγιστο αριθμό εποχών, τον αριθμό των επιπέδων κ.ά. Η επιλογή τους είναι φυσικά πολύ σημαντική και επηρεάζει την τελική απόδοση του ΣΝΔ στην εργασία στην οποία τάχθηκε.

Μία συστηματική μέθοδος για την επιλογή των βέλτιστων τιμών αυτών των υπερπαραμέτρων είναι η μέθοδος του cross-validation. Σύμφωνα με τη μέθοδο αυτή, χωρίζουμε το σύνολο εκπαίδευσης σε S ισομεγέθη τμήματα (folds) (πχ. $S = 10$) και για κάθε fold $i = 0, \dots, S-1$ υπολογίζουμε την απόδοση J_{test}του δικτύου σε αυτό αφού προηγουμένως το εκπαιδεύσουμε χρησιμοποιώντας τα δεδομένα από τα υπόλοιπα $S-1$ τμήματα. Από το πείραμα αυτό υπολογίζουμε τον μέσο όρο των σφαλμάτων ελέγχου. Επαναλαμβάνουμε τα παραπάνω για κάθε μία από τις υποψήφιες τιμές της υπερπαραμέτρου και τελικά επιλέγουμε την τιμή της υπερπαραμέτρου που έδωσε τον καλύτερο μέσο όρο σφαλμάτων ελέγχου.

Το πλεονέκτημα της μεθόδου είναι η καλή εκτίμηση της βέλτιστης τιμής μιας υπερπαραμέτρου λειτουργώντας και με λίγα δεδομένα. Το μειονέκτημα είναι ότι απαιτούνται πολλές επαναλήψεις και άρα χρόνος, μιας και εκπαιδεύουμε το δίκτυο S φορές.

3.2.2 Κανονικοποίηση ΣΝΔ

Με τον όρο κανονικοποίηση εννοούμε την προσαρμογή της διαδικασίας εκπαίδευσης για την αποφυγή του φαινομένου της υπερμοντελοποίησης (overfitting). Στοχεύουμε δηλαδή με κάποιες τεχνικές που περιγράφονται παρακάτω στην αύξηση της ικανότητας γενίκευσης του μοντέλου μας, με αντίτιμο την αύξηση του σφάλματος εκπαίδευσης. Οι τεχνικές αυτές διακρίνονται σε εμπειρικές και σαφείς και είναι οι ακόλουθες:

Εμπειρικές Τεχνικές Κανονικοποίησης ΣΝΔ:

- Τυχαία αποκοπή συνδέσεων (μέθοδος dropout)
  Η μέθοδος αυτή εφαρμόζεται στα πλήρως συνδεδεμένα επίπεδα, που λόγω των πολλών παραμέτρων τους είναι επιρρεπή στην υπερμοντελοποίηση. Σύμφωνα με τη μέθοδο, σε κάθε στάδιο εκπαίδευσης οι νευρώνες των πλήρως συνδεδεμένων επιπέδων αποκόπτονται

με πιθανότητα $1-p$, ή ισοδύναμα διατηρούνται με πιθανότητα p (συνήθως $p=0.5$), με αποτέλεσμα να προκύπτει τελικά ένα μειωμένο δίκτυο. Οι εισερχόμενες και εξερχόμενες ακμές ενός αποκομμένου νευρώνα επίσης αφαιρούνται. Στο στάδιο αυτό, οι ανανέωσεις των βαρών αφορούν μόνο τους νευρώνες του μειωμένου δικτύου. Στην επόμενη επανάληψη, οι νευρώνες που είχαν απομακρυνθεί, εισέρχονται ξανά στο δίκτυο με τα αρχικά τους βάρη. Με τη μέθοδο dropout, εκπαιδεύονται τελικά πολλά νευρωνικά δίκτυα, αφού σε κάθε ζεύγος forward-backward propagation, λόγω της τυχαίας αποκοπής των συνδέσεων η εκπαίδευση αφορά διαφορετικό δίκτυο. Αφού τελειώσει η εκπαίδευση, τα βάρη $\boldsymbol{w}$ που έχουν μαθευτεί κλιμακώνονται κατά τον παράγοντα p, και τελικά τα βάρη $\widehat{\boldsymbol{w}} = p\boldsymbol{w}$ χρησιμοποιούνται στην φάση της ανάκλησης και για την παραγωγή της εξόδου που αντιστοιχεί σε μία άγνωστη εικόνα x. Η κλιμάκωση αυτή γίνεται έτσι ώστε η αναμενόμενη τιμή της εξόδου κάθε κόμβου να είναι η ίδια όπως στα στάδια εκπαίδευσης.

- Επαύξηση συνόλου δεδομένων
  Λόγω της γενικά βαθιάς αρχιτεκτονικής ενός ΣΝΔ, εισάγονται πολλές παράμετροι προς εκπαίδευση και συνεπώς για την αποφυγή του overfitting απαιτείται και ένα εύλογα μεγάλο training set. Αν έχουμε στη διάθεσή μας λίγα δεδομένα, τότε μπορούμε να τα αυξήσουμε είτε παράγοντας νέα δεδομένα από την αρχή αν κάτι τέτοιο είναι δυνατό είτε διαταράσσοντας λίγο τα υπάρχοντα παραδείγματα εκπαίδευσης για την παραγωγή νέων. Για παράδειγμα, μπορούμε να εφαρμόσουμε μετασχηματισμούς μετατόπισης ή περιστροφής, να προσθέσουμε θόρυβο ή να περικόψουμε (crop) μία εικόνα, πάντα όμως με προσοχή ώστε η κατηγορία της νέας εικόνας να είναι η ίδια με πριν.

Σαφείς Τεχνικές Κανονικοποίησης ΣΝΔ:

- Εξασθένηση βαρών (Weight decay)
  Ένας απλός τρόπος κανονικοποίησης είναι η προσθήκη στη σφάλμα εκπαίδευσης J ενός όρου Ω που είναι συνάρτηση των βαρών του δικτύου με αντίστοιχη παράμετρο κανονικοποίησης λ. Ο όρος αυτός θέλουμε να είναι μεγάλος όταν το δίκτυο είναι μεγάλο και μικρός όταν το δίκτυο είναι μικρό. Η παράμετρος λ ορίζει την έμφαση που δίνουμε στην κανονικοποίηση του δικτύου: όσο πιο μεγάλη είναι η σταθερά αυτή τόσο πιο μεγάλη σημασία δίνουμε στην μείωση της πολυπλοκότητας του δικτύου. Συνήθως, παίρνουμε την L2 νόρμα του διανύσματος βαρών (L2 regularization), οπότε τότε:

$$\Omega = \frac{1}{2}\sum_i w_i^2$$

- Περιορισμοί μέγιστης νόρμας (Max norm constraints)
  Μια άλλη μορφή κανονικοποίησης είναι να επιβάλλουμε ένα άνω φράγμα στο μέτρο του διανύσματος βαρών κάθε νευρώνα. Δηλαδή αφού εκτελέσουμε την ανανέωση των βαρών με το συνήθη τρόπο, αν προκύψει $\|\boldsymbol{w}\|_2 > c$, επανακλιμακώνουμε το διάνυσμα $\boldsymbol{w}$ έτσι ώστε

$$\|\boldsymbol{w}\|_2 = c$$

Συνήθως $c = 3$ ή $c = 4$.

3.2.3 Εκπαίδευση ΣΝΔ

Αφού επιλεγούν οι υπερπαράμετροι και προσδιοριστεί η κανονικοποίηση, ακολουθεί η εκπαίδευση του ΣΝΔ. Όπως αναφέραμε και προηγουμένως, οι παράμετροι προς εκπαίδευση που εισάγονται από το δίκτυο είναι τα βάρη και οι πολώσεις των νευρώνων στα συνελικτικά και πλήρως συνδεδεμένα επίπεδα. Η εκπαίδευση γίνεται όπως στα κλασικά νευρωνικά δίκτυα με τον αλγόριθμο της οπισθοδιάδοσης σφάλματος ή backpropagation όπως εξηγήθηκε στην ενότητα 2.6.3 και μπορεί να είναι online, παράδειγμα προς παράδειγμα, χρησιμοποιώντας τον αλγόριθμο της στοχαστικής κατάβασης δυναμικού (Stochastic Gradient Descent – SGD), ή μαζική χρησιμοποιώντας ολόκληρο το σύνολο εκπαίδευσης ή παρτίδες (mini-batches) αυτού.

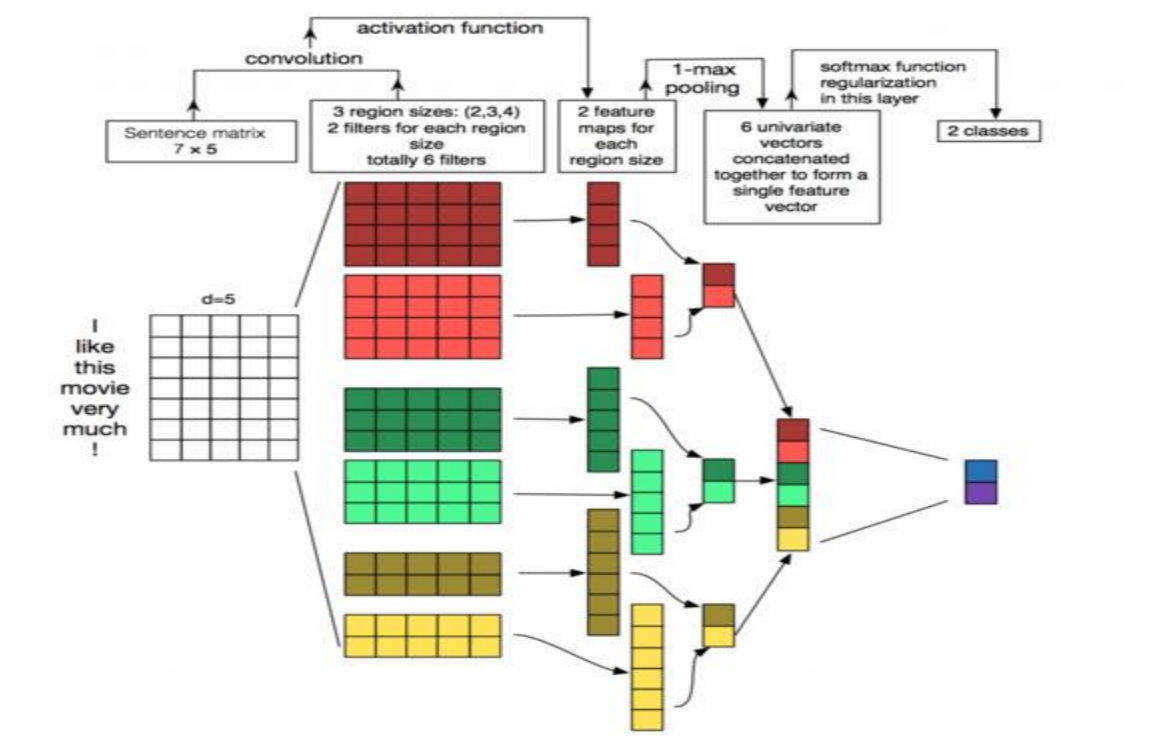
3.3 Χρήση ΣΝΔ στην Επεξεργασία Φυσικής Γλώσσας

Η είσοδος στα διάφορα προβλήματα NLP είναι προτάσεις ή ολόκληρα κείμενα τα οποία θα πρέπει να αναπαρασταθούν μέσω ενός πίνακα για να μπορούν να επεξεργαστούν από τα ΣΝΔ. Κάθε γραμμή του πίνακα αντιστοιχεί σε ένα token, συνήθως μια λέξη αλλά θα μπορούσε να είναι ένας χαρακτήρας. Τα διανύσματα αυτά που αντιστοιχούν σε λέξεις, συνήθως είναι τα λεγόμενα word embeddings (αναπαράσταση λέξεων από διανύσματα πραγματικών αριθμών μικρότερης διάστασης από το μέγεθος του λεξιλογίου) όπως τα διανύσματα word2vec[1] ή Glove[2], αλλά μπορεί να είναι και one-hot vectors με τη μονάδα να δηλώνει το index της λέξης στο λεξιλόγιο. Προτιμάται όμως η επιλογή των word embeddings τα οποία κωδικοποιούν τη σημασιολογική συγγένεια μεταξύ των σχέσεων, όπως θα δούμε στη συνέχεια. Η αναπαράσταση των λέξεων με one-hot vectors αν και δίνει καλά αποτελέσματα σε προβλήματα document classification, δεν αποδίδει εξίσου καλά σε προβλήματα sentence classification, στα οποία προτιμάται η επιλογή των διανυσμάτων word2vec ή Glove.

Στην επεξεργασία φυσικής γλώσσας, όπως αναφέραμε και προηγουμένως, τα φίλτρα τυπικά ολισθαίνουν καλύπτοντας πλήρως τις γραμμές του πίνακα (δηλ. τα word vectors). Άρα, το πλάτος των φίλτρων είναι ίσο με τη διάσταση των word vectors. Το ύψος μπορεί να μεταβάλλεται από εφαρμογή σε εφαρμογή, αλλά τυπικά κυμαίνεται μεταξύ 2 και 5 λέξεων. Έτσι, ένα ΣΝΔ για πρόβλημα NLP μπορεί να έχει τη μορφή του σχήματος 15, όπου το βήμα S του επιπέδου συνέλιξης είναι $S = 1$ και η μέγιστη τιμή στο επίπεδο συγκέντρωσης επιλέγεται από ολόκληρο το feature map.

[1] https://code.google.com/archive/p/word2vec/
[2] http://nlp.stanford.edu/projects/glove/

Ένα φίλτρο ανίχνευσης/εξαγωγής χαρακτηριστικών που χρησιμοποιείται στο συνελικτικό επίπεδο μπορεί να αφορά στην ανίχνευση μίας άρνησης όπως για παράδειγμα “not interesting” και ένα άλλο στην ανίχνευση μίας φράσης που συνοδεύεται από μία λέξη έντασης (intensifier) πχ “very entertaining” . Αν πχ. η $1^{η}$ φράση εμφανιστεί κάπου σε μία πρόταση, τότε η εφαρμογή του φίλτρου στην περιοχή αυτή θα παράγει μία μεγάλη τιμή σχετικά με τις άλλες περιοχές. Στη συνέχεια, εφαρμόζοντας την πράξη max σε ολόκληρο το feature map διατηρείται η πληροφορία σχετικά με το αν το χαρακτηριστικό εμφανίστηκε ή όχι την πρόταση, αλλά χάνεται η πληροφορία της θέσης που εντοπίστηκε ακριβώς. Συνεπώς, η όλη διαδικασία είναι παρόμοια με το μοντέλο n-grams της αναπαράστασης κειμένου. Το πλεονέκτημα των ΣΝΔ έναντι του μοντέλου n-grams είναι ότι τα ΣΝΔ υλοποιούνται και εκπαιδεύονται γρήγορα κάνοντας χρήση GPU, ενώ στο μοντέλο n-grams όταν το λεξιλόγιο είναι μεγάλο ο υπολογισμός των 3-grams και άνω είναι αρκετά “ακριβός”.



Σχήμα 15:Μορφή ΣΝΔ για ταξινόμηση προτάσεων

Η ανάγκη της αναπαράστασης των λέξεων με διανύσματα υπαγορεύτηκε από πολλά προβλήματα NLP. Οι 2 ευρύτερα χρησιμοποιούμενοι αλγόριθμοι που πραγματοποιούν την αναπαράσταση αυτή είναι οι αλγόριθμοι Glove και Word2vec οι οποίοι περιγράφονται στη συνέχεια. Περισσότερη έμφαση δίνεται στον αλγόριθμο Word2vec, τα διανύσματα του οποίου χρησιμοποιήσαμε στην υλοποίηση του ΣΝΔ μας.

3.3.1 Αλγόριθμοι Glove και Word2vec

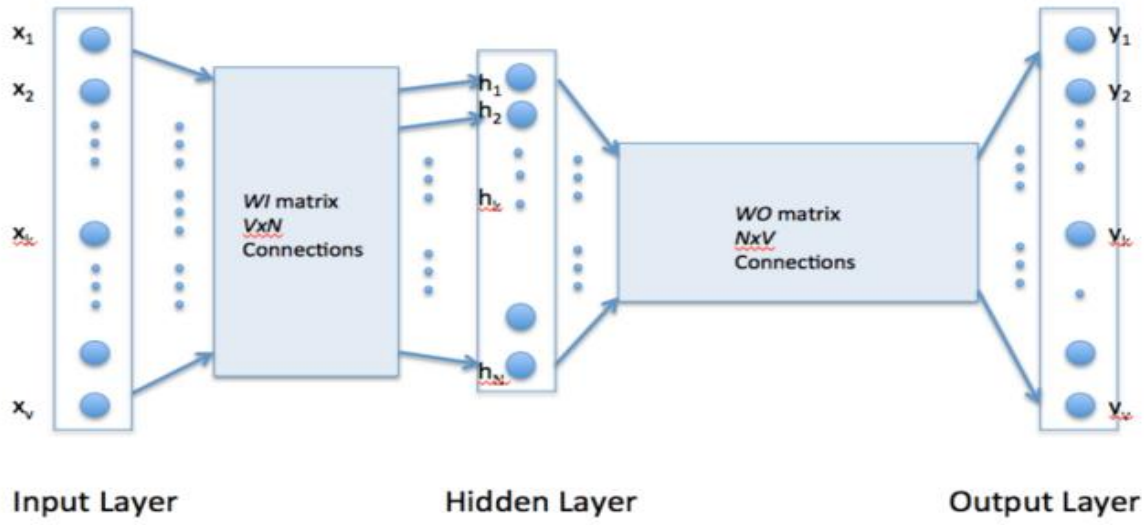
Αλγόριθμος Glove

Ο αλγόριθμος Glove είναι ένας αλγόριθμος μη επιβλεπόμενης μάθησης για την απόκτηση διανυσματικών αναπαραστάσεων των λέξεων. Η εκπαίδευση εφαρμόζεται σε ένα μεγάλο πίνακα μετρήσεων co-occurrence από ένα corpus, και οι αναπαραστάσεις που προκύπτουν εκδηλώνουν χρήσιμες ιδιότητες των λέξεων, όπως σχέσεις αναλογίας ή σχέσεις σημασιολογικής συγγένειας. Για παράδειγμα, παράγει ένα διάνυσμα που προσεγγίζει την αναπαράσταση του vec('Rome') σαν αποτέλεσμα της πράξης vec('Paris') - vec('France') + vec('Italy') και απεικονίζει λέξεις που είναι σημασιολογικά κοντά σε αντίστοιχα κοντινές θέσεις με την έννοια της ευκλείδειας απόστασης ή της απόστασης συνημιτόνου.

Αλγόριθμος Word2vec

Ο αλγόριθμος Word2vec του Mikolov, [9], χρησιμοποιεί και αυτός ένα μεγάλο όγκο κειμένου για τη δημιουργία υψηλής διάστασης (50 με 300) αναπαραστάσεων των λέξεων που κωδικοποιούν τις σχέσεις μεταξύ τους χωρίς τη βοήθεια εξωτερικών επισημειωτών (unsupervised learning). Μία τέτοια αναπαράσταση φαίνεται να συλλαμβάνει πολλές γλωσσικές κανονικότητες, όπως αυτές που αναφέρθηκαν προηγουμένως.

Ο αλγόριθμος Word2vec χρησιμοποιεί ένα 3-layer MLP, όπως φαίνεται στο σχήμα 16. Οι νευρώνες του κρυφού επιπέδου είναι όλοι γραμμικοί. Οι νευρώνες εισόδου είναι όσοι και οι λέξεις του λεξιλογίου της συλλογής κειμένων (corpus) για εκπαίδευση. Το πλήθος των κρυφών νευρώνων είναι ίσο με την επιθυμητή διάσταση των word vectors που θα προκύψουν. Το πλήθος των νευρώνων εξόδου είναι ίσο με το πλήθος των νευρώνων εισόδου. Άρα, υποθέτοντας ότι το λεξιλόγιο για την μάθηση των word vectors αποτελείται από V λέξεις και N είναι η διάσταση των word vectors, οι συνδέσεις μεταξύ της εισόδου και του κρυφού επιπέδου μπορούν να αναπαρασταθούν με ένα πίνακα W_I μεγέθους VxN όπου κάθε γραμμή αντιπροσωπεύει μία λέξη του λεξιλογίου. Επίσης, οι συνδέσεις από το κρυφό επίπεδο στο επίπεδο εξόδου μπορούν να περιγραφούν από τον πίνακα W_O μεγέθους NxV. Στην περίπτωση αυτή, κάθε στήλη του πίνακα W_O αντιπροσωπεύει μία λέξη του λεξιλογίου. Η είσοδος στο δίκτυο κωδικοποιείται χρησιμοποιώντας την αναπαράσταση «1 από V» («1-out of V») ή αλλιώς one-hot.



Σχήμα 16: Το δίκτυο που χρησιμοποιείται από τον αλγόριθμο Word2vec

Σαν παράδειγμα ας θεωρήσουμε ότι το training corpus περιέχει τις ακόλουθες προτάσεις: “the dog saw a cat”, “the dog chased the cat”, “the cat climbed a tree”.

Το λεξιλόγιό μας έχει 8 λέξεις. Μόλις διαταχθούν αλφαβητικά κάθε λέξη μπορεί να αναφερθεί από το index της. Για το παράδειγμα αυτό, το νευρωνικό μας δίκτυο θα έχει 8 νευρώνες εισόδου και 8 νευρώνες εξόδου. Αποφασίζουμε να χρησιμοποιήσουμε 3 νευρώνες στο κρυφό επίπεδο, κάτι που σημαίνει οι πίνακες W_I και W_O θα είναι $8x3$ και $3x8$ αντίστοιχα. Προτού αρχίσει η εκπαίδευση, οι πίνακες αυτοί αρχικοποιούνται σε μικρές τυχαίες τιμές όπως συνηθίζεται στην εκπαίδευση των νευρωνικών δικτύων. Ας υποθέσουμε ότι οι πίνακες W_I and W_O αρχικοποιούνται στις ακόλουθες τιμές:

$W_I =$

```
-0.094491  -0.443977   0.313917
-0.490796  -0.229903   0.065460
 0.072921   0.172246  -0.357751
 0.104514  -0.463000   0.079367
-0.226080  -0.154659  -0.038422
 0.406115  -0.192794  -0.441992
 0.181755   0.088268   0.277574
-0.055334   0.491792   0.263102
```

$W_O =$

```
 0.023074   0.479901   0.432148   0.375480  -0.364732  -0.119840   0.266070  -0.351000
-0.368008   0.424778  -0.257104  -0.148817   0.033922   0.353874  -0.144942   0.130904
 0.422434   0.364503   0.467865  -0.020302  -0.423890  -0.438777   0.268529  -0.446787
```

Ας υποθέσουμε ότι θέλουμε το δίκτυο να μάθει τη σχέση μεταξύ των λέξεων “cat” και “climbed”. Δηλαδή, το δίκτυο πρέπει να δείχνει υψηλή πιθανότητα για τη λέξη “climbed” όταν η είσοδος στο δίκτυο είναι η λέξη “cat” . Στην ορολογία των word embeddings, η λέξη “cat” αναφέρεται ως σημασιολογικό πλαίσιο (context) και η λέξη “climbed” αναφέρεται ως στόχος (target). Στην περίπτωση αυτή, το διάνυσμα εισόδου **x** θα είναι $(0,1,0,0,0,0,0,0)^T$ και το διάνυσμα στόχου θα είναι $(0,0,0,1,0,0,0,0)^T$. Η έξοδος του κρυφού επιπέδου είναι τότε:

$$\boldsymbol{h} = W_I^T \boldsymbol{x} = (-0.490796, -0.229903, 0.065460)^T$$

Λόγω αυτής της μορφής αναπαράστασης της εισόδου, το word vector της λέξης εισόδου (αντίστοιχη γραμμή του πίνακα W_I) αντιγράφεται στην έξοδο του hidden layer. Η ενεργοποίηση του στρώματος εξόδου είναι:

$$W_O^T \boldsymbol{h} =$$

$$\big(0.100934\,, -0.309331,\ -0.122361,\ -0.151399,\ 0.143463,\ -0.051262,$$
$$-0.079686,\ 0.112928\big)^T$$

Αφού ο στόχος είναι η παραγωγή πιθανοτήτων για τις λέξεις στο επίπεδο εξόδου $P(word_k|word_{context})\ για\ k = 1, \dots, V$, ώστε να αντικατοπτρίζουν την σχέση επόμενης λέξης με τη λέξη context στην είσοδο, χρειαζόμαστε το άθροισμα των εξόδων στο επίπεδο εξόδου να ισούται με 1. Αυτό επιτυγχάνεται χρησιμοποιώντας τη συνάρτηση softmax. Άρα, η έξοδος του k νευρώνα υπολογίζεται ως:

$$y_k = P(word_k|word_{context}) = \frac{\exp(activation(k))}{\sum_{n=1}^{V} \exp(activation(n))}$$

Άρα, οι πιθανότητες για τις 8 λέξεις του λεξιλογίου είναι:

0.143073 0.094925 0.114441 ***0.111166*** *0.149289 0.122874 0.119431 0.144800*

Η πιθανότητα με bold είναι για την επιλεγμένη λέξη-στόχο “climbed”. Δοθέντος του διανύσματος στόχου $(0,0,0,1,0,0,0,0)^T$, το διάνυσμα σφάλματος για το επίπεδο εξόδου υπολογίζεται εύκολα αφαιρώντας το διάνυσμα πιθανοτήτων από το διάνυσμα στόχου. Με γνωστό το σφάλμα, τα βάρη

στους πίνακες W_O και W_I μπορούν να ανανεωθούν με τον κανόνα backpropagation. Άρα, η εκπαίδευση μπορεί να προχωρήσει παρουσιάζοντας διαφορετικά ζευγάρια context-στόχου από το corpus. Στην ουσία, έτσι ο αλγόριθμος Word2vec μαθαίνει τις σχέσεις μεταξύ των λέξεων και κατά τη διαδικασία τις διανυσματικές αναπαραστάσεις των λέξεων του corpus.

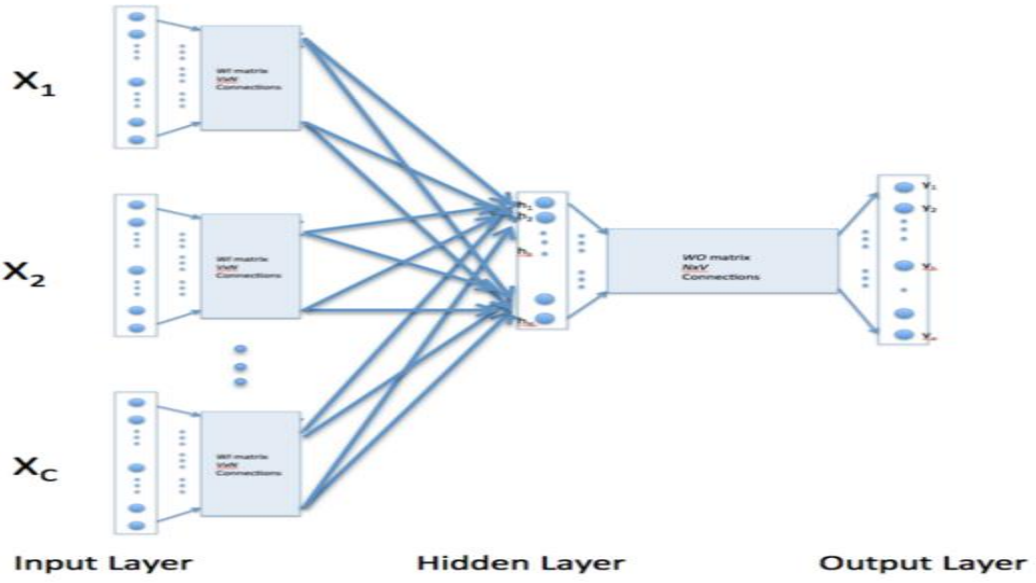
Υπάρχουν 2 κύριοι αλγόριθμοι μάθησης στον Word2vec:

- Ο Αλγόριθμος Continuous Bag of Words (CBOW) Learning
- Ο Αλγόριθμος Skip-gram

Οι αλγόριθμοι αυτοί εξηγούνται στη συνέχεια.

CBOW

Η παραπάνω περιγραφή και αρχιτεκτονική έχει νόημα όταν στόχος είναι η μάθηση σχέσεων μεταξύ ζεύγους λέξεων. Στο μοντέλο CBOW, το context αναπαρίσταται από πολλαπλές λέξεις για ένα δεδομένο στόχο. Για παράδειγμα, θα μπορούσαμε να χρησιμοποιήσουμε τις λέξεις “cat” και “tree” ως context με στόχο τη λέξη “climbed” . Αυτό συνεπάγεται την τροποποίηση της παραπάνω αρχιτεκτονικής. Η αλλαγή αυτή που φαίνεται στο σχήμα 17, αποτελείται από την αναπαραγωγή των συνδέσεων από την είσοδο στο κρυφό επίπεδο *C* φορές, όσες και οι λέξεις του context και την προσθήκη μιας διαίρεσης δια *C* στο κρυφό επίπεδο.



Σχήμα 17: Το δίκτυο που χρησιμοποιείται από τον αλγόριθμο Word2vec και το μοντέλο CBOW

Με τις παραπάνω τροποποιήσεις, η έξοδος του κρυφού επιπέδου είναι ο μέσος όρος των word vectors που αντιστοιχούν στις λέξεις του context στην είσοδο. Το επίπεδο εξόδου διατηρείται το ίδιο και η εκπαίδευση γίνεται με τον τρόπο που αναφέρθηκε προηγουμένως.

Skip-gram

Το μοντέλο skip-gram αντιστρέφει την χρήση της λέξης στόχου και των λέξεων context. Στην περίπτωση αυτή, η λέξη στόχος τροφοδοτείται στην είσοδο και το επίπεδο εξόδου του νευρωνικού αναπαράγεται πολλές φορές για να αντιστοιχιστεί με τις λέξεις context. Παίρνοντας σαν παράδειγμα τις λέξεις "cat" και "tree" ως context και τη λέξη "climbed" ως στόχο, η είσοδος στο μοντέλο skip-gram θα ήταν $(0,0,0,1,0,0,0,0)^T$, ενώ τα 2 επίπεδα εξόδου θα είχαν ως διανύσματα εξόδου τα $(0,1,0,0,0,0,0,0)^T$ και $(0,0,0,0,0,0,0,1)^T$ αντίστοιχα. Τώρα παράγονται 2 διανύσματα πιθανοτήτων στην έξοδο. Το διάνυσμα σφάλματος για κάθε επίπεδο εξόδου παράγεται με τον τρόπο που αναφέρθηκε προηγουμένως. Όμως, τα διανύσματα σφάλματος από όλα τα επίπεδα εξόδου αθροίζονται για να ρυθμιστούν τα βάρη με τον κανόνα backpropagation. Αυτό εξασφαλίζει ότι ο πίνακας βαρών W_O για κάθε επίπεδο εξόδου παραμένει απαράλλαχτος καθ' όλη την εκπαίδευση.

Με λίγα λόγια μπορούμε να πούμε ότι και τα δύο μοντέλα (Glove και Word2vec) μαθαίνουν ενδιαφέρουσες διανυσματικές αναπαραστάσεις των λέξεων χρησιμοποιώντας την πληροφορία της συνύπαρξης (co-occurrence) (πόσο συχνά εμφανίζονται μαζί σε μια μεγάλη συλλογή κειμένων). Διαφέρουν στον τρόπο που το πετυχαίνουν. Το Glove είναι ένα "count-based" μοντέλο, ενώ το Word2vec είναι ένα προβλεπτικό μοντέλο. Και τα 2 μοντέλα ωστόσο συμπεριφέρονται παρόμοια στα διάφορα προβλήματα NLP στα οποία χρησιμοποιούνται.

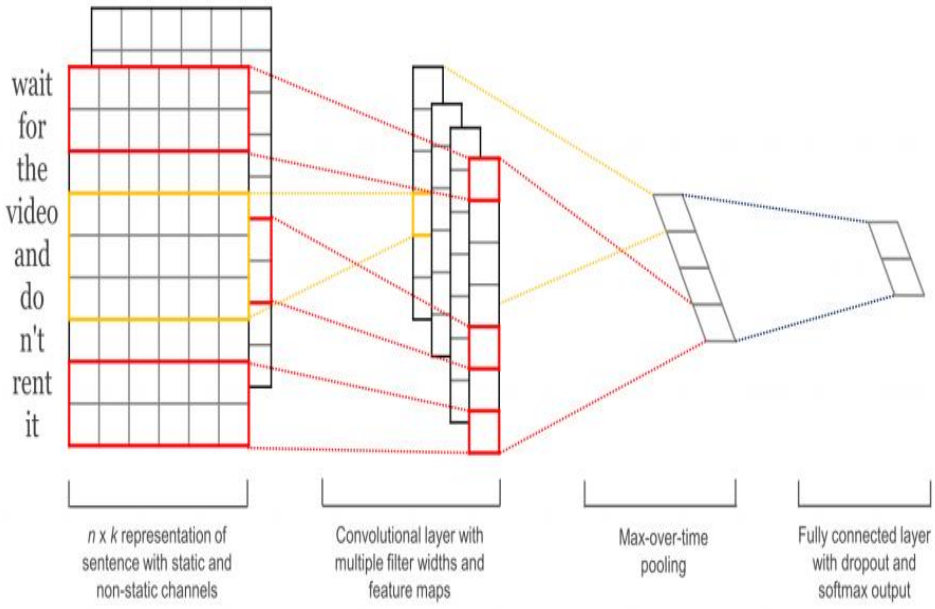
Υπάρχουν δημοσίως διαθέσιμα pre-trained word vectors. Για την υλοποίηση του ΣΝΔ μας, χρησιμοποιήσαμε τα word2vec διανύσματα που εκπαιδεύτηκαν πάνω σε 100 δισεκατομμύρια λέξεις από το dataset του Google News. Τα διανύσματα αυτά έχουν διάσταση 300 και αποκτήθηκαν χρησιμοποιώντας την αρχιτεκτονική CBOW.

3.3.2 Εφαρμογές ΣΝΔ σε Προβλήματα NLP

Τα ΣΝΔ, όπως είπαμε, εφαρμόζονται σε προβλήματα NLP και κυρίως σε προβλήματα ταξινόμησης κειμένου, όπως στην ανάλυση συναισθήματος, στην ανίχνευση spam και στην ταξινόμηση με βάση το θέμα (topic). Οι πράξεις της συνέλιξης και της συγκέντρωσης χάνουν πληροφορία σχετικά με την τοπική σειρά των λέξεων, έτσι η σειριακή επισημείωση όπως η επισημείωση μέρους του λόγου (POS Tagging) ή η εξαγωγή οντοτήτων (Entity Extraction) είναι δυσκολότερο να υλοποιηθούν από ένα ΣΝΔ. Κάποια σημαντικά αποτελέσματα σε προβλήματα NLP με χρήση ΣΝΔ είναι τα ακόλουθα:

- Στο [10], υλοποιείται ένα ΣΝΔ ενός συνελικτικού επιπέδου που ταξινομεί προτάσεις κυρίως με βάση την άποψη (sentiment analysis) ή το θέμα (topic categorization). Η

αρχιτεκτονική αυτή (σχήμα 18) επιτυγχάνει πολύ καλή απόδοση σε διάφορα datasets, και μάλιστα σε κάποια τα αποτελέσματα είναι εντυπωσιακά ξεπερνώντας άλλες μεθόδους. Το γεγονός μάλιστα ότι έχει αρκετά απλή αρχιτεκτονική είναι αυτό που κάνει το δίκτυο τόσο ισχυρό. Η είσοδος στο ΣΝΔ είναι μία πρόταση, η οποία αναλύεται στις λέξεις της με κάποιον αλγόριθμο tokenization και οι λέξεις αναπαριστώνται με τα διανύσματα word2vec. Ο πίνακας της πρότασης αποτελείται από την συνένωση των διανυσμάτων αυτών. Το επίπεδο εισόδου ακολουθείται από ένα συνελικτικό επίπεδο με διάφορα φίλτρα και ακολουθούν ένα επίπεδο συγκέντρωσης που εκτελεί την πράξη max και ένας softmax ταξινομητής. Γίνεται ακόμη πειραματισμός με τα κανάλια της «εικόνας»: δοκιμάζεται και η περίπτωση της αναπαράστασης της «εικόνας» με 2 κανάλια, εκ των οποίο ένα είναι σταθερό και δεν μεταβάλλεται κατά τη διάρκεια της εκπαίδευσης (static channel) και το άλλο προσαρμόζεται κατά τη διάρκεια της εκπαίδευσης (non static channel) επιτρέποντας το fine-tuning των αρχικών word vectors και την προσαρμογή τους στο πιο συγκεκριμένο έργο του sentiment analysis. Στο [11], προστίθεται ένα επιπλέον επίπεδο που εφαρμόζει «semantic clustering” .



Σχήμα 18: Αρχιτεκτονική ΣΝΔ με 2 κανάλια με στόχο την ταξινόμηση προτάσεων, Y.Kim (2014)

- Στο [12], υλοποιείται ένα ΣΝΔ από την αρχή, χωρίς την ανάγκη pre-trained word vectors όπως τα word2vec ή Glove. Χρησιμοποιούνται τα απλά one-hot vectors. Ο συγγραφέας

επίσης προτείνει έναν αποδοτικό ως προς τον χώρο, σαν το BOW, τρόπο αναπαράστασης για τα δεδομένα εισόδου, μειώνοντας τον αριθμό των εκπαιδεύσιμων παραμέτρων.

- Οι υπερπαράμετροι που σχετίζονται με τη σχεδίαση ενός ΣΝΔ είναι αρκετές μεταξύ των οποίων περιλαμβάνονται η αναπαράσταση της εισόδου (word2vec, Glove, one-hot), ο αριθμός και το μέγεθος των συνελικτικών φίλτρων, οι στρατηγικές συγκέντρωσης (max,μέσου όρου) και οι συναρτήσεις ενεργοποίησης (ReLU ,tanh). Στο [13], γίνεται μία εμπειρική αξιολόγηση της επίδρασης των διαφόρων υπερπαραμέτρων, ερευνώντας την επίδρασή τους στην απόδοση και διακύμανση σε πολλαπλές εκτελέσεις. Οι ερευνητές που θέλουν να υλοποιήσουν το δικό τους ΣΝΔ για προβλήματα ταξινόμησης κειμένου, μπορούν να χρησιμοποιήσουν τα αποτελέσματα αυτά σαν σημείο εκκίνησης. Κάποια συμπεράσματα που ξεχωρίζουν είναι ότι η συγκέντρωση μεγίστου είναι πάντοτε ανώτερη της συγκέντρωσης μέσου όρου, τα ιδανικά μεγέθη των φίλτρων είναι σημαντικά αλλά εξαρτώνται από το εκάστοτε task, και η κανονικοποίηση δεν φαίνεται να κάνει μεγάλη διαφορά στα προβλήματα NLP που εξετάστηκαν. Ωστόσο, συνίσταται προσοχή, καθώς όλα τα datasets που θεωρήθηκαν είναι παρόμοια ως προς το μέγεθος των κειμένων τους, και οι ίδιες οδηγίες μπορεί να μην εφαρμόζονται σε δεδομένα που είναι αρκετά διαφορετικά.

Τα παραπάνω μοντέλα βασίζονταν σε λέξεις. Όμως, έχει γίνει έρευνα και για την εφαρμογή ΣΝΔ απευθείας σε χαρακτήρες (character-level ΣΝΔ). Στο [14], εξετάζεται ένα ΣΝΔ το οποίο μαθαίνει διανύσματα χαρακτήρων (character-level embeddings) και τα χρησιμοποιεί μαζί με pre-trained word vectors για POS tagging. Στα [15], [16] διερευνάται η χρήση των ΣΝΔ για τη μάθηση κατευθείαν από χαρακτήρες, χωρίς την ανάγκη για επιπλέον pre-trained word vectors. Ειδικότερα, οι συγγραφείς χρησιμοποιούν ένα βαθύ δίκτυο 9 συνολικά επιπέδων για sentiment analysis και text categorization. Τα αποτελέσματα δείχνουν ότι η μάθηση άμεσα από τους χαρακτήρες δουλεύει πολύ καλά σε μεγάλα datasets (με εκατομμύρια παραδείγματα) αλλά λειτουργεί λιγότερο καλά σε σύγκριση με απλούστερα μοντέλα σε μικρότερα datasets (με εκατοντάδες ή χιλιάδες παραδείγματα).

Κεφάλαιο 4

Ανάλυση Συναισθήματος Βασισμένη σε Λεξικό

Στο κεφάλαιο αυτό, θα αναλύσουμε την δεύτερη βασική μέθοδο προσέγγισης του προβλήματος της ανάλυσης συναισθήματος, τη βασισμένη σε λεξικό προσέγγιση, αναφέροντας κάποιες από τις αδυναμίες της καθώς και τρόπους αντιμετώπισής τους. Θα αναδείξουμε επίσης κάποια λεξικά που είναι δημοσίως διαθέσιμα και θα επισημάνουμε κάποιες ενδιαφέρουσες ερευνητικές προσπάθειες που έχουν γίνει με χρήση λεξικού.

4.1 Ανάλυση Βασισμένης σε Λεξικό Προσέγγισης

Η βασισμένη σε λεξικό προσέγγιση υποθέτει ότι ο συναισθηματικός χαρακτηρισμός του κειμένου μπορεί να προκύψει από τον συναισθηματικό χαρακτηρισμό των επιμέρους λέξεων ή φράσεων του. Αυτές, λέξεις ή φράσεις, έχουν σημειωθεί ως προς το συναισθηματικό τους περιεχόμενο και βρίσκονται κατοχυρωμένες σε ένα λεξικό, το οποίο χαρακτηρίζεται word-level ή concept-level συναισθηματικό λεξικό (sentiment lexicon). Στα λεξικά αυτά οι λέξεις ή φράσεις συνοδεύονται από αντίστοιχες βαθμολογίες οι οποίες εκφράζουν κατά πόσο αυτές ταιριάζουν με μία συγκεκριμένη κατηγορία συναισθήματος: συνήθως οι κατηγορίες αυτές είναι οι δύο βασικές (θετική και αρνητική) και οι βαθμολογίες έχουν το αντίστοιχο πρόσημο, με την απόλυτη τιμή να εκφράζει την κλίμακα (ή «βεβαιότητα») της κατηγορίας. Ωστόσο υπάρχουν και λεξικά με πιο συγκεκριμένες κατηγορίες συναισθημάτων, όπως χαρά, λύπη, θυμός, κ.α.

Η διαδικασία που ακολουθείται εφόσον βρεθεί και επιλεγεί κάποιο συναισθηματικό λεξικό για τον προσδιορισμό της συναισθηματικής κατηγορίας ενός κειμένου εισόδου είναι η εξής: Με έναν αλγόριθμο tokenization, αναλύουμε το κείμενο σε λέξεις ή φράσεις. Κάθε token αναζητάται στο συναισθηματικό λεξικό και σημειώνουμε τη βαθμολογία του. Ο συναισθηματικός χαρακτηρισμός του συνολικού κειμένου προκύπτει αθροίζοντας τις επιμέρους βαθμολογίες. Για ταξινόμηση σε μία από τις δύο κατηγορίες, θετική ή αρνητική, όπως στο πρόβλημά μας, αρκεί η εξέταση του προσήμου του αθροίσματος. Για ταξινόμηση σε μία από πολλές κατηγορίες (πολυεπίπεδη συναισθηματική κατάταξη) (πχ. χαρακτηρισμός αστεριών μίας ταινίας βάσει μιας κριτικής) γίνεται χρήση και των κατάλληλων κατωφλίων.

Ωστόσο, η μέθοδος αυτή, παρά την απλότητά της, έχει κάποιες αδυναμίες που περιορίζουν την απόδοσή της στην ταξινόμηση κειμένου. Οι βασικοί περιορισμοί με τους οποίους έρχεται αντιμέτωπη η μέθοδος αυτή είναι:

- Άρνηση (Negation): Η διαδικασία που περιγράφτηκε παραπάνω, αγνοεί την άρνηση, που στα αγγλικά ανιχνεύεται με τις λέξεις not, never, nothing, nobody κτλ. Η ύπαρξη μίας άρνησης λέξης, επηρεάζει σαφώς το νόημα της λέξης ή των λέξεων που ακολουθούν

αντιστρέφοντας το polarity. Για παράδειγμα, ενώ η πρόταση "*The movie was good*" αντιστοιχεί σε μία θετική κριτική, η πρόταση "The movie was not good" αντιστοιχεί σε μία αρνητική κριτική. Η προφανής επιλογή αντιμετώπισης του φαινομένου αυτού και αυτή που χρησιμοποιείται συνήθως είναι η αντιστροφή του polarity των λέξεων που ακολουθούν την λέξη άρνησης μέχρι το επόμενο σημείο στίξης ή κάποιον αντιθετικό σύνδεσμο, πχ. but, however κ.ά. Όμως η επιλογή αυτή έχει αδυναμίες. Για παράδειγμα, αν υποθέσουμε ότι η λέξη *excellent* έχει βαθμολογία $+5$ και η λέξη *good* έχει βαθμολογία $+3$, τότε προκύπτει ότι η άρνηση *not good* έχει μεγαλύτερη βαθμολογία από την άρνηση *not excellent*, στοιχείο που δεν συνάδει με τη διαίσθησή μας. Οι Taboada et al [17], πρότειναν η βαθμολογία της άρνησης να προκύπτει από την ολίσθηση της βαθμολογίας της επόμενης λέξης κατά μία σταθερή ποσότητα προς την αντίθετη κατεύθυνση.

- Μετατόπιση Έντασης/Σθένους (Valence Shifters): Πέραν της άρνησης, που σαφώς επηρεάζει το νόημα των λέξεων που ακολουθούν, υπάρχουν και οι λέξεις μεταβολής έντασης που αυξάνουν (intensifiers) ή μειώνουν (downtoners) την ένταση της επόμενης λέξης. Παραδείγματα τέτοιων λέξεων στα αγγλικά είναι very, truly, really, slightly, more, less κ.α. Η εξέταση των λέξεων αυτών είναι σημαντική, ιδιαίτερα στην περίπτωση που επιδιώκουμε πολυεπίπεδη συναισθηματική κατάταξη. Κάποιοι ερευνητές, όπως οι Kennedy και Inkpen [18] και οι Polanyi and Zaenen [19], αντιμετώπισαν το φαινόμενο με απλή πρόσθεση και αφαίρεση: Αν βρεθεί κάπου μέσα στο κείμενο μία λέξη intensifier, το polarity της επόμενης λέξης αυξάνεται κατά μία σταθερή ποσότητα, ενώ αντίθετα, αν βρεθεί μία λέξη downtoner, το polarity της επόμενης λέξης μειώνεται κατά την ίδια σταθερή ποσότητα. Το πρόβλημα αυτής της προσέγγισης είναι ότι δεν λαμβάνει υπόψη τις διαφορές μεταξύ των valence shifters. Πχ. η λέξη extraordinarily είναι για παράδειγμα πολύ πιο δυνατός «ενισχυτής» από τη λέξη rather. Οι Taboada et al [17], πρότειναν την αντιστοίχιση ενός ποσοστού σε κάθε valence shifter που θα προστίθεται ή θα αφαιρείται από το 100% ανάλογα με την περίπτωση και θα πολλαπλασιάζεται με το polarity της επόμενης λέξης.
- Σειρά Λέξεων: Η lexicon-based μέθοδος αγνοεί τη σειρά των λέξεων που εμφανίζονται στο κείμενο. Αυτή η BoW (Bag of Words) μοντελοποίηση του κειμένου εμφανίζει αδυναμίες αφού η σειρά των λέξεων μπορεί και να αντιστρέψει το polarity μίας πρότασης. Αν χρησιμοποιήσουμε και πάλι το παράδειγμα της ενότητας 2.2:

  That's not true, I'm a fan of this movie.
  That's true, I'm not a fan of this movie.

  βλέπουμε ότι οι δύο παραπάνω προτάσεις χρησιμοποιούν το ίδιο σύνολο λέξεων αλλά έχουν εντελώς διαφορετικό (αντίθετο) νόημα. Το παραπάνω πρόβλημα αντιμετωπίζεται με την επισήμανση της άρνησης (λέξη not) και τον προσδιορισμό της εμβέλειάς της όπως εξηγήθηκε νωρίτερα.

- Ύπαρξη Αντιθετικών/Εναντιωματικών Συνδέσμων (Adversative conjunctions): Οι αντιθετικοί σύνδεσμοι σε μία πρόταση, όπως υποδηλώνει το όνομά τους, συνδέουν δύο φράσεις αντίθετης πολικότητας. Παραδείγματα αντιθετικών συνδέσμων στα αγγλικά είναι

οι λέξεις but, although, however κ.ά. Συνήθως, το polarity της συνολικής πρότασης καθορίζεται από το δεύτερο συστατικό της πρότασης. Για παράδειγμα, στην πρόταση "The car is nice but expensive" η προδιάθεση του συγγραφέα ή ομιλητή είναι εναντίον της αγοράς του αυτοκινήτου, ενώ στην πρόταση "The car is expensive but nice", η προδιάθεση του συγγραφέα ή ομιλητή ως προς την αγορά του αυτοκινήτου είναι θετική.

- Ιδιωματισμοί: Μία άλλη αστοχία της μεθόδου είναι ότι μελετώντας κάθε λέξη ξεχωριστά, αγνοούμε την ύπαρξη φράσεων των οποίων οι επιμέρους λέξεις προσδίδουν ένα ιδιαίτερο συνολικό νόημα. Παράδειγμα αποτελεί η φράση "once in a blue moon", που σημαίνει πολύ σπάνια. Για την αντιμετώπισή τους απαιτείται η χρήση ειδικού λεξικού που θα περιλαμβάνει τέτοιους ιδιωματισμούς και η ανάλυση του κειμένου όχι σε επίπεδο λέξεων αλλά σε επίπεδο φράσεων.
- Αμφισημία: Το φαινόμενο κατά το οποίο μία λέξη ή φράση έχει διαφορετικό νόημα ανάλογα με το ευρύτερο νοηματικό πλαίσιο στο οποίο χρησιμοποιείται ονομάζεται αμφισημία. Για παράδειγμα, η λέξη "unpredictable" έχει θετική έννοια όταν αναφέρεται σε μία ταινία και συνήθως αρνητική όταν αναφέρεται στον καιρό. Ακόμη, η φράση "go read the book" είναι θετική για μία κριτική βιβλίου αλλά αρνητική για μία κριτική ταινίας. Ο χειρισμός της αμφισημίας είναι ένα από τα δυσκολότερα εμπόδια που καλείται να λύσει η ανάλυση συναισθήματος και έχει συγκεντρώσει το ενδιαφέρον πολλών ερευνητών. Στο [20], οι Poria, Cambria et al, εξετάζουν το πρόβλημα του sentiment analysis σε επίπεδο εννοιών (concept-level) , σύμφωνα με το οποίο κάθε πρόταση εισόδου αναλύεται σε έννοιες και αυτές αναζητούνται σε ένα ειδικά σχεδιασμένο λεξικό, στο λεξικό SenticNet. Στο [21], η ίδια ομάδα ερευνητών, πέτυχε ακόμη καλύτερα αποτελέσματα πάνω στα ίδια dataset, πάλι διεξάγοντας concept-level sentiment analysis αλλά χρησιμοποιώντας πιο σύνθετους κανόνες για την εύρεση των εξαρτήσεων μεταξύ των εννοιών και γλωσσολογικά patterns.
- Ειρωνεία: Πολλές φορές οι άνθρωποι επιστρατεύουν το χιούμορ και τον σαρκασμό για να εκφράσουν την συνήθως αρνητική άποψή τους. Η ανίχνευση της ειρωνείας είναι πολλές φορές δύσκολη από τον άνθρωπο, πόσο μάλλον από μία μηχανή. Για παράδειγμα, η πρόταση "The restaurant was great in that it will make all future meals seem more delicious" εμπεριέχει ειρωνεία που μπορεί να μη γίνει αντιληπτή από κάποιο αναγνώστη που την «προσπερνά» γρήγορα. Το πρόβλημα λοιπόν της ανίχνευσης ειρωνείας κειμένου με αυτοματοποιημένο τρόπο είναι δύσκολο και η επιστημονική έρευνα έχει επικεντρωθεί σε μεθόδους εντοπισμού της με διάφορες μεθόδους μελετώντας το δυαδικό πρόβλημα της ταξινόμησης μίας πρότασης ως σαρκαστική ή όχι.
- Πολλαπλοί στόχοι: Πολλές φορές σε μία κριτική γίνεται αναφορά σε παραπάνω από μία οντότητες (πρόσωπα, προϊόντα, γεγονότα) ή ακόμα και σε διαφορετικά χαρακτηριστικά (aspects) της ίδιας οντότητας. Στην περίπτωση αυτή συνήθως δεν ενδιαφέρει η ταξινόμηση του κειμένου συνολικά ως μία θετική ή αρνητική άποψη αλλά εξετάζεται το κείμενο σε επίπεδο χαρακτηριστικών (aspect-level). Η ανάλυση συναισθήματος σε επίπεδο χαρακτηριστικών (Aspect-level SA) στοχεύει στον προσδιορισμό του συναισθήματος ως προς συγκεκριμένα aspects των οντοτήτων και διαφέρει από την ανάλυση συναισθήματος σε επίπεδο κειμένου (Document-level SA). Το πρώτο βήμα είναι η αναγνώριση των οντοτήτων και των χαρακτηριστικών τους. Στη συνέχεια εξάγονται οι γλωσσικές

εκφράσεις που αναφέρονται στα ζεύγη (Entity, Aspect) και με χρήση συνωνύμων και λεξικού αποδίδεται το αντίστοιχο συναίσθημα. Για παράδειγμα, στην πρόταση ‘‘The voice quality of this phone is not good, but the battery life is long”, η Aspect-level SA θα έδινε:

{(phone, voice quality), negative}

{(phone, battery), positive}

Μία ενδιαφέρουσα εφαρμογή του sentiment analysis είναι η ανάλυση άποψης, σχετικά με μία οντότητα, από κείμενα χρηστών ενός κοινωνικού μέσου, για παράδειγμα του twitter. Στο πρόβλημα όμως αυτό εισέρχονται ακόμη περισσότερες δυσκολίες (πέραν αυτών που αναφέρθηκαν προηγουμένως) που σχετίζονται με την ιδιαίτερη μορφή των tweets. Συγκεκριμένα, το μικρό τους μέγεθος (έως 140 χαρακτήρες) υποχρεώνει τον χρήστη να εκφράζει την άποψή του με λίγες χρωματισμένες λέξεις οι οποίες μπορεί να μην βρίσκονται στο λεξικό. Τότε, η εξαγωγή συμπεράσματος ως προς την πολικότητα της άποψης είναι εξαιρετικά δύσκολη αν όχι αδύνατη. Επίσης, στα tweets, είναι συχνή η χρήση συντομογραφιών (λόγω του περιορισμού μεγέθους), όπως και συχνά είναι και τα ορθογραφικά λάθη λόγω επιπολαιότητας. Ακόμη πολλές φορές τα μηνύματα περιέχουν λέξεις από περισσότερες της μίας γλώσσας, ενώ ειδικότερα για τους Έλληνες χρήστες η χρήση greeklish είναι ευρέως διαδεδομένη. Όλα αυτά, απαιτούν ένα λεξικό το οποίο θα αφομοιώνει τις τάσεις των χρηστών του διαδικτύου, ενσωματώνοντας συχνά misspellings, συντομογραφίες και λέξεις και άλλων γλωσσών κάτι, λεξικό που όμως είναι δύσκολο να κατασκευαστεί.

Σε αντίθεση με τη μηχανική μάθηση, η βασισμένη σε λεξικό μέθοδος δεν απαιτεί την εκπαίδευση ενός ταξινομητή πάνω σε επισημειωμένα δεδομένα εξοικονομώντας έτσι σημαντικό χρόνο. Ωστόσο, απαιτεί ένα συναισθηματικό λεξικό, περιορίζοντας την εφαρμογή της μεθόδου στην ανάλυση κειμένων γραμμένων στην γλώσσα του λεξικού και την απόδοσή της στην ποιότητα ή πληρότητα του λεξικού. Από την άλλη πλευρά, οι μέθοδοι μηχανικής μάθησης πετυχαίνουν συνήθως καλά αποτελέσματα με αντίτιμο την ανάγκη εκπαίδευσης που μπορεί να είναι χρονοβόρα. Γι’ αυτό, η επιστημονική έρευνα τα τελευταία χρόνια προσανατολίζεται στη χρήση υβριδικών μεθόδων που συνδυάζουν λεξικό με μηχανική μάθηση ώστε να επωφεληθούν από τα πλεονεκτήματα των επιμέρους μεθόδων, δηλαδή της ταχύτητας της lexicon-based προσέγγισης και της ακρίβειας της machine learning προσέγγισης.

4.2 Συναισθηματικά Λεξικά

Κάποια δημοσίως διαθέσιμα λεξικά στα αγγλικά που χρησιμοποιούνται σε εφαρμογές sentiment analysis είναι:

- Bing Liu's Opinion Lexicon: Το λεξικό αυτό περιέχει 2006 θετικές και 4783 αρνητικές λέξεις. Περιέχει ορθογραφικά λάθη, μορφολογικές παραλλαγές, αργκό και σημάνσεις social-media (πχ. twitter).
- MPQA Subjectivity Lexicon: Το MPQA (Multi-Perspective Question Answering) subjectivity lexicon διατηρείται από τους Theresa Wilson, Janyce Wiebe και Paul

Hoffmann. Το λεξικό αυτό περιέχει 5097 αρνητικές και 2533 θετικές λέξεις, οι οποίες επισημειώνονται ως λέξεις με ισχυρή ή ασθενή πολικότητα. Ο πίνακας 1 δείχνει πώς είναι η δομή του.

	Strength	Length	Word	Part-of-speech	Stemmed	Polarity
1.	type=weaksubj	len=1	word1=abandoned	pos1=adj	stemmed1=n	priorpolarity=negative
2.	type=weaksubj	len=1	word1=abandonment	pos1=noun	stemmed1=n	priorpolarity=negative
3.	type=weaksubj	len=1	word1=abandon	pos1=verb	stemmed1=y	priorpolarity=negative
4.	type=strongsubj	len=1	word1=abase	pos1=verb	stemmed1=y	priorpolarity=negative
5.	type=strongsubj	len=1	word1=abasement	pos1=anypos	stemmed1=y	priorpolarity=negative
6.	type=strongsubj	len=1	word1=abash	pos1=verb	stemmed1=y	priorpolarity=negative
7.	type=weaksubj	len=1	word1=abate	pos1=verb	stemmed1=y	priorpolarity=negative
8.	type=weaksubj	len=1	word1=abdicate	pos1=verb	stemmed1=y	priorpolarity=negative
9.	type=strongsubj	len=1	word1=aberration	pos1=adj	stemmed1=n	priorpolarity=negative
10.	type=strongsubj	len=1	word1=aberration	pos1=noun	stemmed1=n	priorpolarity=negative
...						
8221.	type=strongsubj	len=1	word1=zest	pos1=noun	stemmed1=n	priorpolarity=positive

Πίνακας 1: Ένα απόσπασμα του MPQA subjectivity lexicon

- SentiWordNet: Το λεξικό αυτό αποδίδει θετικούς και αρνητικούς πραγματικούς αριθμούς ως βαθμολογίες συναισθήματος στα σύνολα συνωνύμων του WordNet. Το WordNet είναι μία μεγάλη λεξιλογική βάση δεδομένων στα αγγλικά που δημιουργήθηκε στο Πανεπιστήμιο Princeton το 1985. Ουσιαστικά, ρήματα, επίθετα και επιρρήματα εντάσσονται σε ομάδες-σύνολα συνωνύμων (synsets) με το καθένα να εκφράζει μία διακριτή έννοια. Το WordNet δίνει σύντομους ορισμούς των λέξεων και παραδείγματα χρήσης τους και περιλαμβάνει σχέσεις μεταξύ των synsets όπως σχέσεις υπερωνυμίας (hyperonymy) ή υπωνυμίας (hyponymy) ή και αντωνυμίας μεταξύ των επιθέτων (antonymy). Ο παρακάτω πίνακας συνοψίζει τη δομή του SentiWordNet.

POS	ID	PosScore	NegScore	SynsetTerms	Gloss
a	00001740	0.125	0	able#1	(usually followed by `to') having the necessary means or [...]
a	00002098	0	0.75	unable#1	(usually followed by `to') not having the necessary means or [...]
a	00002312	0	0	dorsal#2 abaxial#1	facing away from the axis of an organ or organism; [...]
a	00002527	0	0	ventral#2 adaxial#1	nearest to or facing toward the axis of an organ or organism; [...]
a	00002730	0	0	acroscopic#1	facing or on the side toward the apex
a	00002843	0	0	basiscopic#1	facing or on the side toward the base
a	00002956	0	0	abducting#1 abducent#1	especially of muscles; [...]
a	00003131	0	0	adductive#1 adducting#1 adducent#1	especially of muscles; [...]
a	00003356	0	0	nascent#1	being born or beginning; [...]
a	00003553	0	0	emerging#2 emergent#2	coming into existence; [...]

Πίνακας 2: Ένα απόσπασμα του SentiWordNet

- Harvard General Inquirer: Το λεξικό αυτό παρέχει πληροφορίες συντακτικού και σημασιολογικού περιεχομένου σε POS επισημειωμένες λέξεις. Στον πίνακα 3 φαίνεται ενδεικτικά ο πλούτος και η πολυπλοκότητα του λεξικού.

	Entry	Positiv	Negativ	Hostile	...184 classes ...	Othtags	Defined
1	A					DET ART	...
2	ABANDON		Negativ			SUPV	
3	ABANDONMENT		Negativ			Noun	
4	ABATE		Negativ			SUPV	
5	ABATEMENT					Noun	
...							
35	ABSENT#1		Negativ			Modif	
36	ABSENT#2					SUPV	
...							
11788	ZONE					Noun	

Πίνακας 3: Ένα απόσπασμα του Harvard General Inquirer

- LIWC: Το λεξικό LIWC (Linguistic Inquiry and Word Counts) είναι μία ιδιόκτητη βάση δεδομένων που αποτελείται από πολλές κατηγοριοποιημένες κανονικές εκφράσεις. Οι ταξινομήσεις του είναι υψηλά συσχετισμένες με εκείνες του Harvard General Inquirer. Στον πίνακα 4 δίνονται κάποιες σχετικές ως προς το συναίσθημα κατηγορίες με παραδείγματα κανονικών εκφράσεων.

Category	Examples
Negate	aint, ain't, arent, aren't, cannot, cant, can't, couldnt, ...
Swear	arse, arsehole*, arses, ass, asses, asshole*, bastard*, ...
Social	acquainta*, admit, admits, admitted, admitting, adult, adults, advice, advis*
Affect	abandon*, abuse*, abusi*, accept, accepta*, accepted, accepting, accepts, ache*
Posemo	accept, accepta*, accepted, accepting, accepts, active*, admir*, ador*, advantag*
Negemo	abandon*, abuse*, abusi*, ache*, aching, advers*, afraid, aggravat*, aggress*,
Anx	afraid, alarm*, anguish*, anxi*, apprehens*, asham*, aversi*, avoid*, awkward*
Anger	jealous*, jerk, jerked, jerks, kill*, liar*, lied, lies, lous*, ludicrous*, lying, mad

Πίνακας 4: Ένα απόσπασμα του LIWC

- WordNet Affect: Το λεξικό αυτό βασίζεται στα synsets και στο σημασιολογικό γράφο του WordNet. Σε κάθε συναισθηματική έννοια-synset του WordNet αποδίδεται από το λεξικό μία ή περισσότερες συναισθηματικές ετικέτες (a-labels). Στον πίνακα 5 φαίνεται το σύνολο των επιγραφών μαζί με κάποια παραδείγματα synsets.

A-Labels	Examples
EMOTION	*noun anger#1, verb fear#1*
MOOD	*noun animosisy#1, adjective amiable#1*
TRAIT	*noun aggressiveness#1, adjective competitive#1*
COGNITIVE STATE	*noun confusion#2, adjective dazed#2*
PHYSICAL STATE	*noun illness#1, adjective all in#1*
HEDONIC SIGNAL	*noun hurt#3, noun suffering#4*
EMOTION-ELICITING SITUATION	*noun awkwardness#3, adjective out of danger#1*
EMOTIONAL RESPONSE	*noun cold sweat#1, verb tremble#2*
BEHAVIOUR	*noun offense#1, adjective inhibited#1*
ATTITUDE	*noun intolerance#1, noun defensive#1*
SENSATION	*noun coldness#1, verb feel#3*

Πίνακας 5: Ένα απόσπασμα του WordNet Affect

- SenticNet: Το λεξικό αυτό παρέχει ένα σύνολο semantics, sentics και polarity για 30000 έννοιες της αγγλικής γλώσσας. Συγκεκριμένα, με τον όρο semantics (σημασιολογία) εννοούμε τις έννοιες που είναι σημασιολογικά κοντά με την έννοια εισόδου (δηλαδή τις 5 έννοιες που μοιράζονται τα περισσότερα σημασιολογικά χαρακτηριστικά με την έννοια εισόδου), με τον όρο sentics τις τιμές αισθηματικής κατηγοριοποίησης εκφρασμένες στις 4 διαστάσεις συναισθήματος της ευχαρίστησης (pleasantness), της προσοχής (attention), της ευαισθησίας (sensitivity) και της ικανότητας (aptitude) και τέλος με τον όρο polarity, εννοούμε την πολικότητα της έννοιας που είναι ένας πραγματικός αριθμός ανάμεσα στο -1 και στο +1 (όπου με -1 δηλώνεται ακραία αρνητική έννοια και με +1 ακραία θετική). Στον πίνακα 6 φαίνονται οι πληροφορίες που παρέχονται από το λεξικό για μία έννοια εισόδου, πχ. τη λέξη good.

```
▼<rdf:RDF xmlns:rdf="http://w3.org/1999/02/22-rdf-syntax-ns#">
 ▼<rdf:Description rdf:about="http://sentic.net/api/en/concept/good">
    <rdf:type rdf:resource="http://sentic.net/api/concept"/>
    <text xmlns="http://sentic.net">good</text>
    <semantics xmlns="http://sentic.net" rdf:resource="http://sentic.net/api/en/concept/uncommon"/>
    <semantics xmlns="http://sentic.net" rdf:resource="http://sentic.net/api/en/concept/niceness"/>
    <semantics xmlns="http://sentic.net" rdf:resource="http://sentic.net/api/en/concept/like_candy"/>
    <semantics xmlns="http://sentic.net" rdf:resource="http://sentic.net/api/en/concept/hard_find"/>
    <semantics xmlns="http://sentic.net" rdf:resource="http://sentic.net/api/en/concept/pleasant"/>
    <pleasantness xmlns="http://sentic.net" rdf:datatype="http://w3.org/2001/XMLSchema#float">0.92</pleasantness>
    <attention xmlns="http://sentic.net" rdf:datatype="http://w3.org/2001/XMLSchema#float">0.98</attention>
    <sensitivity xmlns="http://sentic.net" rdf:datatype="http://w3.org/2001/XMLSchema#float">0</sensitivity>
    <aptitude xmlns="http://sentic.net" rdf:datatype="http://w3.org/2001/XMLSchema#float">0.75</aptitude>
    <polarity xmlns="http://sentic.net" rdf:datatype="http://w3.org/2001/XMLSchema#float">0.883</polarity>
  </rdf:Description>
</rdf:RDF>
```

Πίνακας 6: Ένα απόσπασμα του SenticNet

Στο λεξικό αυτό βασιστήκαμε για την διεξαγωγή concept-level sentiment analysis, που περιγράφεται στο κεφάλαιο 5.

4.3 Ενδιαφέρουσες Ερευνητικές Προσπάθειες

Κάποιες ενδιαφέρουσες ερευνητικές προσπάθειες ανάλυσης συναισθήματος βάσει λεξικού είναι:

- Οι Balahur et al., [22], εφάρμοσαν sentiment analysis με λεξικό για τον χαρακτηρισμό - κατάταξη εκφράσεων σε εισαγωγικά (παραθέσεις) που έχουν αντληθεί από άρθρα ειδήσεων. Η κατάταξη έγινε ως προς τέσσερις κατηγορίες (positive, negative, high positive, high negative) και χρησιμοποιήθηκαν τέσσερα διαφορετικά λεξικά, τα JRC, WordNet Affect, SentiWordNet, MicroWNOp αλλά και συνδυασμός τους. Σύμφωνα με τα αποτελέσματα, η ποιότητα της ταξινόμησης εξαρτάται σαφώς από την ποιότητα του κάθε λεξικού και ο συνδυασμός των λεξικών οδηγεί στη βέλτιστη απόδοση. Ακόμη το φιλτράρισμα των δεδομένων με έλεγχο της υποκειμενικότητά τους, οδηγεί σε καλύτερα αποτελέσματα.
- Οι Ngoc και Yoo, [23], εφάρμοσαν sentiment analysis με λεξικό για να αξιολογήσουν fan pages στο Facebook. Οι παραδοσιακές μέθοδοι κατάταξης fan pages στο Facebook βασίζονται στην απλή καταμέτρηση των posts, σχολίων και "likes". Η πολικότητα κάθε σχολίου, η οποία μπορεί να είναι θετική, αρνητική ή ουδέτερη αγνοείται σε αυτές τις μεθόδους. Στην εργασία αυτή, οι Ngoc και Yoo υπολογίζουν 2 βαθμολογίες για κάθε σελίδα και η τελική βαθμολογία προκύπτει από τον συνδυασμό τους. Η πρώτη προκύπτει από την καταμέτρηση των likes και η δεύτερη από την πολικότητα των σχολίων των χρηστών. Το λεξικό που χρησιμοποιείται για τον χαρακτηρισμό των σχολίων είναι το αγγλικό λεξικό AFINN το οποίο αποδίδει ακέραιες βαθμολογίες σε λέξεις και φράσεις από -5 (negative) μέχρι και +5 (positive).
- Οι Kolchyna et al., [24], εφάρμοσαν sentiment analysis με μεθόδους machine learning και λεξικού σε δεδομένα από το twitter. Έδειξαν ότι εμπλουτίζοντας τα συναισθηματικά λεξικά με emoticons, συντομογραφίες και εκφράσεις αργκό που χρησιμοποιούνται συχνά στα μέσα κοινωνικής δικτύωσης αυξάνεται η ακρίβεια ταξινόμησης tweets. Έδειξαν επίσης ότι η κατάλληλη μέθοδος εξαγωγής χαρακτηριστικών για ταξινόμηση με μεθόδους machine learning, όπως οι Naive Bayes και SVM, υπερισχύει του λεξικού. Πρότειναν τέλος μία συγχώνευση (fusion) των 2 τεχνικών (λεξικού και machine learning) μέσω της εισαγωγής του lexicon-based sentiment score σαν χαρακτηριστικό εισόδου για την προσέγγιση με machine learning. Ο συνδυασμός αυτός έδειξε να παράγει πιο ακριβείς ταξινομήσεις.

Κεφάλαιο 5

Υλοποίηση

Όπως αναφέραμε και στα προηγούμενα κεφάλαια, στην εργασία αυτή εξετάσαμε την αυτόματη ανάλυση συναισθήματος με χρήση λεξικού, αλγορίθμων επιβλεπόμενης μηχανικής μάθησης αλλά και συνδυασμού αυτών. Σαν δεδομένα, χρησιμοποιήθηκαν οι κριτικές ταινιών που χρησιμοποιήθηκαν από τους Pang και Lee στην εργασία [4], ένα σύνολο 10662 μικρών σε έκταση (με μέγεθος μιας περιόδου) κριτικών ταινιών, από τις οποίες οι μισές ανήκουν στην θετική και οι υπόλοιπες μισές στην αρνητική κλάση. Το dataset αυτό χρησιμοποιήθηκε έκτοτε σε πλήθος δημοσιεύσεων διευκολύνοντας τις συγκρίσεις μεταξύ των διαφορετικών υλοποιήσεων.

Η υλοποίηση των πειραμάτων έγινε κατά κύριο ρόλο στη γλώσσα προγραμματισμού Python, λόγω της απλότητας χρήσης της και των διαθέσιμων εργαλειοθηκών (toolkits) που επιτρέπουν την επεξεργασία φυσικής γλώσσας (Natural Language Toolkit - NLTK[3]), τον υπολογισμό στοιχείων γραμμικής άλγεβρας (NumPy[4]), την υλοποίηση αλγορίθμων μηχανικής μάθησης (scikit-learn – sklearn[5]) και τον ορισμό συμβολικών μαθηματικών εκφράσεων καθώς και την αποδοτική διαφόριση αυτών (Theano[6]). Για την υλοποίηση του MLP χρησιμοποιήθηκε η γλώσσα προγραμματισμού Matlab και το αντίστοιχο toolbox.

5.1 Δεδομένα

Τα δεδομένα που χρησιμοποιήσαμε για την ταξινόμηση και την εκτέλεση των πειραμάτων μας είναι το σύνολο κριτικών ταινιών σε επίπεδο πρότασης που συστήθηκε από τους Pang και Lee το 2005 (sentence polarity dataset v1.0). Στο dataset αυτό όμως, δεν ήταν όλες οι κριτικές στα αγγλικά όπως αναμέναμε. Υπήρχαν και κάποιες λίγες κριτικές που ήταν γραμμένες στα ισπανικά, όπως πχ. η πρόταση “la cinta comienza intentando ser un drama , rápidamente se transforma en una comedia y termina por ser una parodia absolutamente predecible”.

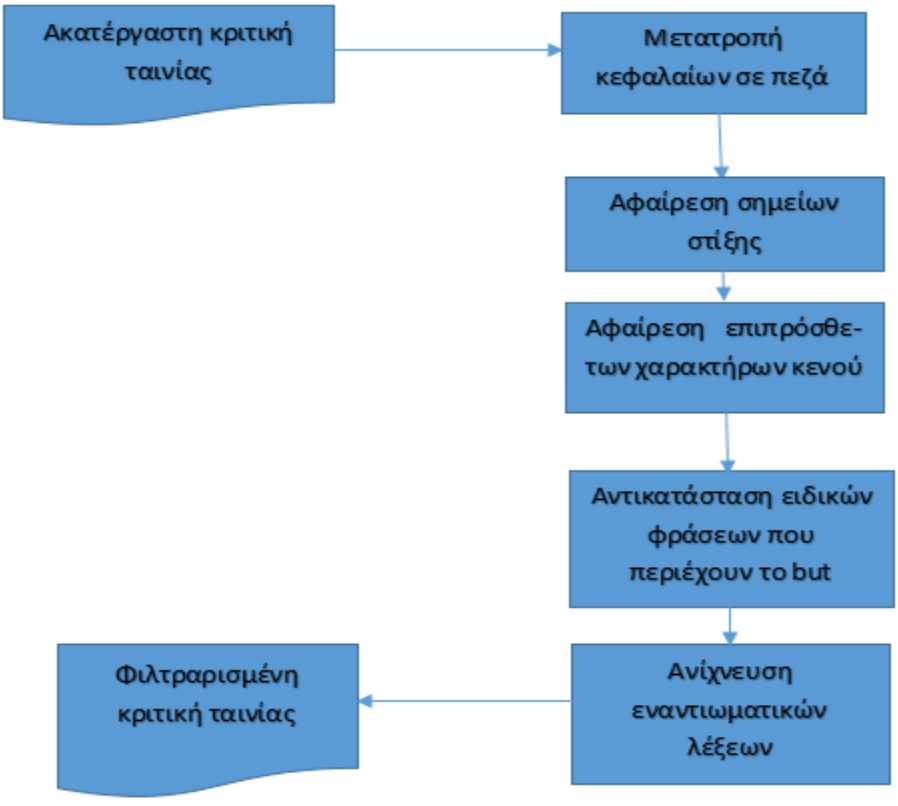
Αφαιρέσαμε αρχικά από το dataset των 10662 κριτικών ταινιών τις 56 προτάσεις που ήταν γραμμένες στην ισπανική γλώσσα. Τέτοιες προτάσεις θα αγνοούνταν από ένα οποιαδήποτε συναισθηματικό λεξικό αγγλικής γλώσσας, όπως το SenticNet που χρησιμοποιήσαμε, και συνεπώς η εξέτασή τους θα είχε νόημα μόνο αν χρησιμοποιούσαμε ένα αρκετά πλούσιο λεξικό με λέξεις και στα ισπανικά ή για την κατασκευή των features vectors για machine learning χρησιμοποιούσαμε corpus-based τεχνικές που θα επέτρεπαν την αναπαράσταση των αντίστοιχων

[3] http://www.nltk.org/
[4] http://www.numpy.org/
[5] http://scikit-learn.org/stable/
[6] http://deeplearning.net/software/theano/

ισπανικών λέξεων. Ωστόσο, για την κατασκευή των features vectors στα πειράματά μας χρησιμοποιήσαμε ένα υποσύνολο των καταχωρήσεων του SenticNet όπως παρουσιάζεται στη συνέχεια. Έτσι, το dataset μας αποτελείται από 5292 θετικές και 5314 αρνητικές κριτικές ταινιών.

5.2 Προεπεξεργασία Δεδομένων

Πριν την κατασκευή των feature vectors συνήθως εφαρμόζουμε κάποιες τεχνικές προεπεξεργασίας των δεδομένων, όπως αναφέρθηκε στην ενότητα 2.2. Ο σκοπός είναι η κανονικοποίηση της εισόδου για τη μείωση της αναπαράστασης με ταυτόχρονη απόρριψη της πλεονάζουσας-άχρηστης πληροφορίας που καλείται θόρυβος. Έτσι, μειώνεται η διάσταση του προβλήματος και ο κίνδυνος του overfitting και ταυτόχρονα μειώνεται και ο απαιτούμενος χρόνος ταξινόμησης. Ο αλγόριθμος προεπεξεργασίας των δεδομένων που εφαρμόσαμε φαίνεται στο ακόλουθο διάγραμμα ροής.



Σχήμα 19: Διάγραμμα ροής του αλγορίθμου προεπεξεργασίας των δεδομένων

Τα επιμέρους βήματα περιγράφονται ακολούθως.

- Μετατροπή κεφαλαίων σε πεζά

  Μετατρέπουμε όλα τα κεφαλαία γράμματα σε πεζά διότι η διαφοροποίηση πχ. των λέξεων Great και great δεν παίζει κάποιο ρόλο στο πρόβλημα του sentiment analysis, μιας και δεν αλλάζει η πολικότητα των λέξεων παρά μόνο ίσως η έντασή της (με τη χρήση κεφαλαίων). Η επεξεργασία αυτή γίνεται επίσης διότι οι λέξεις του κειμένου που αναλύεται αναζητούνται σε λεξικό, εν προκειμένω στο SenticNet, όπου η αναπαράσταση των λέξεων είναι με όλους τους χαρακτήρες πεζούς. Οι κριτικές του dataset μας είναι ήδη γραμμένες με πεζά γράμματα, έτσι το βήμα αυτό έχει νόημα κυρίως στη φάση της ανάκλησης και της ταξινόμησης μιας νέας άγνωστης κριτικής.

- Αφαίρεση σημείων στίξης

  Όπως τα κεφαλαία γράμματα, έτσι και τα σημεία στίξης δεν επιδρούν άμεσα την πολικότητα του συναισθήματος . Μερικές φορές απλώς, η χρήση τους (πχ. του «!») μπορεί να ενισχύσει το συναίσθημα. Έτσι, μπορούν να αγνοηθούν, για το πρόβλημα της ταξινόμησής μας. Επειδή στη συνέχεια βασιζόμαστε στα σημεία στίξης για τον χειρισμό της άρνησης, στο βήμα αυτό αγνοούμε μόνο τα ακόλουθα σημεία στίξης: *#,_,:,(,)*

- Αφαίρεση επιπρόσθετων χαρακτήρων κενού

  Σε περίπτωση που στη συμβολοσειρά εισόδου υπάρχουν μεταξύ δύο λέξεων 2 ή περισσότεροι χαρακτήρες κενού, εξαλείφουμε τους πλεονάζοντες χαρακτήρες κενού.

- Αντικατάσταση ειδικών φράσεων που περιέχουν το but

  Αντικαθιστούμε στο βήμα αυτό κάποιες ειδικές φράσεις που περιέχουν τον σύνδεσμο but με το ισοδύναμο νόημα τους. Ο λόγος είναι ότι στις φράσεις αυτές ο όρος but δεν επιτελεί το συνηθισμένο αντιθετικό του χαρακτήρα. Στην εργασία αυτή, αντικαθιστούμε την φράση “anything but” με τη λέξη “not” και την φράση “nothing but” με τη λέξη “only”.

- Ανίχνευση εναντιωματικών λέξεων

  Στο βήμα αυτό ανιχνεύουμε μέσα στην συμβολοσειρά εισόδου εναντιωματικές λέξεις όπως είναι οι αντιθετικοί σύνδεσμοι but, although, however ή οι φράσεις in spite of, despite, even though. Η ανίχνευση αυτή βασίζεται στη σκέψη ότι μία πρόταση που περιέχει αντιθετικό σύνδεσμο, έχει συνήθως πολικότητα που καθορίζεται από το δεύτερο συστατικό της πρότασης (δεξιά του συνδέσμου), όπως είδαμε στην ενότητα 4.1 το οποίο και εξάγουμε για περαιτέρω ανάλυση που περιγράφεται στην ενότητα 5.3. Αυτό ειδικότερα συμβαίνει με τους συνδέσμους but και however. Αν σε μία πρόταση περιέχεται κάποια από τις υπόλοιπες προαναφερθείσες εναντιωματικές λέξεις ή φράσεις, τότε το ρόλο του δεύτερου συστατικού παίζει το μέρος της πρότασης που είναι δεξιά του πρώτου χαρακτήρα κόμματος αν η εναντιωματική λέξη ή φράση βρίσκεται στην αρχή της πρότασης, αλλιώς χωρίζουμε την πρόταση στην εναντιωματική αυτή φράση και τότε η πολικότητα της πρότασης καθορίζεται από το πρώτο συστατικό (αριστερά της εναντιωματικής φράσης). Για παράδειγμα, η πρόταση

"Although the movie was boring in the beginning, I finally liked it."

θεωρείται από την ανάλυσή μας θετική. Παρομοίως η πρόταση

"I found the movie interesting despite the fact that I don't really like the leading actor"

θεωρείται και αυτή από την ανάλυσή μας θετική.

Αν το «δεύτερο συστατικό» μιας πρότασης έχει μηδενική αναπαράσταση λόγω του ότι κανένα από τα επιμέρους concepts της δεν βρέθηκε στο SenticNet, αναλύεται το «πρώτο συστατικό της» και το polarity της πρότασης προκύπτει αντίθετο από το polarity που βρίσκεται από το λεξικό. Αντίστοιχα για την προσέγγιση με machine learning, δηλώνουμε την ύπαρξη αντίθεσης προσθέτοντας ως επιπλέον χαρακτηριστικό την λέξη but.

Ακολουθούν κάποια παραδείγματα προεπεξεργασίας κριτικών ταινιών από τις οποίες οι 4 πρώτες αντλήθηκαν από το dataset μας και η 5η είναι μία νέα άγνωστη κριτική που μπορεί να δώσει ένας χρήστης.

1. *sent = "effective but too-tepid biopic"* →

   s = str2 = "too-tepid biopic", str1 = "effective"

2. *sent = "although it bangs a very cliched drum at times, this crowd-pleaser's fresh dialogue, energetic music, and good-natured spunk are often infectious."* →

   str1 = "although it bangs a very cliched drum at times"
   s = str2 = "this crowd-pleaser's fresh dialogue, energetic music, and good-natured spunk are often infectious."

3. *sent = "the film seems a dead weight. the lack of pace kills it, although, in a movie about cancer, this might be apt."* →

   s = str2 = "the film seems a dead weight. the lack of pace kills it"
   str1 = ", in a movie about cancer, this might be apt."

4. *sent = "this is nothing but familiar territory."* →

   s = "this is only familiar territory."

5. *sent = "I want to say just this: The movie was awesome!!!"* →

   s = "i want to say just this the movie was awesome!!!"

Η προεπεξεργασία που περιγράψαμε εκτελείται σε πρώτο βήμα, προτού ακολουθήσει η κυρίως ανάλυση με λεξικό και αλγορίθμους machine learning, προσεγγίσεις που περιγράφονται στις ενότητες 5.3 και 5.4 αντίστοιχα.

5.3 Ανάλυση με Λεξικό

Αφού φιλτράραμε τις κριτικές ταινιών από το dataset μας μειώνοντας το θόρυβο που δυσχεραίνει το έργο της ανάλυσης συναισθήματος, μένει να εξάγουμε τους πιθανές έννοιες-concepts της κάθε πρότασης ώστε να διεξάγουμε ανάλυση συναισθήματος σε επίπεδο εννοιών, όπως για παράδειγμα γίνεται στις εργασίες [20] και [21].

Σαν συναισθηματικό λεξικό χρησιμοποιήσαμε το SenticNet, που περιγράφηκε στην ενότητα 4.2. Το SenticNet είναι ένα λεξικό 30000 εννοιών και για κάθε μία από αυτές παρέχει ένα σκορ για τις 4 διαστάσεις συναισθήματος (Pleasantness, Attention, Sensitivity, Aptitude) και ένα σκορ για το polarity που κυμαίνεται από το -1 έως στο +1. Εμείς επικεντρωθήκαμε στο σκορ για το polarity.

Πριν αναζητήσουμε στο λεξικό τα concepts που εκφράζονται σε κάθε πρόταση, θα πρέπει να κανονικοποιήσουμε τη μορφή τους εφαρμόζοντας stemming (πχ με τον αλγόριθμο Porter) ή λημματοποίηση (lemmatization). Και οι δύο τεχνικές αποσκοπούν στην αποκοπή των μορφολογικών καταλήξεων των λέξεων. Η διαφορά τους είναι ότι η τεχνική του stemming εφαρμόζει μία πιο ακατέργαστη (crude) αποκοπή, συχνά δεν δίνει ως αποτέλεσμα έγκυρες λέξεις και εξαρτάται μόνο από τη λέξη αγνοώντας το context, ενώ η τεχνική του lemmatization από την άλλη επιστρέφει το λήμμα μίας λέξης, δηλαδή μία έγκυρη λέξη και εξαρτάται από το μέρος του λόγου (POS tag) της λέξης. Η διαφορά μεταξύ stemming και lemmatization φαίνεται στα παρακάτω παραδείγματα.

$$replacements \rightarrow stem: replac$$
$$replacements \rightarrow lemma: replacement$$
$$better \rightarrow stem: better$$
$$better\ (adjective) \rightarrow lemma: good$$
$$went \rightarrow stem: went$$
$$went\ (verb) \rightarrow lemma: go$$
$$meeting\ (noun) \rightarrow lemma: meeting$$
$$meeting\ (verb\ gerund) \rightarrow lemma: meet$$

Εμείς επιλέξαμε την λημματοποίηση κυρίως διότι το αποτέλεσμά της θα πρέπει να συγκρίνεται με τις έγκυρες έννοιες από το λεξικό SenticNet. Για την λημματοποίηση χρησιμοποιήσαμε τον WordNet Lemmatizer από το NLTK ο οποίος λαμβάνει υπόψη την POS ετικέτα των λέξεων.

Πιο συγκεκριμένα, εκμεταλλευόμενοι τις συναρτήσεις που περιέχουν τα modules του NLTK, εφαρμόσαμε πρώτα τον word tokenizer του NLTK πάνω στην πρόταση s που ορίστηκε στην προηγούμενη ενότητα και έπειτα χρησιμοποιήσαμε τον POS tagger του NLTK για τον

χαρακτηρισμό του μέρους του λόγου των επιμέρους λέξεων. Στη συνέχεια, χρησιμοποιήσαμε τον WordNet Lemmatizer για να πάρουμε τα λήμματα των ρημάτων και των ουσιαστικών με βάση την ετικέτα που παράχθηκε προηγουμένως.

Οι έννοιες που παράγουμε από μία πρόταση είναι στην ουσία κάποιοι συνδυασμοί των λέξεων που προκύπτουν μετά τη λημματοποίηση. Αφού ορίσουμε σαν ονοματική φράση (noun phrase – NP) την κανονική έκφραση που αποτελείται από ένα άρθρο (προαιρετικά), ένα ή περισσότερα επίθετα και ένα ή περισσότερα ουσιαστικά και σαν ρηματική φράση (verb phrase – VP) την κανονική έκφραση που αποτελείται από ένα ή περισσότερα ρήματα (για την αντιμετώπιση πχ. της φράσης have taken) και προαιρετικά από μία πρόθεση ή μετοχή (για την αντιμετώπιση phrasal verbs, δηλαδή περιπτώσεων της μορφής give up) κάνουμε τη συντακτική ανάλυση (parsing) της πρότασης. Στη συνέχεια, βασιζόμενοι στο συντακτικό δέντρο της πρότασης, εξάγουμε:

- *event concepts*

  Για την κατασκευή της λίστας αυτής συσχετίζουμε κάθε (φραστικό) ρήμα της πρότασης με κάθε ουσιαστικό που ακολουθεί στην αντίστοιχη ονοματική φράση, ή με το επόμενο επίρρημα.

- *noun phrase list*

  Στη λίστα αυτή ανήκουν οι φράσεις που αποτελούνται από την διαδοχή ενός η περισσότερων επιθέτων από ένα ή περισσότερα ουσιαστικά.

- *noun list*

  Στη λίστα αυτή ανήκουν όλα τα ουσιαστικά της πρότασης ανεξάρτητα από το αν χαρακτηρίζονται από επίθετο ή όχι. Ο λόγος που συμπεριλαμβάνουμε στη λίστα αυτή ουσιαστικά που ανήκουν και στην προηγούμενη λίστα είναι επειδή αν «χάσουμε» (λόγω ανεπάρκειας του λεξικού SenticNet) έννοιες όπως good work να μπορούμε να βρούμε τουλάχιστον τη λέξη work.

- *verb list*

  Η λίστα αυτή αποτελείται από όλα τα (φραστικά) ρήματα της πρότασης. Αν σε μία ρηματική φράση υπάρχει το ρήμα have (για τον σχηματισμό ρήματος παρακείμενου χρόνου) ή το ρήμα be (λήμμα του was,am κτλ.) αυτά απομακρύνονται από την ανάλυσή μας γιατί δεν συνεισφέρουν κάπως στο έργο μας. Τα ρήματα που απομένουν μετά την απομάκρυνση αυτή εισάγονται στη λίστα.

- *lone adjectives*

Η λίστα αυτή αποτελείται από τα επίθετα της πρότασης που δεν ακολουθούνται αμέσως μετά από ουσιαστικό, όπως γίνεται στην περίπτωση της ονοματικής φράσης που ορίσαμε προηγουμένως.

- *adjectives from np*

  Στη λίστα αυτή εισάγονται τα επίθετα που είναι μέρος μίας ονοματικής φράσης.

- *adverb list*

  Στη λίστα αυτή εισάγονται όλα τα επιρρήματα της πρότασης.

Για τον χειρισμό της άρνησης κατασκευάζουμε μία επιπλέον λίστα, την *invert polarity*, στην οποία εισάγουμε όλες τις έννοιες (προκύπτουν με τους παραπάνω τρόπους) που ακολουθούν μία λέξη άρνησης και βρίσκονται πριν από το επόμενο σημείο στίξης. Σαν λέξεις άρνησης, στην ανάλυσή μας, θεωρήσαμε τις *λέξεις not, n't, never,without, no* και *nothing.* Τα σημεία στίξης που οριοθετούν την εμβέλεια της άρνησης είναι τα *"!", "?", ".", ";", ",", "..."*. Στις έννοιες που ανήκουν στην λίστα αυτή θα αντιστοιχισθεί αντίθετο polarity από αυτό του SenticNet.

Ιδιαίτερη προσοχή πρέπει να δώσουμε στο εξής: Φράσεις της μορφής *not only* ή *not merely* δεν συνιστούν άρνηση αλλά χρησιμοποιούνται για να αποδώσουν έμφαση όπως φαίνεται για παράδειγμα στην πρόταση "The movie is not only boring but also offensive". Το στοιχείο αυτό λήφθηκε υπόψη τόσο κατά την ανίχνευση της άρνησης όσο και κατά την ερμηνεία του συνδέσμου but που εδώ λειτουργεί σαν and.

Αφού κατασκευάσαμε τις παραπάνω λίστες, αναζητούμε τα στοιχεία τους στο SenticNet. Η αναζήτηση αυτή γίνεται με προσοχή ώστε να μην αναζητήσουμε όρους που έχουν βρεθεί προηγουμένως ως στοιχεία εννοιών. Για παράδειγμα, αν βρούμε την ονοματική φράση good movie, δεν θα πρέπει μετά να αναζητήσουμε στο λεξικό και τις λέξεις good (από τη λίστα *adjectives from np*) και movie (από τη λίστα *noun list*). Έτσι, πρώτα αναζητούμε τα στοιχεία των *event concepts* και *noun phrases* στο λεξικό. Με βάση τα στοιχεία που βρέθηκαν από τις αναζητήσεις αυτές «φιλτράρουμε» την λίστα *noun list* και ακολούθως αναζητούμε τα στοιχεία της στο SenticNet. Στη συνέχεια, φιλτράρουμε με βάση τα ευρήματα από τη *noun phrase* την λίστα *adjectives from np* την οποία συνενώνουμε με την *lone adjectives* δημιουργώντας τη λίστα *adjective list* και με βάση τα ευρήματα από την *event concepts* φιλτράρουμε τις *verb list* και *adverb list*. Αναζητούμε, τέλος, στο λεξικό και τα στοιχεία των τριών νέων αυτών λιστών. Αφού αντιστρέψουμε το polarity των όρων που βρέθηκαν και ανήκουν και στην *invert polarity,* παίρνουμε τον μέσο όρο των σκορ και το αποδίδουμε σαν συνολικό polarity στην πρόταση εισόδου. Η πρόταση χαρακτηρίζεται θετική αν το συνολικό polarity είναι θετικό, αρνητική αν το συνολικό polarity είναι αρνητικό και ουδέτερη (άρα εξ' ορισμού λανθασμένα ταξινομημένη εφόσον έχουμε δυαδικό πρόβλημα ταξινόμησης) αν το συνολικό polarity είναι 0 που στην πράξη σημαίνει ότι δεν βρέθηκε καμία έννοια της πρότασης στο SenticNet.

5.4 Ανάλυση με Μηχανική Μάθηση

Για την κατασκευή των feature vectors και την εκτέλεση των διάφορων αλγορίθμων machine learning, θεωρήσαμε την Bag of Concepts αναπαράσταση του κειμένου εισόδου. Πιο συγκεκριμένα, κατά την εκτέλεση του αλγορίθμου ταξινόμησης με λεξικό που περιγράφηκε προηγουμένως, για κάθε πρόταση εισόδου από τις 10606 εισάγαμε σε μία λίστα τις «σημαντικές» έννοιες με polarity που συμφωνεί με την ετικέτα της πρότασης. Ως «σημαντικές» εννοούμε τις έννοιες αυτές που έχουν απόλυτη τιμή polarity μεγαλύτερη ή ίση του 50% της μέγιστης απόλυτης τιμής polarity, μεταξύ των εννοιών που ταιριάζουν με την ετικέτα κλάσης της πρότασης. Έτσι, προέκυψε ένα λεξιλόγιο 3082 εννοιών. Σε αυτό προσθέσαμε τις λέξεις άρνησης not, never, no, without και τον αντιθετικό σύνδεσμο but, με αποτέλεσμα τελικά το λεξιλόγιο για την κατασκευή των feature vectors να αποτελείται από 3087 στοιχεία.

Αφού αναλύσουμε τις προτάσεις και κατασκευάσουμε τις λίστες με τις έννοιες όπως πριν, αναζητούμε τώρα τις έννοιες όχι στο SenticNet αλλά στο νέο λεξιλόγιο των 3087 εννοιών. Σε 231 από τις 10606 προτάσεις δεν βρέθηκε καμία έννοια σε αυτό το νέο λεξιλόγιο των 3087 εννοιών και απορρίφθηκαν από τις επόμενες αναλύσεις καθώς ο συνυπολογισμός τους θα δυσκόλευε το έργο του ταξινομητή. Τελικά, για κάθε πρόταση από τις εναπομείναντες 10375, το διάνυσμα χαρακτηριστικών $\boldsymbol{x}$ θα έχει μήκος 3087 με την ύπαρξη μίας έννοιας να κωδικοποιείται με 1 στην αντίστοιχη θέση και την απουσία της με 0. Από τις 10375 αυτές προτάσεις, οι 5224 είναι θετικές και οι 5151 είναι αρνητικές.

Εξετάσαμε στη συνέχεια την επίδραση και άλλων χαρακτηριστικών, ανεξάρτητων από το SenticNet, στην απόδοση του Multinomial Naive Bayes που είναι απλός στην εκπαίδευσή του. Συγκεκριμένα εξετάσαμε τη χρήση unigrams, bigrams αλλά και συνδυασμού αυτών, με και χωρίς τη χρήση stemming.

5.4.1 Υλοποίηση Naive Bayes, Maximum Entropy, SVM

Για την υλοποίηση των παραπάνω αλγορίθμων supervised machine learning χρησιμοποιήσαμε το scikit-learn, μία βιβλιοθήκη της Python που περιέχει απλά και αποδοτικά εργαλεία για την εξόρυξη και ανάλυση δεδομένων. Η εκπαίδευση και αξιολόγηση των ταξινομητών είναι πολύ απλή χάρη στις έτοιμες (built-in) μεθόδους fit και score. Για την επιλογή των υπερπαραμέτρων των αλγορίθμων και συγκεκριμένα:

- της παραμέτρου ομαλοποίησης C που σχετίζεται με τον Maximum Entropy ταξινομητή (η παράμετρος αυτή ερμηνεύεται ως το αντίστροφο της ομαλοποίησης, όπως στα SVMs)
- της παραμέτρου ομαλοποίησης C που σχετίζεται με τον SVM ταξινομητή με γραμμικό πυρήνα
- της παραμέτρου ομαλοποίησης C και της παραμέτρου πυρήνα γ που σχετίζεται με τον SVM ταξινομητή με rbf πυρήνα

εφαρμόσαμε εξαντλητική αναζήτηση πάνω σε πεπερασμένα σύνολα λογικών τιμών για κάθε υπερπαράμετρο και τελικά επιλέξαμε τις τιμές

$$C(Maximum\ Entropy) = 0.2, C(LinearSVM) = 0.03, C(RbfSVM) = 100, \gamma = 0.001$$

Σε περίπτωση που δεν δίνονται ρητά σύνολα εκπαίδευσης και ελέγχου, όπως στην περίπτωσή μας, μπορούμε πέρα την κλασικής επιλογής του τυχαίου διαχωρισμού των δεδομένων στα 2 σύνολα με κάποια αναλογία (πχ. $train = 90\%$, $test = 10\%$) να εκτελέσουμε k-fold cross-validation (πχ k=10), επιλογή που δίνει μια πιο ακριβή εκτίμηση της ορθότητας του μοντέλου. Για την αξιολόγηση των αλγορίθμων χρησιμοποιήσαμε 10-fold cross-validation μέσω της συνάρτησης cross_val_score.

5.4.2 Υλοποίηση MLP

Η υλοποίηση του multilayer perceptron (MLP) έγινε στη γλώσσα προγραμματισμού Matlab κάνοντας χρήση της συνάρτησης patternet.

Το νευρωνικό μας δίκτυο αποτελείται από 3 επίπεδα, το επίπεδο εισόδου, ένα κρυφό επίπεδο και το επίπεδο εξόδου και έχει τη μορφή του σχήματος 9. Οι διαφορετικές αρχιτεκτονικές που εξετάσαμε εξαρτώνται από τη διάσταση του κρυφού επιπέδου d_{hidden} και φαίνονται στον πίνακα 7.

Το επίπεδο εισόδου αποτελείται από 3087 νευρώνες, έναν για κάθε χαρακτηριστικό. Το κρυφό επίπεδο αποτελείται όπως είπαμε από μεταβλητό αριθμό νευρώνων που αποτελεί αντικείμενο πειραματισμού. Η συνάρτηση μεταφοράς των νευρώνων του κρυφού επιπέδου είναι η υπερβολική εφαπτομένη

$$\varphi\left(v_i^{hid}\right) = tanhv = \frac{e^{v_i^{hid}} - e^{-v_i^{hid}}}{e^{v_i^{hid}} + e^{-v_i^{hid}}} \quad , i = 1, \dots, d_{hidden}$$

Το επίπεδο εξόδου αποτελείται από 2 νευρώνες, έναν για κάθε κλάση, με συνάρτηση μεταφοράς την συνάρτηση softmax

$$\varphi(v_i^{out}) = \frac{e^{v_i^{out}}}{\sum_{j=1}^{2} e^{v_j^{out}}} \quad , i = 1,2$$

Τα παραδείγματα εκπαίδευσης $\boldsymbol{x_i}$ συνοδεύονται από διανύσματα στόχων της μορφής

$$\boldsymbol{d_i} = (1,0)^T \text{ ή } \boldsymbol{d_i} = (0,1)^T$$

ανάλογα με το αν το πρότυπο x_i ανήκει στην κατηγορία 1 (θετική) ή στην κατηγορία 2 (αρνητική).

	Διάσταση επιπέδου εισόδου	Διάσταση κρυφού επιπέδου	Διάσταση επιπέδου εξόδου
Αρχιτεκτονική 1	3087	5	2
Αρχιτεκτονική 2	3087	10	2
Αρχιτεκτονική 3	3087	15	2
Αρχιτεκτονική 4	3087	20	2
Αρχιτεκτονική 5	3087	25	2
Αρχιτεκτονική 6	3087	30	2
Αρχιτεκτονική 7	3087	50	2
Αρχιτεκτονική 8	3087	100	2
Αρχιτεκτονική 9	3087	150	2
Αρχιτεκτονική 10	3087	500	2

Πίνακας 7: Οι αρχιτεκτονικές του 3-layer MLP που εξετάσαμε

Η εκπαίδευση γίνεται με τον αλγόριθμο backpropagation και χρησιμοποιώντας σαν συνάρτηση κριτηρίου το κόστος διεντροπίας.

Η αξιολόγηση της απόδοσης κάθε μίας από τις ανωτέρω αρχιτεκτονικές γίνεται με τη μέθοδο 10-fold cross-validation. Σε κάθε επανάληψη της μεθόδου, χωρίζουμε με τυχαίο τρόπο τα δεδομένα εκπαίδευσης σε 2 σύνολα, training και validation, με αναλογία 90% προς 10%. Τα δεδομένα του πρώτου συνόλου χρησιμοποιούνται αποκλειστικά για την εκπαίδευση του μοντέλου, ενώ τα δεδομένα του δεύτερου συνόλου χρησιμοποιούνται για τον έλεγχο του overfitting επιτρέποντας το Early Stopping.

5.5 Ανάλυση με Συνελικτικά Νευρωνικά Δίκτυα

Η υλοποίηση του συνελικτικού νευρωνικού δικτύου (ΣΝΔ) έγινε στη γλώσσα προγραμματισμού Python κάνοντας χρήση και της βιβλιοθήκης Python Theano.

Το ΣΝΔ αποτελείται από το επίπεδο εισόδου, όπου γίνεται η αναπαράσταση της πρότασης σε μορφή πίνακα, ένα συνελικτικό επίπεδο, όπου γίνεται το φιλτράρισμα και η εξαγωγή των χαρακτηριστικών, ένα επίπεδο max-pooling, όπου γίνεται υποδειγματοληψία μέσω της πράξης max και από κάθε feature map διατηρείται μία μόνο τιμή και τέλος από ένα πλήρως συνδεδεμένο επίπεδο με softmax έξοδο, όπου εκτελείται η τελική ταξινόμηση. Η μορφή του ΣΝΔ φαίνεται στο σχήμα 18, με τη διαφορά ότι θα χρησιμοποιήσουμε ένα κανάλι για την είσοδο, ένα σύνολο από word vectors τα οποία προσαρμόζονται κατά την εκπαίδευση (non-static model).

Για την αναπαράσταση μιας πρότασης σε μορφή πίνακα, εκτελούμε πρώτα την ακόλουθη προεπεξεργασία των δεδομένων. Αντικαθιστούμε κάθε χαρακτήρα που δεν ανήκει στη λίστα

$[A-Z, a-z, 0-9, (,), !, ?, ', `]$ με τον κενό χαρακτήρα, προσθέτουμε σε κάθε έκφραση που περιέχει απόστροφο ('), πχ. n't, ένα κενό χαρακτήρα στην αρχή και αντικαθιστούμε 2 ή περισσότερα διαδοχικούς κενούς χαρακτήρες με ένα χαρακτήρα κενού. Κάθε πρόταση ανατίθεται με τυχαίο τρόπο σε ένα από 10 folds (θα χρησιμοποιήσουμε 10-fold CV) και αφού «καθαριστεί» όπως περιγράψαμε πριν, αναλύεται στις λέξεις της. Με τον τρόπο αυτό κατασκευάζεται το λεξιλόγιο του dataset. Για την αναπαράσταση των λέξεων με διανύσματα επιλέχθηκαν τα pre-trained word vectors του αλγορίθμου word2vec πάνω σε 100 δισεκατομμύρια λέξεις από τη Google News. Τα διανύσματα αυτά έχουν διάσταση ίση με 300. Σε κάθε λέξη του λεξιλογίου αντιστοιχούμε έναν αριθμό που δηλώνει τη γραμμή ενός πίνακα W που περιέχει το αντίστοιχο word vector. Λέξεις που δεν ανήκουν στο σύνολο των pre-trained word vectors αρχικοποιούνται με τυχαία διανύσματα word vectors.

Σαν τεχνικές κανονικοποίησης (regularization) χρησιμοποιήσαμε την τεχνική dropout στο πλήρως συνδεδεμένο επίπεδο, με έναν περιορισμό της μέγιστης l_2 νόρμας των διανυσμάτων βαρών.

Για το πείραμά μας χρησιμοποιήσαμε ReLU συναρτήσεις ενεργοποίησης για τους νευρώνες του συνελικτικού επιπέδου, μεγέθη φίλτρων (h) ίσα με 3,4 και 5 και 100 feature maps το καθένα, πιθανότητα dropout (p) ίση με 0.5, l_2 περιορισμό (s) ίσο με 3 και παρτίδες (mini-batches) μεγέθους 50. Οι τιμές αυτές αποκτήθηκαν με εξαντλητική αναζήτηση πάνω σε ένα προκαθορισμένο σύνολο τιμών των παραμέτρων (grid search).

Η εκπαίδευση γίνεται κατά τα γνωστά με τον αλγόριθμο backpropagation. Συνεπώς, πρέπει να υπολογίσουμε τις μερικές παραγώγους της συνάρτησης κόστους, που εδώ ορίζεται ως το μέσο κόστος διεντροπίας (ή ισοδύναμα η αρνητική λογαριθμική συνάρτηση πιθανοφάνειας) πάνω σε ένα mini-batch, ως προς τις προσαρμόσιμες παραμέτρους του μοντέλου στις οποίες ανήκουν τα word vectors, τα βάρη και οι πολώσεις των νευρώνων του συνελικτικού επιπέδου και τα βάρη και οι πολώσεις των νευρώνων του πλήρως συνδεδεμένου επιπέδου. Ο υπολογισμός αυτός διευκολύνεται χρησιμοποιώντας τη βιβλιοθήκη Theano που επιτρέπει την συμβολική διαφόριση συναρτήσεων. Έτσι, αφού ορίσουμε τη δομή του ΣΝΔ και την εμπρόσθια λειτουργία του, ο υπολογισμός της κλίσης του κόστους διεντροπίας ως προς τις προσαρμόσιμες παραμέτρους γίνεται πολύ εύκολα χρησιμοποιώντας τις δυνατότητες της Theano.

Σαν μέγιστο αριθμό εποχών εκπαίδευσης ορίσαμε τις 25 επαναλήψεις. Τα δεδομένα εκπαίδευσης χωρίστηκαν σε δεδομένα αποκλειστικά για εκπαίδευση και δεδομένα επαλήθευσης (validation set) με αναλογία 90% προς 10%. Η εκπαίδευση πραγματοποιείται στα δεδομένα μόνο του πρώτου συνόλου και στο τέλος κάθε εποχής εξετάζεται η απόδοση του ταξινομητή στα δεδομένα επαλήθευσης. Όταν σημειώνεται η μέγιστη μέχρι στιγμής απόδοση στα δεδομένα επαλήθευσης, εξετάζεται η απόδοση του ταξινομητή και στα δεδομένα ελέγχου (test set). Στο τέλος κάθε επανάληψης του αλγορίθμου 10-fold CV επιστρέφεται σαν μέτρο απόδοσης η ακρίβεια του ταξινομητή στα δεδομένα ελέγχου όταν η ακρίβεια στα δεδομένα επαλήθευσης ήταν η μέγιστη μεταξύ των 25 εποχών. Τελικά παίρνοντας το μέσο όρο αυτών των 10 ακριβειών έχουμε ένα μέτρο εκτίμησης της απόδοσης του ΣΝΔ.

5.6 Συνδυασμός Τεχνικών

Αφού εξετάσαμε τη χρήση λεξικού αλλά και αλγορίθμων μηχανικής μάθησης (συμπεριλαμβανομένων του ΣΝΔ) στο πρόβλημα της ανάλυσης συναισθήματος, πειραματιστήκαμε και με τον συνδυασμό των διαφορετικών αυτών τεχνικών. Ο συνδυασμός των τεχνικών μάθησης καλείται συνολική μάθηση (ensemble learning) και χρησιμοποιείται συχνά για να βελτιώσει τα αποτελέσματα των επιμέρους ταξινομητών.

Η συνολική μάθηση συνήθως πραγματοποιείται με έναν από τους παρακάτω δύο τρόπους:

- Κανόνας της πλειοψηφίας (majority voting)

  Κάθε μοντέλο ταξινόμησης συνεισφέρει κατά μία ισοβαρή ψήφο στην ταξινόμηση ενός άγνωστου προτύπου. Τελικά το πρότυπο ταξινομείται στην κατηγορία με τις περισσότερες ψήφους. Είναι προφανές ότι ο κανόνας αυτός εφαρμόζεται για περιττό αριθμό ταξινομητών ώστε να αποφεύγονται οι ισοπαλίες.

- Κανόνας της σταθμισμένης ψηφοφορίας (weighted voting)

  Κάθε μοντέλο ταξινόμησης συνεισφέρει κατά μία σταθμισμένη ψήφο στην ταξινόμηση ενός άγνωστου προτύπου. Το βάρος της ψήφου μπορεί να καθορίζεται ανάλογα με την απόδοση του μοντέλου σε ένα σύνολο επαλήθευσης (validation set) ή την βεβαιότητά του (πιθανότητα εξόδου) στην περίπτωση των πιθανοτικών ταξινομητών. Τελικά προσθέτουμε όλες τις σταθμισμένες ψήφους και το πρότυπο ταξινομείται στην κατηγορία με το μεγαλύτερο «σκορ». Ο κανόνας αυτός δεν είναι υποχρεωτικό να εφαρμόζεται για περιττό πλήθος ταξινομητών.

Στην εργασία μας πειραματιστήκαμε και με τους δύο κανόνες. Συγκεκριμένα, για τον κανόνα της πλειοψηφίας εξετάσαμε όλους τους συνδυασμούς περιττού πλήθους ταξινομητών από τους Multinomial Naive Bayes, Bernoulli Naive Bayes, Maximum Entropy, LinearSVM και λεξικό (ο ταξινομητής RbfSVM αγνοήθηκε εδώ λόγω της αργής εκπαίδευσής του), συνδυασμοί που φαίνονται στο παρακάτω πίνακα:

Multinomial Naive Bayes	+	+	+	+	+	+					+
Bernoulli Naive Bayes	+	+	+				+	+	+		+
Maximum Entropy	+			+	+		+	+		+	+
LinearSVM		+		+		+	+		+	+	+
Λεξικό			+		+	+		+	+	+	+

Πίνακας 8: Οι συνδυασμοί περιττού πλήθους ταξινομητών που εξετάσαμε για την εφαρμογή του κανόνα της πλειοψηφίας

Για τον κανόνα της σταθμισμένης ψηφοφορίας εξετάσαμε τον συνδυασμό των πιθανοτικών ταξινομητών Multinomial Naive Bayes, Bernoulli Naive Bayes, Maximum Entropy και λεξικού.

Σαν βεβαιότητα της lexicon-based προσέγγισης θεωρήσαμε την κανονικοποίηση ως προς 2 του μέσο όρου των σκορ των concepts που βρέθηκαν στο SenticNet. Έτσι, για παράδειγμα αν ο μέσος όρος των polarities των εννοιών μίας πρότασης που βρέθηκαν στο SenticNet είναι 0.3, η βεβαιότητα του λεξικού είναι 0.65 ως προς τη θετική κλάση, ενώ αν ο μέσος όρος είναι -0.4, η βεβαιότητα του λεξικού είναι 0.7 ως προς την αρνητική κλάση. Ο γενικός κανόνας μετατροπής του μέσου polarity $average\_polarity$ σε βεβαιότητα $confidence\_score$ είναι:

$$confidence\_score = sign(average\_polarity) \cdot \frac{1 + abs(average\_polarity)}{2}$$

με το πρόσημο να δηλώνει την απόφαση της προσέγγισης αυτής για ταξινόμηση στην αντίστοιχη κατηγορία.

Για τον υπολογισμό του μέσου όρου των σκορ των concepts που βρέθηκαν στο SenticNet, έγινε χρήση του τύπου:

$$average\_polarity = \frac{polarity_1^N + \cdots + polarity_m^N}{m}$$

όπου m το πλήθος των concepts της πρότασης που βρέθηκαν στο λεξικό και N παράμετρος προς πειραματισμό. Αντίστοιχα, οι πιθανότητες εξόδου των αλγορίθμων μηχανικής μάθησης υψώνονται στην ίδια δύναμη N, για να προκύψει η απόφαση του σύνθετου μοντέλου ως

$$decision\_of\_fusion\_model = sgn(\frac{\sum_{i=1}^{4} decision_i \cdot probabilty_i}{4})$$

όπου $i = 1 \rightarrow MNB, i = 2 \rightarrow BNB, i = 3 \rightarrow Max\ Ent, i = 4 \rightarrow lexicon - based\ ταξινομητής$ και το πρόσημο της απόφασης δηλώνει ταξινόμηση στην αντίστοιχη κατηγορία.

Η διαίσθηση πίσω από την επιλογή της ύψωσης σε δύναμη N, είναι η εύρεση ενός κατωφλίου, που θα εξαλείφει τα μη σημαντικά (με την έννοια του μικρού polarity) concepts της πρότασης και θα ενισχύει την πιθανότητα εξόδου (confidence_score) του λεξικού με την ελπίδα να διορθώνει τα λάθη που προκύπτουν από τους αλγορίθμους μηχανικής μάθησης. Για τα πειράματά μας δοκιμάσαμε τις ακόλουθες τιμές του N:

$$N = 1,2,3,5,10,20,30,40,50$$

Τα παραπάνω πειράματα έγιναν με χωρισμό των 10375 δειγμάτων σε σύνολα εκπαίδευσης και ελέγχου με αναλογία 90% προς 10% και επανάληψη 10 φορών ώστε να προκύψει ο μέσος όρος accuracy σαν τελικό μέτρο απόδοσης.

Κεφάλαιο 6

Πειραματικά Αποτελέσματα

Στο κεφάλαιο αυτό παρουσιάζονται τα αποτελέσματα από την εκτέλεση των πειραμάτων της προηγούμενης ενότητας. Η μετρική απόδοσης που μας ενδιαφέρει εδώ είναι κυρίως η ακρίβεια ταξινόμησης (accuracy) και για την απόκτησή της χρησιμοποιήθηκαν οι τεχνικές του 10-fold cross-validation και του επαναληπτικού διαχωρισμού του συνόλου δεδομένων σε σύνολα εκπαίδευσης και ελέγχου, όπως εξηγήθηκε προηγουμένως.

6.1 Εφαρμογή Λεξικού

Από τις 10606 αγγλικές προτάσεις του αρχικού dataset, ο lexicon-based ταξινομητής ταξινομεί σωστά τις 6347 προτάσεις, οδηγώντας σε

$$accuracy = \frac{6347}{10606} \cdot 100\% = 59.84\%$$

Αν συνυπολογίσουμε και το γεγονός ότι σε 162 προτάσεις, όπως για παράδειγμα στις προτάσεις “the spiderman rocks”, “a solidly seaworthy chiller”, το λεξικό SenticNet δεν βρίσκει καμία έννοια, το accuracy μεταξύ των προτάσεων που έχουν κάποια έννοια στο λεξικό είναι

$$accuracy = \frac{6347}{10444} \cdot 100\% = 60.77\ \%$$

Από τις 6347 προτάσεις που ταξινομούνται σωστά, οι 4105 ανήκουν στη θετική κλάση και οι 2242 στην αρνητική. Άρα εποπτικά, ανά κατηγορία αλλά και συνολικά, τα αποτελέσματα από τη βασισμένη σε λεξικό προσέγγιση φαίνονται στον παρακάτω πίνακα:

Θετική κατηγορία		Αρνητική κατηγορία		Συνολικά
Precision	Recall	Precision	Recall	Accuracy
57.20%	77.57%	65.38%	42.19%	59.84%

Πίνακας 9: Αποτελέσματα από τη βασισμένη σε λεξικό προσέγγιση

Από τα παραπάνω αποτελέσματα φαίνεται ότι το μοντέλο μας είναι προκατειλημμένο ως προς τη θετική κατηγορία. Η θετική κατηγορία έχει μεγάλο σχετικά recall και αρκετά μικρότερο precision και για την αρνητική κατηγορία ισχύει το αντίστροφο. Αυτό σημαίνει το μοντέλο αποφασίζει συχνότερα από ότι πρέπει την θετική κατηγορία.

Το γεγονός ότι η ακρίβεια δεν είναι πολύ υψηλή οφείλεται στην απλότητα της ανάλυσής μας. Η ενσωμάτωση περισσότερων κανόνων, σαν αυτόν της άρνησης ή της αντίθεσης, για παράδειγμα εξετάζοντας τη χρήση λέξεων μεταβολής έντασης ή εξετάζοντας την εξάρτηση των επιμέρους σχέσεων της κάθε πρότασης ([20],[21]), μπορεί να οδηγήσει σε σημαντική αύξηση της ορθότητας του μοντέλου.

6.2 Εφαρμογή Αλγορίθμων Μηχανικής Μάθησης

6.2.1 Εφαρμογή Naive Bayes, Maximum Entropy, SVM

Χρησιμοποιώντας σαν χαρακτηριστικά την Bag-of-Concepts αναπαράσταση κειμένου, η οποία αναλύθηκε στο προηγούμενο κεφάλαιο, πραγματοποιήθηκαν πειράματα τύπου 10-fold cross-validation. Τα αποτελέσματα που προκύπτουν φαίνονται στον παρακάτω πίνακα

Αλγόριθμοι μηχανικής μάθησης	Θετική κατηγορία		Αρνητική κατηγορία		Συνολικά
	Precision	Recall	Precision	Recall	Accuracy
Multinomial Naive Bayes	72.21%	72.09%	71.74%	71.85%	71.97%
Bernoulli Naive Bayes	72.72%	71.19%	71.39%	72.90%	72.04%
Maximum Entropy	72.09%	71.47%	71.30%	71.89%	71.68%
SVM (linear kernel)	72.17%	71.21%	71.18%	72.12%	71.66%
SVM (rbf kernel)	72.64%	70.54%	70.97%	73.01%	71.77%

Πίνακας 10: Αποτελέσματα από την εφαρμογή αλγορίθμων μηχανικής μάθησης

Η ακρίβεια των αλγορίθμων μηχανικής μάθησης είναι αισθητά μεγαλύτερη από αυτήν της βασισμένης σε λεξικό προσέγγισης, στοιχείο που αποτελεί και το πλεονέκτημα της μεθόδου machine learning. Το μειονέκτημά της, όπως αναφέραμε και νωρίτερα, είναι η ανάγκη εκπαίδευσης του μοντέλου η οποία μπορεί να είναι αρκετά χρονοβόρα, όπως στην περίπτωση του αλγορίθμου SVM με rbf πυρήνα. Ακόμη, η απόδοση των ταξινομητών είναι παρόμοια και για τις δύο κατηγορίες, επομένως δεν είναι προκατειλημμένοι σε αντίθεση με τον lexicon-based ταξινομητή.

Παρατηρούμε ότι η απόδοση των ταξινομητών είναι παρόμοια, στοιχείο που μας οδηγεί στη σκέψη ότι οι ταξινομητές αυτοί ίσως να είναι συσχετισμένοι κάνοντας παρόμοια λάθη. Αυτό είναι πιθανό διότι όλοι οι παραπάνω αλγόριθμοι χρησιμοποιούν τα ίδια χαρακτηριστικά. Πράγματι, αν εκπαιδεύσουμε τους 3 πρώτους ταξινομητές χρησιμοποιώντας ένα τυχαίο υποσύνολο 9338 προτάσεων, και ελέγξουμε τις εξόδους τους πάνω στις εναπομείναντες 1037 προτάσεις, διαπιστώνουμε ότι κατά μέσο όρο ο Multinomial Naive Bayes διαφωνεί με τον Bernoulli Naive Bayes σε μόλις 21 περιπτώσεις, ο Bernoulli με τον Maximum Entropy σε 102 περιπτώσεις και ο Multinomial με τον Maximum Entropy σε 104 περιπτώσεις.

Για να τονίσουμε την εξάρτηση του αλγορίθμου από τα χαρακτηριστικά και άρα τη σημασία της προσεκτικής και σωστής εξαγωγής χαρακτηριστικών, επαναλάβαμε την υλοποίηση του Multinomial Naive Bayes (επιλέξαμε αυτόν τον ταξινομητή λόγω της απλότητας, της ταχύτητας εκπαίδευσης και της απόδοσής του) με χαρακτηριστικά τα συχνότερα χρησιμοποιούμενα unigrams ή/και bigrams (με ή χωρίς stemming κάνοντας χρήση του αλγορίθμου Porter) και αφού μετατρέψουμε τα κεφαλαία σε πεζά και απομακρύνουμε τα σημεία στίξης. Τα αποτελέσματα ήταν τα ακόλουθα:

Χαρακτηριστικά	Accuracy
4000 top unigrams	74.63%
4000 top unigrams (stems)	74.88%
3000 top bigrams	68.64%
3000 top bigrams (stems)	69.56%
4000 top unigrams + 3000 top bigrams	75.80%
4000 top unigrams+ 3000 top bigrams (stems)	76.48%

Πίνακας 11: Ποσοστά ορθότητας από την εφαρμογή του Multinomial Naive Bayes

Από τον πίνακα 11 παρατηρούμε ότι αυτά τα corpus-based χαρακτηριστικά δίνουν καλύτερα αποτελέσματα σε σχέση με την ανάλυση σε επίπεδο εννοιών, τουλάχιστον όσον αφορά τον αλγόριθμο Multinomial Naive Bayes. Πιο συγκεκριμένα, βλέπουμε ότι η χρήση stemming (με τον Porter stemmer) με τον συνδυασμό unigrams με bigrams δίνει την καλύτερη απόδοση, πετυχαίνοντας accuracy ίσο με 76.48%.

6.2.2 Εφαρμογή MLP

Στο πείραμα αυτό υλοποιήσαμε τις διάφορες αρχιτεκτονικές 3-layer MLP του πίνακα 7, όπου το νευρωνικό δέχεται στην είσοδο την αναπαράσταση του κειμένου εισόδου σε επίπεδο εννοιών κατά τα γνωστά. Τα αποτελέσματα που προκύπτουν με τη μέθοδο 10-fold cross-validation και χωρισμό των δεδομένων εκπαίδευσης σε training και validation sets με αναλογία 90% προς 10% φαίνονται στον παρακάτω πίνακα:

	Διάσταση κρυφού επιπέδου	Accuracy
Αρχιτεκτονική 1	5	72.45%
Αρχιτεκτονική 2	10	70.70%

Αρχιτεκτονική 3	15	74.68%
Αρχιτεκτονική 4	20	69.20%
Αρχιτεκτονική 5	25	70.45%
Αρχιτεκτονική 6	30	69.23%
Αρχιτεκτονική 7	50	67.39%
Αρχιτεκτονική 8	100	68.69%
Αρχιτεκτονική 9	150	72.38%
Αρχιτεκτονική 10	500	67.21%

Πίνακας 12: Ποσοστά ορθότητας από την εφαρμογή διαφόρων αρχιτεκτονικών 3-layer MLP

Παρατηρούμε ότι την καλύτερη απόδοση, με ποσοστό accuracy 74.68% σημειώνει η αρχιτεκτονική 3 με τους 15 κρυφούς νευρώνες και την χειρότερη, με ποσοστό accuracy 67.21% η αρχιτεκτονική 10 με τους 500 κρυφούς νευρώνες. Επίσης, αυξάνοντας τη διάσταση του κυφού επιπέδου, αυξάνουμε την πολυπλοκότητα του νευρωνικού και άρα και τον χρόνο εκπαίδευσής του, χωρίς να κερδίζουμε κάτι ουσιαστικά σε απόδοση.

Το μέσο ποσοστό accuracy όλων των αρχιτεκτονικών είναι 70.24%, μικρότερο από το ποσοστό του 71.97% που πετυχαίνει ο απλούστερος ταξινομητής Multinomial Naive Bayes. Έτσι, προτιμάται το μοντέλο του Naive Bayes που επιδεικνύει ενδιαφέροντα αποτελέσματα και συγχρόνως εκπαιδεύεται πολύ γρηγορότερα.

6.3 Εφαρμογή ΣΝΔ

Στο πείραμα αυτό υλοποιήσαμε το ΣΝΔ του σχήματος 18, όπου το νευρωνικό δέχεται στην είσοδο την «εικόνα» του κειμένου εισόδου χρησιμοποιώντας τα pre-trained word2vec διανύσματα. Τα διανύσματα αυτά ενσωματώνονται στις προσαρμόσιμες παραμέτρους του μοντέλου και συνεπώς εξειδικεύονται στο πρόβλημα του sentiment analysis. Τα αποτελέσματα που προκύπτουν με τη μέθοδο 10-fold cross-validation, με μαζική μάθηση σε παρτίδες μεγέθους 50 και χωρισμό των δεδομένων εκπαίδευσης σε training και validation sets με αναλογία 90% προς 10% φαίνονται στον παρακάτω πίνακα:

Μοντέλο	Accuracy
ΣΝΔ με μη σταθερό επίπεδο εισόδου	81.86%

Πίνακας 13: Ποσοστό ορθότητας από την εφαρμογή ΣΝΔ

Παρατηρούμε ότι η ακρίβεια ταξινόμησης που επιτυγχάνεται με το ΣΝΔ είναι αρκετά μεγαλύτερη από αυτή που επιτυγχάνεται με τους προηγούμενους αλγορίθμους της (παραδοσιακής) μηχανικής μάθησης. Η χρήση πολλών χαρτών χαρακτηριστικών μεταβλητού μεγέθους, επιτρέπει την κατασκευή πολύτιμων χαρακτηριστικών που συνεισφέρουν σημαντικά στο έργο της ανίχνευσης πολικότητας από κείμενο. Το μειονέκτημα βέβαια της μεθόδου αυτής είναι η ενσωμάτωση πολλών παραμέτρων προς εκπαίδευση, κάτι που επιβραδύνει σημαντικά την εκπαίδευση του δικτύου.

Όσον αφορά τα διανύσματα λέξεων που μαθεύτηκαν κατά την εκπαίδευση, σημειώνουμε τον προσανατολισμό τους στην σύλληψη σχέσεων συναισθηματικού περιεχομένου μεταξύ των λέξεων. Για παράδειγμα, ενώ αρχικά, πριν την εκπαίδευση του ΣΝΔ, η λέξη good ήταν η πιο κοντινή λέξη στη λέξη bad κατά την cosine απόσταση, μετά την εκπαίδευση το μοντέλο έμαθε τη συσχέτιση της λέξης bad με τη λέξη lousy. Επίσης, διανύσματα λέξεων που ήταν τυχαία αρχικοποιημένα λόγω του ότι δεν ήταν στο αρχικό σύνολο των pre-trained διανυσμάτων, μετά την εκπαίδευση αποκτούν χρήσιμες αναπαραστάσεις. Για παράδειγμα, το θαυμαστικό συσχετίζεται με διαχυτικές εκφράσεις όπως τις λέξεις beautiful και terrific, και το κόμμα με συνδέσμους όπως οι λέξεις and και but.

6.4 Συνδυασμός Τεχνικών

Τα αποτελέσματα (ποσοστά accuracy) που προκύπτουν από τους διάφορους συνδυασμούς ταξινομητών του πίνακα 8, εφαρμόζοντας τον κανόνα της πλειοψηφίας, φαίνονται στον ακόλουθο πίνακα:

Συνδυασμοί	Multinomial Naive Bayes	Bernoulli Naive Bayes	Max Ent	LinearSVM	Λεξικό	Συνδυασμός
1	71.98%	71.83%	72.13%	–	–	71.95%
2	71.59%	71.67%	–	69.39%	–	71.70%
3	71.71%	71.62%	–	–	61.84%	71.78%
4	70.82%	–	71.62%	69.26%	–	71.45%
5	72.20%	–	72.16%	–	61.58%	73.05%
6	72.20%	–	–	69.78%	62.30%	72.30%
7	–	71.95%	71.84%	69.46%	–	72.08%
8	–	71.68%	71.72%	–	60.77%	72.73%
9	–	72.31%	–	69.85%	61.93%	72.12%
10	–	–	71.88%	69.54%	61.75%	71.18%
11	71.53%	71.51%	71.49%	69.53%	60.55%	72.28%

Πίνακας 14: Ποσοστά ορθότητας με εφαρμογή του κανόνα της πλειοψηφίας

Από τον πίνακα 12 παρατηρούμε ότι οι αλγόριθμοι μηχανικής μάθησης δίνουν συσχετισμένους ταξινομητές που κάνουν παρόμοια λάθη. Έτσι, το συνδυασμένο μοντέλο που βασίζεται σε περιττό συνδυασμό των αλγορίθμων αυτών καταλήγει να έχει την ίδια ή λίγο χειρότερη απόδοση από την απόδοση του καλύτερου μεμονωμένου ταξινομητή. Αν όμως συμπεριλάβουμε στην εξέτασή μας και τον βασισμένο σε λεξικό ταξινομητή, η απόδοση αυξάνεται έστω και λίγο. Η μέγιστη βελτίωση σημειώνεται από τον συνδυασμό των Bernoulli Naive Bayes και Maximum Entropy με τον βασισμένο σε λεξικό ταξινομητή, ενώ σημαντική είναι η βελτίωση της απόδοσης που πετυχαίνουμε όταν συνδυάζουμε και τους 5 ταξινομητές.

Ακολούθως εστιάσαμε στους πιθανοτικούς ταξινομητές για την εφαρμογή του κανόνα της σταθμισμένης ψηφοφορίας. Τα αποτελέσματα που προκύπτουν για τις διάφορες τιμές του N φαίνονται στον παρακάτω πίνακα:

Τιμές του N	Αλγόριθμοι μηχανικής μάθησης				
	MNB	BNB	Max Ent	Λεξικό	Συνδυασμός
$N = 1$	71.97%	71.87%	72.08%	61.43%	72.80%
$N = 2$	71.31%	71.30%	71.56%	59.68%	72.00%
$N = 3$	72.20%	72.25%	72.09%	60.63%	72.83%
$N = 5$	72.82%	72.62%	72.16%	59.90%	70.97%
$N = 10$	71.62%	71.83%	71.63%	59.17%	66.78%
$N = 20$	71.27%	71.49%	71.10%	58.83%	63.48%
$N = 30$	71.72%	71.87%	71.73%	59.41%	62.65%
$N = 40$	72.32%	72.28%	71.98%	58.96%	61.69%
$N = 50$	71.41%	71.46%	71.74%	59.06%	61.46%

Πίνακας 15: Ποσοστά ορθότητας με εφαρμογή του κανόνα της σταθμισμένης ψηφοφορίας

Αυξανομένης της τιμής του N, η ακρίβεια του σύνθετου μοντέλου δεν αυξάνεται, αντιθέτως μειώνεται. Συνεπώς, δεν προκύπτει κανένα όφελος από την ύψωση στην N-οστή δύναμη των confidence_scores των αλγορίθμων μηχανικής μάθησης και των polarities από τον βασισμένο σε λεξικό ταξινομητή. Μάλιστα η μέγιστη βελτίωση που σημειώνεται για N=1, είναι ανάλογη με τη βελτίωση που επιτεύχθηκε προηγουμένως από τον συνδυασμό των Bernoulli Naive Bayes, Maximum Entropy και του βασισμένο σε λεξικό ταξινομητή εφαρμόζοντας τον κανόνα της πλειοψηφίας.

6.5 Σύνοψη της Εργασίας

Στην παρούσα διπλωματική εργασία ασχοληθήκαμε με το πρόβλημα της ανάλυσης συναισθήματος από κείμενο και πιο συγκεκριμένα με αυτό της αυτόματης ανάλυσης άποψης και του χαρακτηρισμό της ως θετική ή αρνητική με διάφορες μεθόδους. Σαν dataset για την εκτέλεση των διάφορων πειραμάτων χρησιμοποιήσαμε το σύνολο κριτικών ταινιών σε έκταση πρότασης που συστήθηκε από τους Pang και Lee το 2005 στην εργασία [4].

Αρχικά δοκιμάσαμε τη βασισμένη σε λεξικό προσέγγιση για την ανάλυση συναισθήματος. Σαν λεξικό χρησιμοποιήθηκε το SenticNet, ένα λεξικό 30000 εννοιών της αγγλικής γλώσσας και συνεπώς για την αξιοποίησή του διεξήγαμε ανάλυση του κειμένου εισόδου (πρότασης ουσιαστικά) σε επίπεδο εννοιών (concept-level sentiment analysis). Τα αποτελέσματα που προέκυψαν από την προσέγγιση αυτή δεν ήταν ικανοποιητικά. Συγκεκριμένα, η ακρίβεια ταξινόμησης που προέκυψε ήταν ίση με 59.84% και η χαμηλή τιμή της οφείλεται στην απλότητα της ανάλυσής μας.

Στη συνέχεια εξετάσαμε την χρήση διάφορων αλγορίθμων μηχανικής μάθησης: του Naive Bayes (Multinomial και Bernoulli), του Maximum Entropy, των μηχανών διανυσμάτων υποστήριξης (SVM) με γραμμικό και rbf πυρήνα και ενός 3-layer MLP με μεταβλητό αριθμό νευρώνων στο κρυφό επίπεδο. Σαν χαρακτηριστικά χρησιμοποιήσαμε την Bag-of-Concepts αναπαράσταση κειμένου και σαν concepts χρησιμοποιήθηκε ένα υποσύνολο των καταχωρήσεων του SenticNet. Προέκυψαν με αυτό τον τρόπο διανύσματα χαρακτηριστικών μεγέθους 3087. Τα αποτελέσματα που προέκυψαν από πειράματα τύπου 10-fold cross-validation έδειξαν ότι οι ταξινομητές είχαν παρόμοια απόδοση μεταξύ τους, απόδοση εμφανώς ανώτερη από τον βασισμένο σε λεξικό ταξινομητή, με τα καλύτερα αποτελέσματα να δίνει ο Bernoulli Naive Bayes με ποσοστό accuracy 72.04%. Όσον αφορά το 3-layer MLP, η βέλτιστη απόδοση σημειώθηκε με τους 15 κρυφούς νευρώνες, με accuracy 74.68% και αυξανομένης της πολυπλοκότητας του κρυφού επιπέδου, το accuracy δεν ανέβαινε πάνω από το ποσοστό αυτό. Ο συνδυασμός απλότητας, καλής απόδοσης και μικρού χρόνου εκπαίδευσης του Multinomial Naive Bayes μας οδήγησε στον πειραματισμό του ταξινομητή αυτού και με άλλου είδους χαρακτηριστικά για σύγκριση. Σαν χαρακτηριστικά χρησιμοποιήσαμε τα 4000 και 3000 πιο συνηθισμένα unigrams και bigrams αντίστοιχα, με stemming και χωρίς, από κοινού και μεμονωμένα και τελικά καταλήξαμε στο συμπέρασμα ότι ο συνδυασμός unigrams και bigrams με την τεχνική stemming είναι το πιο πετυχημένο είδος χαρακτηριστικών με ποσοστό ορθότητας 76.48%.

Έπειτα δοκιμάσαμε και την εφαρμογή ενός συνελικτικού νευρωνικού δικτύου της απλής σχετικά αρχιτεκτονικής του σχήματος 18 με την διαφορά ότι χρησιμοποιείται ένα κανάλι για την είσοδο, το οποίο προσαρμόζεται και αυτό κατά την εκπαίδευση. Χρησιμοποιήθηκαν παράθυρα μεγέθους (h) 3,4, 5 με 100 χάρτες χαρακτηριστικών το καθένα και max-pooling στο επίπεδο συγκέντρωσης. Το μέσο ποσοστό accuracy που επιστράφηκε μετά από πειραματισμό τύπου 10-fold cross-validation είναι 81.68% φανερώνοντας γιατί τα μοντέλα deep learning χρησιμοποιούνται όλο και περισσότερο σε προβλήματα επεξεργασίας φυσικής γλώσσας πέρα από τον τομέα της όρασης υπολογιστών. Ακόμη, η προσαρμογή (fine-tuning) των word vectors επιτρέπει την εξειδίκευση τους στον τομέα του συναισθήματος.

Σαν επιστέγασμα της εργασίας μας εξετάσαμε τον συνδυασμό τεχνικών για την πραγματοποίηση συνολικής μάθησης (ensemble learning). Πειραματιστήκαμε και με τους δύο πιθανούς τρόπους εφαρμογής της, τον κανόνα της πλειοψηφίας και τον κανόνα της σταθμισμένης ψηφοφορίας. Για τον κανόνα της πλειοψηφίας πειραματιστήκαμε με όλους τους πιθανούς συνδυασμούς περιττού πλήθους ταξινομητών μεταξύ των Multinomial Naive Bayes, Bernoulli Naive Bayes, Maximum Entropy, SVM με γραμμικό πυρήνα και βασισμένου σε λεξικό ταξινομητή. Η μεγαλύτερη βελτίωση απόδοσης σημειώθηκε όταν συνδυάσαμε τους Bernoulli Naive Bayes, Maximum Entropy με τον βασισμένο σε λεξικό ταξινομητή και το accuracy του σύνθετου μοντέλου έφτασε στο 72.73%. Για τον κανόνα της σταθμισμένης ψηφοφορίας πειραματιστήκαμε με τους πιθανοτικούς ταξινομητές Multinomial Naive Bayes, Bernoulli Naive Bayes, Maximum Entropy και τον βασισμένο σε λεξικό ταξινομητή. Υψώσαμε τις πιθανότητες εξόδου των ταξινομητών αυτών σε διάφορες δυνάμεις για την εξάλειψη τυχόν πολώσεων που ενδέχεται να υπάρχει στον ταξινομητή που βασίζεται στο SenticNet. Η μεγαλύτερη βελτίωση απόδοσης σημειώθηκε όταν υψώσαμε στην $1^{η}$ δύναμη και το accuracy του σύνθετου μοντέλου έφτασε στο 72.80%. Υψώνοντας σε μεγαλύτερες δυνάμεις μειώνεται το accuracy του σύνθετου μοντέλου γεγονός που σημαίνει ότι και τα αμελητέα concepts, σύμφωνα με το λεξικό SenticNet, θα πρέπει να λαμβάνονται υπόψη και όχι να εξαλείφεται η επίδρασή τους. Τα παραπάνω πειράματα πραγματοποιήθηκαν με τον (τυχαίο) χωρισμό του συνόλου των δεδομένων σε train και test σύνολα με αναλογία 90% προς 10% και επανάληψη 10 φορών για τη μείωση της διακύμανσης.

Συμπεραίνουμε ότι για την ανάλυση άποψης σε επίπεδο κειμένου, η μέθοδος Naive Bayes με τις δύο παραλλαγές της, έδωσε πολύ ενδιαφέροντα αποτελέσματα παρά την απλότητά της. Από την άλλη, η χρήση τεχνητών νευρωνικών δικτύων με τρία επίπεδα δεν απέφερε εξίσου καλά αποτελέσματα ταξινόμησης, τουλάχιστον με αυτά που θα αναμέναμε λόγω της πολυπλοκότητας τους. Τα συνελικτικά νευρωνικά δίκτυα κατόρθωσαν να αυξήσουν σημαντικά την ακρίβεια ταξινόμησης με αντίτιμο όμως την αργή εκπαίδευσή τους. Τέλος, η προσπάθεια συνδυασμού των επιμέρους τεχνικών για την ενίσχυση της απόδοσης βελτίωσε κατά μικρό ποσοστό την ακρίβεια σε σύγκριση με τους επιμέρους ταξινομητές και το μικρό αυτό ποσοστό της βελτίωσης οφείλεται στο ότι οι ταξινομητές αυτοί «έμαθαν» παρόμοια πράγματα και έκαναν παρόμοια λάθη.

Βιβλιογραφία

[1] C.M. Bishop (2006). "Pattern Recognition and Machine Learning". Springer.

[2] Peter D Turney (2002). "Thumbs up or thumbs down? Semantic Orientation Applied to Unsupervised Classification of Reviews". In *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics. pp. 417-424.

[3] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan (2002). "Thumbs up? Sentiment classification using Machine learning Techniques". In *Proceedings of the conference on empirical methods in natural language processing*. Association for Computational Linguistics. pp. 79–86.

[4] Bo Pang and Lillian Lee (2005). "Seeing Stars: Exploiting class relationships for sentiment categorization with respect to rating scales". In *Proceedings of the 43rd annual meeting on association for computational linguistics*. pp. 115-124

[5] S.Rustamov, E.Mustafayev, M.A. Clements (2013). "Sentiment analysis using Neuro-Fuzzy and Hidden Markov models of text". In *Proceedings of IEEE 2013*. pp.1-6.

[6] Tetsuji Nakagawa, Kentaro Inui, and Sadao Kurohashi (2010). "Dependency tree-based sentiment classification using CRFs with hidden variables". In *Proceedings of the 2010 annual conference of the north american chapter of the association for computational linguistics*. pp. 786-794

[7] Moshe Koppel and Jonathan Schler (2005). "The Importance of Neutral Examples for Learning Sentiment". In *workshop on the analysis of informal and formal information exchange during negotiations.*

[8] Simon Haykin (2009). "Neural Networks and Learning Machines". 3rd edition.

[9] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean (2013). "Efficient Estimation of Word Representations in Vector Space". *In Proceedings of Workshop at ICLR.*

[10] Yoon Kim (2014). "Convolutional Neural Networks for Sentence Classification". In *Proceedings of the 2014 conference on empirical methods in natural language processing.* Association for Computational Linguistics. pp. 1746-1751.

[11] Peng Wang, Jiaming Xu, Bo Xu, Cheng-Lin Liu, Heng Zhang, Fangyuan Wang and Hongwei Hao (2015). "Semantic Clustering and Convolutional Neural Network for Short Text Categorization." In *Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing.* ACL. pp. 352–357.

[12] Rie Johnson, Tong Zhang (2015). "Effective Use of Word Order for Text Categorization with Convolutional Neural Networks". In *NAACL.*

[13] Ye Zhang and Byron C. Wallace (2016). “A Sensitivity Analysis of (and Practitioners’ Guide to) Convolutional Neural Networks for Sentence Classification”. arXiv:1510.03820v4.

[14] Cicero Nogueira dos Santos, Bianca Zadrozny (2011). “Learning Character-level Representations for Part-of-Speech Tagging”. In *Proceedings of the 31st International Conference on Machine Learning*. pp. 1818–1826.

[15] Xiang Zhang, Zunbo Zhao and Yann LeCun (2015). “Character-level Convolutional Networks for Text Classification”. In *advanced in neural information processing systems 28.*

[16] Xiang Zhang and Yann LeCun (2016). “Text Understanding from Scratch”. arXiv:1502.01710v5.

[17] Taboada et al. (2011). “Lexicon-Based Methods for Sentiment Analysis”. ACL. pp. 267-307.

[18] Kennedy and Inkpen (2006). “Sentiment classification of movie and product reviews using contextual valence shifters”. Computational Intelligence. pp. 110-125.

[19] Polanyi and Zaenen (2006). “Contextual valence shifters”. In *Janyce Wiebe, editor, Computing Attitude and Affect in Text: Theory and Applications*. Springer. pp. 1-10.

[20] Poria et al. (2014). “Sentic patterns: Dependency-based rules for concept-level sentiment analysis”. In *knowledge-based systems 69*. pp. 45-63.

[21] Poria et al. (2015). “Sentiment Data Flow Analysis by Means of Dynamic Linguistic Patterns”. In *IEEE Computational Intelligence Magazine 10*. pp. 26-36.

[22] Balahur et al. (2009). “Opinion Mining on Newspaper Quotations”. In *Web Intelligence and Intelligent Agent Technologies.* pp. 523-536.

[23] Ngoc and Yoo (2014). “The Lexicon-based sentiment analysis for fan page ranking in Facebook”. In *The International Conference on Information Networking 2014. pp. 444-448.*

[24] Kolchyna et al. (2015). “Twitter Sentiment Analysis: Lexicon Method, Machine Learning Method and Their Combination”. arXiv:1507.00955v3.