ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΕΦΑΡΜΟΣΜΕΝΩΝ ΜΑΘΗΜΑΤΙΚΩΝ ΚΑΙ ΦΥΣΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΔΙΑΤΜΗΜΑΤΙΚΟ ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

«ΜΑΘΗΜΑΤΙΚΗ ΠΡΟΤΥΠΟΠΟΙΗΣΗ σε ΣΥΓΧΡΟΝΕΣ ΤΕΧΝΟΛΟΓΙΕΣ και την ΟΙΚΟΝΟΜΙΑ»

Ανάλυση συναισθήματος στο Twitter με χαρακτηριστικά διανυσμάτων λέξεων

ΧΑΡΑΛΑΜΠΟΥΣ ΧΑΡΑΛΑΜΠΟΣ

ΑΡΙΘΜΟΣ ΜΗΤΡΩΟΥ: 09313044

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: Καθηγητής Σταφυλοπάτης Ανδρέας-Γεώργιος

ΑΘΗΝΑ, Τετάρτη 11/10/2017

Contents

Εισαγωγή

Η μηχανική μάθηση είναι υποπεδίο της επιστήμης των υπολογιστών που αναπτύχθηκε από τη μελέτη της αναγνώρισης προτύπων και της υπολογιστικής θεωρίας μάθησης στην τεχνητή νοημοσύνη. Το 1959, ο Arthur Samuel ορίζει τη μηχανική μάθηση ως "Πεδίο μελέτης που δίνει στους υπολογιστές την ικανότητα να μαθαίνουν, χωρίς να έχουν ρητά προγραμματιστεί". Η μηχανική μάθηση διερευνά τη μελέτη και την κατασκευή αλγορίθμων που μπορούν να μαθαίνουν από τα δεδομένα και να κάνουν προβλέψεις σχετικά με αυτά. Τέτοιοι αλγόριθμοι λειτουργούν κατασκευάζοντας μοντέλα από πειραματικά δεδομένα, προκειμένου να κάνουν προβλέψεις βασιζόμενες στα δεδομένα ή να εξάγουν αποφάσεις που εκφράζονται ως το αποτέλεσμα.
Η μηχανική μάθηση εφαρμόζεται σε μια σειρά από υπολογιστικές εργασίες, όπου τόσο ο σχεδιασμός όσο και ο ρητός προγραμματισμός των αλγορίθμων είναι ανέφικτος. Παραδείγματα εφαρμογών αποτελούν τα φίλτρα spam (spam filtering), η οπτική αναγνώριση χαρακτήρων (OCR), οι μηχανές αναζήτησης και η υπολογιστική όραση.
Στο πεδίο της ανάλυσης δεδομένων, η μηχανική μάθηση είναι μια μέθοδος που χρησιμοποιείται για την επινόηση πολύπλοκων μοντέλων και αλγορίθμων που οδηγούν στην πρόβλεψη. Τα αναλυτικά μοντέλα επιτρέπουν στους ερευνητές, τους επιστήμονες δεδομένων, τους μηχανικούς και τους αναλυτές να παράγουν αξιόπιστες αποφάσεις και αποτελέσματα και να αναδείξουν αλληλοσυσχετίσεις μέσω της μάθησης από ιστορικές σχέσεις και τάσεις στα δεδομένα.

Οι εργασίες μηχανικής μάθησης συνήθως ταξινομούνται σε τρεις μεγάλες κατηγορίες, ανάλογα με τη φύση του εκπαιδευτικού «σήματος» ή την «ανατροφοδότηση» που είναι διαθέσιμα σε ένα σύστημα εκμάθησης. Αυτές είναι:

- Επιτηρούμενη μάθηση (ή αλλιώς επιβλεπόμενη μάθηση ή μάθηση με επίβλεψη δηλ. supervised learning): Το υπολογιστικό πρόγραμμα δέχεται τις παραδειγματικές εισόδους καθώς και τα επιθυμητά αποτελέσματα από έναν «δάσκαλο», και ο στόχος είναι να μάθει έναν γενικό κανόνα προκειμένου να αντιστοιχίσει τις εισόδους με τα αποτελέσματα.

- Μη επιτηρούμενη μάθηση (ή μάθηση χωρίς επίβλεψη δηλ. unsupervised learning): Χωρίς να παρέχεται κάποια εμπειρία στον αλγόριθμο μάθησης, πρέπει να βρει την δομή των δεδομένων εισόδου. Η μη επιτηρούμενη μάθηση μπορεί να είναι αυτοσκοπός (ανακαλύπτοντας κρυμμένα μοτίβα σε δεδομένα) ή μέσο για ένα τέλος (χαρακτηριστικό της μάθησης).

- Ενισχυτική μάθηση (reinforcement learning): Ένα πρόγραμμα υπολογιστή αλληλεπιδρά με ένα δυναμικό περιβάλλον στο οποίο πρέπει να επιτευχθεί ένας συγκεκριμένος στόχος (όπως η οδήγηση ενός οχήματος), χωρίς κάποιος δάσκαλος να

του λέει ρητά αν έχει φτάσει κοντά στο στόχο του. Ένα άλλο παράδειγμα είναι να μάθει να παίζει ένα παιχνίδι εναντίον κάποιου αντιπάλου.

Μία μηχανή διανυσμάτων υποστήριξης (SVM), όπου τα δεδομένα ταξινομούνται σε δύο κλάσεις, που χωρίζονται από ένα γραμμικό σύνορο. Εδώ, έχει μάθει να διακρίνει τους μαύρους από τους άσπρους κύκλους.

Μια άλλη κατηγοριοποίηση των προβλημάτων μηχανικής μάθησης προκύπτει όταν κάποιος θεωρήσει το επιθυμητό αποτέλεσμα του συστήματος μηχανικής μάθησης:

- Στην ταξινόμηση, τα δεδομένα εισόδου χωρίζονται σε δύο ή περισσότερες κλάσεις, και η μηχανή πρέπει να κατασκευάσει ένα μοντέλο, το οποίο θα αντιστοιχίζει τα δεδομένα σε μία ή περισσότερες (multi-label ταξινόμηση) κλάσεις. Αυτό συνήθως εμπίπτει στην επιτηρούμενη μάθηση. Τα φίλτρα spam είναι ένα παράδειγμα ταξινόμησης, όπου οι είσοδοι είναι τα emails ή άλλα μηνύματα και οι κλάσεις είναι "spam" και "όχι spam".

- Στην παλινδρόμηση, επίσης πρόβλημα επιτηρούμενης μάθησης, τα αποτελέσματα είναι συνεχή και όχι διακριτά.

- Στην συσταδοποίηση, ένα σύνολο εισόδων πρόκειται να χωριστεί σε ομάδες. Σε αντίθεση με την ταξινόμηση, οι ομάδες δεν είναι γνωστές εκ των προτέρων, καθιστώντας αυτόν τον διαχωρισμό τυπική εργασία μη επιτηρούμενης μάθησης.

- Στην εκτίμηση πυκνότητας βρίσκει την κατανομή των δεδομένων εισόδου σε κάποιο χώρο.

- Σε προβλήματα μείωσης διαστασιμότητας (dimensionality reduction), τα δεδομένα απλοποιούνται και αντιστοιχίζονται σε ένα χώρο λιγότερων διαστάσεων. Το στατιστικό μοντέλο θεμάτων (Topic modeling) είναι ένα σχετικό πρόβλημα, όπου η μηχανή καλείται να βρει έγγραφα που καλύπτουν παρόμοια θέματα από ένα σύνολο εγγράφων γραμμένων σε φυσική γλώσσα.

Προσεγγίσεις

Εκμάθηση με δέντρο απόφασης

Η εκμάθηση με δέντρο απόφασης χρησιμοποιεί ένα δέντρο απόφασης ως προγνωστικό μοντέλο, το οποίο αντιστοιχίζει παρατηρήσεις σχετικά με ένα στοιχείο σε συμπεράσματα σχετικά με την τιμή στόχο του αντικειμένου.

Εκμάθηση με κανόνες συσχέτισης

Η εκμάθηση με κανόνες συσχέτισης είναι μια μέθοδος ανακάλυψης ενδιαφερουσών σχέσεων μεταξύ των μεταβλητών σε μεγάλες βάσεις δεδομένων.

Τεχνητά νευρωνικά δίκτυα

Ένας αλγόριθμος εκμάθησης τεχνητού νευρωνικού δικτύου, που συνήθως ονομάζεται "νευρωνικό δίκτυο" (NN), είναι ένας αλγόριθμος μάθησης, που εμπνέεται από τη δομή και τις λειτουργικές πτυχές των βιολογικών νευρωνικών δικτύων. Η δομή των υπολογισμών βασίζεται σε μια ομάδα εσωτερικά διασυνδεδεμένων τεχνητών νευρώνων, οι οποίοι επεξεργάζονται την πληροφορία και εκτελούν υπολογισμούς επικοινωνώντας μεταξύ τους. Τα σύγχρονα νευρωνικά δίκτυα είναι εργαλεία μη γραμμικής στατιστικής μοντελοποίησης δεδομένων. Συνήθως χρησιμοποιούνται για τη μοντελοποίηση σύνθετων σχέσεων μεταξύ δεδομένων εισόδου και εξόδου, για την ανακάλυψη προτύπων στα δεδομένα, ή για τον εντοπισμό στατιστικής δομής σε μία άγνωστη κοινή κατανομή πιθανότητας μεταξύ των παρατηρούμενων μεταβλητών.

Βαθιά Μάθηση

Η πτώση των τιμών του υλικού των τελευταίων ετών καθώς και η ανάπτυξη των GPU για προσωπική χρήση, οδήγησε στην ανάπτυξη της ιδέας της Βαθιάς Μάθησης. Αυτή η προσέγγιση προσπαθεί να μοντελοποιήσει τον τρόπο που ο ανθρώπινος εγκέφαλος επεξεργάζεται το φως και τον ήχο και τα μετατρέπει σε όραση και ακοή. Ορισμένες επιτυχείς εφαρμογές της Βαθιάς μάθησης είναι η μηχανική όραση και η αναγνώριση ομιλίας.

Επαγωγικός λογικός προγραμματισμός

Ο επαγωγικός λογικός προγραμματισμός (ILP) είναι μια προσέγγιση που διέπει την μάθηση και χρησιμοποιεί λογικό προγραμματισμό ως τρόπο παρουσίασης των παραδειγμάτων εισόδου, του γνωστικού υποβάθρου και των υποθέσεων. Δεδομένης μιας κωδικοποίησης του γνωστικού υποβάθρου και ενός συνόλου παραδειγμάτων που παρουσιάζονται σαν λογική βάση γεγονότων, το σύστημα ΕΛΠ παράγει το υποτιθέμενο λογικό πρόγραμμα που περιέχει όλα τα θετικά και κανένα αρνητικό παράδειγμα. Ο επαγωγικός προγραμματισμός είναι ένας σχετικός τομέας που λαμβάνει υπόψιν κάθε είδος προγραμματιστικής γλώσσας για την αναπαράσταση υποθέσεων (και όχι μόνο λογικό προγραμματισμό), όπως τα συναρτησιακά προγράμματα.

Μηχανές διανυσμάτων υποστήριξης

Οι μηχανές διανυσμάτων υποστήριξης είναι ένα σύνολο μεθόδων επιτηρούμενης μάθησης που χρησιμοποιούνται για την ταξινόμηση και την παλινδρόμηση. Σ' αυτήν την περίπτωση δίνεται ένα σύνολο παραδειγμάτων εκπαίδευσης και κάθε φορά δηλώνεται σε ποια από τις δύο κατηγορίες ανήκει το παράδειγμα. Μία μηχανή διανυσμάτων υποστήριξης κατασκευάζει ένα μοντέλο που προβλέπει αν το νέο παράδειγμα εμπίπτει στην μία κατηγορία ή την άλλη.

Ομαδοποίηση

Η ομαδοποίηση είναι η διαδικασία κατά την οποία ένα σύνολο παρατηρήσεων χωρίζεται σε υποσύνολα έτσι ώστε οι παρατηρήσεις που ανήκουν στην ίδια ομάδα (cluster) είναι όμοιες, σύμφωνα με κάποιο ή κάποια προκαθορισμένα κριτήρια, ενώ οι παρατηρήσεις που προέρχονται από διαφορετικά υποσύνολα είναι ανόμοιες. Διαφορετικές τεχνικές κατηγοριοποίησης οδηγούν σε διαφορετικές υποθέσεις σχετικά με τη δομή των δεδομένων, οι οποίες συχνά καθορίζονται από κάποιο μέτρο ομοιότητας και αξιολογούνται για παράδειγμα ως προς την εσωτερική συνοχή (ομοιότητα μεταξύ των μελών του ίδιου cluster) και το διαχωρισμό ανάμεσα σε διαφορετικές ομάδες. Άλλες μέθοδοι βασίζονται στην εκτιμώμενη πυκνότητα και την συνεκτικότητα των γραφημάτων. Η ομαδοποίηση είναι μία μέθοδος μη επιτηρούμενης μάθησης και μία τεχνική η οποία χρησιμοποιείται επίσης στην στατιστική ανάλυση δεδομένων.

Δίκτυα Bayes

Ένα δίκτυο Bayes, ένα δίκτυο εμπιστοσύνης ή ένα άκυκλο γραφικό μοντέλο είναι ένα πιθανοθεωρητικό γραφικό μοντέλο που απεικονίζει ένα σύνολο τυχαίων μεταβλητών και την μεταξύ τους υποθετική ανεξαρτησία διαμέσου ενός κατευθυνόμενου άκυκλου γράφου . Για παράδειγμα, ένα δίκτυο Bayes μπορεί να αναπαραστήσει την πιθανοθεωρητική σχέση μεταξύ ασθενειών και συμπτωμάτων. Δεδομένων των συμπτωμάτων, το δίκτυο μπορεί να χρησιμοποιηθεί για να υπολογίσει τις πιθανότητες παρουσίας διαφόρων ασθενειών.

Ενισχυτική μάθηση

Η ενισχυτική μάθηση ασχολείται με το πώς ένα υποκείμενο (πράκτορας) θα πρέπει να δράσει σε ένα περιβάλλον , έτσι ώστε να μεγιστοποιηθεί κάποια έννοια μακροπρόθεσμης ανταμοιβής. Οι αλγόριθμοι ενισχυτικής μάθησης προσπαθούν να βρούν μια πολιτική που αντιστοιχίζει τις καταστάσεις του περιβάλλοντος με τις ενέργειες που ο πράκτορας θα πρέπει να επιτελέσει σε αυτές τις καταστάσεις. Η

ενισχυτική μάθηση διαφέρει από τα προβλήματα επιτηρούμενης μάθησης αφού τα σωστά ζεύγη δεδομένων εισόδου/εξόδου ζεύγη δεν παρουσιάστηκαν ποτέ, ούτε οι βέλτιστες δυνατές ενέργειες έχουν ρητά διορθωθεί.

Εκμάθηση με μέτρο ομοιότητας

Σε αυτή την κατηγορία προβλημάτων δίνονται στην μηχανή μάθησης ζεύγη παραδειγμάτων που θεωρούνται όμοια και ζεύγη που θεωρούνται ανόμοια. Τότε η μηχανή μάθησης πρέπει να μάθει μια συνάρτηση ομοιότητας (ή μια συνάρτηση μετρικής απόστασης), που μπορεί να προβλέψει αν δύο καινούρια αντικείμενα είναι όμοια. Πρόκειται για μια τεχνική που χρησιμοποιείται σε συστήματα σύστασης.

Γενετικοί αλγόριθμοι

Ένας γενετικός αλγόριθμος (GA) είναι μια ευρετική αναζήτηση που μιμείται τη διαδικασία της φυσικής επιλογής, και χρησιμοποιεί μεθόδους όπως αυτή της μετάλλαξης και της διασταύρωσης προκειμένου να δημιουργήσει καινούρια γονότυπα με την ελπίδα εύρεσης αποτελεσματικών λύσεων σε ένα συγκεκριμένο πρόβλημα. Στη μηχανική μάθηση, γενετικοί αλγόριθμοι χρησιμοποιήθηκαν τη δεκαετία του 1980 και του 1990. Αντίστροφα, τεχνικές μηχανικής μάθησης έχουν χρησιμοποιηθεί για την βελτίωση της απόδοσης γενετικών και εξελικτικών αλγορίθμων.

Εξαγωγή Χαρακτηριστικών σε Δεδομένα Κειμένου

Σε αυτό το κεφάλαιο λοιπόν θα δούμε τρόπους εξαγωγής χαρακτηριστικών από δεδομένα κειμένου. Υπενθυμίζεται ότι το πρόβλημα είναι η ταξινόμηση ενός συνόλου από tweets σε δύο κλάσεις, θετικό ή αρνητικό συναίσθημα. Συνεπώς το ζητούμενο είναι η αντιστοίχιση κάθε tweet σε μία διανυσματική αναπαράσταση $\mathbf{x} = (x1, x2, \dots, xn)$ όπου n η διάσταση του χώρου χαρακτηριστικών.

Δεδομένων κάποιων σημείων σε ένα n-διάστατο χώρο συνοδευόμενων από κάποια κλάση, το ζητούμενο είναι η υλοποίηση ενός μαθηματικού μοντέλου που θα έχει τη δυνατότητα να ταξινομεί επιτυχώς νέα σημεία στη σωστή κλάση. Για το σκοπό αυτό είδαμε αλγορίθμους επιβλεπόμενης μάθησης που μαθαίνουν από τα δεδομένα εκπαίδευσης, όπου η μάθηση στην ουσία είναι η χρήση των δεδομένων αυτών για την

υλοποίηση κάποιας διαμέρισης του χώρου σε περιοχές, ώστε σημεία της ίδιας περιοχής να αντιστοιχούν στην ίδια κλάση.

Όταν όμως θέλουμε να ταξινομήσουμε αντικείμενα σε κλάσεις, αφού προηγουμένως εφαρμοστεί ο αλγόριθμος ταξινόμησης, είναι απαραίτητο να αναπαραστήσουμε με κάποιο τρόπο τα αντικείμενα αυτά σε διανύσματα ενός χώρου χαρακτηριστικών. Η διαδικασία αυτή καλείται εξαγωγή χαρακτηριστικών (feature extraction) και το γενικότερο πεδίο μελέτης feature engineering.

Χρειαζόμαστε ένα τρόπο να απεικονίζουμε λέξεις, φράσεις, προτάσεις και ολόκληρα κείμενα σε ένα διανυσματικό χώρο λίγων διαστάσεων, έτσι ώστε κοντινά σημεία στο χώρο να αντιπροσωπεύουν παρόμοιο νόημα (semantic similarity). Επίσης είναι χρήσιμο, για υπολογιστικούς λόγους, τα διανύσματα να είναι πυκνά (dense) με συνεχή χαρακτηριστικά και όχι διακριτά και αραιά όπως στην περίπτωση των αναπαραστάσεων term frequency ή term occurrence. Επιπλέον οι αναπαραστάσεις αυτές πρέπει να έχουν καθολικό χαρακτήρα έτσι ώστε να μπορούν να χρησιμοποιηθούν σε ποικίλες εφαρμογές επεξεργασίας κειμένου.

Εισαγωγή στο μοντέλο Word2Vec

Οι διανυσματικές αναπαραστάσεις στη βιβλιογραφία απαντώνται με τους όρους embeddings, representations ή απλά vectors. Συγκεκριμένα για τις αναπαραστάσεις λέξεων, συναντάμε τους όρους word embeddings, word representations ή word vectors. Η ιδέα της αναπαράστασης λέξεων με σημεία ενός διανυσματικού χώρου μικρής διάστασης είναι σχετικά παλιά, ωστόσο τα τελευταία χρόνια έχει γίνει σημαντική πρόοδος με την ανάπτυξη αλγορίθμων που έχουν την ικανότητα να εξ άγουν υψηλής ποιότητας αναπαραστάσεις από πολύ μεγάλους όγκους κειμένου.

Αυτό το κεφάλαιο θα επικεντρωθεί στη χρήση κατανεμημένων διανυσμάτων λέξεων (Distributed Word Vectors) που δημιουργούνται από τον αλγόριθμο Word2Vec. Το Word2vec, που δημοσιεύεται από την Google το 2013, είναι μια εφαρμογή νευρωνικού δικτύου που μαθαίνει κατανεμημένες αναπαραστάσεις για λέξεις. Άλλες αρχιτεκτονικές νευρωνικών δικτύων είχαν προταθεί για την εκμάθηση λέξεων, πριν από αυτό το μοντέλο, αλλά το σημαντικότερο πρόβλημα τους ήταν ο μακρύς χρόνος που απαιτείται για την εκπαίδευση των μοντέλων. Το Word2vec μαθαίνει γρήγορα σε σχέση με άλλα μοντέλα. Αλλά ενδιαφέροντα χαρακτηριστικά της κατασκευής των μοντέλων αυτών είναι η μη χρήση τεχνικών επίβλεψης (unsupervised) και η μη εξάρτηση από γλωσσολογικά χαρακτηριστικά, γεγονός που τα καθιστά -από πλευράς υλοποίησης- ανεξάρτητα από τη φυσική γλώσσα ως προς την οποία εφαρμόζονται (language-agnostic).

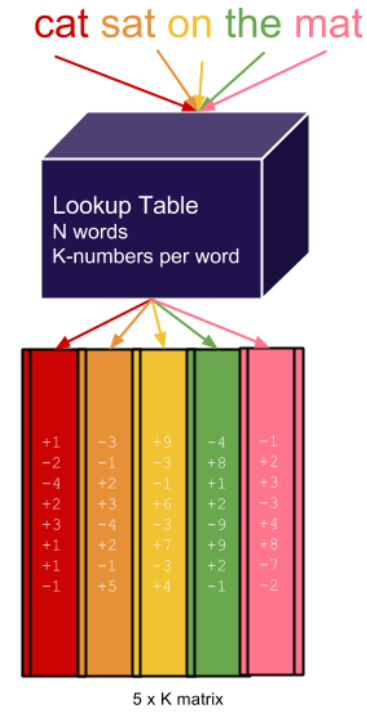


Figure 1: Word embeddings are usually stored in a simple lookup table. Given a word, the word vector of numbers is returned. Given a sentence, a matrix of vectors for each word in the sentence is returned.

Πως φτιάχνεται το διάνυσμα λέξεων.

Τι εννοούμε με τον όρο διάνυσμα λέξεων

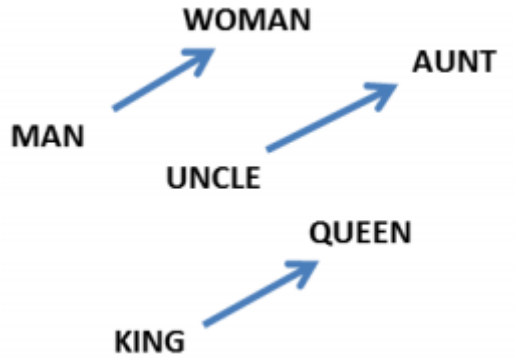
Ένα διάνυσμα λέξης (word embedding) $W:\rightarrow n$ είναι μια συνάρτηση που αντιστοιχεί μία λέξη σε διανύσματα μεγάλης διαστάσεως (πχ 50 έως 500 διαστάσεις). Για παράδειγμα, θα μπορούσαμε να βρούμε:

$$W(\text{‘‘cat"})=(0.2, -0.4, 0.7, ...)$$

$$W(\text{‘‘mat"})=(0.0, 0.6, -0.1, ...)$$

Το Word2Vec δεν χρειάζεται τα δεδομένα να έχουν επισημανθεί (labeled) για να δημιουργήσει σημαντικές αναπαραστάσεις. Αυτό είναι χρήσιμο, αφού τα περισσότερα δεδομένα στον πραγματικό κόσμο δεν έχουν επισημανθεί. Εάν δοθούν στο μοντέλο αρκετά δεδομένα (σώμα κειμένου (corpus) με δεκάδες δισεκατομμύρια λέξεις), παράγει ενδιαφέροντα χαρακτηριστικά. Λέξεις με παρόμοιες έννοιες εμφανίζονται σε συστάδες και οι συστάδες είναι τοποθετημένες σε απόσταση έτσι ώστε μερικές λέξεις μπορούν να αναπαραχθούν χρησιμοποιώντας μαθηματικά διανύσματα. Το διάσημο παράδειγμα είναι ότι, με πολύ καλά εκπαιδευμένα διάνυσματα λέξεων, "βασιλιάς - άντρας + γυναίκα = βασίλισσα". Η βασική ιδέα που υλοποιούν αυτά τα μοντέλα είναι

ότι “κοντινές” σημασιολογικά λέξεις θα έχουν και κοντινές διανυσματικές αναπαραστάσεις.



Σχήμα από Mikolov (2013) [3]

Relationship	Example 1	Example 2	Example 3
France - Paris	Italy: Rome	Japan: Tokyo	Florida: Tallahassee
big - bigger	small: larger	cold: colder	quick: quicker
Miami - Florida	Baltimore: Maryland	Dallas: Texas	Kona: Hawaii
Einstein - scientist	Messi: midfielder	Mozart: violinist	Picasso: painter
Sarkozy - France	Berlusconi: Italy	Merkel: Germany	Koizumi: Japan
copper - Cu	zinc: Zn	gold: Au	uranium: plutonium
Berlusconi - Silvio	Sarkozy: Nicolas	Putin: Medvedev	Obama: Barack
Microsoft - Windows	Google: Android	IBM: Linux	Apple: iPhone
Microsoft - Ballmer	Google: Yahoo	IBM: McNealy	Apple: Jobs
Japan - sushi	Germany: bratwurst	France: tapas	USA: pizza

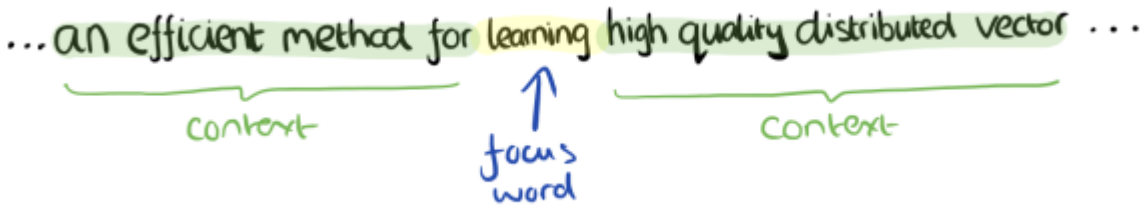
Ζευγάρια λέξεων που προκύπτουν από διανύσματα λέξεων. Σχήμα από Mikolov (2013) [3]

Οι Mikolov στο [3] δίνουν δύο διαφορετικές υλοποιήσεις του μοντέλου word2vec όπου η πρώτη καλείται Continuous Bag-of-Words (CBOW) και βασίζεται στην πρόβλεψη της κεντρικής λέξης (central word) δεδομένων των λέξεων γύρω από αυτή (context words) και η δεύτερη Skip-Gram (SG) και βασίζεται στην πρόβλεψη των context words δεδομένης της λέξης στο κέντρο. Για την θεωρητική ανάλυση του αλγορίθμου στη συνέχεια , δανείζονται στοιχεία από την δημοσίευση του Xin Rong [22] πάνω στον αλγόριθμο word2vec.

Αλγόριθμος Word2vec

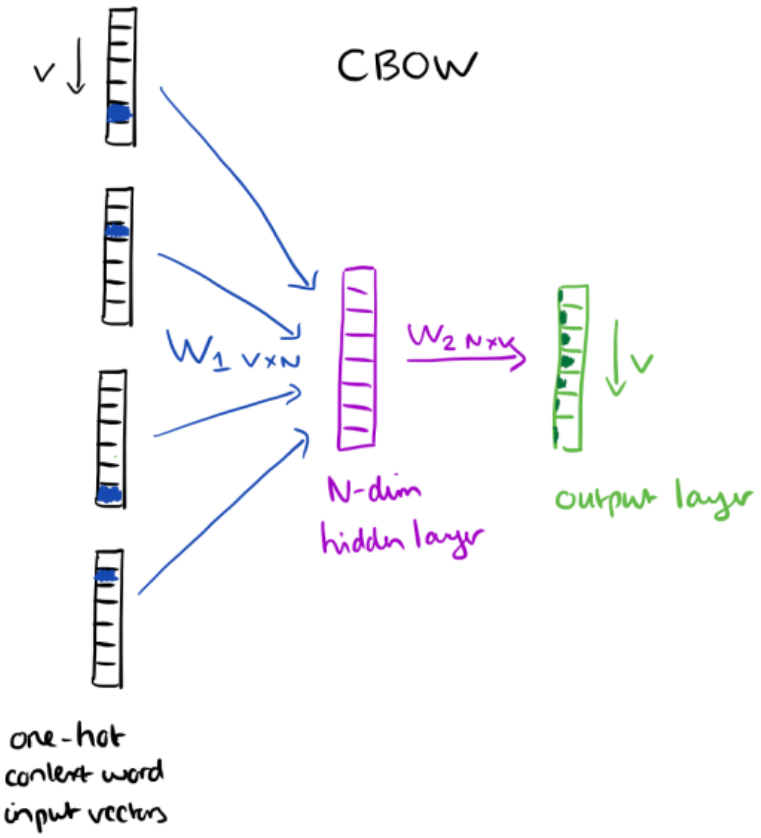
Θεωρείστε την ακόλουθη φράση “*The recently introduced continuous Skip-gram model is an efficient method for learning high-quality distributed vector representations that*

capture a large number of precises syntatic and semantic word relationships.” Ας επιλέξουμε μία λέξη - στόχο και ένα παράθυρο γύρω από τη λέξη αυτή, συγκεκριμένα 4 λέξεις πριν και 4 λέξεις μετά από την λέξη στόχο. Αυτές οι 4 λέξεις αποτελούν το context όπως ονομάζεται.



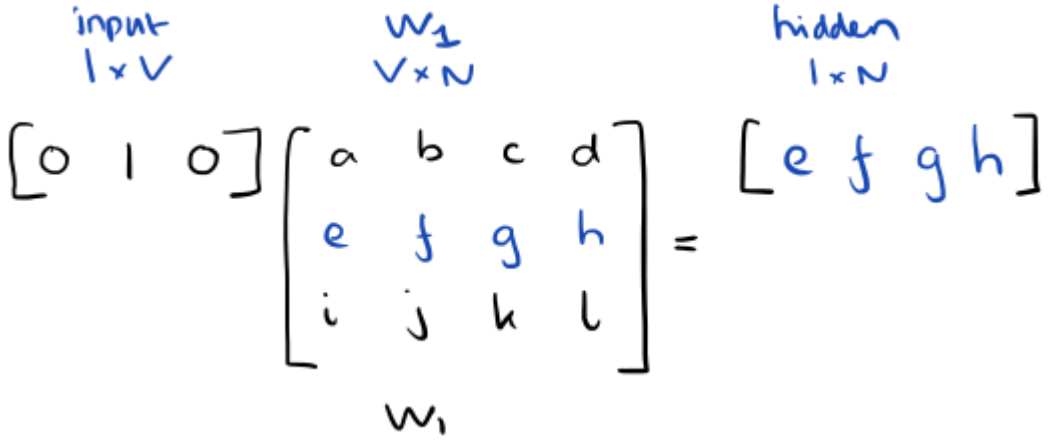
Το μοντέλο χρησιμοποιεί ένα νευρωνικό δίκτυο που αποτελείται από 3 επίπεδα (layers). Εκτός από το επίπεδο εισόδου υπάρχει ένα κρυφό επίπεδο και ένα επίπεδο εξόδου. Οι context λέξεις μπαίνουν στο επίπεδο εισόδου (input layer). Το πλήθος των κρυφών νευρώνων είναι ίσο με την επιθυμητή διάσταση των word vectors που θα προκύψουν. Το πλήθος των νευρώνων εξόδου είναι ίσο με το πλήθος των νευρώνων εισόδου.

Κάθε λέξη κωδικοποιείται χρησιμοποιώντας την αναπαράσταση «1 από V» («1-out of V») ή αλλιώς one-hot. Επομένως, εάν το μέγεθος του λεξικού είναι V, αυτοί θα είναι πίνακες διάστασης V όπου μόνο ένα στοιχείο θα είναι 1 και όλα τα άλλα θα είναι μηδενικά. 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 WI 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 $W2$ μεγέθους NxV. Στην περίπτωση αυτή, κάθε στήλη του πίνακα W2 αντιπροσωπεύει μία λέξη του λεξιλογίου. Υπάρχουν 2 κύριοι αλγόριθμοι μάθησης στον Word2vec: Ο Αλγόριθμος Continuous Bag of Words (CBOW) Learning και ο Αλγόριθμος Skip-gram.



Ο στόχος της εκπαίδευσης είναι να μεγιστοποιηθεί η υπό όρους πιθανότητα παρατήρησης της πραγματικής λέξης εξόδου (λέξη στόχος) δεδομένου των context λέξεων εισόδου. Στο παράδειγμά μας, δεδομένου της εισόδου (“an”, “efficient”, “method”, “for”, “high”, “quality”, “distributed”, “vector”) θέλουμε να μεγιστοποιήσουμε την πιθανότητα να πάρουμε την λέξη “learning” στην έξοδο.

Εφόσον οι πίνακες στην είσοδο είναι one-hot, πολλαπλασιάζοντας τον πίνακα ειδόσοδο με τον πίνακα βαρών **W1** ισοδυναμεί με το να πάρουμε μία γραμμή από τον πίνακα **W1**.



Η έξοδος του κρυφού επιπέδου είναι ***h*** (activation function for the hidden layer):

h=W1T*x

Όπου x: input vector

Λόγω αυτής της μορφής αναπαράστασης της εισόδου, το word vector της λέξης εισόδου (αντίστοιχη γραμμή του πίνακα $W1$) αντιγράφεται στην έξοδο του hidden layer. Η ενεργοποίηση του στρώματος εξόδου δίνεται από:

$$Act = W2T*h$$

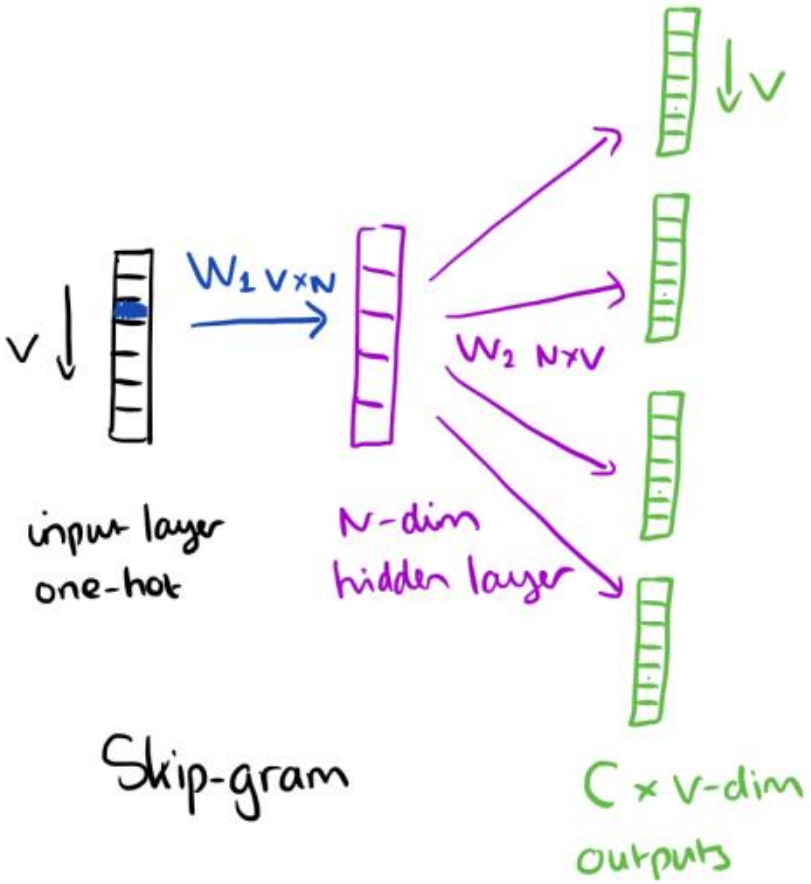
Αφού ο στόχος είναι η παραγωγή πιθανοτήτων για τις λέξεις στο επίπεδο εξόδου $P(wordk\ |wordcontext)\ για\ k = 1, \dots, V$, ώστε να αντικατοπτρίζουν την σχέση επόμενης λέξης με τη λέξη context στην είσοδο, χρειαζόμαστε το άθροισμα των εξόδων να ισούται με 1. Αυτό επιτυγχάνεται χρησιμοποιώντας τη συνάρτηση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

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

Στον αλγόριθμο CBOW έχουμε *C* στο πλήθος input word vectors (όσες και οι λέξεις του context), και η συνάρτηση **h** αποτελείται από το άθροισμα των 'hot' γραμμών στον πίνακα W1 και την προσθήκη μίας διαίρεσης με το *C* στο κρυφό επίπεδο.

Οι πίνακες W1 και W2 μπορούν να ανανεωθούν με τον κανόνα backpropagation. Άρα, η εκπαίδευση μπορεί να προχωρήσει παρουσιάζοντας διαφορετικά ζευγάρια context-στόχου από το corpus.

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. (the target context words are now at the output layer):



Σε αυτή την περίπτωση η συνάρτηση στο κρυφό επίπεδο (activation function) υπολογίζεται με αντιγραφή της αντίστοιχης γραμμής από τον πίνακα βαρών $\mathbf{W_1}$ (linear) όπως είδαμε πριν στο μοντέλο CBOW. Στο επίπεδο εξόδου αυτή τη φορά έχουμε *C* στο πλήθος πιθανότητες (multinomial distributions) αντί για μία. Ο στόχος της εκπαίδευσης για αυτό το μοντέλο είναι να ελαχιστοποιηθεί το σφάλμα πρόβλεψης σε όλες τις λέξεις που υπάρχουν στο context, στο επίπεδο εξόδου. Στο παράδειγμά μας η είσοδος είναι "learning", και ελπίζουμε στην έξοδο να δούμε: ("an", "efficient", "method", "for", "high", "quality", "distributed", "vector").

Όλα όσα περιγράψαμε στο κεφάλαιο αυτό υλοποιούνται στον αλγόριθμο word2vec που δίνει η Google (γραμμένος σε γλώσσα προγραμματισμού C) . Υπάρχουν δημοσίως διαθέσιμα pre-trained word vectors.

Για την υλοποίηση μας, χρησιμοποιήσαμε τα word2vec διανύσματα που εκπαιδεύτηκαν πάνω σε ένα μεγάλο σύνολο από tweets (βλ. ενότητα «Δεδομένα»). Τα διανύσματα αυτά έχουν διάσταση 300 και αποκτήθηκαν χρησιμοποιώντας την αρχιτεκτονική CBOW.

Εξαγωγή χαρακτηριστικών από ένα tweet

Για κάθε tweet φτιάχνουμε ένα πίνακα χαρακτηριστικών. Τα χαρακτηριστικά αυτά προέρχονται είτε από τα διανύσματα λέξεων (word embedding) είτε από λεξικά. Τα χαρακτηριστικά μπορούν να χωριστούν σε ομάδες όπως δείχνουμε στον παρακάτω πίνακα.

FEATURE SETS
Thrird party lexica :affin, nrc, nrctag
Subjectivity lexica
w2vec (size=300, 200 etc.)
morphology
hashtags

Προεπεξεργασία

Ειδικά στην περίπτωση του Twitter λόγω της ειδικής φύσης των tweets η διαδικασία της προεπεξεργασίας είναι ιδιαίτερα σημαντική.

Ας δούμε ενδεικτικά τη μορφή μερικών tweets από το σύνολο δεδομένων:

Gas by my house hit $3.39!!!! I'm going to Chapel Hill on Sat. :)

Looks like Andy the Android may have had a little too much fun yesterday. http://t.co/…

@Jen I have studied all day but tomorrow I'm going out with friends! :D Omg Jennette did?!!!! I'm gonna look! <3

#NowPlaying: BEP, Ricky Martin and KT Tunstall!
Great songs to get you through your Sunday! Hate the rain!! http://t.co/...

Στα παραπάνω tweets φαίνονται μερικά από τα ιδιαίτερα tokens που χρίζουν ειδικής μεταχείρισης όπως το emoticon ":)", το url "http://t.co/...", το username "@Jen" και το hashtag "#NowPlaying".

Η προεπεξεργασία γίνεται με τη βοήθεια κανονικών εκφράσεων (regular expressions) και του re module της python. Αναζητούνται σειρές χαρακτήρων που επαληθεύουν

συγκεκριμένες κανονικές εκφράσεις και αντικαθίστανται από strings ή άλλες κανονικές εκφράσεις.

Ηλεκτρονικές διευθύνσεις - URLs

Οι ηλεκτρονικές διευθύνσεις αντικαθίστανται με το string . Για παράδειγμα http://t.co/... → <url>

Δηλαδή ένα tweet από τα παραδείγματα θα γίνει :
looks like andy the android may have had a little too much fun yesterday <url>.

Ειδικοί χαρακτήρες html

Στο σώμα κειμένου, ειδικά των tweets που προέρχονται από το SemEval περιέχονται διάφοροι ειδικοί χαρακτήρες, κατάλοιπο της html επεξεργασίας που υφίστανται τα tweets. Οι ειδικοί αυτοί χαρακτήρες αφαιρούνται από τα tweets καθώς θεωρήσαμε ότι δεν παίζουν σημαντικό ρόλο στον καθορισμό του συνασθήματος.

Usernames

Τα usernames, αρχίζουν με το σύμβολο @ και περιλαμβάνουν γράμματα, αριθμούς ή underscores, αντικαθίστανται απλά από τον όρο <user> καθώς δεν περιέχουν σημαντική πληροφορία που μπορεί να ωφελήσει την ανάλυση συναισθήματος.

<user> i have studied all day but tomorrow i'm going out with friends!

Hashtags

Τα hashtags διακρίνονται από τη χρήση του χαρακτήρα #. Δεν αφαιρούνται ούτε γίνεται κάποια άλλη επεξεργασία σε αυτά και όπως θα δούμε στη συνέχεια χρησιμοποιούνται ως χαρακτηριστικά ενός tweet.

Emoticons

Η επιτυχής αναγνώριση των emoticons είναι μία απαιτητική εργασία εξαιτίας του θορύβου που ενυπάρχει στον τρόπο γραφής τους σε πραγματικά tweets. Επειδή τα tweets του SemEval [13] δεν περιέχουν σημαντικό αριθμό από emoticons, τα χρησιμοποιούμε όπως θα δούμε στην συνέχεια για να εξάγουμε κάποια στατιστικά από το σύνολο των tweets και δεν κάνουμε κάποια επιπλέον επεξεργασία.

Retweets - RTs

Τα string RT που δηλώνουν κοινοποίηση tweet άλλου χρήστη αντικαθίστανται με τον όρο <rt> .

POS-tagging / Tokenization

Το επόμενο βήμα πριν την εξαγωγή των χαρακτηριστικών είναι το POS-tagging / Tokenization. Αυτό έγινε χρησιμοποιώντας το εργαλείο ARK NLP twitter tagger [23].

Χρησιμοποιώντας το ARK NLP tagger μπορούμε να χαρακτηρίσουμε τις λέξεις που υπάρχουν μέσα σε ένα tweet ανάλογα με το αν είναι noun,verb,adjective κτλ.
Για παράδειγμα:

word	*tag*	*confidence*
ikr	!	0.8143
smh	G	0.9406
he	O	0.9963
asked	V	0.9979
fir	P	0.5545
yo	D	0.6272
last	A	0.9871
name	N	0.9998
so	P	0.9838
he	O	0.9981
can	V	0.9997
add	V	0.9997
u	O	0.9978
on	P	0.9426
fb	^	0.9453
lololol	!	0.9664

Επειδή το APK NLP έχει γραφτεί ειδικά για το twitter βλέπουμε από τα παραδείγματα απάνω ότι αναγνωρίζει επιτυχώς τις ιδιωματικές λέξεις που χρησιμοποιούνται στο twitter

- *"ikr"* σημαίνει "I know, right?", το οποίο σημειώνεται ώς interjection.
- *"fb"* σημαίνει "Facebook", ένα πολύ κοινό “proper noun” (^).

Επίσης επισημειώνονται και τα emoticons που μπορεί να περιέχει ένα tweet.

word	*tag*	*confidence*
:o	E	0.9387
:/	E	0.9983
:'(	E	0.9975
>:o	E	0.9964
(:	E	0.9994
:)	E	0.9997

>.< E 0.9952
XD E 0.9938
-__- E 0.9956
o.O E 0.9899
;D E 0.9995
:-) E 0.9992
@_@ E 0.9964
:P E 0.9996
8D E 0.9961
: E 0.6925
1 $ 0.9194
>:(E 0.9715
:D E 0.9996
=| E 0.9963
" , 0.6125
) , 0.9078
: , 0.7460
> G 0.7490
... , 0.5223
. , 0.9946

Word Vectors και Ανάλυση Συναισθήματος

Σε αυτή την ενότητα θα εξεταστεί ο τρόπος εξαγωγής χαρακτηριστικών που θεμελιώθηκε θεωρητικά στις προηγούμενες ενότητες. Θα χρησιμοποιηθούν trained word vectors του μοντέλου Skip-gram, και για να εκπαιδεύσουμε το μοντέλο θα χρησιμοποιήσουμε το corpus των 50εκ. tweets.
Μετά το πέρας αυτής της εκπαίδευσης κάθε μία λέξη στο corpus αυτό θα αντιστοιχεί σε ένα διάνυσμα έστω με 300 τιμές.

Ας εξετάσουμε ένα tweet ως μία πρόταση.

“Tix going fast: IA's Bash "20th C Time Machine" celebrating the music, film, web & books we're saving 4 a new era.”

Για κάθε μία λέξη στο παραπάνω tweet προσθέτουμε το αντίστοιχο διάνυσμα που προκύπτει από το μοντέλο word2vec που μόλις εκπαιδεύσαμε. Αν μία λέξη δεν υπάρχει στο μοντέλο μας (out-of-corpus) τότε για τη λέξη αυτή φτιάχνουμε ένα τυχαίο διάνυσμα με 300 τιμές στο διάστημα [0,1].

$$\mathbf{v}_d = \sum_{i=1}^{n} \mathbf{v}_i \quad , \quad v_{dj} = \sum_{i=1}^{n} v_{ij}$$

Στο τέλος, αφού προσθέσουμε όλα τα επιμέρους διανύσματα, το διάνυσμα της πρότασης (Sentence matrix) θα είναι ο μέσος όρος των διανυσμάτων των λέξεων της πρότασης. Δηλαδή:

$$\mathbf{v}_d = \frac{1}{n}\sum_{i=1}^{n} \mathbf{v}_i \quad , \quad v_{dj} = \frac{1}{n}\sum_{i=1}^{n} v_{ij}$$

Είναι αξιοσημείωτο το πώς η απλή πρόσθεση διανυσματικών αναπαραστάσεων που κωδικοποιούν τη γενικότερη σημασιολογική σχέση μεταξύ λέξεων, αποδίδει τόσο καλά στο πρόβλημα της ανάλυσης συναισθήματος. Τα word vectors προκύπτουν από unsupervised κείμενο και δεν είναι task specific για το πρόβλημα της ανάλυσης συναισθήματος, δηλαδή δεν είναι προσαρμοσμένα ώστε να συλλαμβάνουν τη σχέση των λέξεων σε επίπεδο συναισθηματικής πόλωσης. Αντ' αυτού είναι globally tuned (αφού προέρχονται από unsupervised κείμενο) για να μπορούν να χρησιμοποιηθούν σε διάφορα προβλήματα επεξεργασίας φυσικού λόγου. Εξαιτίας των παραπάνω λόγων, είναι ενδιαφέρον το πως η πρόσθεση των word vectors δίνει καλά αποτελέσματα συγκρίσιμα μίας μεθόδου Bag of words [34].

Προσθήκη χαρακτηριστικών στο διάνυσμα λέξης

Προσπαθώντας να έχουμε καλύτερη απόδοση στο πρόβλημα που εξετάζουμε σκεφτήκαμε να εξετάσουμε την προσθήκη επιπλέον χαρακτηριστικών στο διάνυσμα κάθε λέξης. Για παράδειγμα το **AFFIN** λεξικό (Nielsen, 2011), περιέχει ένα σύνολο από 2477 λέξεις από το twitter για τις οποίες έχει δοθεί μία τιμή συναισθήματος (valence) που είναι μία συνεχής τιμή στο διάστημα [-1,1] . **Πηγαίνοντας πίσω στο διάνυσμα της πρότασης μπορούμε εκτός από τις 300 τιμές από το word2vec να προσθέσουμε και ένα επιπλέον χαρακτηριστικό, την μέση τιμή valence για το κάθε tweet.** Με αυτό τον τρόπο ο πίνακας για την πρόταση (Sentence matrix) θα έχει τώρα 301 τιμές. **Η βελτίωση ή όχι της συγκεκριμένης τεχνικής μελετάται στην ενότητα των αποτελεσμάτων.**
Με τη ίδια λογική (της μέσης τιμής ανά πρόταση) χρησιμοποιήσαμε και άλλα δύο διαθέσιμα λεξικά , το **NRC** και το **NRCTAG** [20, 21].

Επιπλέον χρησιμοποιήσαμε το **Subjectivity lexicon** (list of subjectivity clues) (Wilson , 2005). Το λεξικό αυτό αποτελείται από ένα σύνολο λέξεων οι οποίες έχουν επισημανθεί ως strong positive/negative, weak positive/negative. **Τα χαρακτηριστικά που χρησιμοποιούμε από αυτό το λεξικό είναι να μετρήσουμε τη συχνότητα των strong/weak λέξεων σε κάθε tweet και τη μέση τιμή εμφάνισης για κάθε subjective κατηγορία**.

Επιπλέον, χρησιμοποιούμε και τα παρακάτω χαρακτηριστικά τα οποία χωρίζονται ανάλογα με το part of speech tag. Ονομάζουμε το σύνολο των χαρακτηριστικών αυτών ***morphology features***: Συγκεκριμένα υπολογίζουμε μήκος σε χαρακτήρες (length), καθώς **και άλλα statistics** όπως min, max, max amplitude, sum, average, range (max minus min), standard deviation και variance. **Τα αποτελέσματα είναι στατιστικές τιμές ανά part of speech tag**, δηλαδή "maximum valence among adjectives", "mean valence among proper nouns", "number of verbs and nouns", κτλ.

Τα **hashtags** παίζουν σημαντικό ρόλο πολλές φορές στην συναισθηματική αναγνώριση ενός tweet. Παρόλο που μπορεί κάποιος να αναλύσει τα hashtags, π.χ. να σπάσει κάθε hashtag στις πιθανές λέξεις που περιέχει, εμείς χρησιμοποιήσαμε μόνο στατιστικά από τα hashtags που βρήκαμε στα tweets του SemEval [13]. Συγκεκριμένα χρησιμοποιήσαμε μέση συχνότητα εμφάνισης hashtags καθώς και μία δυαδική τιμή (binary indicator) ότι ένα tweet περιέχει η όχι ένα hashtag.

Τέλος στην ομάδα των χαρακτηριστικών αυτών (morphology features) προσθέσαμε και **στατιστικά για τα emoticons** καθώς και για τα σημεία στίξης, καθώς θεωρούμε ότι και αυτά είναι ενδεικτικά του συναισθήματος που μπορεί να εκφράζει μία πρόταση.

Stopwords and collocations

Η βελτίωση της εξαγωγής χαρακτηριστικών μπορεί συχνά να έχει σημαντικά θετική επίδραση στην ακρίβεια ταξινομητή . Σε αυτή την ενότητα θα εξετάσουμε **δύο τροποποιήσεις της μεθόδου εξαγωγής χαρακτηριστικών:**

1. Φιλτράρουμε τις λέξεις που ονομάζονται stopwords
2. Εισάγουμε χαρακτηριστικά που αφορούν τα bigrams

Σε αυτή τη διαδικασία χρησιμοποιούμε ένα τρόπο εξαγωγής χαρακτηριστικών ο οποίος παίρνει τις λέξεις από ένα αρχείο και επιστρέφει το "λεξικό χαρακτηριστικών". Θα χρησιμοποιήσουμε αυτά τα χαρακτηριστικά για την εκπαίδευση ενός Naive Bayes Classifier.

Τα **stopwords** είναι λέξεις που γενικά δεν θεωρούνται χρήσιμες στην επεξεργασία, όπως πχ a, and, or κτλ. Οι περισσότερες μηχανές αναζήτησης αγνοούν αυτές τις λέξεις επειδή είναι τόσο συνηθισμένες. Επιλέξαμε να φιλτράρουμε τις 128 αγγλικές λέξεις που περιέχονται στο πακέτο NLTK .

Η υπόθεση που κάνουμε στη συνέχεια είναι ότι οι άνθρωποι λένε φράσεις όπως "not great", που είναι μια αρνητική έκφραση που θα μπορούσε να ερμηνεύσει ως θετική αν δούμε τη λέξη "great" ως ξεχωριστή λέξη. Επομένως η **χρήση των bigrams** σαν χαρακτηριστικά θα μπορούσε να βελτιώσει την απόδοση στην ανάλυση συνασθήματος.

Για να βρούμε τα "σημαντικά" bigrams, μπορούμε να χρησιμοποιήσουμε το nltk.collocations.BigramCollocationFinder της python μαζί με τα nltk.metrics.BigramAssocMeasures. Ο BigramCollocationFinder διατηρεί 2 εσωτερικά FreqDists, ένα για συχνότητες λέξεων, ένα άλλο για συχνότητες bigram. Αφού έχει αυτές τις κατανομές συχνότητας, μπορεί να βαθμολογήσει μεμονωμένα bigrams χρησιμοποιώντας μια λειτουργία βαθμολόγησης που παρέχεται από το BigramAssocMeasures, όπως το chi-square. Αυτές οι λειτουργίες βαθμολόγησης μετρούν τη συσχέτιση μεταξύ των 2 λέξεων, βασικά αν το bigram συμβαίνει περίπου τόσο συχνά όσο κάθε μεμονωμένη λέξη.

```
from nltk.collocations import BigramCollocationFinder
bigram_finder = BigramCollocationFinder.from_words(words)
```

Για παράδειγμα χρησιμοποιώντας τα bigrams ως χαρακτηριστικά σε έναν Naive Bayes Classifier, με train/test τα δεδομένα του SemEval [13], βλέπουμε για παράδειγμα ότι η λέξη " Erdogan" είναι προφανώς ένας από τους καλύτερους παράγοντες πρόβλεψης για αρνητικό συναίσθημα.

Most Informative Features

```
                Erdogan = True          negati : positi =     32.1 : 1.0
                  hurts = True          negati : positi =     30.5 : 1.0
       ('Jeb', 'Bush') = True           negati : positi =     27.7 : 1.0
                awesome = True          positi : neutra =     26.2 : 1.0
                excited = True          positi : neutra =     24.4 : 1.0
                Trayvon = True          neutra : positi =     24.2 : 1.0
                    Jeb = True          negati : positi =     23.3 : 1.0
                Trump's = True          negati : positi =     21.1 : 1.0
                   poor = True          negati : positi =     21.1 : 1.0
('absolutely', 'no') = True             neg : pos    =     10.6 : 1.0
```

Μέθοδοι ταξινόμησης

Σε αυτό το κεφάλαιο θα επιχειρηθεί μία σύντομη εισαγωγή των αλγορίθμων μηχανικής μάθησης που χρησιμοποιήθηκαν σε αυτή την εργασία.

Random forests

Σε αυτό το κεφάλαιο θα επιχειρηθεί μία σύντομη εισαγωγή του αλγορίθμου μηχανικής μάθησης Random Forest που χρησιμοποιήθηκε σε αυτή την εργασία. Ο αλγόριθμος αυτος είναι αλγόριθμος επιβλεπόμενης (supervised) μάθησης και επιλύει ουσιαστικά ένα πρόβλημα ταξινόμησης.

Ας δούμε όμως αρχικά το γενικό πλαίσιο του προβλήματος της ταξινόμησης. Έστω σημεία ενός χώρου διάστασης n

$\mathbf{x} = (x1, x2, \dots, xn)$

Τα $x1, x2 \dots$ ονομάζονται χαρακτηριστικά (features) και το **x** διάνυσμα χαρακτηριστικών (feature vector).

Το σύνολο δεδομένων εκπαίδευσης αποτελείται από πολλά τέτοια διανύσματα στον n-διάστατο χώρο χαρακτηριστικών (feature space), κάθε ένα από τα οποία ανήκει σε μία από τις k κλάσεις:

$C1, C2, \dots, Ck \in C$

όπου C το σύνολο των κλάσεων $\{C1, C2, \dots, Ck\}$.

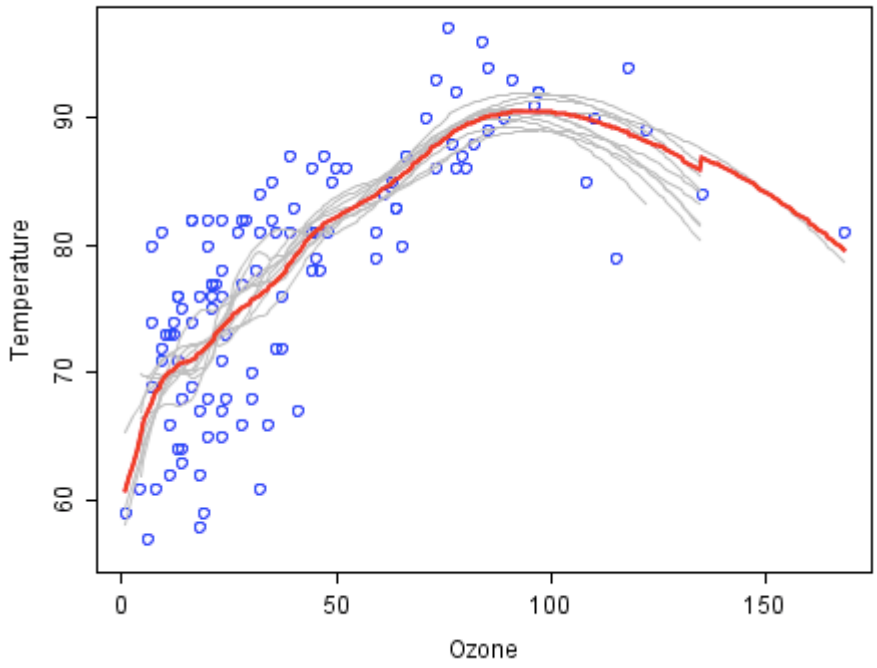
Η κλάση κάθε σημείου είναι γνωστή και στόχος είναι η πρόβλεψη της κλάσης νέων σημείων.

Κάποιοι αλγόριθμοι όπως οι μηχανές διανυσμάτων υποστήριξης, αντιμετωπίζουν το πρόβλημα γεωμετρικά και αναζητούν διαμερίσεις του χώρου χαρακτηριστικών σε διαστήματα, ώστε σημεία του ίδιου διαστήματος να ανήκουν στην ίδια κλάση. Άλλοι αλγόριθμοι όπως οι Μπεϋζιανοί ταξινομητές (Bayesian classifiers) προσεγγίζουν το πρόβλημα στατιστικά.

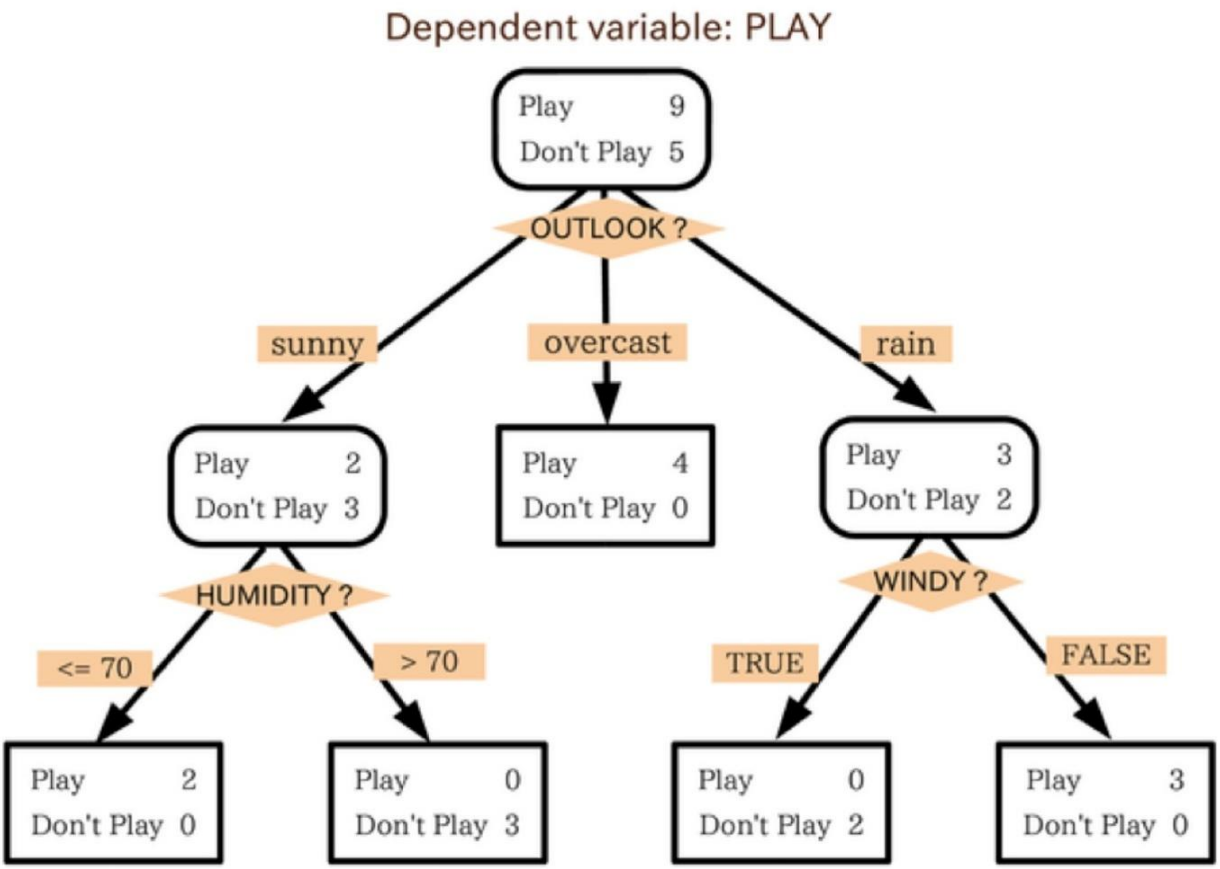
Το random forest (Breiman, 2001) είναι μια μέθοδος συνόλου που μπορεί επίσης να θεωρηθεί ως μια μορφή nearest neighbor predictor. Πρόκειται δηλ. για μέθοδο divide-and-conquer που χρησιμοποιείται για τη βελτίωση της απόδοσης. Η βασική αρχή πίσω

από τις μεθόδους αυτές είναι ότι μια ομάδα «αδύναμων μαθητών» μπορεί να έρθει μαζί για να σχηματίσει ένα «ισχυρό μαθητή». Το παρακάτω σχήμα (που λαμβάνεται από εδώ) αποτελεί παράδειγμα. Κάθε ταξινομητής, ξεχωριστά, είναι ένας "αδύναμος μαθητής", ενώ όλοι οι ταξινομητές που λαμβάνονται μαζί είναι ένας "ισχυρός μαθητής".

Τα δεδομένα που θα μοντελοποιηθούν είναι οι μπλε κύκλοι. Υποθέτουμε ότι αντιπροσωπεύουν κάποια υποκείμενη λειτουργία συν θόρυβο. Κάθε μεμονωμένος μαθητής εμφανίζεται ως γκρι καμπύλη. Κάθε γκρίζα καμπύλη (ένας ασθενής μαθητής) είναι μια δίκαιη προσέγγιση με τα υποκείμενα δεδομένα. Η κόκκινη καμπύλη (το σύνολο "ισχυρός μαθητής") μπορεί να θεωρηθεί ότι είναι πολύ καλύτερη προσέγγιση των υποκείμενων δεδομένων.



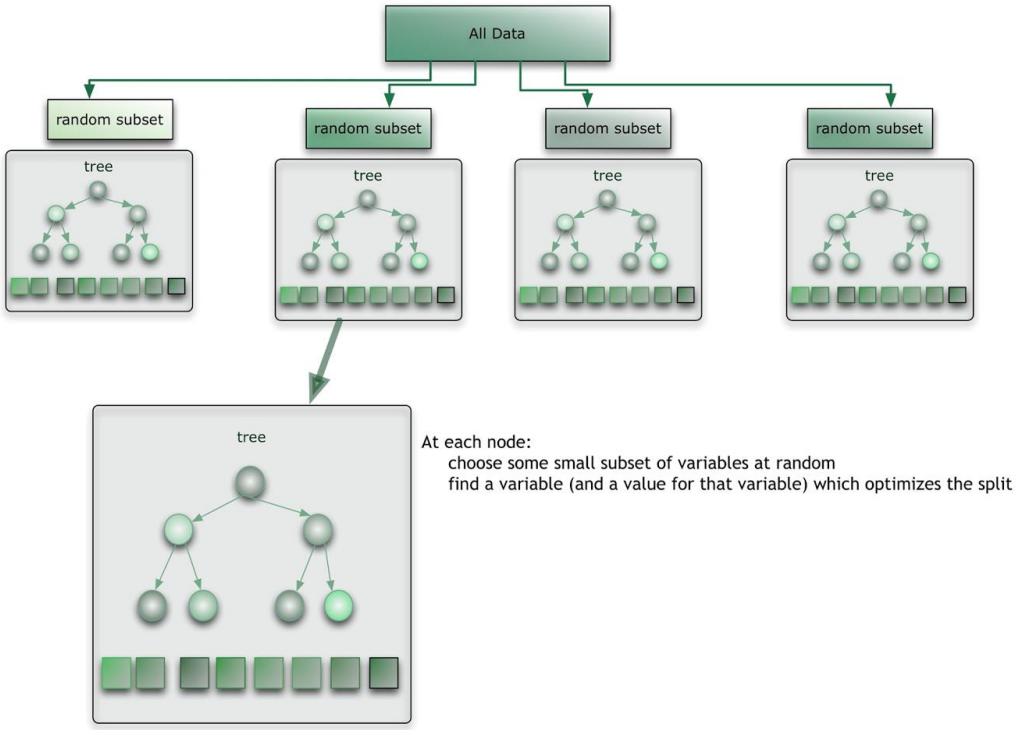
Δέντρα και δάση. Το random forest ξεκινά με μια τυπική τεχνική μηχανικής μάθησης που ονομάζεται "δέντρο απόφασης", η οποία, σε όρους συνόλων, αντιστοιχεί στον αδύναμο μαθητή μας. Σε ένα δέντρο απόφασης, μια είσοδος εισάγεται στην κορυφή και καθώς διασχίζει το δέντρο τα δεδομένα ομαδοποιούνται σε μικρότερα και μικρότερα σετ.



Σχήμα από http://blog.citizennet.com/blog/2012/11/10/random-forests-ensembles-and-performance-metrics)

Σε αυτό το παράδειγμα, το δέντρο μας συμβουλεύει, ανάλογα με τις καιρικές συνθήκες, να παίξουμε ή όχι μπάλα. Για παράδειγμα, εάν οι προοπτικές είναι ηλιόλουστες και η υγρασία είναι μικρότερη ή ίση με 70, τότε πιθανόν να είναι ok το να παίξουμε.
Το random forest (βλέπε σχήμα παρακάτω) παίρνει αυτή την έννοια στο επόμενο επίπεδο συνδυάζοντας τα δέντρα με την έννοια ενός συνόλου. Έτσι, σε όρους συνόλων, τα δέντρα είναι αδύναμοι μαθητές και το random forest είναι ένας ισχυρός μαθητής.

-



Εδώ βλέπουμε το πώς εκπαιδεύτηκε ένα τέτοιο σύστημα για κάποιο αριθμό δέντρων T:

1. Κάνουμε δειγματοληψία N περιπτώσεων τυχαία με αντικατάσταση για να δημιουργήσουμε ένα υποσύνολο των δεδομένων (βλ. κορυφαίο στρώμα του σχήματος παραπάνω). Το υποσύνολο πρέπει να είναι περίπου το 66% του συνολικού συνόλου.
2. Σε κάθε κόμβο:
    1. Για κάποιο αριθμό m (βλέπε παρακάτω), m μεταβλητές πρόβλεψης επιλέγονται τυχαία από όλες τις μεταβλητές πρόβλεψης.
    2. Η μεταβλητή πρόβλεψης που παρέχει την καλύτερη διαίρεση, σύμφωνα με κάποια αντικειμενική συνάρτηση, χρησιμοποιείται για να κάνει ένα δυαδικό διαχωρισμό σε αυτόν τον κόμβο.
    3. Στον επόμενο κόμβο, επιλέγουμε τυχαία άλλες m μεταβλητές από όλες τις μεταβλητές πρόβλεψης και κάνουμε το ίδιο.

Ανάλογα με την τιμή του m, υπάρχουν τρία ελαφρώς διαφορετικά συστήματα:

- Τυχαία επιλογή διαχωριστή: m =1
- Breiman’s bagger: m = συνολικός αριθμός μεταβλητών πρόβλεψης

- Random forest: m << αριθμός μεταβλητών πρόβλεψης. Ο Brieman προτείνει 3 πιθανές τιμές για το m: $\frac{1}{2}\sqrt{m}$, $\sqrt{m}$, and $2\sqrt{m}$

Εκτέλεση του random forest: Όταν εισάγεται μια νέα είσοδος στο σύστημα, τρέχει σε όλα τα δέντρα. Το αποτέλεσμα μπορεί να είναι είτε ο μέσος όρος είτε ο σταθμισμένος μέσος όρος όλων των τελικών κόμβων στους οποίου φτάνουμε ή (στην περίπτωση κατηγορηματικών μεταβλητών) η πλειοψηφία των ψηφοφόρων.

Σημειώνουμε ότι:

- Με ένα μεγάλο αριθμό προγνωστικών, το επιλέξιμο σύνολο προγνωστικών θα είναι αρκετά διαφορετικό από κόμβο σε κόμβο.
- Όσο μεγαλύτερη είναι η συσχέτιση μεταξύ των δέντρων, τόσο μεγαλύτερος είναι ο τυχαίος ρυθμός σφάλματος, επομένως μια βελτίωση στο μοντέλο είναι να έχουμε τα δένδρα όσο το δυνατόν πιο uncorrelated.
- Καθώς το m μειώνεται, τόσο η συσχέτιση μεταξύ δέντρων όσο και η ισχύς μεμονωμένων δέντρων μειώνονται. Πρέπει λοιπόν να βρεθεί κάποια βέλτιστη τιμή.

Παρακάτω αναφέρονται συνοπτικά οι ιδιότητες και τα κύρια πλεονεκτήματα των Random Forests:

- Μπορούν να εκπαιδευτούν σε σύνολα δεδομένων υψηλής διάστασης όπως είναι τα κείμενα και οι εικόνες, χωρίς να εμφανίσουν σημαντικό βαθμό υπερεκπαίδευσης.
- Εξαιτίας του μεγάλου πλήθους δέντρων στο δάσος, το σφάλμα γενίκευσης είναι περιορισμένο. Αυτό έχει ως αποτέλεσμα τη μη εμφάνιση φαινομένων υπερεκπαίδευσης.
- Μη επαναληπτική διαδικασία εκπαίδευσης, ο αλγόριθμος ολοκληρώνεται σε σταθερό αριθμό βημάτων.
- Η τυχαία επιλογή ενός υποσυνόλου των χαρακτηριστικών για τη διαμέριση των παραδειγμάτων κάθε ενδιάμεσου κόμβου ελαττώνει τη συσχέτιση ανάμεσα στα δέντρα και διατηρεί την πόλωση (bias) σε χαμηλά επίπεδα καθώς τα δέντρα αναπτύσσονται χωρίς κλάδεμα. Χρησιμοποιώντας ένα σύνολο δέντρων απόφασης μειώνεται και η διακύμανση (variance).
- Η διάσχιση ενός δέντρου από ένα παράδειγμα ξεκινώντας από τη ρίζα και καταλήγοντας σε έναν από τους τερματικούς κόμβους γίνεται σε λογαριθμικό ως προς το πλήθος των φύλλων του.
- Παρουσιάζουν ανεκτικότητα ως προς το θόρυβο και αριθμητικών σφαλμάτων στα δεδομένα εκπαίδευσης (π.χ. απόκρυψη μέρους του αντικειμένου, ελλιπή δεδομένα).

- Για την επαγωγή κάθε δέντρου περίπου το 1/3 των παραδειγμάτων δεν επιλέγεται για εκπαίδευση. Αυτά τα παραδείγματα καλούνται Out-of-Bag παραδείγματα και μπορούν να χρησιμοποιηθούν για την εκτίμηση της πιθανότητας σφάλματος, εξαλείφοντας την ανάγκη ύπαρξης ενός συνόλου ελέγχου ή εφαρμογής της τεχνικής cross-validation.
- Παράγει μια εσωτερική αμερόληπτη εκτίμηση του σφάλματος γενίκευσης καθώς εξελίσσεται η διαδικασία κατασκευής του δέντρου.
- Υπάρχει η δυνατότητα παράλληλης επαγωγής των δέντρων, σε αντίθεση με την μέθοδο Boosting.
- Αναζητά το καλύτερο διαχωρισμό σε ένα μικρό υποσύνολο των χαρακτηριστικών και δεν κάνει εξαντλητική αναζήτηση όπως ο αλγόριθμος Boosting.
- Μπορεί να χρησιμοποιηθεί για ομαδοποίηση.
- Επιτρέπει τη δημιουργία παραλλαγών της βασικής τεχνικής ως προς την κατασκευή του μοντέλου ταξινόμησης π.χ. χρήση διαφορετικών τεχνικών διαμέρισης των παραδειγμάτων των ενδιάμεσων κόμβων.

Τα τυχαία δάση παρουσιάζουν όμως και κάποια σημαντικά μειονεκτήματα ως προς την εφαρμογή τους τα οποία αναφέρονται συνοπτικά παρακάτω:

- Υψηλό υπολογιστικό κόστος.
- Υπάρχει σημαντικό πλήθος ελεύθερων παραμέτρων τις οποίες πρέπει να προσδιορίσει ο χρήστης π.χ. πλήθος δέντρων, βαθμός κόμβων, πλήθος παραδειγμάτων εκπαίδευσης, συνθήκη τερματισμού διαμέρισης των κόμβων.
- Για την επέκταση ενός μοντέλου με στόχο την εισαγωγή μιας ακόμα κατηγορίας απαιτείται η κατασκευή του μοντέλου από την αρχή.
- Κάθε νέο παράδειγμα πρέπει να διασχίσει όλα τα δέντρα του δάσους για την εκτίμηση της κατηγορίας του.

Ταξινομητής SVM

Ο **κλασσικός ταξινομητής SVM** (Smola , 2004) που χρησιμοποιήσαμε ονομάζεται **γραμμικού πυρήνα**. Χρησιμοποιεί τη βιβλιοθήκη libsvm για βελτιστοποίηση.

```
from sklearn.svm import SVC
classifier = SVC(kernel='linear')
```

Η εκπαίδευση και αξιολόγηση των ταξινομητών είναι πολύ απλή χάρη στις έτοιμες (built-in) μεθόδους fit και score.

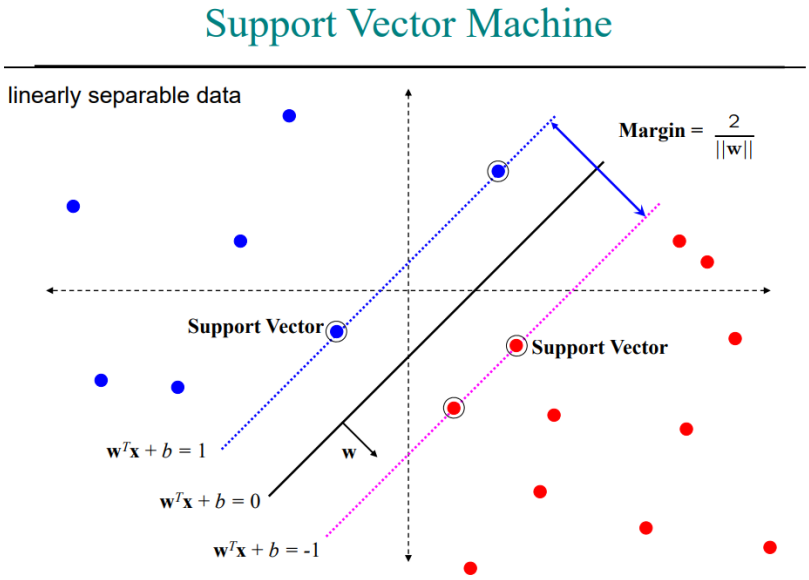
Τα **πλεονεκτήματα** των support vector machines περιλαμβάνουν:

- Πολύ καλή απόδοση σε χώρους υψηλών διαστάσεων.
- Είναι αποδοτικά ακόμα και αν ο αριθμός των διαστάσεων είναι μεγαλύτερος από των αριθμό των δειγμάτων.
- Χρησιμοποιούν ένα υποσύνολο των δειγμάτων εκπαίδευσης (που ονομάζονται support vectors) επομένως είναι και αποδοτικά ως προς την χρήση της μνήμης.
- Διαφορετικές συναρτήσεις kernel functions μπορούν να χρησιμοποιηθούν, ακόμα και custom kernels.

Τα **μειονεκτήματα** τους περιλαμβάνουν:

- Αν ο αριθμός των χαρακτηριστικών είναι αρκετά μεγαλύτερος του αριθμού των δειγμάτων πρέπει να δωθεί προσοχή στην επιλογή του κατάλληλου kernel function για να αποθευχθεί το over-fitting .
- Τα SVMs δεν παρέχουν εκτιμήσεις πιθανότητας, αυτές πρέπει να υπολογιστούν μέσα από ένα πιο δαπανηρό five-fold cross-validation.

Οι Μηχανές Διανυσμάτων Υποστήριξης αποτελούν μία από τις πιο ακριβείς προσεγγίσεις διακρινουσών συναρτήσεων για ταξινόμηση. Ο ταξινομητής SVM προσπαθεί να βρει ένα υπερεπίπεδο απόφασης το οποίο να διαχωρίζει το σύνολο των παραδειγμάτων εκπαίδευσης με τέτοιο τρόπο ώστε τα παραδείγματα που ανήκουν στην ίδια κατηγορία να είναι στη ίδια πλευρά του υπερεπιπέδου. Μεταξύ όλων των πιθανών υπερεπιπέδων αναζητά εκείνο για το οποίο η απόσταση από το κοντινότερο παράδειγμα είναι μέγιστη, δηλ. αναζητά υπερεπίπεδο μέγιστου περιθωρίου (maximal margin hyperplane).



Σχήμα από http://www.robots.ox.ac.uk/~az/lectures/ml/lect2.pdf

Το υπερεπίπεδο απόφασης για ένα σύνολο N παραδειγμάτων εκπαίδευσης

$$\{[x_i, y_i]\}_{i=1}^{N}$$

και δύο κατηγορίες $y_i \in \{-1,1\}$ ορίζεται ως εξής w• x + b = 0 [15] , όπου w και b είναι οι παράμετροι του μοντέλου και το $x_i = \{x_{i1}, x_{i2}, x_{i3}, \dots, x_{id})$ αντιστοιχεί στο σύνολο χαρακτηριστικών του ι-οστού παραδείγματος εκπαίδευσης. Το περιθώριο του υπερεπιπέδου υπολογίζεται ως εξής:

$$d = \frac{2}{||w||}$$

Γενικά, ο ταξινομητής SVM είναι μια μέθοδος βελτιστοποίησης πολλαπλών κριτηρίων [16,17]:

- Μεγιστοποιεί την απόσταση μεταξύ των διανυσμάτων υποστήριξης (support vectors) και ενός υπερεπιπέδου απόφασης. Τα διανύσματα υποστήριξης είναι τα παραδείγματα εκπαίδευσης που βρίσκονται πιο κοντά στο υπερεπίπεδο και καθορίζουν το περιθώριό του (margin). Η μέθοδος μεγιστοποιεί το

περιθώριο, το οποίο αποτελεί μέτρο της γενικευτικής ικανότητας του ταξινομητή, καθώς ταξινομητές που παράγουν όρια απόφασης με μικρά περιθώρια είναι ευάλωτοι σε φαινόμενα υπερεκπαίδευσης (model overfitting).

$min \frac{||w||^2}{2}$ y (w • x + b) ≥ 1 i i , i = 2,1 ,...,N

- Στα περισσότερα προβλήματα ταξινόμησης τα παραδείγματα εκπαίδευσης δεν είναι γραμμικά διαχωρίσιμα. Σ' αυτή την περίπτωση, τα SVM απεικονίζουν το αρχικό σύνολο χαρακτηριστικών σε ένα σύνολο μεγαλύτερης διάστασης χρησιμοποιώντας μια συνάρτηση Φ(x).

$min \frac{||w||^2}{2}$ y (w • Φ(x) + b) ≥ 1 i i , i = 2,1 ,...,N

Το πρόβλημα βελτιστοποίησης επιλύεται με την χρήση πολλαπλασιαστών Lagrange και καταλήγει στο ακόλουθο δυικό πρόβλημα βελτιστοποίησης χωρίς περιορισμούς [18].

$$L_D = \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i,j} \lambda_i \lambda_j y_i y_j \Phi(x_i) \bullet \Phi(x_j)$$

Η συνάρτηση απόφασης για κάθε παράδειγμα ελέγχου x μπορεί να εκφραστεί ως εξής:

$$f(x) = sign(\sum_{i=1}^{N} \lambda_i y_i \Phi(x_i) \bullet \Phi(x) + b)$$

Το εσωτερικό γινόμενο στον χώρο Φ(x) καλείται συνάρτηση πυρήνας (kernel function).

- Επειδή το πρόβλημα της ταξινόμησης μπορεί να επιλυθεί χρησιμοποιώντας σύνθετες διακρίνουσες συναρτήσεις, τα SVM επιδιώκουν να ελαχιστοποιούν το μέγεθος της λύσης (το άθροισμα των βαρών των χαρακτηριστικών) [19].
- Παρόλο που ο μετασχηματισμός των χαρακτηριστικών είναι μη γραμμικός, κάποια από τα παραδείγματα δεν ταξινομούνται σωστά. Τα SVM τείνουν να μειώνουν το πλήθος των εσφαλμένων ταξινομήσεων. Σ' αυτή την περίπτωση αναζητάνε ένα υπερεπίπεδο «χαλαρού» περιθωρίου, δηλ. μια επιφάνεια απόφασης η οποία διαχωρίζει τα δεδομένα εκπαίδευσης κάνοντας τα λιγότερα λάθη. Οι βοηθητικές μεταβλητές ξi≥0 (slack variables) αποτελούν μια εκτίμηση του σφάλματος του επιπέδου απόφασης για το ι-οστό παράδειγμα

εκπαίδευσης και η μεταβλητή C καθορίζει πόσο αυστηροί είμαστε με τα λάθη [15,16,19].

$$\min\left(\frac{\|w\|^2}{2}+C\sum_{i=1}^{N}\xi_i\right)$$

$$\text{subject to} \quad \xi_i \geq 0 \text{ and } y_i\left(w\bullet\Phi(x_i)+b\right)\geq 1-\xi_i, \quad i=1,2,...,N$$

Τα SVM χρησιμοποιούν διάφορους αλγόριθμους βελτιστοποίησης ώστε να ικανοποιούνται όλα τα κριτήρια ταυτόχρονα, παρόλα αυτά χρειάζεται να σταθμίσουμε κατάλληλα όλα τα παραπάνω κριτήρια ώστε να πετύχουμε την καλύτερη ταξινόμηση. Για προβλήματα ταξινόμησης πολλών κατηγοριών υπάρχουν δύο προσεγγίσεις:

(α) ένας-εναντίον-όλων (one-against-all) και

(β) ένας-εναντίον-ένα (one-against-one). Στην πρώτη προσέγγιση για ένα σύνολο K κατηγοριών C={C1, C2, …, CK} απαιτείται η εκπαίδευση K δυαδικών SVM. Κάθε SVM υπολογίζει ένα υπερεπίπεδο απόφασης το οποίο διαχωρίζει τα παραδείγματα της κατηγορίας i από τα παραδείγματα των υπολοίπων K-1 κατηγοριών. Ένα παράδειγμα ελέγχου x ανατίθεται στην κατηγορία Ci αν η έξοδος του ταξινομητή i είναι μεγαλύτερη από τις εξόδους των υπόλοιπων SVM.

Στη δεύτερη προσέγγιση για ένα σύνολο K κατηγοριών C={C1, C2, …, CK} απαιτείται η εκπαίδευση K(K-1)/2 δυαδικών SVM. Οι ταξινομητές υπολογίζουν ένα υπερεπίπεδο απόφασης το οποίο διαχωρίζει τα παραδείγματα της κατηγορίας i από τα παραδείγματα κάθε μίας από τις υπόλοιπες κατηγορίες. Κάθε παράδειγμα ελέγχου x ανατίθεται σε μια από τις κατηγορίες με βάση το αποτέλεσμα μιας «ψηφοφορίας», δηλ. ανατίθεται στην κατηγορία η οποία εμφανίζεται πιο πολλές φορές στην έξοδο των δυαδικών ταξινομητών.

Round robin ταξινόμηση (classification)

Το πρόβλημα της αξιοποίησης ενός labeled coprus για την εκμάθηση μοντέλων για ανάλυση συναισθημάτων έχει προσελκύσει μεγάλο ενδιαφέρον τα τελευταία χρόνια [25, 24,14, 1]. Ένα **κοινό χαρακτηριστικό** σχεδόν όλων αυτών των εργασιών ήταν η τάση να οριστεί η εργασία ως **πρόβλημα δύο κατηγοριών**: θετικό έναντι αρνητικού. Σε όλα σχεδόν τα προβλήματα της πολικότητας, συμπεριλαμβανομένης της ανάλυσης των αισθήσεων, υπάρχουν τουλάχιστον τρεις κατηγορίες που πρέπει να διακρίνονται: θετικές, αρνητικές και ουδέτερες. Βέβαια, σε καμία περίπτωση κάθε σχόλιο για ένα προϊόν ή εμπειρία εκφράζει καθαρά θετικό ή αρνητικό συναίσθημα. Κάποια (σε πολλές περιπτώσεις τα περισσότερα) σχόλια μπορεί να αναφέρουν αντικειμενικά γεγονότα χωρίς να εκφράζουν κάποιο συναίσθημα, ενώ άλλα μπορεί να εκφράζουν μεικτό ή αντιφατικό συναίσθημα. Οι ερευνητές γνωρίζουν βεβαίως την ύπαρξη ουδέτερων εγγράφων. Το σκεπτικό για την αγνόησή τους ήταν η εξάρτηση από **δύο σιωπηρές υποθέσεις**:

• Η επίλυση του δυαδικού θετικού και αρνητικού προβλήματος, λύνει αυτόματα το πρόβλημα των τριών κατηγοριών, αφού τα ουδέτερα έγγραφα θα βρίσκονται απλά κοντά στο όριο του δυαδικού μοντέλου
• Υπάρχει λιγότερη εκμάθηση από ουδέτερα έγγραφα παρά από έγγραφα με σαφώς καθορισμένο συναίσθημα

Η round robin ταξινόμηση [7,9,10] (π.χ. ταξινόμηση κατά ζεύγη), μια τεχνική για τον χειρισμό προβλημάτων πολλαπλών κατηγοριών με δυαδικούς ταξινομητές, με την εκμάθηση ενός ταξινομητή για κάθε ζεύγος κλάσεων. Αν και τα προβλήματα του πραγματικού κόσμου συχνά έχουν πολλαπλές κλάσεις, πολλοί αλγόριθμοι εκμάθησης είναι εγγενώς δυαδικοί, δηλ. είναι ικανοί να διακρίνουν μόνο μεταξύ δύο τάξεων. Οι λόγοι για αυτό μπορεί να είναι περιορισμοί που επιβάλλονται από τη γλώσσα της υπόθεσης (π.χ. linear discriminants ή support vector machines), την αρχιτεκτονική μάθησης (π.χ. νευρωνικά δίκτυα με κόμβους εξόδου) ή το πλαίσιο μάθησης (π.χ. προς την έννοια της μάθησης, δηλαδή, το πρόβλημα της εκμάθησης μιας έννοιας από θετικά και αρνητικά παραδείγματα).

Υπάρχουν δύο βασικές προσεγγίσεις για την εφαρμογή τέτοιων αλγορίθμων σε προβλήματα πολλαπλών τάξεων: μία προσέγγιση είναι η γενίκευση του αλγορίθμου - όπως έχει γίνει για support vector machines [8, 11 ,9,10] και η **άλλη** είναι η χρήση τεχνικών διμερισμού (binarization) κλάσης, οι οποίες μειώνουν το πρόβλημα πολλών τάξεων σε μια σειρά δυαδικών προβλημάτων. Μια από τις πιο συνηθισμένες προσεγγίσεις της κατηγοριοποίησης είναι η εκμάθηση ξεχωριστών περιγραφών ιδεών

για κάθε μεμονωμένη κατηγορία, δηλ. h δημιουργία μιας σειράς προβλημάτων, μία για κάθε τάξη, όπου όλα τα παραδείγματα αυτής της τάξης θεωρούνται θετικά παραδείγματα, ενώ όλα τα άλλα παραδείγματα θεωρούνται αρνητικά.

Η κατά ζεύξη ταξινόμηση (pairwise classification) είναι μια εναλλακτική τεχνική διμερισμού κατηγορίας, η οποία έχει προσελκύσει πρόσφατα κάποια προσοχή στα δίκτυα νευρωνικών δικτύων και στα support vector machines. Η βασική του ιδέα είναι να μειώσει ένα πρόβλημα πολλαπλών τάξεων σε πολλαπλά προβλήματα δύο κατηγοριών, μαθαίνοντας έναν ταξινομητή για κάθε ζευγάρι τάξεων, χρησιμοποιώντας μόνο παραδείγματα εκπαίδευσης για αυτές τις δύο κατηγορίες και αγνοώντας όλα τα άλλα.

Πολλοί αλγόριθμοι μηχανικής μάθησης είναι εγγενώς σχεδιασμένοι για δυαδικά (δύο τάξεων) προβλήματα λήψης αποφάσεων. Σημαντικά παραδείγματα είναι τα perceptrons, τα support vector machines, ο αρχικός αλγόριθμος AdaBoost [29] και η separate-and-conquer rule learning. Επιπλέον, όλοι οι αλγόριθμοι παλινδρόμησης μπορούν, κατ' αρχήν, να χρησιμοποιηθούν για προβλήματα δυαδικής απόφασης, αλλά όχι για προβλήματα πολλαπλών τάξεων (εκτός αν αν οι τάξεις μπορούν να έχουν διάταξη). Από την άλλη πλευρά, τα προβλήματα του πραγματικού κόσμου συχνά έχουν πολλαπλές τάξεις. Ευτυχώς, υπάρχουν αρκετές απλές τεχνικές για την μετατροπή των προβλημάτων πολλών τάξεων σε ένα σύνολο δυαδικών
προβλημάτων. Τέτοιες τεχνικές ονομάζονται τεχνικές διμερισμού κατηγορίας.

Ορισμός 1 (διάρθρωση κλάσης, αποκωδικοποίηση, βασικός μαθητής) Μια διάρθρωση της κατηγορίας είναι μια χαρτογράφηση ενός μαθησιακού προβλήματος πολλαπλών τάξεων σε αρκετά μαθησιακά προβλήματα δύο τάξεων με τρόπο που επιτρέπει μια λογική αποκωδικοποίηση της πρόβλεψης, δηλ. επιτρέπει την εξαγωγή μια πρόβλεψης για το πρόβλημα πολλών τάξεων από τις προβλέψεις του συνόλου ταξινομητών δύο τάξεων. Ο αλγόριθμος μάθησης που χρησιμοποιείται για την επίλυση των δύο κατηγοριών προβλημάτων ονομάζεται βασικός μαθητής (base learner). Η πιο δημοφιλής τεχνική διμερισμού μεταξύ κατηγοριών είναι η unordered ή one-against-all class binarization, όπου παίρνουμε κάθε τάξη με τη σειρά και μαθαίνουμε δυαδικές έννοιες που διακρίνουν αυτή την τάξη από όλες τις άλλες τάξεις. Έχει προταθεί ανεξάρτητα για την εκμάθηση κανόνων ([26]), για νευρωνικά δίκτυα [30], και support vector machines [31].

Ορισμός 2 (unordered/one-against-all class binarization) Η unordered class binarization μετασχηματίζει ένα πρόβλημα c κατηγοριών σε c προβλήματα κατηγορίας. Αυτά κατασκευάζονται χρησιμοποιώντας τα παραδείγματα της κλάσης i ως θετικά παραδείγματα και τα παραδείγματα των κλάσεων j (j = 1 ... c, j = i) ως αρνητικά παραδείγματα. Το όνομα "unordered" προέρχεται από τους Clark και Boswell

[26], οι οποίοι πρότειναν αυτή την προσέγγιση ως εναλλακτική λύση στην προσέγγιση μάθησης των αποφάσεων που χρησιμοποιήθηκε αρχικά στο CN2 (Clark και Niblett [27], Rivest, [28]). Καθώς η κύρια μας ανησυχία είναι η μάθηση κατά κανόνα, θα επικεντρωθούμε κυρίως στην ορολογία που χρησιμοποιείται εκεί, αλλά θα την περιγράψουμε επίσης περιστασιακά ως one-against-all, το οποίο φαίνεται να κυριαρχεί σε άλλους τομείς.

Ορισμός 3 (ordered class binarization) Η ordered class binarization μετασχηματίζει ένα πρόβλημα c κατηγοριών σε c - 1 δυαδικά προβλήματα. Αυτά κατασκευάζονται χρησιμοποιώντας τα παραδείγματα της κλάσης i (i = 1 ... c - 1) ως τα θετικά παραδείγματα και τα παραδείγματα των κλάσεων j> i ως τα αρνητικά παραδείγματα. Σημειώστε ότι η ordered class binarization των τάξεων επιβάλλει μια σειρά στους επαγόμενους ταξινομητές, τους οποίους πρέπει να ακολουθήσετε κατά τον χρόνο ταξινόμησης: ο ταξινομητής που έμαθε για τη διάκριση κατηγορίας 1 από τις κλάσεις 2 ... c πρέπει να καλείται πρώτα. Εάν αυτός ο ταξινομητής ταξινομεί το παράδειγμα ως ανήκει στην κλάση 1, δεν καλείται κανένας άλλος ταξινομητής. εάν όχι, το παράδειγμα μεταβιβάζεται στον επόμενο ταξινομητή. Αντίθετα, η μη εξουσιοδοτημένη διάρθρωση των κλάσεων πρέπει να καλέσει καθένα από τους δυαδικούς ταξινομητές της και να απαιτήσει κάποιο εξωτερικό κριτήριο για το συνδυασμό των μεμονωμένων προβλέψεων σε μια τελική πρόβλεψη. Οι τυπικοί κανόνες αποκωδικοποίησης ψηφίζουν τις προβλέψεις των μεμονωμένων ταξινομητών, ενδεχομένως λαμβάνοντας υπόψη την εμπιστοσύνη των προβλέψεων.

Round Robin Classification Σε αυτή την ενότητα θα συζητήσουμε μια πιο σύνθετη διαδικασία ταξινόμησης των κλάσεων, τον ταξινομητή ανά ζεύγη. Η βασική ιδέα είναι πολύ απλή, δηλαδή να μάθουμε έναν ταξινομητή για κάθε ζευγάρι τάξεων. Σε αναλογία με τα Round Robin αθλητικά τουρνουά, στα οποία ο κάθε συμμετέχων έχει αντιστοιχιστεί με κάθε άλλο συμμετέχοντα, ονομάζουμε αυτή τη διαδικασία συσχέτιση binarization robin.

Ορισμός 4 (round robin/pairwise binarization) Η round robin ή pairwise class binarization μετατρέπει ένα πρόβλημα κατηγορίας c σε c(c-1) / 2 προβλήματα δύο κατηγοριών, ένα για κάθε σύνολο κλάσεων {i, j}, i = 1 ... c-1, j = i + 1 ... c. Ο δυαδικός ταξινομητής για το πρόβλημα εκπαιδεύεται με παραδείγματα κλάσεων i και j, ενώ τα παραδείγματα των κλάσεων k = i, j αγνοούνται για αυτό το πρόβλημα.

Θεωρητικές εκτιμήσεις

Σε αυτό το κεφάλαιο θα δούμε ότι αν και η (single) round robin classification μετατρέπει ένα μαθησιακό πρόβλημα κατηγορίας c σε c (c - 1) / 2 προβλήματα δύο κατηγοριών, η συνολική προσπάθεια κατάρτισης είναι γραμμική μόνο στον αριθμό των κατηγοριών και τα πειράματα είναι λιγότερα από την προσπάθεια που απαιτείται για unordered binarization. Τα πειραματικά μας αποτελέσματα δείχνουν ότι σε

σύγκριση με conventional ordered ή unordered binarization, η round robin προσέγγιση μπορεί να αποφέρει σημαντικά κέρδη στην ακρίβεια χωρίς να κινδυνεύει από κακή απόδοση.

Υλοποίηση και Αποτελέσματα

Δεδομένα

Για την υλοποίηση χρησιμοποιήθηκαν δεδομένα από διαφορετικές πηγές για να μελετήσουμε διαφορετικές μεθόδους ταξινόμησης. Τα δεδομένα μας αποτελούνται από tweets τα οποία έχουν επισημανθεί ώς θετικά, αρνητικά η ουδέτερα από τους διοργανωτές του ετήσιου διαγωνισμού **SemEval** [13], καθώς και δεδομένα που αποκτήσαμε από μία ψηφιακή βιβλιοθήκη που ονομάζεται **Internet Archive** (https://archive.org/) η οποία παρέχει έναν μεγάλο όγκο δεδομένων από το twitter που εκτείνονται χρονολογικά από το 2013 μέχρι το 2016. Για τη συγκεκριμένη εργασία, και λόγω των περιορισμών σε πόρους που επιβάλει ένας προσωπικός υπολογιστής, από την συλλογή του Internet Archive κρατήσαμε **50 εκατ. μοναδικά tweets**. (Να σημειώσουμε εδώ ότι η βιβλιοθήκη αυτή από tweets φτάνει περίπου στα 250 εκ. μοναδικά μηνύματα).

Το σύνολο δεδομένων που διατίθενται από το διαγωνισμό SemEval [13], αποτελείται από 20,811 tweets εκ των οποίων τα 10,408 είναι θετικά και τα 10,403 αρνητικά.

Τα tweets που αποκτήσαμε από το Internet Archive δεν είναι αναγνωρισμένα ως προς το συναίσθημα που περιέχουν (unlabeled). Χρησιμοποιούνται όμως για να εκπαιδεύσουμε το μοντέλο word2vec που περιγράψαμε σε προηγούμενη ενότητα, ώστε να εξάγουμε στη συνέχεια χαρακτηριστικά για τις προτάσεις με τον τρόπο που θα περιγράψουμε στην συνέχεια.

Από τα tweets που έχουμε στη διάθεσή μας από το SemEval [13] εξάγονται χαρακτηριστικά και έτσι κάθε tweet αντιστοιχίζεται σε ένα σύνολο χαρακτηριστικών και μία ετικέτα θετικού ή αρνητικού συναισθήματος. Το σύνολο των tweets χωρίζεται σε δεδομένα εκπαίδευσης (training data) και δεδομένα δοκιμής (testing data).

Για τις ανάγκες της παρούσας διπλωματικής εργασίας χρησιμοποιήθηκαν τα ακόλουθα σύνολα δεδομένων που παρέχει το SemEval για την επίλυση του Subtask A. *Given a message, determine whether it expresses a positive, a negative or a neutral sentiment.*

- Τα training, development και development-test δεδομένα του 2016 (SemEval-2016 task 4 train, dev and devtest data).
- Τα training δεδομένα του 2013 (SemEval-2013 task 2 train data).
- Τα development δεδομένα του 2013 (SemEval-2013 task 2 dev data).

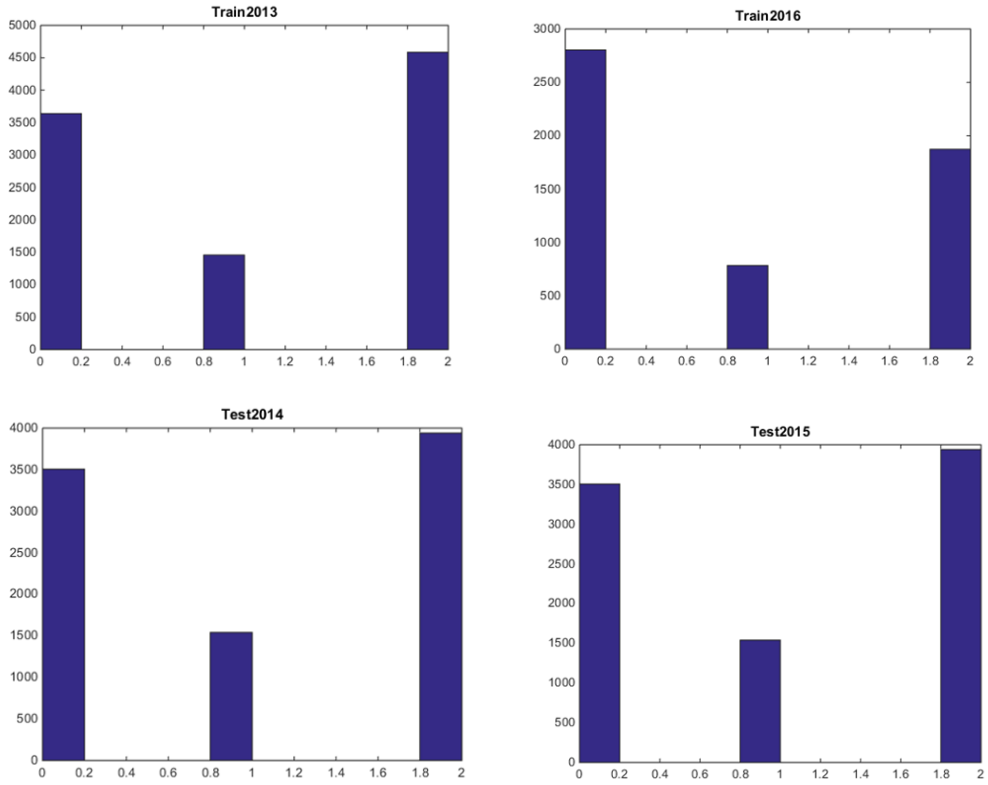
- Τα development-test δεδομένα του 2013 (SemEval-2013 task 2 devtest data).
- Τα development-test δεδομένα του 2014 (SemEval-2014 task 9 devtest data).

Datasets	positive	negative	neutral	Total
Twitter2013-train	5,895	3,131	471	9,497
Twitter2013-dev	648	430	57	1,135
Twitter2016-train	3,094	863	2,043	6,000
Twitter2016-dev	844	765	391	2,000
Twitter2016-devtest	994	681	325	2,000
Twitter2013-test	2,734	1,541	160	4,435
SMS2013-test	1,071	1,104	159	2,334
Twitter2014-test	1,807	578	88	2,473
Twitter2014-sarcasm	82	37	5	124
LiveJournal2014-test	660	511	144	1,315
Twitter2015-test	1,899	1,008	190	3,097
Twitter2016-test	7,059	10,342	3,231	20,632

Data statistics (Subtask A) Τα δεδομένα train, dev, devtest χρησιμοποιήθηκαν ως δεδομένα για εκπαίδευση των αλγορίθμων που παρουσιάζονται σε αυτή την εργασία.

Τα δεδομένα παρέχονται με τη μορφή αρχείων κειμένου που περιέχουν τις μοναδικές ταυτότητες (IDs) των tweets και τα labels για κάθε tweet. Με τη βοήθεια των αρχείων αυτών, των python scripts που παρέχει το SemEval και ενός Twitter account κατεβάσαμε τα δεδομένα απευθείας από το Twitter. Από το σύνολο των tweets αφαιρέθηκαν τα μη διαθέσιμα (Not Available).

SEMEVAL DATASETS - HISTOGRAMS – We want to show class imbalance
0:positive, 1:negative, 2:neutral



Οι αλγόριθμοι μηχανικής μάθησης εκπαιδεύονται στα χαρακτηριστικά και τις ετικέτες των tweets εκπαίδευσης και αξιολογούνται σε αυτά των tweets δοκιμής. Η μετρική επίδοσης που χρησιμοποιείται είναι το F-score των θετικών και αρνητικών κλάσεων . Τυπικά ορίζουμε έναν πίνακα απόδοσης λαθών:

		Gold Standard		
		POSITIVE	NEUTRAL	NEGATIVE
Predicted	POSITIVE	PP	PU	PN
	NEUTRAL	UP	UU	UN
	NEGATIVE	NP	NU	NN

Πίνακας απόδοσης λαθών (confusion matrix)

Κάθε σειρά του πίνακα αντιπροσωπεύει τις περιπτώσεις σε μια προβλεπόμενη κλάση ενώ κάθε στήλη αντιπροσωπεύει τις περιπτώσεις σε μια πραγματική κλάση (ή αντίστροφα)

Και η μετρική F1 στην συνέχεια ορίζεται ως εξής:

$$F_1^{PN} = \frac{F_1^{Pos} + F_1^{Neg}}{2}$$

$$F_1^{Pos} = \frac{2\pi^{Pos}\rho^{Pos}}{\pi^{Pos} + \rho^{Pos}} \quad (\alpha), \quad \rho^{Pos} = \frac{PP}{PP + PU + NP} \quad (\beta), \quad \pi^{Pos} = \frac{PP}{PP + PU + PN}$$

$$F_1^{Neg} = \frac{2\pi^{Neg}\rho^{Neg}}{\pi^{Neg} + \rho^{Neg}} \quad (\alpha), \quad \rho^{Neg} = \frac{NN}{NN + NU + PN} \quad (\beta), \quad \pi^{Neg} = \frac{NN}{NN + NU + NP}$$

Ανάλυση συναισθήματος - πειραματικές μέθοδοι

Αφού εξάγουμε όλα τα χαρακτηριστικά από ένα tweet το επόμενο βήμα είναι να δοκιμάσουμε την την απόδοσή τους στα προβλήματα που θέτει ο διαγωνισμός SemEval. **Στόχος μας δεν είναι να ξεπεράσουμε την μέγιστη απόδοση των νικητών του διαγωνισμού αλλά να μελετήσουμε την ποιότητα των χαρακτηριστικών, τα οποία άλλωστε χρησιμοποιούνται αρκετά συχνά από τις ομάδες που συμμετέχουν στον εν λόγω διαγωνισμό**. Για το λόγο αυτό θα χρησιμοποιήσουμε τις παρακάτω μεθόδους ταξινόμησης συνδυάζοντας τες με τα ανάλογα χαρακτηριστικά.

Για την υλοποίηση των παρακάνω 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.

1. Μέθοδος ταξινόμησης με Random Forest και μεταβλητό μέγεθος από διανύσματα λέξεων (word embeddings).
2. Μέθοδος ταξινόμησης με SVM ταξινομητή και μεταβλητό μέγεθος από διανύσματα λέξεων (word embeddings).
3. Μέθοδος ταξινόμησης με Random Forest, με μεταβλητό μέγεθος από διανύσματα λέξεων (word embeddings) και προσθήκη ενός επιπλέον χαρακτηριστικού συναισθήματος (valence score).
4. Μέθοδος ταξινόμησης με SVM ταξινομητή, με μεταβλητό μέγεθος από διανύσματα λέξεων (word embeddings) και προσθήκη ενός επιπλέον χαρακτηριστικού συνασθήματος (valence score).
5. Μέθοδος ταξινόμησης με Random Forest σε πολλαπλά στάδια (stacking) και μεταβλητό μέγεθος από διανύσματα λέξεων (word embeddings).

6. Μέθοδος ταξινόμησης με SVM ταξινομητή σε πολλαπλά στάδια (stacking) και μεταβλητό μέγεθος από διανύσματα λέξεων (word embeddings).
7. **Για να αποτιμήσουμε τη συνεισφορά των χαρακτηριστικών μορφολογίας που έχουμε επιλέξει όταν συνδυασμούν με τα word embeddings στο πρόβλημα κατηγοριοποίησης που μελετάμε, εκτελέσαμε μία leave-one out τεχνική εξαιρώντας κάθε φορά από τον ταξινομητή μία ομάδα χαρακτηριστικών. Για αυτά τα πειράματα χρησιμοποιήθηκε το εργαλείο WEKA και ένας NaiveBayesTree ταξινομητής.**

Αποτελέσματα

Η απόδοση υπολογίζεται με το F1 score όπως ακριβώς και στον διαγωνισμό (macroaveraged F-score of the positive and negative classes).

word2vec size	Random forest: Single step system						
	LiveJournal2014	SMS2013	Twitter2013	Twitter2014	Twitter2014Sarcasm	Twitter2015	Twitter2016
BEST	0.741	0.641	0.813	0.759	0.566	0.671	0.633
300	0.419	0.291	0.327	0.375	0.279	0.372	0.364
200	0.459	0.360	0.363	0.383	0.335	0.357	0.389
100	0.466	0.343	0.376	0.403	0.367	0.388	0.429
50	0.464	0.338	0.406	0.432	0.429	0.421	0.477

Πίνακας 1: Αποτελέσματα με την μέθοδο Random Forest για διαφορετικές διαστάσεις διανυσμάτων λέξεων.

word2vec size	**Stacking method and random forest**, the same evaluation metric is used and we are also testing the system with different word2vec sizes.						
	LiveJournal2014	SMS2013	Twitter2013	Twitter2014	Twitter2014Sarcasm	Twitter2015	Twitter2016
BEST	0.741	0.641	0.813	0.759	0.566	0.671	0.633

300	0.544	0.422	0.44	0.469	0.364	0.502	0.505
200	0.536	0.462	0.455	0.513	0.393	0.499	0.504
100	0.566	0.448	0.474	0.508	0.387	0.541	0.518
50	0.534	0.441	0.49	0.514	0.462	0.505	0.522

Πίνακας 2: Αποτελέσματα με την μέθοδο Random Forest και 2-step μέθοδο , για διαφορετικές διαστάσεις διανυσμάτων λέξεων.

word2vec size	SVM classifier-simple-no extra features-no extra data						
	LiveJournal2014	SMS2013	Twitter2013	Twitter2014	Twitter2014Sarcasm	Twitter2015	Twitter2016
BEST	0.741	0.641	0.813	0.759	0.566	0.671	0.633
300	0.675	0.588	0.512	0.579	0.461	0.561	0.563
200	0.651	0.581	0.514	0.572	0.503	0.563	0.555
100	0.623	0.533	0.486	0.559	0.487	0.538	0.542
50	0.575	0.491	0.426	0.491	0.480	0.474	0.484

Πίνακας 3: Αποτελέσματα με την μέθοδο SVM για διαφορετικές διαστάσεις διανυσμάτων λέξεων.

word2vec size	SVM classifier (linear)+ features from AFFIN,NRC,NRCTAG lexica						
	LiveJournal2014	SMS2013	Twitter2013	Twitter2014	Twitter2014Sarcasm	Twitter2015	Twitter2016
BEST	0.741	0.641	0.813	0.759	0.566	0.671	0.633
300	0.589	0.471	0.575	0.643	0.614	0.592	0.552
200	0.585	0.470	0.576	0.646	0.611	0.590	0.546
100	0.579	0.464	0.583	0.641	0.569	0.589	0.542
50	0.575	0.461	0.560	0.630	0.555	0.565	0.533

Πίνακας 4: Αποτελέσματα με την μέθοδο SVM για διαφορετικές διαστάσεις διανυσμάτων λέξεων και προσθήκη έξτρα χαρακτηριστικών.

word2vec size	stacking-SVM classifier- linear						
	LiveJournal2014	SMS2013	Twitter2013	Twitter2014	Twitter2014Sarcasm	Twitter2015	Twitter2016
BEST	0.741	0.641	0.813	0.759	0.566	0.671	0.633
300	0.615	0.508	0.545	0.594	0.503	0.563	0.529
200	0.598	0.518	0.520	0.582	0.443	0.550	0.525
100	0.570	0.495	0.505	0.559	0.526	0.528	0.509
50	0.527	0.442	0.476	0.510	0.475	0.516	0.486

Πίνακας 4: Αποτελέσματα με την μέθοδο SVM 2-step, για διαφορετικές διαστάσεις διανυσμάτων λέξεων.

BigramCollocation function of NLTK - parameter n here is the number of significant bigrams to keep	NaiveBayesTree classifier						
	LiveJournal2014	SMS2013	Twitter2013	Twitter2014	Twitter2014Sarcasm	Twitter2015	Twitter2016
BEST	0.741	0.641	0.813	0.759	0.566	0.671	0.633
n=100	0.547	0.426	0.504	0.569	0.456	0.619	0.517
n=200	0.539	0.414	0.514	0.555	0.455	0.616	0.512
n=300	0.540	0.407	0.516	0.577	0.425	0.617	0.516

Πίνακας 5: Αποτελέσματα με την μέθοδο BigramCollocation για διαφορετικό αριθμό από bigrams.

Αποτελέσματα με leave-one-out μέθοδο στο WEKA.

F1 SCORE	LiveJournal2014	SMS2013	Twitter2013	Twitter2014	Twitter2014Sarcasm	Twitter2015	Twitter2016
BEST SCORE ACHIEVED per dataset (amongst all teams)	**0.741**	**0.641**	**0.813**	**0.759**	**0.566**	**0.671**	**0.633**
Winner of 2016 scores	**0.695**	**0.637**	**0.7**	**0.716**	**0.566**	**0.671**	**0.633**
ALL features	0.686	0.61	0.68	0.696	0.494	0.594	0.588
ALL features - remove morphology	0.688	0.599	0.673	0.694	0.478	0.591	0.585
ALL features - remove hashtags	0.684	0.613	0.679	0.695	0.474	0.591	0.587
ALL features - remove w2vec	0.594	0.466	0.523	0.543	0.47	0.526	0.513
ALL features - remove subjectivity	0.688	0.602	0.673	0.689	0.497	0.587	0.583
ALL features - remove lexica	0.646	0.543	0.636	0.647	0.509	0.562	0.563
ALL features - remove affin	0.696	0.63	0.664	0.69	0.49	0.588	0.573

ALL features - remove nrc	0.699	0.613	0.68	0.698	0.521	0.59	0.586
ALL features - remove nrctag	0.694	0.61	0.674	0.695	0.482	0.589	0.587

Πίνακας 6: Αξιολόγηση των χαρακτηριστικών

Συμπεράσματα

Από τα αποτελέσματα των πειραμάτων μας είναι ξεκάθαρο ότι **η χρήση περισσότερων χαρακτηριστικών**, των **μορφολογικών** αλλά και των **συναισθηματικών λεξικών βελτίωσε τα αποτελέσματα** ανεξάρτητα από τον ταξινομητή που χρησιμοποιήθηκε.

Η round robin μέθοδος ταξινόμησης βελτίωσε σημαντικά την απόδοση συγκρινόμενη με την κλασσική μέθοδο ταξινόμησης με Random Forest. Παρόλα αυτά ο **SVM** είναι **καλύτερος** συγκρινόμενος με τις προαναφερόμενες μεθόδους.

Αν θέλαμε να ταξινομήσουμε τα χαρακτηριστικά που μελετήσαμε, ως προς το πόσο συνεισφέρουν στο τελικό αποτέλεσμα, είναι ξεκάθαρο από τον πίνακα 6 παραπάνω, ότι το πιο σημαντικό ρόλο στην σωστή ταξινόμηση έχουν τα **διανύσματα λέξεων**, καθώς βλέπουμε ότι ρίχνουν σημαντικά την απόδοση, πχ 0.588 σε 0.513 στο Twitter 2016. **Αμέσως μετά ακολουθούν** τα **χαρακτηριστικά** που βασίζονται στα **λεξικά**. Τα **χαρακτηριστικά μορφολογίας δεν συνεισφέρουν σημαντικά στην απόδοση**, παρόλα αυτά χρησιμοποιούνται πάντα από όλες τις ομάδες που συμμετέχουν στο διαγωνισμό.

Προκειμένου να αποκτήσουμε κάποια στοιχεία σχετικά με το τι χρειάζεται για την περαιτέρω βελτίωση των συστημάτων μας, εξετάσαμε σε ένα από αυτά τα **σφάλματα** που έγιναν. Συγκεκριμένα, εξετάσαμε τα σφάλματα στο καλύτερο σύστημα το οποίο χρησιμοποιεί όλα τα χαρακτηριστικά και τον Naive Bayes Tree ταξινομητή. Είναι προφανές ότι ορισμένα είδη σφαλμάτων ήταν πιο διαδεδομένα. Η δομή της σημασιολογίας / πρότασης συχνά περιλαμβάνει αστεία, σαρκασμό, ερωτήσεις ή ρητορικές ερωτήσεις, που έχουν τεκμηριωθεί στο παρελθόν ως προβλήματα για τα εργαλεία ανάλυσης του συναισθήματος. Τα λανθασμένα ερμηνευμένα tweets αποδίδεται επίσης στην παρουσία όρων που χρησιμοποιούνται σε ένα πλαίσιο που δεν είναι κυριολεκτικό. Για παράδειγμα, η χρήση υβριστικών λέξεων για να δείξουν ένα θετικό αποτέλεσμα. Άλλη κοινή περίπτωση σφαλμάτων συναντήσαμε τα tweets που περιέχουν μικτά συναισθήματα, όπου οι συντάκτες ενσωματώνουν θετικά και αρνητικά συναισθήματα μέσα στους 140 χαρακτήρες. Αυτό το εύρημα υποδηλώνει ότι το όριο των 140 χαρακτήρων μπορεί να περιορίσει την ικανότητα του χρήστη να εκφράσει σύνθετες απόψεις.

Ο πίνακας που ακολουθεί, δείχνει μερικά παραδείγματα των tweets που ταξινομούνται ψευδώς σε οποιαδήποτε από τις τρεις κλάσεις για τα Subtask A. Παραδείγματα από τα tweets που έχουν ταξινομηθεί εσφαλμένα ως θετική επίδειξη περιέχουν θετικές λέξεις (π.χ. good, fun, lol κ.λπ.). Από την άλλη πλευρά, τα tweets που ταξινομούνται ως αρνητικά μπορούν να περιέχουν τόσο θετικές όσο και αρνητικές λέξεις, όπως "pleasure" και "guilty", επομένως είναι σημαντικό να ταξινομηθούν σωστά οι περιπτώσεις αυτές στη σωστή κατηγορία. Τέλος, τα tweets έχουν ταξινομηθεί εσφαλμένως ουδέτερα δεν μπορεί να περιέχουν λέξεις που μεταφέρουν το συναίσθημα.

Είναι αρκετά δύσκολο για το μοντέλο μας να ενσωματώνει περίπλοκες γνώσεις όπως για παράδειγμα ιδιωματικές εκφράσεις ή μεταφορικές εκφράσεις που μπορεί να περιέχουν το συναίσθημα.

Πραγματικό label	**Predicted label**	**tweet**
positive	negative	Work tomorrow is gonna be so fun #tired
positive	neutral	Oomf is gay. He may not know it yet˘002c but I do lol
negative	positive	Well kid’s it’s Friday. So rejoice. Another week is done. The Hell with trump I’m voting for Amy Schumer :-) White house bound @amyschumer

Βιβλιογραφία

[1] 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.

[2] 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.

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

[4] 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.

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

[6] Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5-32.

[7] Fürnkranz, J. (2002). Round robin classification. *Journal of Machine Learning Research*, *2*(Mar), 721-747.

[8] J. Weston and C. Watkins. Support vector machines for multi-class pattern recognition. In M. Verleysen (ed.) Proceedings of the 7th European Symp

[9] R. E. Schapire. Using output codes to boost multiclass learning problems. In D. H. Fisher (ed.) Proceedings fo the 14th International Conference on Machine Learning (ICML-97), pp. 313–321, Nashville, TN, 1997. Morgan Kaufmann. R. E.

[10] Schapire and Y. Singer. Improved boosting algorithms using confidence-rated predictions. Machine Learning, 37(3):297–336, 1999.

[11] E. Mayoraz and E. Alpaydin. Support vector machines for multi-class classification. In J. Mira and J. V. S´anchez-Andr´es (eds.) Engineering Applications

of Bio-Inspired Artifi- cial Neural Networks: Proceedings of the International Work-Conference on Artificial and Natural Neural Networks (IWANN-99), Volume II, pp. 833–842, Alicante, Spain, 1999. Springer-Verlag.

[12] Wilson, T., Wiebe, J., & Hoffmann, P. (2005, October). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*(pp. 347-354). Association for Computational Linguistics.

[13] Rosenthal, S., Farra, N., & Nakov, P. (2017). SemEval-2017 task 4: Sentiment analysis in Twitter. In Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017) (pp. 502-518).

[14] G. Grefenstette, Y. Qu, J.G. Shanahan, and D.A. Evans. 2001. Coupling niche browsers and affect analysis for an opinion mining application. In RIAO-2004

[15] "A Tutorial on Support Vector Regression", Alex J. Smola, Bernhard Schölkopf - Statistics and Computing archive Volume 14 Issue 3, August 2004, p. 199-222.

[16] Elements of Statistical Learning by T. Hastie, R. Tibshirani and J. H. Friedman.

[17] Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992, July). A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory* (pp. 144-152). ACM.

[18] C-W. Hsu, C-C. Chang, C-J. Lin, "A Practical Guide to Support Vector Classification".

[19] S. Tong, D. Koller, "Support vector machine active learning with applications to text classification", ICML, 2000.

[20] Mohammad, S. M., Kiritchenko, S., & Zhu, X. (2013). NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets. *arXiv preprint arXiv:1308.6242*.

[21] Mohammad, S. M., & Turney, P. D. (2013). *Nrc emotion lexicon*. NRC Technical Report.

[22] Rong, X. (2014). word2vec parameter learning explained. *arXiv preprint arXiv:1411.2738*.

[23] Owoputi, O., O'Connor, B., Dyer, C., Gimpel, K., Schneider, N., & Smith, N. A. (2013). Improved part-of-speech tagging for online conversational text with word clusters. Association for Computational Linguistics.

[24] Pang, B., Lee, L., & Vaithyanathan, S. (2002, July). Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10* (pp. 79-86). Association for Computational Linguistics.

[25] Dave, K., Lawrence, S., & Pennock, D. M. (2003, May). Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th international conference on World Wide Web* (pp. 519-528). ACM.

[26] Clark, P., & Boswell, R. (1991, March). Rule induction with CN2: Some recent improvements. In *European Working Session on Learning* (pp. 151-163). Springer, Berlin, Heidelberg.

[27] Clark, P., & Niblett, T. (1989). The CN2 induction algorithm. Machine learning, 3(4), 261-283.

[28] Rivest, R. L. (1987). Learning decision lists. Machine learning, 2(3), 229-246.

[29] Hu, W., Hu, W., & Maybank, S. (2008). Adaboost-based algorithm for network intrusion detection. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 38(2), 577-583.

[30] Anand, R., Mehrotra, K., Mohan, C. K., & Ranka, S. (1995). Efficient classification for multiclass problems using modular neural networks. IEEE Transactions on Neural Networks, 6(1), 117-124.

[31] Cortes, Corinna, and Vladimir Vapnik. "Support-vector networks." Machine learning 20, no. 3 (1995): 273-297.

[32] Wikipedia: https://el.wikipedia.org/wiki/Μηχανική_μάθηση

[33] Kevin P. Murphy – Machine Learning, A Probabilistic Perspective, The MIT Press.

[34] Γ. Μαστραπάς - Ανάλυση Συναισθήματος Σε Δεδομένα Του Κοινωνικού Δικτύου Twitter Με Μεθόδους Μηχανικής Μάθησης, http://artemis-new.cslab.ece.ntua.gr:8080/jspui/handle/123456789/7843