ΙΟΝΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Προπτυχιακό Πρόγραμμα Σπουδών στην Πληροφορική

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΑΥΤΟΜΑΤΗ ΑΝΑΓΝΩΡΙΣΗ ΟΡΘΟΓΡΑΦΙΚΩΝ ΛΑΘΩΝ ΣΤΗΝ ΕΛΛΗΝΙΚΗ ΓΛΩΣΣΑ

Φοιτήτρια: Παναγιώτα Σμπιλίρη

Επιβλέπουσα: κα. Κάτια Λήδα Κερμανίδου

Κέρκυρα, Φεβρουάριος 2019

Επιβλέπουσα

κα. Κάτια Λήδα Κερμανίδου

Τριμελής Επιτροπή

κα. Κάτια Λήδα Κερμανίδου

κ. Σπυρίδων Σιούτας

κ. Φοίβος Μυλωνάς

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΠΕΡΙΛΗΨΗ

Η παρούσα πτυχιακή εργασία μελετά την αυτόματη διόρθωση ορθογραφικών λαθών στην ελληνική γλώσσα. Πιο συγκεκριμένα, εξετάζει τα πολύ συνηθισμένα λάθη στον ελληνικό γραπτό λόγο που προέρχονται από τη λανθασμένη τοποθέτηση των λέξεων «πού-που» και «πώς-πως» σε μια φράση. Αν και η χρήση της λέξης «πού» στη θέση του «που», ή αντίστροφα, είναι λανθασμένη, όπως και η χρήση του «πώς» στη θέση του «πως», ή αντίστροφα, τα λάθη αυτά δεν είναι ανιχνεύσιμα από ένα λογισμικό επεξεργασίας κειμένου, καθώς δεν αφορούν αυστηρά την ορθογραφία των λέξεων αλλά προκύπτουν από τη σημασία τους. Έτσι, με βάση το ελληνικό επισημειωμένο σώμα κειμένων DELOS, επιχειρείται η εξόρυξη γνώσης για τα λάθη αυτά.

Για το σκοπό αυτό, το σώμα κειμένων τοποθετήθηκε σε μια βάση δεδομένων, για την ευκολότερη πρόσβαση στις πληροφορίες αυτού οι οποίες ήταν χρήσιμες για τη μελέτη. Έπειτα, δημιουργήθηκαν τα datasets, πάνω στα οποία πραγματοποιήθηκε η μηχανική μάθηση, μέσω του λογισμικού WEKA. Οι ταξινομητές που χρησιμοποιήθηκαν ήταν οι ZeroR, Naive Bayes, Id3, J48 και Random Forest. Επίσης, υπήρξε πειραματισμός ανάμεσα στα αρχικά datasets και την εξισορροπημένη μορφή τους, η οποία δημιουργήθηκε από την εφαρμογή του φίλτρου SMOTE, αλλά και μεταξύ των παραθύρων λέξεων, καθώς κάποια datasets περιείχαν την προηγούμενη και την επόμενη λέξη της λέξης-στόχου και άλλα περιείχαν τις δύο προηγούμενες και τις δύο επόμενες λέξεις της λέξης-στόχου.

ΛΕΞΕΙΣ – ΚΛΕΙΔΙΑ

Εξόρυξη Γνώσης, Μηχανική Μάθηση, SQL, WEKA, SMOTE, ZeroR, Naive Bayes, Id3, J48, Random Forest

1 ΕΞΟΡΥΞΗ ΔΕΔΟΜΕΝΩΝ

1.1 ΟΡΙΣΜΟΣ ΕΞΟΡΥΞΗΣ ΔΕΔΟΜΕΝΩΝ

Ένα από τα ζητήματα τα οποία έχει κληθεί να αντιμετωπίσει ο επιστημονικός κλάδος της Πληροφορικής τις τελευταίες δεκαετίες είναι αυτό της διαχείρισης του μεγάλου όγκου δεδομένων που συλλέγονται για την πραγματοποίηση ερευνών, αλλά και της εξαγωγής χρήσιμης πληροφορίας από αυτόν. Και αυτό, καθώς, όλος αυτός ο όγκος των δεδομένων δεν είναι χρήσιμος εάν τα δεδομένα δεν επεξεργαστούν κατάλληλα.

Ωστόσο, συχνά η επεξεργασία αυτή είναι εξαιρετικά πολύπλοκη, τόσο εξαιτίας του μεγάλου όγκου των δεδομένων όσο και εξαιτίας των πολύπλοκων και ίσως δύσκολα ανιχνεύσιμων σχέσεων μεταξύ των δεδομένων. Μια ακόμα πρόκληση δημιουργείται από το γεγονός πως τα παραδοσιακά εργαλεία ανάλυσης δεδομένων σταδιακά παραγκωνίζονται λόγω της δυσκολίας εφαρμογής τους σε δεδομένα τέτοιου μεγάλου όγκου. Δεν είναι λίγες οι φορές που είτε οι παραδοσιακές αυτές τεχνικές ανάλυσης δεδομένων δε μπορούν να απαντήσουν σε σύγχρονα ερωτήματα, είτε η μη παραδοσιακή φύση των δεδομένων κάνει τις τεχνικές αυτές ακατάλληλες προς εφαρμογή, με αποτέλεσμα την επιτακτική ανάγκη για εύρεση νέων μεθόδων και τεχνικών που να ανταποκρίνονται στις καταστάσεις αυτές.

Η ανάγκη αυτή δημιούργησε ένα νέο επιστημονικό και ερευνητικό πεδίο, το οποίο ονομάστηκε «Εξόρυξη Δεδομένων» (Data Mining), και το οποίο χρησιμοποιείται όλο και περισσότερο από εταιρίες, οργανισμούς και επιστήμες, καθώς ο όγκος των δεδομένων τα οποία καλούνται να διαχειριστούν όλο και αυξάνεται. Η εξόρυξη δεδομένων, λοιπόν, στοχεύει στην επίλυση των δυσκολιών που παραπάνω περιγράφηκαν με την αξιοποίηση προτύπων, δομών, μοντέλων κ.ά. με έναν αυτοματοποιημένο τρόπο.

Εν συνεχεία όσον αναφέρθηκαν παραπάνω, θα μπορούσαμε να προσδιορίσουμε την εξόρυξη δεδομένων ως τη διαδικασία της αυτόματης ανακάλυψης χρήσιμων πληροφοριών μέσα από μεγάλες δεξαμενές δεδομένων, ή όπως αλλιώς αυτές ονομάζονται «αποθήκες δεδομένων» (data warehouses). Έτσι, οι τεχνικές της χρησιμοποιούνται είτε για την ανακάλυψη κρυμμένων προτύπων (patterns) σε ήδη υπάρχουσες βάσεις δεδομένων, τα οποία δε θα μπορούσαν να ανακαλυφθούν με τις παραδοσιακές μεθόδους, είτε για την πρόβλεψη του αποτελέσματος μιας μελλοντικής παρατήρησης.

1.2 ΣΤΟΧΟΙ ΕΞΟΡΥΞΗΣ ΔΕΔΟΜΕΝΩΝ

Η εξόρυξη δεδομένων έχεις ως κύριους στόχους την εφαρμογή τεχνικών πρόβλεψης και συμπεριφοράς τάσεων, την αναγνώριση, την περιγραφή σε μεγάλες βάσεις δεδομένων (Fayyad et al., 1996) καθώς και την ταξινόμηση και την βελτιστοποίηση των πόρων της. Πιο συγκεκριμένα:

- **Πρόβλεψη**
  Κατά τη διαδικασία της πρόβλεψης, γίνεται μια προσπάθεια για την πρόβλεψη μελλοντικών συμπεριφορών, μέσω συμπερασμάτων που προκύπτουν από τα διαθέσιμα δεδομένα. Πιο συγκεκριμένα, επιχειρείται η πρόβλεψη άγνωστων τιμών κάποιων μεταβλητών μέσω των διαδικασιών πρόβλεψης της εξόρυξης δεδομένων (predictive data mining tasks) και χρησιμοποιώντας μεταβλητές ή χαρακτηριστικά από τα ήδη διαθέσιμα δεδομένα.
- **Αναγνώριση**
  Κατά τη διαδικασία της αναγνώρισης, ερευνάται η ύπαρξη ή μη ενός γεγονότος μέσω των τυποποιημένων μορφών των δεδομένων.
- **Περιγραφή**
  Κατά τη διαδικασία της περιγραφής, επιχειρείται η όσο το δυνατόν πιο κατανοητή αναπαράσταση των περίπλοκων δεδομένων, τα οποία υπάρχουν στη βάση δεδομένων, μέσω νέων προτύπων και μοντέλων. Έτσι, οι ιδιότητες των διαθέσιμων δεδομένων μπορούν να περιγραφούν μέσω των περιγραφικών διαδικασιών της εξόρυξης δεδομένων (descriptive data mining tasks).
- **Ταξινόμηση**
  Κατά τη διαδικασία της ταξινόμησης, τα δεδομένα διαχωρίζονται σε διαφορετικές κλάσεις ή κατηγορίες. Ο διαχωρισμός αυτός χρησιμοποιείται στη συνέχεια για τη λήψη περαιτέρω αποφάσεων.
- **Βελτιστοποίηση**
  Κατά τη διαδικασία της βελτιστοποίησης επιχειρείται η επίτευξη ενός ακόμα σημαντικού στόχου για την εξόρυξη δεδομένων, αυτού της επίτευξης της βέλτιστης χρήσης κάποιων πόρων υπό περιορισμούς, πόρων όπως ο χρόνος, η μεγιστοποίηση μεγεθών όπως οι πωλήσεις για μια επιχείρηση κ.ά.

1.3 ΜΕΘΟΔΟΙ ΕΞΟΡΥΞΗΣ ΔΕΔΟΜΕΝΩΝ

Οι στόχοι της εξόρυξης δεδομένων που παρουσιάστηκαν στην προηγούμενη υποενότητα, επιτυγχάνονται μέσω διάφορων μεθόδων της εξόρυξης δεδομένων, οι κυριότερες από τις οποίες είναι η κατηγοριοποίηση, η συσταδοποίηση και η παλινδρόμηση (Fayyad et al., 1996). Πιο συγκεκριμένα:

- **Κατηγοριοποίηση**

  Πρόκειται για τη πιο διαδεδομένη και ευρέως χρησιμοποιήσιμη μέθοδο εξόρυξης δεδομένων. Κατά την κατηγοριοποίηση, ή αλλιώς «ταξινόμηση» (classification), όπως αυτή ονομάζεται εναλλακτικά, πραγματοποιείται ο διαχωρισμός ενός συνόλου αντικειμένων, τα οποία περιγράφονται από ένα σύνολο χαρακτηριστικών, σε προκαθορισμένες κλάσεις, μέσω μεθόδων επιβλεπόμενης μάθησης. Έτσι, οι τεχνικές κατηγοριοποίησης χρησιμοποιούν ένα σύνολο εκπαίδευσης, στο οποίο κάθε αντικείμενο είναι ήδη συνδεδεμένο με την κλάση στην οποία ανήκει, ώστε ο αλγόριθμος μηχανικής μάθησης να εκπαιδευτεί μέσα από το σύνολο εκπαίδευσης αυτό. Το αποτέλεσμα της διαδικασίας μάθησης είναι η κατασκευή ενός μοντέλου βάσει του οποίου θα κατηγοριοποιηθούν τα νέα αντικείμενα στις κατάλληλες κλάσεις.

- **Συσταδοποίηση**

  Κατά τη συσταδοποίηση, ή αλλιώς «ομαδοποίηση» (clustering), όπως αυτή ονομάζεται εναλλακτικά, πραγματοποιείται ο διαχωρισμός αντικειμένων σε ομοιογενείς και ανεξάρτητες ομάδες. Στόχος είναι τα αντικείμενα κάθε ομάδας να είναι όσο το δυνατόν πιο όμοια μεταξύ τους και τα αντικείμενα διαφορετικών ομάδων να είναι όσο το δυνατόν ανόμοια μεταξύ τους. Έτσι, οι αλγόριθμοι συσταδοποίησης επιχειρούν αφενός τη μεγιστοποίηση της ομοιότητας μεταξύ των αντικειμένων της ίδιας ομάδας, και αφετέρου την ελαχιστοποίηση της ομοιότητας μεταξύ των αντικειμένων διαφορετικών ομάδων.

  Η βασικότερη διαφορά μεταξύ της κατηγοριοποίησης και της συσταδοποίησης είναι πως στη συσταδοποίηση η δομή και το πλήθος των ομάδων καθορίζονται κατά τη διάρκεια της έρευνας από τον αλγόριθμο συσταδοποίησης που χρησιμοποιείται, καθώς αρχικά είναι άγνωστο τόσο το πλήθος όσο και η δομή τους. Για το λόγο αυτό, στη συσταδοποίηση χρησιμοποιείται πάντα μη επιβλεπόμενη μάθηση, αφού τα αντικείμενα ομαδοποιούνται σύμφωνα με το εκάστοτε κριτήριο το οποίο ο ερευνητής

θέτει, σε αντίθεση με την κατηγοριοποίηση, στην οποία χρησιμοποιείται πάντα επιβλεπόμενη μάθηση, αφού οι κλάσεις είναι καθορισμένες εξαρχής.

- **Ανάλυση Συσχέτισης**

  Κατά την ανάλυση συσχέτισης (association analysis) επιχειρείται η ανακάλυψη σχέσεων μεταξύ των χαρακτηριστικών μια βάσης δεδομένων. Οι σχέσεις αυτές ονομάζονται «κανόνες συσχέτισης» (association rules) και χαρακτηρίζονται από το «κατώφλι στήριξης» (support threshold) και το «κατώφλι εμπιστοσύνης» (confidence threshold). Το μεν πρώτο, αναγνωρίζει τα στοιχεία εκείνα των βάσεων δεδομένων με τις περισσότερες εμφανίσεις ενώ, το μεν δεύτερο, εκφράζει την υπό συνθήκη πιθανότητα (conditional probability) ένα στοιχείο να εμφανίζεται σε μια διαδικασία όταν ένα άλλο επίσης εμφανίζεται.

- **Παλινδρόμηση**

  Πρόκειται για την παλαιότερη και δημοφιλέστερη στατιστική μέθοδο εξόρυξης δεδομένων. Κατά την παλινδρόμηση (regression), δημιουργείται μια μαθηματική σχέση η οποία περιγράφει τα δεδομένα μιας βάσης αριθμητικών δεδομένων. Η μαθηματική αυτή σχέση εφαρμόζεται σε νέα αριθμητικά δεδομένα με σκοπό την πρόβλεψη μιας μελλοντικής συμπεριφοράς, καθώς κύριος στόχος της παλινδρόμησης είναι η πρόβλεψη της τιμής μιας μεταβλητής βάσει των τιμών που είχε στο παρελθόν. Ωστόσο, αξίζει να σημειωθεί πως η μέθοδος αυτή δε λειτουργεί ορθά σε κατηγορικά δεδομένα καθώς λειτουργεί ορθά μόνο σε συνεχή ποσοτικά δεδομένα, όπως είναι το βάρος, η ηλικία κ.ά.

1.4 ΕΞΟΡΥΞΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΑΝΑΚΑΛΥΨΗ ΓΝΩΣΗΣ

Μεταξύ των εννοιών της Εξόρυξης Δεδομένων και της Ανακάλυψης Γνώσης (Knowledge Discovery) επικρατεί μια σύγχυση και συχνά μια λανθασμένη ταύτιση, ωστόσο αξίζει να σημειωθεί πως η Εξόρυξη Δεδομένων αποτελεί τμήμα της Ανακάλυψης γνώσης και μάλιστα εξαιρετικά σημαντικό. Έτσι, η Εξόρυξη Γνώσης είναι ο πυρήνας της Ανακάλυψης Γνώσης, η οποία είναι η συνολική διαδικασία μετατροπής των ακατέργαστων δεδομένων σε χρήσιμες πληροφορίες. Η διαδικασία αυτή αποτελείται από τα εξής τρία στάδια:

- **Προεπεξεργασία**

Κατά την προεπεξεργασία, τα ακατέργαστα δεδομένα εισόδου, τα οποία μπορεί είτε να κατανέμονται σε ποικίλες θέσεις, είτε να βρίσκονται αποθηκευμένα σε μια ποικιλία μορφών, είτε σε μια κεντρική αποθήκη δεδομένων, επεξεργάζονται κατάλληλα ώστε να φθάσουν σε μορφή συμβατή με την επεξεργασία που θα ακολουθήσει. Πιο συγκεκριμένα, το στάδιο αυτό περιλαμβάνει την συγχώνευση των δεδομένων, τον καθαρισμό των δεδομένων ώστε να απαλειφθούν τυχόν διπλότυπες παρατηρήσεις ή γενικότερος θόρυβος και την επιλογή των γνωρισμάτων και των εγγραφών εκείνων βάσει των οποίων θα γίνει αποδοτικότερη η ανάλυση που θα ακολουθήσει. Σύνηθες είναι η διαδικασία της προεπεξεργασίας να είναι χρονοβόρα και περίπλοκη, λόγω της ποικιλίας τεχνικών αποθήκευσης και συλλογής των δεδομένων εισόδου.

- **Εξόρυξη δεδομένων**
- **Εκ των υστέρων επεξεργασία**

  Κατά την εκ των υστέρων επεξεργασία, στα αποτελέσματα της εξόρυξης δεδομένων μπορούν να εφαρμοστούν μέθοδοι ελέγχου υποθέσεων και στατιστικά μέτρα. Επιπλέον, τα χρήσιμα και έγκυρα αποτελέσματα της εξόρυξης δεδομένων μπορούν να χρησιμοποιηθούν στα συστήματα υποστήριξης αποφάσεων ενός οργανισμού ή μιας επιχείρησης αν αυτοί το επιθυμούν στα πλαίσια της εκ των υστέρων επεξεργασίας. Τέλος, η οπτικοποίηση των αποτελεσμάτων είναι μια συνηθισμένη τεχνική εκ των υστέρων επεξεργασίας.

1.5 ΕΞΟΡΥΞΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΕΞΟΡΥΞΗ ΓΝΩΣΗΣ ΑΠΟ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

Σήμερα, η έννοια της Εξόρυξης Δεδομένων είναι συνώνυμη με αυτή της Εξόρυξης Γνώσης από Βάσεις Δεδομένων (Knowledge Discovery in Databases), η οποία, ωστόσο, επικεντρώνεται περισσότερο στη διαδικασία ανάλυσης των δεδομένων, παρά στις συγκεκριμένες μεθόδους ανάλυσης των δεδομένων (Fayyad et al.,1996). Η Εξόρυξη Γνώσης από Βάσεις Δεδομένων αποτελείται από 5 στάδια, ένα από τα οποία είναι η εξόρυξη δεδομένων. Πιο συγκεκριμένα:

1. **Συλλογή (Selection)**

Στο πρώτο στάδιο, τα δεδομένα συγκεντρώνονται και επιλέγονται εκείνα τα οποία θα αποτελέσουν το σύνολο δεδομένων πάνω στο οποίο θα πραγματοποιηθεί η εξόρυξη δεδομένων, σε ένα από τα παρακάτω στάδια.

2. **Προεπεξεργασία (Preprocessing)**
   Στο δεύτερο στάδιο, γίνεται η κατάλληλη προεπεξεργασία του συνόλου δεδομένων και ο καθαρισμός του, με σκοπό να απομείνει μόνο η πληροφορία εκείνη η οποία θεωρείται χρήσιμη για την εξόρυξη δεδομένων και η υπόλοιπη άχρηστη πληροφορία να εξαφανιστεί από το σύνολο δεδομένων.
3. **Μετασχηματισμός (Transformation)**
   Στο τρίτο στάδιο, γίνεται ο μετασχηματισμός των δεδομένων, μέσω διαφόρων τεχνικών, καθεμία από τις οποίες εξυπηρετεί κάποιο συγκεκριμένο σκοπό, όπως η μείωση του μεγέθους του συνόλου δεδομένων κ.ά.

Ανάμεσα στο τρίτο και το τέταρτο στάδιο, επιλέγεται η μέθοδος εξόρυξης δεδομένων που είναι κατάλληλη για την επίτευξη των στόχων που έχουν τεθεί ήδη από τα πρώτο στάδιο, όπως για παράδειγμα η κατηγοριοποίηση ή η συσταδοποίηση ή η παλινδρόμηση. Έτσι, για τη μετάβαση στο τέταρτο στάδιο, είναι απαραίτητη η επιλογή του αλγορίθμου εξόρυξης γνώσης και της μεθόδου που θα εφαρμοστεί καθώς και η σωστή παραμετροποίησή τους.

4. **Εξόρυξη δεδομένων (Data Minion)**
   Στο τέταρτο στάδιο, πραγματοποιείται η εξόρυξη των δεδομένων.

Ανάμεσα στο τέταρτο και το πέμπτο στάδιο, γίνεται η ερμηνεία των αποτελεσμάτων της εξόρυξης δεδομένων με ποικίλους τρόπους οι οποίοι μπορεί να εξυπηρετούν, ανάλογα με τη φύση του προβλήματος που εξετάζεται και αυτό που επιθυμούν να αποκομίσουν όσοι διεξάγουν τη μελέτη. Ένας εύκολα κατανοητός τρόπος ερμηνείας των αποτελεσμάτων μπορεί να είναι η οπτικοποίησή τους.

5. **Ερμηνεία / Αξιολόγηση (Interpretation / Evaluation)**
   Στο πέμπτο και τελευταίο στάδιο, προκύπτει τελικά η γνώση και αξιολογείται.

Μετά το πέρας και των πέντε σταδίων, η γνώση μπορεί πια να χρησιμοποιηθεί για την επίλυση ενός ζητήματος. (Fayyad et al., 1996)

1.6 ΕΞΟΡΥΞΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΑΛΛΑ ΕΠΙΣΤΗΜΟΝΙΚΑ ΠΕΔΙΑ

Η εξόρυξη δεδομένων δημιουργήθηκε έπειτα από τη συνεργασία ερευνητών διαφόρων επιστημονικών κλάδων. Οι ερευνητές αυτοί προσπάθησαν να αντιμετωπίσουν κάποιες δυσκολίες που είχαν προκύψει για το έργο τους καθώς οι παραδοσιακές πρακτικές ανάλυσης δεδομένων, που χρησιμοποιούνταν έως τότε, δεν ανταποκρίνονταν πια στις νέες μορφές συνόλων δεδομένων. Έτσι, στηριζόμενοι σε πρακτικές όπως η δειγματοληψία, η εκτίμηση και ο έλεγχος υποθέσεων από τον τομέα της στατιστικής, αλλά και σε ιδέες όπως οι θεωρίες μάθησης, οι αλγόριθμοι αναζήτησης και οι τεχνικές μοντελοποίησης από τον τομέα της μηχανικής μάθησης, της τεχνητής νοημοσύνης και της αναγνώρισης προτύπων, δημιούργησαν την εξόρυξη δεδομένων. Ωστόσο, η εξέλιξη της εξόρυξης δεδομένων δε σταμάτησε στη δημιουργία της καθώς σύντομα ενσωμάτωσε χρήσιμα για εκείνη στοιχεία και ιδέες και από άλλες επιστημονικές περιοχές, όπως βελτιστοποίηση, ανάκτηση πληροφοριών, επεξεργασία σήματος, οπτικοποίηση, εξελικτικό υπολογισμό και θεωρία της πληροφορίας.

Παρόλα αυτά, πλήθος επιπρόσθετων τομέων έχουν σημαντικό υποστηρικτικό ρόλο ώστε να είναι δυνατή η εφαρμογή των πρακτικών της εξόρυξης δεδομένων. Αρχικά, κύριο ρόλο έχουν τα συστήματα βάσεων δεδομένων, τα οποία είναι απαραίτητα για την αποδοτική αποθήκευση, την ευρετηριοποίηση και την επεξεργασία των ερωτημάτων. Επιπλέον, οι τεχνικές των παράλληλων συστημάτων (συστήματα υψηλής απόδοσης) συχνά είναι απαραίτητες ώστε να αντιμετωπιστούν σύνολα δεδομένων μεγάλου όγκου. Τέλος, όταν τα δεδομένα δεν μπορούν να συγκεντρωθούν σε μια τοποθεσία, απαραίτητη είναι η χρήση κατανεμημένων τεχνικών.

Συγκεντρωτικά, τις ρίζες της εξόρυξη δεδομένων μπορούμε να τις εντοπίσουμε στα επιστημονικά πεδία των Βάσεων Δεδομένων, της Στατιστικής καθώς και Τεχνητής Νοημοσύνης και της Μηχανικής Μάθησης.

2 ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

2.1 ΟΡΙΣΜΟΣ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ

Η μηχανική μάθηση εισήχθηκε ως όρος τη δεκαετία του 1980, αν και προϋπήρχε ως επιστημονικός τομέας, και είναι ένας από τους ταχύτερα αναπτυσσόμενους τομείς της ΤΝ. Όπως είναι φυσικό, δε θα μπορούσε να μην κερδίσει το ενδιαφέρον και να μην αποτελέσει πρόκληση για τους ερευνητές της ΤΝ η ικανότητα του ανθρώπου να μαθαίνει και να εξελίσσεται μέσω της γνώσης που αποκτά. Έτσι, η μηχανική μάθηση ασχολείται με την ικανότητα ενός συστήματος να αντιλαμβάνεται και να αλληλεπιδρά με το περιβάλλον του και, μέσω της εμπειρίας που θα αποκτά από την αλληλεπίδραση αυτή, να έχει τη δυνατότητα να προσαρμόζεται καλύτερα στο περιβάλλον αυτό και να βελτιώνει τις ενέργειές του.

Ο Αμερικάνος πρωτοπόρος της τεχνητής νοημοσύνης και σχεδιαστής παιχνιδιών Arthur Samuel επινόησε τον όρο «μηχανική μάθηση» το 1959 και όρισε τη μηχανική μάθηση ως «*πεδίο μελέτης που δίνει στους υπολογιστές την ικανότητα να μαθαίνουν, χωρίς να έχουν ρητά προγραμματιστεί*» ενώ, ένας άλλος ορισμός ορίζει πως «*ένα πρόγραμμα υπολογιστή θεωρείται ότι μαθαίνει από την εμπειρία Ε σε σχέση με μια κατηγορία εργασιών Τ και μια μετρική απόδοσης Ρ, αν η απόδοσή του σε εργασίες της Τ, όπως μετριούνται με την Ρ, βελτιώνονται με την εμπειρία Ε*» (Mitchell, 1997).

Η μηχανική μάθηση είναι ένα διεπιστημονικό πεδίο το οποίο συνεχώς εξελίσσεται και διευρύνεται. Στην εξέλιξή του αυτή συμβάλλει καθοριστικά η αλληλεπίδραση της μηχανικής μάθησης με επιστήμες όπως η στατιστική, η επιστήμη των υπολογιστών, η μηχανική, η γνωστική επιστήμη κ.ά. Εξάλλου, από τις αρχές ακόμη της εμφάνισής της, η μηχανική μάθηση δανείστηκε λεξιλόγιο αλλά και μεθόδους από τη στατιστική, τη γνωστική επιστήμη, τη λογική κ.ά. Τέλος, σημαντικό ρόλο στη μεταφορά του τρόπου μάθησης του ανθρώπου στους υπολογιστές είχε η κατανόηση του μέσω της μελέτης της λειτουργίας του εγκεφάλου των έμβιων όντων από ειδικούς ερευνητές και μελετητές.

2.2 ΙΣΤΟΡΙΑ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ

Αν και η ιστορία της μηχανικής μάθησης είναι αλληλένδετη με εκείνη της τεχνητής νοημοσύνης, δεν είναι δυο πορείες ταυτόσημες. Από τη μία πλευρά, η μηχανική μάθηση χρησιμοποιείται σε καθημερινές διαδικασίες, όπως η υποστήριξη εργασιών, η

κατηγοριοποίηση ειδήσεων κ.ά. Από την άλλη πλευρά, η τεχνητή νοημοσύνη στοχεύει στην επίλυση πιο πολύπλοκων προβλημάτων καθώς και στη λήψη αποφάσεων μέσω της κατασκευής συστημάτων μίμησης της ανθρώπινης συμπεριφοράς.

Στην ιστορία της μηχανικής μάθησης [12] σταθμός υπήρξε το 1950, όταν ο Άγγλος Alan Turing, ο οποίος από πολλούς θεωρείται «πατέρας της επιστήμης των υπολογιστών», πρότεινε την αντικατάσταση της ευφυούς μηχανής με μια μηχανή μίμησης της ανθρώπινης συμπεριφοράς. Η πρόταση αυτή υπήρξε σταθμός καθώς ήταν η αρχή για τη δημιουργία αλγορίθμων και την παραγωγή προγραμμάτων με κατεύθυνση την επίτευξη της δημιουργίας της μηχανής μίμησης αυτής. Έτσι, δημιούργησε το γνωστό «Turing Test» στο οποίο ένας υπολογιστής μπορεί να επιτύχει αν ένας άνθρωπος – εξεταστής δεν μπορέσει να συμπεράνει αν ο γραπτές απαντήσεις που θα έχει δώσει ο υπολογιστής σε ερωτήσεις του ανθρώπου – εξεταστή προέρχονται από άνθρωπο ή όχι. Για να περάσει τη δοκιμασία αυτή ένας υπολογιστής θα πρέπει να έχει τις εξής ικανότητες:

- **επεξεργασία φυσικής γλώσσας** (natural language processing), ώστε να μπορεί να επικοινωνεί σε μία γλώσσα όπως π.χ. η αγγλική,
- **αναπαράσταση γνώσης** (knowledge representation), ώστε να αποθηκεύει όσα αναφέρονται στη συζήτηση με τον άνθρωπο – εξεταστή,
- **μηχανική μάθηση** (machine learning), ώστε να προσαρμόζεται σε κάθε νέα ερώτηση και να συμπεραίνει αυτόνομα τί θα απαντήσει μέσω του εντοπισμού και του συμπερασμού προτύπων,
- **μηχανική όραση** (computer vision), ώστε να αντιλαμβάνεται αντικείμενα και τέλος
- **ρομποτική** (robotics), ώστε να μετακινείται στο χώρο και να χειρίζεται αντικείμενα.

Έπειτα, το 1952, ο Αμερικάνος πρωτοπόρος της τεχνητής νοημοσύνης Arthur Samuel, δημιούργησε ένα από τα πρώτα επιτυχημένα προγράμματα υπολογιστή που μπορούσαν να μαθαίνουν από μόνα τους. Το συγκεκριμένο λογισμικό μελετούσε στρατηγικές στο παιχνίδι της Ντάμας και προσπαθούσε να συμπεράνει τα χαρακτηριστικά των στρατηγικών που κερδίζουν. Τέλος, το 1957, ο Αμερικάνος ψυχολόγος Frank Rosenblatt δημιούργησε το πρώτο νευρωνικό δίκτυο για υπολογιστές, το οποίο επιχειρούσε την προσομοίωση των διαδικασιών σκέψης του ανθρώπινου εγκεφάλου και το 1967 αναπτύχθηκε ο Αλγόριθμος Πλησιέστερου Γείτονα (Nearest Neighbour Algorithm). Τα γεγονότα αυτά δεν είναι τα μοναδικά σημαντικά στην πορεία της μηχανικής μάθησης μέσα στα χρόνια, ωστόσο είναι καταλυτικά σημεία αυτής καθώς έπεισαν ακόμα περισσότερους ερευνητές να καταπιαστούν με έναν νέο τότε κλάδο μελέτης, απέναντι στον οποίο ίσως κάποιοι από αυτούς

στέκονταν επιφυλακτικά, αφού τους απέδειξαν πως η μηχανική μάθηση ήταν ένας ενδιαφέρον τομέας για ενασχόληση ο οποίος θα μπορούσε να έχει αποτελέσματα ιδιαίτερα χρήσιμα για τον άνθρωπο στο μέλλον.

2.3 ΚΑΤΗΓΟΡΙΟΠΟΙΗΣΗ ΤΕΧΝΙΚΩΝ ΜΑΘΗΣΗΣ

2.3.1 ΕΠΙΒΛΕΠΟΜΕΝΗ ΜΑΘΗΣΗ

Στην επιβλεπόμενη μάθηση (supervised learning), ή όπως αλλιώς μπορούμε να τη συναντήσουμε στη βιβλιογραφία «επιτηρούμενη μάθηση» ή «μάθηση συνάρτησης», ο αλγόριθμος που χρησιμοποιείται έχει στη διάθεσή του τα δεδομένα γνωρίζοντας εκ των προτέρων σε ποια κατηγορία ανήκει καθένα από αυτά και επιδιώκει να «μαντέψει» σωστά την κατηγορία αυτή. Έτσι, στην επιβλεπόμενη μάθηση, στόχος είναι η κατασκευή ενός μοντέλου το οποίο θα είναι σε θέση να προβλέψει σε νέα, άγνωστα αντικείμενα ιδιότητες άγνωστες έως τότε, οι οποίες όμως ιδιότητες είναι ήδη γνωστές στο σώμα δεδομένων εκπαίδευσης το οποίο έχει στη διάθεσή του το μοντέλο.

Ας δούμε τώρα τον ορισμό της επιβλεπόμενης μάθησης λίγο πιο αυστηρά, μέσα από κάποια βασικά σημεία της.

- Η επιβλεπόμενη μάθηση στοχεύει στη γνώση της «συνάρτησης στόχο» (target function), η οποία ουσιαστικά χαρακτηρίζει το μοντέλο που περιγράφει τα δεδομένα.
- Οι τιμές που μπορεί να δεχτεί η συνάρτηση αυτή ως είσοδο, το πεδίο ορισμού της δηλαδή, ονομάζεται «σύνολο περιπτώσεων».
- Μέσω της συνάρτησης στόχου και βάσει των τιμών ενός συνόλου μεταβλητών που ονομάζονται «χαρακτηριστικά» (attributes ή features) ή «μεταβλητές εισόδου» ή «ανεξάρτητες μεταβλητές», προβλέπεται η τιμή μιας μεταβλητής που ονομάζεται «μεταβλητή εξόδου» ή «εξαρτημένη μεταβλητή».
- Κάθε σύνολο χαρακτηριστικών προσδιορίζει μια περίπτωση ή αλλιώς ένα «στιγμιότυπο».
- Ένα σύνολο στιγμιότυπων για τα οποία γνωρίζουμε την έξοδο της συνάρτησης στόχου ονομάζεται «παραδείγματα» ή «σύνολο δεδομένων εκπαίδευσης».

- Έτσι, το σύστημα, βάσει του συνόλου δεδομένων εκπαίδευσης, επιχειρεί την όσο το δυνατόν καλύτερη προσέγγιση της συνάρτησης στόχου μέσα από τον πειραματισμό με διαφορετικές συναρτήσεις, οι οποίες ονομάζονται «υποθέσεις».

Στην επιβλεπόμενη μάθηση βρίσκουν λύση δύο κυρίως κατηγορίες προβλημάτων, τα προβλήματα στα οποία γίνεται πρόβλεψη αριθμητικών τιμών και είναι γνωστά ως προβλήματα παλινδρόμησης (regression) και τα προβλήματα στα οποία γίνεται πρόβλεψη μοντέλων διακριτών κλάσεων και είναι γνωστά ως προβλήματα κατηγοριοποίησης (classification). Τέτοια παραδείγματα προβλημάτων είναι, αντίστοιχα, η εκτίμηση του βάρους ενός ανθρώπου βάσει των υπόλοιπων βιομετρικών χαρακτηριστικών του και η εκτίμηση του αν θα βρέξει ή όχι μια μέρα βάσει των υπόλοιπων χαρακτηριστικών του καιρού, όπως πχ την ένταση του αέρα, την υγρασία κ.ά.

Κάποιες από τις βασικότερες τεχνικές επιβλεπόμενης μάθησης είναι οι εξής:

- Μάθηση κατά Bayes
- Νευρωνικά δίκτυα (neural networks)
- Μηχανές διανυσμάτων υποστήριξης (support vector machines – γνωστή και ως SVMs)
- Δέντρα απόφασης (decision trees)
- Μάθηση κανόνων (rule learning)
- Μάθηση εννοιών (concept learning)
- Μάθηση βάσει περιπτώσεων (instance based learning)
- Γενετικοί αλγόριθμοι (genetic algorithms)

2.3.2 ΜΗ ΕΠΙΒΛΕΠΟΜΕΝΗ ΜΑΘΗΣΗ

Σε αντίθεση με την επιβλεπόμενη μάθηση, στη μη επιβλεπόμενη μάθηση (unsupervised learning), ή όπως αλλιώς μπορούμε να τη συναντήσουμε στη βιβλιογραφία «μη επιτηρούμενη μάθηση», ο αλγόριθμος πρέπει αυτόνομα να βρει δομή στα δεδομένα εισόδου που διαθέτει. Έτσι, στη μη επιβλεπόμενη μάθηση, στα στιγμιότυπα που απαρτίζουν το σώμα δεδομένων εκπαίδευσης δεν είναι γνωστή εκ των προτέρων η τιμή της ιδιότητας η οποία πρέπει να προβλεφθεί, αφήνοντας έτσι το σύστημα περισσότερο «ελεύθερο» σε σχέση με την επιβλεπόμενη μάθηση.

Στη μη επιβλεπόμενη μάθηση ανήκει, και σχεδόν ταυτίζεται, το πρόβλημα της ομαδοποίησης, αφού εδώ δεν υπάρχει γνώση σχετικά με το εάν υπάρχουν ομάδες, πόσες και ποιες ακριβώς είναι αυτές. Πιο συγκεκριμένα, ως ομαδοποίηση καλείται η διαδικασία κατά την οποία γίνεται ο διαχωρισμός δεδομένων σε ομάδες με στόχο να υπάρχουν στην ίδια ομάδα όσο το δυνατόν πιο όμοια δεδομένα. Οι αλγόριθμοι ομαδοποίησης κατηγοριοποιούνται στις τρεις παρακάτω κατηγορίες:

1. **Ιεραρχικοί αλγόριθμοι**
   Μέσω των αλγορίθμων αυτών προκύπτει τόσο ο αριθμός όσο και η δομή των ομάδων με ιεραρχικό τρόπο και, σύμφωνα με τον τρόπο με τον οποίο αναπτύσσεται η ιεραρχία αυτή, οι αλγόριθμοι της κατηγορίας αυτής διαχωρίζονται περαιτέρω στους αλγορίθμους διαίρεσης και στους αλγορίθμους συγχώνευσης.
2. **Πιθανοκρατικοί αλγόριθμοι**
   Οι αλγόριθμοι αυτοί βασίζονται σε πιθανοτικά μοντέλα, όπως τη θεωρία του Bayes.
3. **Αλγόριθμοι βασισμένοι σε διαχωρισμούς**
   Μέσω των αλγορίθμων αυτών επιχειρείται ο βέλτιστος διαμοιρασμός των δεδομένων σε έναν προκαθορισμένο αριθμό ομάδων. Ένας τέτοιος αλγόριθμος είναι ο K-means.

2.3.3 ΕΝΙΣΧΥΤΙΚΗ ΜΑΘΗΣΗ

Η ενισχυτική μάθηση (reinforcement learning) διαφέρει από την επιβλεπόμενη και μη επιβλεπόμενη μάθηση σε δύο βασικά σημεία. Το πρώτο είναι πως στην ενισχυτική μάθηση υπάρχει επιβράβευση βάσει της έκβασης της εργασίας που παράγει ένας αλγόριθμος και αυτό διότι η ενισχυτική μάθηση επηρεάστηκε καταλυτικά από τη μάθηση με επιβράβευση και τιμωρία, που αποτελεί μοντέλο μάθησης για τα έμβια όντα. Το δεύτερο είναι πως δεν υπάρχουν εξ αρχής ετικέτες στα δεδομένα που εξετάζονται. Έτσι, πιο συγκεκριμένα, στην ενισχυτική μάθηση ανήκουν τεχνικές μάθησης οι οποίες επιτρέπουν στο σύστημα μάθησης να μαθαίνει αυτόνομα, αλληλεπιδρώντας με το περιβάλλον του. Στόχος των συστημάτων ενισχυτικής μάθησης, λοιπόν, είναι να ανακαλύψουν από μόνα τους, χωρίς τη βοήθεια κάποιου εξωτερικού επιβλέποντα, τις ενέργειες εκείνες που θα αποφέρουν το μεγαλύτερο δυνατό κέρδος, θα μεγιστοποιήσουν δηλαδή τη συνάρτηση του αριθμητικού σήματος ενίσχυσης. Τέτοια συστήματα ενισχυτικής μάθησης δραστηριοποιούνται στη βελτιστοποίηση εργασιών σε εργοστασιακές μονάδες, στη μάθηση επιτραπέζιων παιχνιδιών κ.ά.

3 ΠΡΟΣΕΓΓΙΣΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ

3.1 ΚΑΤΗΓΟΡΙΕΣ ΛΕΞΕΩΝ ΕΛΛΗΝΙΚΗΣ ΓΛΩΣΣΑΣ

Η ελληνική γλώσσα είναι μια ιδιαιτέρως πλούσια γλώσσα, με ιδιαιτέρως πλούσιο λεξιλόγιο. Όλες οι λέξεις που την απαρτίζουν κατατάσσονται σε δέκα μεγάλες κατηγορίες, οι οποίες ονομάζονται μέρη του λόγου. Πιο συγκεκριμένα, τα μέρη του λόγου της ελληνικής γλώσσας είναι τα παρακάτω:

- Άρθρα
- Ουσιαστικά
- Επίθετα
- Αντωνυμίες
- Ρήματα
- Μετοχές
- Επιρρήματα
- Προθέσεις
- Σύνδεσμοι
- Επιφωνήματα

Επιπρόσθετα, υπάρχουν λέξεις της ελληνικής γλώσσας οι οποίες δεν συναντώνται πάντα με την ίδια μορφή στον γραπτό ή τον προφορικό λόγο, καθώς το τελευταίο μέρος τους κάποιες φορές αλλάζει ανάλογα με το νόημα που εκείνες εκφράζουν. Οι λέξεις αυτές λέμε πως κλίνονται και τις χαρακτηρίζουμε ως κλιτές, σε αντίθεση με εκείνες που συναντώνται πάντα με την ίδια μορφή, δηλαδή δεν κλίνονται, και τις χαρακτηρίζουμε ως άκλιτες. Έτσι, οδηγούμαστε σε μια ευρύτερη κατηγοριοποίηση των λέξεων της ελληνικής γλώσσας και κατηγοριοποιούμε τα μέρη του λόγου σε κλιτά και άκλιτα. Στα κλιτά μέρη του λόγου συμπεριλαμβάνονται τα άρθρα, τα ουσιαστικά, τα επίθετα, οι αντωνυμίες, τα ρήματα και οι μετοχές ενώ στα άκλιτα τα επιρρήματα, οι προθέσεις, οι σύνδεσμοι και τα επιφωνήματα.

3.2 ΟΜΩΝΥΜΑ, ΠΑΡΩΝΥΜΑ ΚΑΙ ΤΟΠΙΚΑ ΠΑΡΩΝΥΜΑ

Η ελληνική γλώσσα, λόγω του πλούτου της, έχει και αρκετές ιδιαιτερότητες. Παραδείγματα τέτοιων ιδιαιτεροτήτων αποτελούν οι λέξεις που χαρακτηρίζονται ως

ομώνυμα (ή ομόηχα), εκείνες που χαρακτηρίζονται ως παρώνυμα και εκείνες που χαρακτηρίζονται ως τονικά παρώνυμα.

Πιο συγκεκριμένα, ομώνυμες ή ομόηχες ονομάζονται δύο λέξεις οι οποίες προφέρονται με τον ίδιο τρόπο αλλά έχουν διαφορετική σημασία και συχνά διαφορετική ορθογραφία. Παραδείγματα ομώνυμων / ομόηχων λέξεων είναι:

- όμως (σύνδεσμος) – (ο) ώμος (μέρος του σώματος),
- (το) κλίμα (καιρός) – (το) κλήμα (αμπέλι),
- (οι) καιροί – (το) κερί,
- κλείνω (την πόρτα) – κλίνω (το ρήμα),
- πήρα (αόριστος του ρήματος παίρνω) – πείρα (εμπειρία) κ.ά.

Από την άλλη πλευρά, παρώνυμες ονομάζονται δύο λέξεις των οποίων η προφορά μοιάζει, έχουν δηλαδή παρόμοια προφορά, ωστόσο έχουν διαφορετική σημασία και ορθογραφία. Παραδείγματα παρώνυμων λέξεων είναι:

- ωφέλεια – αφέλεια,
- αμαρτωλός – αρματολός,
- σφήκα (έντομο) – σφίγγα (τέρας της ελληνικής μυθολογίας) κ.ά.

Τέλος, τα τονικά παρώνυμα είναι λέξεις οι οποίες έχουν ίδια προφορά όμως διαφέρουν στο τονισμό, έχουν διαφορετική σημασία και τις περισσότερες φορές διαφορετική ορθογραφία. Παραδείγματα τονικά παρώνυμων λέξεων είναι:

- παίρνω – περνώ,
- χώρος – χορός,
- ράφι – ραφή,
- νόμος – νομός,
- έκτος – εκτός κ.ά.

3.3 ΣΕΤ ΛΕΞΕΩΝ «ΠΟΎ-ΠΟΥ» ΚΑΙ «ΠΏΣ-ΠΩΣ»

Τα σετ μονοσύλλαβων λέξεων «πού-που» και «πώς-πως» συχνά μπερδεύουν όσους επιχειρούν να εκφραστούν γραπτά μέσω της ελληνικής γλώσσας και αυτό διότι, ενώ οι λέξεις κάθε σετ ακούγονται με τον ίδιο τρόπο, σε κάποιες περιπτώσεις είναι σωστή η τονισμένη εκδοχή τους και σε κάποιες άλλες η μη τονισμένη εκδοχή τους. Πιο συγκεκριμένα:

- **Πού**

  Η λέξη «πού» ανήκει στα τοπικά επιρρήματα, στις λέξεις εκείνες, δηλαδή, που συνοδεύουν τα ρήματα και φανερώνουν τόπο. Το ερωτηματικό «πού» μπορεί να αντικατασταθεί από «σε ποιόν», «σε ποιά», «σε ποιό» και συναντάται σε ευθείες ή πλάγιες ερωτήσεις ή στην έκφραση «πού και πού». Παραδείγματα προτάσεων όπου χρησιμοποιείται το «πού» είναι τα εξής:

  - **Πού** βρίσκεται το σπίτι; (ευθεία ερώτηση)
  - Με ρώτησε **πού** άφησα τα έγγραφά του. (πλάγια ερώτηση)
  - **Πού** και **πού** συναντιόμαστε. (έκφραση)

- **Που**

  Η λέξη «που» χρησιμοποιείται ως ειδικός ή συμπερασματικός σύνδεσμος, συνδέει δηλαδή άλλες λέξεις ή προτάσεις μεταξύ τους. Επίσης, μπορεί να χρησιμοποιηθεί και ως αναφορική αντωνυμία και να αντικατασταθεί είτε από «ο οποίος», «η οποία», «το οποίο» (και τον πληθυντικό τους) όταν αναφέρεται στο υποκείμενο της πρότασης, είτε από «τον οποίο», «την οποία», «το οποίο» (και τον πληθυντικό τους) όταν αναφέρεται στο αντικείμενο της πρότασης. Παραδείγματα προτάσεων όπου χρησιμοποιείται το «που» είναι τα εξής:

  - Μου είπε **που** πήγες από εκεί να τον δεις. (σύνδεσμος)
  - Οι άνθρωποι **που** γυμνάζονται… (Οι άνθρωποι **οι οποίοι** γυμνάζονται…) (αντωνυμία)
  - Τα δικαιολογητικά **που** προσκόμισε... (Τα δικαιολογητικά **τα οποία** προσκόμισε…) (αντωνυμία)

- **Πώς**

  Η λέξη «πώς» ανήκει στα τροπικά επιρρήματα, τις λέξεις εκείνες, δηλαδή, που συνοδεύουν τα ρήματα και φανερώνουν τρόπο. Το ερωτηματικό «πώς» συναντάται σε ευθείες ή πλάγιες ερωτήσεις ή στην έκφραση «πώς και πώς». Παραδείγματα προτάσεων όπου χρησιμοποιείται το «πώς» είναι τα εξής:

  - **Πώς** λύνεται αυτή η άσκηση; (ευθεία ερώτηση)
  - Αναρωτιέμαι **πώς** θα τα προλάβω όλα αυτά σε μόνο μισή ώρα. (πλάγια ερώτηση)
  - Περίμενε αυτή την ημέρα **πώς** και **πώς**! (έκφραση)

- **Πως**

  Η λέξη «πως» είναι ειδικός σύνδεσμος, δηλαδή συνδέει άλλες λέξεις ή προτάσεις μεταξύ τους και ισοδυναμεί με το «ότι». Παραδείγματα προτάσεων όπου χρησιμοποιείται το «πως» είναι τα εξής:

- Μου είπαν **πως** με ζητούσες. (Μου είπαν **ότι** με ζητούσες.)
- Έμαθα **πως** θα γυρίσεις σύντομα. (Έμαθα **ότι** θα γυρίσεις σύντομα.)

Έτσι, συγκεντρωτικά, και οι τέσσερις λέξεις «πώς», «πως», «πού» και «που», γραμματικά, ανήκουν στα άκλιτα μέρη του λόγου και οι μεν «πώς» και «πού» ανήκουν στα επιρρήματα ενώ οι δε «πως» και «που» στους συνδέσμους.

Ωστόσο, όπως είναι εύκολα κατανοητό σύμφωνα με όσα αναφέρονται στην αμέσως προηγούμενη παράγραφο, τα σετ λέξεων «πώς-πως» και «πού-που» αποτελούν σετ ομόηχων λέξεων, καθώς οι λέξεις κάθε σετ έχουν την ίδια ακριβώς προφορά, όμως διαφέρει η μεταξύ τους ορθογραφία αλλά και σημασία μέσα στο λόγο.

3.4 ΠΡΟΣΔΙΟΡΙΣΜΟΣ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ

Στο σημείο αυτό, είναι εύκολος ο προσδιορισμός του προβλήματος το οποίο αποτέλεσε πρόκληση για την παρούσα πτυχιακή. Το πρόβλημα αυτό προκύπτει τόσο από τη συχνά λανθασμένη χρήση των ομόηχων λέξεων «πού» και «που» καθώς και των ομόηχων λέξεων «πώς» και «πως» στον ελληνικό γραπτό λόγο, όσο και από την αδυναμία των λογισμικών επεξεργασίας κειμένων να ανιχνεύσουν και να διορθώσουν λάθη τέτοιας φύσης.

3.5 ΣΧΕΤΙΚΕΣ ΠΡΟΣΠΑΘΕΙΕΣ ΑΛΛΩΝ ΕΡΕΥΝΗΤΩΝ

Δεν είναι λίγες οι φορές όπου ερευνητές έχουν ασχοληθεί με την αυτόματη διόρθωση ορθογραφικών λαθών σε συστήματα επεξεργασίας κειμένου, τα οποία λάθη, όμως, δεν αφορούν τη λανθασμένη ορθογραφία των λέξεων, καθώς αυτό είναι ένα ζήτημα επιλυμένο για τα συστήματα επεξεργασίας κειμένου, αλλά αφορούν τη σωστή χρήση των λέξεων μέσα στο λόγο από σημασιολογικής άποψης. Για παράδειγμα, αν και οι λέξεις «σήκω» και «σύκο» είναι σωστές ορθογραφικά, η πρόταση «Σύκο από τη θέση σου.» είναι λανθασμένη καθώς η σωστή της εκδοχή είναι «Σήκω από τη θέση σου.», γεγονός δύσκολο να αναγνωριστεί από έναν επεξεργαστή κειμένου αφού αφορά τη σημασία κάθε μίας λέξης. Έτσι, στο συγκεκριμένο παράδειγμα, ο επεξεργαστής κειμένου, ενώ θα ανατρέξει στο λεξικό της γλώσσας το οποίο διαθέτει και θα θεωρήσει σωστή ορθογραφικά τη λέξη «σύκο» στην πρόταση «Σύκο από τη θέση σου.», παρόλα αυτά η λέξη «σύκο» εκφράζει ένα φρούτο και δεν θα έπρεπε να βρίσκεται στη πρόταση αυτή.

Οι περισσότερες μελέτες, όσον αφορά τα λάθη αυτά, έχουν γίνει στην αγγλική γλώσσα. Και αυτό καθώς και στην αγγλική γλώσσα υπάρχουν σετ λέξεων τα οποία εκφράζουν ομώνυμες λέξεις, όπως τα σετ λέξεων «know(ξέρω) - no(όχι)», «peace(ειρήνη) - piece(κομμάτι)» κ.ά., παρώνυμες λέξεις, όπως τα σετ λέξεων «adopt(ενστερνίζομαι) - adapt(προσαρμόζω)», «affect(επηρεάζω) - effect(αποτέλεσμα)» κ.ά. και πλήθος άλλων λέξεων οι οποίες είναι σωστές ορθογραφικά αλλά αν χρησιμοποιηθούν λανθασμένα σε μια πρόταση πρέπει να διορθωθούν.

Ο Andrew R. Golding το 1995 [2] παρουσίασε μια υβριδική μέθοδο, βασισμένη στους ταξινομητές Bayes, η οποία στόχευε στη βελτίωση των επιδόσεων των λιστών αποφάσεων (decision lists) και τη χρήση τους στην επίλυση ορθογραφικών λαθών που προκύπτουν από τη σημασία των λέξεων. Η έως τότε προσέγγιση του ζητήματος της λεξιλογικής ασάφειας στην αγγλική γλώσσα είχε προσεγγιστεί με δύο κυρίως μεθόδους: είτε εξετάζοντας την παρουσία συγκεκριμένων λέξεων σε μια ορισμένη απόσταση πριν και μετά τη λέξη-στόχο, της οποίας η ορθογραφία εξεταζόταν, είτε εξετάζοντας το μέρος του λόγου και άλλα στοιχεία της γραμματικής των λέξεων γύρω από τη λέξη-στόχο. Ο Yarowsky προσπάθησε με τις λίστες απόφασης να συνδυάσει τις δύο αυτές μεθόδους επίλυσης της λεξιλογικής ασάφειας και, εν συνεχεία, ο Golding, έχοντας ως αφετηρία τη μελέτη αυτή του Yarowsky, προσπάθησε με βελτιωμένες λίστες απόφασης να δώσει λύση σε ορθογραφικά λάθη σχετικά με τη σημασία των λέξεων.

Οι Andrew R. Golding και Yves Schabes [3] το 1996 προσπάθησαν να αντιμετωπίσουν λάθη της αγγλικής γλώσσας που οφείλονται είτε σε λέξεις ομώνυμες ή παρώνυμες, όπως πχ «peace-piece», «quiet-quite» κ.ά., είτε σε λέξεις οι οποίες λανθασμένα χρησιμοποιούνται η μια στη θέση της άλλης λόγω των παραπλήσιων σημασιών τους, όπως πχ οι λέξεις «amount»(ποσό) και «number»(αριθμός), οι λέξεις «among»(ανάμεσα/μεταξύ, χρησιμοποιείται για περισσότερες από δύο έννοιες) και «between»(ανάμεσα/μεταξύ, χρησιμοποιείται για δύο μόνο έννοιες) κ.ά. Στη μελέτη τους αυτή παρουσίασαν μια υβριδική μέθοδο, την οποία ονόμασαν Tribayes και η οποία ουσιαστικά συνδύαζε τα καλύτερα στοιχεία δύο άλλων μεθόδων, της μεθόδου Trigrams και της μεθόδου Bayes. Η πρώτη, εξέταζε τα μέρη του λόγου τριγράμμων ώστε να κωδικοποιήσει τα συμφραζόμενα της λέξης-στόχου, ενώ η δεύτερη ήταν αποδοτικότερη σε περιπτώσεις όπου εξετάζονταν λέξεις που είχαν το ίδιο μέρος του λόγου, περιπτώσεις όπου η πρώτη μέθοδος δεν ήταν τόσο ισχυρή. Οι Golding και Schabes παρατήρησαν πως η μέθοδος Tribayes είχε τη δυνατότητα να επιλύει αυτά τα προβλήματα, τα οποία δεν αντιμετωπίζονταν επαρκώς από τα γνωστά έως τότε ορθογραφικά συστήματα, με επίδοση σημαντικά καλύτερη ακόμα και από το σύστημα αναγνώρισης ορθογραφικών λαθών του Microsoft Word, με το οποίο συγκρίθηκε.

Οι Andrew R. Golding και Dan Roth [4] το 1998 ασχολήθηκαν με την επίλυση ορθογραφικών λαθών που προκύπτουν από τη σημασία των λέξεων, για σετ λέξεων της αγγλικής όπως τα «to-too», «casual-causal» κ.ά. Κατασκεύασαν για το σκοπό αυτό έναν αλγόριθμο ο οποίος συνδύαζε παραλλαγές του αλγορίθμου Winnow, μιας τεχνικής της μηχανικής μάθησης για την εκμάθηση ενός γραμμικού ταξινομητή (linear classifier) μέσα από επισημειωμένα παραδείγματα, και της ψήφου σταθμισμένης πλειοψηφίας (weighted-majority voting), τον οποίο ονόμασαν WinSpell, και για την αξιολόγησή του τον σύγκριναν με τον BaySpell, έναν αλγόριθμο βασισμένο στη στατιστική. Το σημαντικότερο το οποίο παρατήρησαν ήταν πως ο WinSpell, όταν εφαρμοζόταν σε κάποιο σετ παραδειγμάτων ελέγχου το οποίο είχε δημιουργηθεί από διαφορετικό σώμα κειμένων από εκείνο από το οποίο είχε δημιουργηθεί το σετ παραδειγμάτων εκπαίδευσής του, προσαρμοζόταν πολύ καλύτερα από τον BaySpell, χρησιμοποιώντας μια στρατηγική η οποία συνδύαζε την επιβλεπόμενη μάθηση για το σετ παραδειγμάτων εκπαίδευσης με τη μη επιβλεπόμενη μάθηση στο σετ παραδειγμάτων ελέγχου. Επίσης, σε σύγκριση με τον BaySpell, ο WinSpell πετύχαινε σημαντικά υψηλότερη ορθότητα (accuracy) ακόμα και όταν χρησιμοποιούνταν όλα τα χαρακτηριστικά του μοντέλου και το μοντέλο δεν είχε υποστεί κλάδεμα (prunning), από ό, τι μπορούσε ο BaySpell να επιτύχει στο ίδιο κλαδεμένο ή μη μοντέλο. Επιπρόσθετα, η αρχιτεκτονική του WinSpell του έδινε υπεροχή έναντι του BaySpell στην εκμάθηση ενός γραμμικού ταξινομητή. Τέλος, διαπίστωσαν πως ο WinSpell είχε καλύτερες επιδόσεις από πολλές άλλες τεχνικές που είχαν μελετήσει στην έως τότε βιβλιογραφία.

Οι Andrew Carlson και Ian Fette [5] το 2007 μελέτησαν τη διόρθωση ορθογραφικών λαθών της αγγλικής χρησιμοποιώντας μεθόδους μάθησης βασισμένες στη μνήμη (memory-based) και μια μεγάλη βάση δεδομένων με ν-γραμμα λέξεων, προερχόμενα από κείμενα που υπάρχουν στο διαδίκτυο, ως δεδομένα εκπαίδευσης. Η μελέτη τους επιτύγχανε υψηλότερη ορθότητα από ό,τι προηγούμενες μελέτες στη διόρθωση των λαθών αυτών που ονόμασαν «real-word errors», και επίσης πολύ υψηλή ορθότητα στο νέο πεδίο μελέτης που εισήγαγαν και ασχολήθηκαν με τα λάθη αυτά που ονόμασαν «non-word errors». Στα «real-word errors» ανήκουν σετ λέξεων όπως τα «here(εδώ) – hear(ακούω)» κ.ά. ενώ στα «non-word errors» ανήκουν λάθη που προκύπτουν από τη λανθασμένη πληκτρολόγηση μιας λέξης από έναν χρήστη και η μελέτη τους προσπάθησε να επιτύχει την αντικατάστασή της από τη σωστή λέξη πχ εάν κάποιος λανθασμένα πληκτρολόγησε τη λέξη «funn», αυτή θα πρέπει να αντικατασταθεί από κάποια από τις λέξεις «fun», «funny», «funk» ή κάποια άλλη, σύμφωνα με το τι θέλει ο χρήστης να εκφράσει με την πρότασή του.

Οι Johannes Schaback και Fang Li [6], επίσης το 2007, επιχείρησαν να διορθώσουν και εκείνοι real-word errors αλλά και non-word errors χρησιμοποιώντας χαρακτηριστικά από

το φωνολογικό, το συντακτικό, το σημασιολογικό κ.ά. επίπεδα των λέξεων, σε μια προσπάθειά τους να συμπεριλάβουν όση το δυνατόν περισσότερη πληροφορία. Τα επίπεδα αυτά αξιολογούνταν από μια Μηχανή Διανυσμάτων Υποστήριξης (Support Vector Machine-SVM) με σκοπό την πρόβλεψη της σωστής λέξης που θα αντικαθιστούσε τη λανθασμένη. Η προσέγγιση της μελέτης αυτής είχε την ιδιαιτερότητα πως δεν περιοριζόταν στη διόρθωση λέξεων μέσα από ένα προκαθορισμένο κατάλογο «σύγχυσης» λέξεων. Όπως συνέβαινε σε παλαιότερες προσεγγίσεις και μελέτες, και ξεπέρασε σε επιδόσεις το Microsoft Word, την Google κ.ά.

Αυτές είναι κάποιες από τις μελέτες που έχουν γίνει στο πεδίο της διόρθωσης ορθογραφικών λαθών σχετικών με τη σημασία των λέξεων στην αγγλική γλώσσα. Στην ελληνική γλώσσα οι προσπάθειες είναι σαφώς λιγότερες. Μια μελέτη η οποία ξεχωρίζει είναι η προσπάθεια των Gasteratos Petros, Dragonas Vasileios et al. οι οποίοι στο paper τους με τίτλο «Knowledge-poor Context-sensitive Spelling Correction for Modern Greek» παρουσιάζουν μια μεθοδολογία για την επίλυση λαθών της ελληνικής που προέρχονται από δύο συγκεκριμένες κατηγορίες ομώνυμων λέξεων. Η πρώτη αφορά ρήματα που σε κάποιες μορφές της ενεργητικής και της παθητικής φωνής τους ακούγονται με τον ίδιο τρόπο, ωστόσο στο γραπτό λόγο η κατάληξή τους αλλάζει., όπως για παράδειγμα οι λέξεις «γράφετε» (ενερφητική φωνή) και «γράφεται» (παθητική φωνή). Η δεύτερη αφορά επίθετα τα οποία όταν βρίσκονται στον ενικό αριθμό και είναι θηλυκού γένους ακούγονται με τον ίδιο τρόπο όπως όταν βρίσκονται στον πληθυντικό αριθμό και είναι αρσενικού γένους, ωστόσο στο γραπτό λόγο η κατάληξή τους δεν είναι η ίδια, όπως για παράδειγμα οι λέξεις «όμορφη» (θηλυκό/ενικός αριθμός) και «όμορφοι» (αρσενικό/πληθυντικός αριθμός). Στη μελέτη παρουσιάζεται μια μεθοδολογία για την οποία δε χρειάζεται ιδιαίτερη γλωσσολογική πληροφορία για κάθε λέξη αφού η μελέτη στηρίζεται στις καταλήξεις και μόνο των λέξεων. Έπειτα από την εφαρμογή κάποιων αλγορίθμων μηχανικής μάθησης, όπως οι Naive Bayes, Random Forest, Id3 κ.ά. η μελέτη φέρνει σημαντικά επιτυχημένα αποτελέσματα, αφού η επιτυχία της μεθοδολογίας κυμαίνεται μεταξύ 90% και 95%.

4 ΣΤΑΔΙΟ ΕΠΙΛΥΣΗΣ 1^{Ο}: ΣΥΛΛΟΓΗ ΔΕΔΟΜΕΝΩΝ

4.1 ΠΗΓΗ ΔΕΔΟΜΕΝΩΝ

Το δείγμα το οποίο υπήρχε αρχικά στη διάθεσή μας ήταν το DELOS. Το DELOS είναι ένα επισημειωμένο σώμα κειμένων της Νέας Ελληνικής Γλώσσας, όσον αφορά τη μορφολογία και τις επιφανειακές συντακτικές σχέσεις που εμφανίζονται σε αυτά. Με τον όρο «επισημειωμένο σώμα κειμένων» αναφερόμαστε σε κείμενα τα οποία εμπεριέχουν κάποιου είδους γλωσσολογική επισημείωση, ανάθεση δηλαδή γλωσσολογικής πληροφορίας σε ένα σώμα κειμένων με τη βοήθεια των κατάλληλων ετικετών σήμανσης (tags).

Τα κείμενα του DELOS είναι οικονομικής φύσης, και η κατασκευή του ευρύτερου έργου DELOS πραγματοποιήθηκε υπό τη χρηματοδότηση του Ελληνικού Υπουργείου Ανάπτυξης. Τα κείμενα συγκεντρώθηκαν μέσω της ημερήσιας οικονομικής εφημερίδας ΕΞΠΡΕΣ, εκθέσεων του Ιδρύματος Οικονομικών και Βιομηχανικών Ερευνών (ΙΟΒΕ), ερευνητικών εργασιών του Οικονομικού Πανεπιστημίου Αθηνών (ΟΠΑ) και κάποιων ακόμη εκθέσεων της Εθνικής Τράπεζας της Ελλάδας (ΕΤΕ). Έτσι, τα είδη των κειμένων που συγκεντρώθηκαν ποικίλλουν, καθώς πρόκειται για άρθρα εφημερίδων, συνεντεύξεις, επιστημονικές μελέτες κ.ά. τα οποία καλύπτουν τις βασικότερες περιοχές ενδιαφέροντος του οικονομικού κλάδου.

Πρωταρχικός σκοπός του έργου DELOS ήταν η δημιουργία ενός ελληνο-αγγλικού λεξικού όρων οικονομικής ορολογίας, να χρησιμοποιηθεί, δηλαδή, το DELOS ως μια λεξικογραφική εφαρμογή για οικονομική ορολογία. Έπειτα, το DELOS αναδιαρθρώθηκε, με σκοπό να είναι μια εύκολα χρησιμοποιήσιμη, τυποποιημένη και ομοιόμορφα επισημειωμένη κειμενική βάση δεδομένων για όσους επιχειρούν μελέτες σχετικές με την επεξεργασία κειμένου. Η χρήση του αυτή λαμβάνει χώρα και στην παρούσα πτυχιακή, καθώς τα παραδείγματα πάνω στα οποία πραγματοποιείται η μηχανική μάθηση προέκυψαν από τα κατάλληλα σημεία του σώματος κειμένων DELOS.

Όλες οι πληροφορίες που αφορούν το DELOS στην παρούσα πτυχιακή, καθώς και εκτενή περιγραφή αυτού, παρουσιάζεται στο paper με τίτλο «DELOS: An Automatically Tagged Economic Corpus for Modern Greek» των Katia Lida Kermanidis, Nikos Fakotakis, George Kokkinakis.

4.2 ΑΡΧΙΚΗ ΜΟΡΦΗ .TXT ΑΡΧΕΙΟΥ DELOS

PP[Στα<SpT-npa*o> ενοποιημένα<A----a*unknown> *αποτελέσματα<N-npa*αποτέλεσμα> ,<F>] PP[εκτός<R0--*εκτός> από<Sp*από> τη<T-fsa*o> *μητρική<A--fsa*μητρικός> ,<F>] VP[περιλαμβάνονται<V-i-3p--p-*περιλαμβάνω> και<C*και>] NP[οι<T-mpn*o> *εταιρίες<N-fpn*εταιρία> Νίκας<N-fsg*unknown>] NP[*Θεσσαλονίκη<N-fsa*Θεσσαλονίκη> *A.E.<Xa> .<F>] CON[,<F>] NP[Νίκας<A--fsn*unknown> *Σπάρτη<N-fsn*Σπάρτη> *A.E.<Xa> .<F>] CON[,<F>] NP[*Κως<N-fs-*unknown> *A.E.<Xa> .<F>] CON[,<F>] NP[*Νίκας<A--fsn*unknown>] NP[*Κρήτη<N-fsa*Κρήτη> *ABET<Xf> και<C*και>] NP[Νίκας<A--fsg*unknown> *Λάρισα<N-fsg*unknown> *A.E.<Xa> .<F>]#

NP[*ΚΕΡΔΗ<N-fsa*unknown> ύψους<N-nsg*ύψος>] ADP[άνω<R0--*άνω>] NP[των<T-mpg*o> *6<X> *δισ<Xb> *δρχ<Xb>] VP[αναμένεται<V-i-3s--p-*αναμένω>] VP[ότι<C*ότι> θα<Uf*θα> έχει<V-i-3s--a-*έχω>] NP[η<T-fsn*o> *ETBA<Xf>] NP[το<T-nsn*o> *1998<X>] CON[,<F> ενώ<C*ενώ>] NP[βελτιωμένα<A----n*unknown>] VP[προβλέπεται<V-i-3s--p-*προβλέπω>] VP[ότι<C*ότι> θα<Uf*θα> είναι<V-i-3s--a-*είμαι>] NP[όλα<A--npn*όλος> τα<T-npn*o> *οικονομικά<A--npn*οικονομικός>] NP[*στοιχεία<N-npa*στοιχείο> της<T-fsg*o> Τράπεζας<N-fsg*τράπεζα> ,<F>] NP[το<T-nsn*o> *πρόγραμμα<N-nsn*πρόγραμμα> εξυγίανσης<N-fsg*εξυγίανση> ,<F>] NP[της<T-fsg*o> *οποίας<pr-fsg-*οποίος>] VP[προχωρεί<V-i-3s--a-*προχωρώ>] ADP[σταθερά<R0--*σταθερά> και<C*και>] PP[με<Sp*με> *ταχείς<N-fsa*unknown>] NP[*ρυθμούς<N-mpa*ρυθμός>]#

NP[Οι<T-mpn*o> θετικές<A--fpa*θετικός> αυτές<pp3fpa-*εγώ> *εξελίξεις<N-fpa*εξέλιξη> των<T-mpg*o> οικονομικών<N-fpg*unknown> της<T-fsg*o> ETBA<Xf> ,<F>] PP[σε<Sp*σε> *συνδυασμό<N-msa*συνδυασμός> ,<F>] PP[με<Sp*με> την<T-fsa*o> *απόφαση<N-fsa*απόφαση>] CON[που<C*που>] VP[έχει<V-i-3s--a-*έχω> ληφθεί<V---3s----*unknown>] ADP[ήδη<R0--*ήδη>] PP[από<Sp*από> *τη<T-fsa*o>] NP[*διοίκηση<N-fsn*διοίκηση> της<T-fsg*o> Τράπεζας<N-fsg*τράπεζα>] PP[για<Sp*για> *δημιουργία<N-fsa*δημιουργία> και<C*και> εμπορικού<A--nsg*εμπορικός> δικτύου<N-nsg*δίκτυο>] ADP[παράλληλα<R0--*παράλληλα>] PP[με<Sp*με> τις<T-fpa*o> *αναπτυξιακές<N-fpa*unknown> της<pp3fsg-*εγώ>] NP[*δραστηριότητες<A--fpa*unknown> ,<F>] VP[δημιουργουν<V---3p----*unknown>] NP[νέες<A--fpa*νέος> *προοπτικές<N-fpa*προοπτική>]#

VP[Ετσι<V---3s----*unknown>] VP[φαίνεται<V-i-3s--p-*φαίνομαι>] CON[ότι<C*ότι>] ADP[ήδη<R0--*ήδη>] VP[συζητείται<V---3s----*unknown>] PP[από<Sp*από> το<T-msa*o> οικονομικό<A--nsa*οικονομικός> *επιτελείο<N-nsa*επιτελείο> της<T-fsg*o> κυβέρνησης<N-fsg*κυβέρνηση> ,<F>] NP[το<T-nsn*o> *ενδεχόμενο<N-nsa*ενδεχόμενο> και<C*και>] NP[οι<T-mpn*o> *δυνατότητες<N-fpn*δυνατότητα> ,<F> εισαγωγής<N-fsg*εισαγωγή> της<T-fsg*o> ETBA<Xf>] PP[στο<SpT-msa*o> *Χρηματιστήριο<N-nsa*χρηματιστήριο>]#

…και 60.418 ακόμη εγγραφές της μορφής αυτής.

4.3 ΕΠΙΣΗΜΕΙΩΣΗ ΛΕΞΕΩΝ ΤΟΥ DELOS

Όπως προαναφέρθηκε, το σώμα κειμένων του DELOS είναι επισημειωμένο τόσο σε επίπεδο λέξεων όσο και σε επίπεδο προτάσεων. Ωστόσο, στην παρούσα πτυχιακή χρησιμοποιείται μόνο η επισημείωση των λέξεων του για την εξαγωγή των παραδειγμάτων εκπαίδευσης των μοντέλων της μηχανικής μάθησης.

Όσον αφορά, λοιπόν, την επισημείωση των λέξεων του DELOS, οι ετικέτες σήμανσής τους έχουν την προκαθορισμένη μορφή: word<tag*lemma>. Στη θέση word υπάρχει η λέξη, ακριβώς με τη μορφή της με την οποία βρίσκεται μέσα στο σώμα κειμένων. Στη θέση tag υπάρχει ένα σετ χαρακτήρων, τα οποία δίνουν τη μορφολογική πληροφορία της λέξης word, σύμφωνα με το μέρος του λόγου στο οποίο ανήκει αυτή. Και αυτό καθώς, κάθε μέρος του λόγου, όπως θα δούμε και παρακάτω, έχει τα δικά του χαρακτηριστικά, τα οποία πρέπει να δηλωθούν, και τα οποία δεν είναι ίδια για όλα τα μέρη του λόγου. Τέλος, στη θέση lemma υπάρχει το λήμμα της λέξης word όπου, με τη λέξη λήμμα, λεξικογραφικά, αναφερόμαστε στον συνηθέστερο τύπο μια λέξης στον οποίο ανήκουν ή από τον οποίο παράγονται νέες λέξεις.

Σχετικά με τα μέρη του λόγου, στη Νέα Ελληνική Γλώσσα αυτά είναι έντεκα. Στο DELOS, κάθε λέξη επισημειώνεται σύμφωνα με το μέρος του λόγου στο οποίο ανήκει. Κάποια χαρακτηριστικά είναι κοινά για κάποια μέρη του λόγου, όπως το γένος, ο αριθμός και οι πτώσεις για τα ουσιαστικά, τα άρθρα και τα επίθετα. Από την άλλη, υπάρχουν χαρακτηριστικά μοναδικά για κάποια μέρη του λόγου, όπως ο χρόνος, η φωνή και η διάθεση για τα ρήματα. Για κάθε τιμή που μπορεί να πάρει κάθε χαρακτηριστικό του κάθε μέρους του λόγου, υπάρχει ένα σύμβολο, γράμμα ή αριθμός, το οποίο συμβολίζει τη τιμή αυτή στο tag της ετικέτας επισήμανσης. Για όποιο χαρακτηριστικό του DELOS δεν υπάρχει ακριβής τιμή, στη θέση του υπάρχει μια παύλα.

5 ΣΤΑΔΙΟ ΕΠΙΛΥΣΗΣ 2^{Ο}: ΠΡΟΕΠΕΞΕΡΓΑΣΙΑ ΚΑΙ ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ ΔΕΔΟΜΕΝΩΝ

5.1 ΕΡΓΑΛΕΙΑ

Τα εργαλεία τα οποία χρησιμοποιήθηκαν για την περάτωση της προεπεξεργασίας και του μετασχηματισμού των δεδομένων της παρούσας πτυχιακής εργασίας είναι ένας κώδικας σε γλώσσα προγραμματισμού Java, για τον κατάλληλο μετασχηματισμό του αρχικού .txt αρχείου DELOS, καθώς και το λογισμικό Microsoft Access και ο Microsoft SQL Server, για τη δημιουργία και τη διαχείριση της βάσης δεδομένων.

5.2 ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ ΑΡΧΙΚΟΥ .TXT ΑΡΧΕΙΟΥ DELOS

Αρχικά, το .txt αρχείο το οποίο περιέχει το σώμα κειμένων DELOS μετασχηματίζεται κατάλληλα και έτσι να ξεκινήσει η διαδικασία δημιουργίας της βάσης δεδομένων. Για τον μετασχηματισμό του χρησιμοποιείται ο κώδικας Java που παρουσιάζεται στη συνέχεια και ο οποίος δημιουργήθηκε ώστε να «σπάσει» την πληροφορία του αρχικού αρχείου στις επιμέρους λέξεις του σώματος κειμένου καθώς και στις μορφολογικές πληροφορίες που έχουμε για καθεμία από αυτές. Και αυτό διότι η έρευνα, όπως έχει προαναφερθεί, συμβαίνει σε επίπεδο λέξεων.

Έτσι, ο παρακάτω κώδικας απομονώνει κάθε λέξη του σώματος κειμένου ξεχωριστά και για καθεμία από αυτές συγκρατεί την ετικέτα επισημείωσής της καθώς και τον χαρακτηρισμό (επισημείωση) και τον αύξοντα αριθμό της πρότασης στην οποία ανήκει.

5.2.1 ΚΩΔΙΚΑΣ JAVA

```
let fs = require("fs");
//import vivliothikis gia na diaxwrizei ta strings
var DelimiterStream = require('delimiter-stream');
//setarisma vivliothikis gia diaxwrismo me ton xaraktira #
let linestream = new DelimiterStream({
    delimiter: '#'
});
//decoder import
let StringDecoder = require('string_decoder').StringDecoder;
```

```
//decode me charset utf-8
let decoder = new StringDecoder('utf8');
//anoigma stream gia input arxeio
let input = fs.createReadStream('data.txt');
//anoigma stream gia output arxeio
let logger = fs.createWriteStream('output.txt')

//parakatw einai 3 regular expressions
//arxika afou exoume kanei delimite to arxeio me ton xaraktira # ka8e
fora epexergazomaste alli protasi
//opote me to 1o r.e regexBrackets pairnoume ton xaraktirismo tis
frasis px NP/CON/VP..
//me to 2o regexItems,pairnoume ta upoloipa lexi,riza,ktl..

const regexBrackets = /(\S+)\[(.*?)\]/gm;
const regexItems = /([^\[
*]+)(?:[<])([^*<>]+)(?:[*])([^*<>]+)(?:[>])/gm;

//arxikopoihsh metavlitwn
let m1,m2,m3;
let matchesArr=[];
let sentencesArr=[];
let sentenceCounter=0;
let currentBrId;
let str="";


//ekmetaleuomenoi ta regular expressions me loopes sta groups pou
prokuptoun pernoume to apotelesma pou 8eloume
linestream.on('data', function(chunk) {
    let match =  decoder.write(chunk);

    while ((m2 = regexBrackets.exec(match)) !== null) {
        if (m2.index === regexBrackets.lastIndex) {
            regexBrackets.lastIndex++;
        }

        m2.forEach((match2, groupIndex2) => {

            if(groupIndex2==1) currentBrId=match2;

            if(groupIndex2==2){
                while ((m3 = regexItems.exec(match2)) !== null) {
                    if (m3.index === regexItems.lastIndex) {
                        regexItems.lastIndex++;
                    }

                    // let
row={lexi:m3[1],idLexis:m3[2],riza:m3[3],idGroup:currentBrId,numOfSen
tence:sentenceCounter}
                    // matchesArr.push(row);

                    //teliko apotelesma pou kanoume save sto output
arxeio
                    let sentence=`${m3[1]} ${m3[2]} ${m3[3]}
${currentBrId} ${sentenceCounter}\n`
                    logger.write(sentence)
                }
            }
        });
    }
```

```
    sentenceCounter++;
},
console.log("finished"));

input.pipe(linestream);
```

5.2.2 ΜΟΡΦΗ ΜΕΤΑΣΧΗΜΑΤΙΣΜΕΝΟΥ .TXT ΑΡΧΕΙΟΥ DELOS

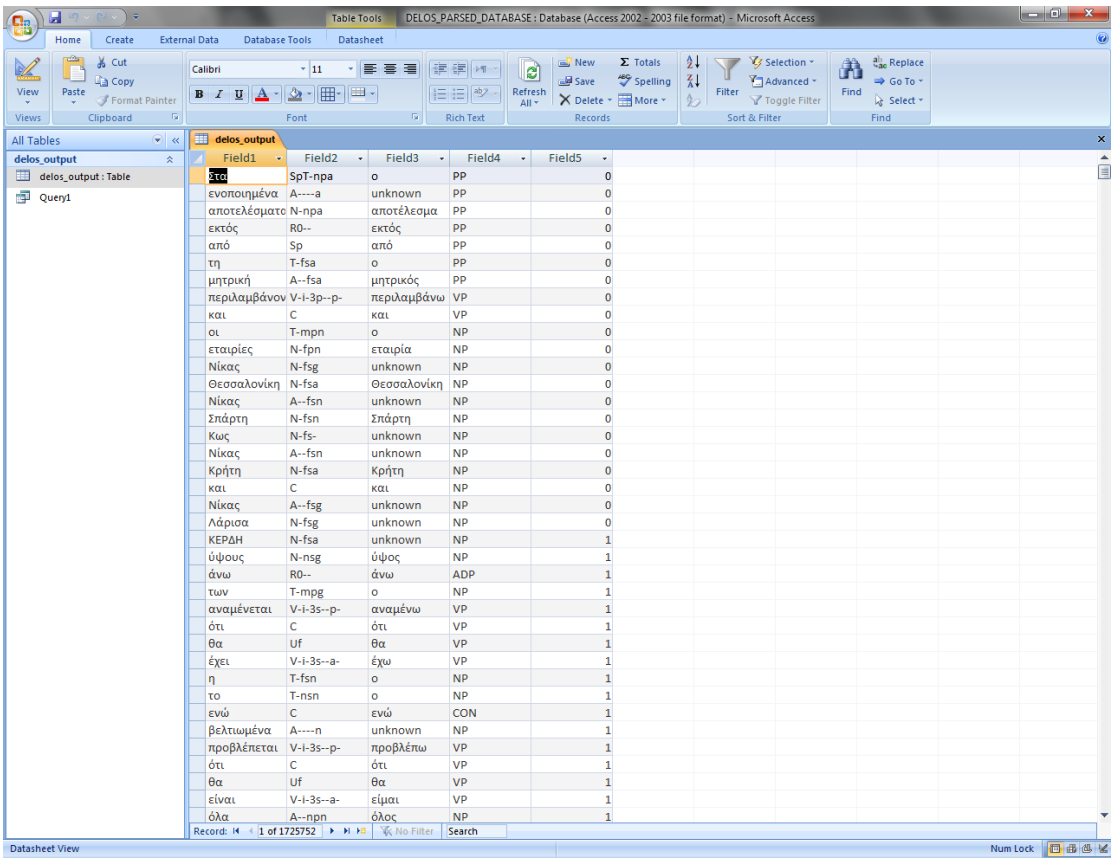
Στα SpT-npa ο PP 0
ενοποιημένα A----a unknown PP 0
αποτελέσματα N-npa αποτέλεσμα PP 0
εκτός R0-- εκτός PP 0
από Sp από PP 0
τη T-fsa ο PP 0
μητρική A--fsa μητρικός PP 0
περιλαμβάνονται V-i-3p--p- περιλαμβάνω VP 0
και C και VP 0
οι T-mpn ο NP 0
εταιρίες N-fpn εταιρία NP 0
Νίκας N-fsg unknown NP 0
Θεσσαλονίκη N-fsa Θεσσαλονίκη NP 0
Νίκας A--fsn unknown NP 0
Σπάρτη N-fsn Σπάρτη NP 0
Κως N-fs- unknown NP 0
Νίκας A--fsn unknown NP 0
Κρήτη N-fsa Κρήτη NP 0
και C και NP 0
Νίκας A--fsg unknown NP 0
Λάρισα N-fsg unknown NP 0
ΚΕΡΔΗ N-fsa unknown NP 1
ύψους N-nsg ύψος NP 1
άνω R0-- άνω ADP 1
των T-mpg ο NP 1
αναμένεται V-i-3s--p- αναμένω VP 1
ότι C ότι VP 1
θα Uf θα VP 1
έχει V-i-3s--a- έχω VP 1
η T-fsn ο NP 1
το T-nsn ο NP 1
ενώ C ενώ CON 1

βελτιωμένα A----n unknown NP 1
προβλέπεται V-i-3s--p- προβλέπω VP 1
ότι C ότι VP 1
θα Uf θα VP 1
είναι V-i-3s--a- είμαι VP 1
όλα A--npn όλος NP 1
τα T-npn ο NP 1
οικονομικά A--npn οικονομικός NP 1
στοιχεία N-npa στοιχείο NP 1
της T-fsg ο NP 1
Τράπεζας N-fsg τράπεζα NP 1
το T-nsn ο NP 1
πρόγραμμα N-nsn πρόγραμμα NP 1
εξυγίανσης N-fsg εξυγίανση NP 1
της T-fsg ο NP 1
οποίας pr-fsg- οποίος NP 1
προχωρεί V-i-3s--a- προχωρώ VP 1
σταθερά R0-- σταθερά ADP 1
και C και ADP 1
με Sp με PP 1
ταχείς N-fsa unknown PP 1
ρυθμούς N-mpa ρυθμός NP 1

…και 1.725.752 ακόμη εγγραφές της μορφής αυτής.

5.3 ΕΙΣΑΓΩΓΗ ΔΕΔΟΜΕΝΩΝ ΣΤΗΝ MICROSOFT ACCESS

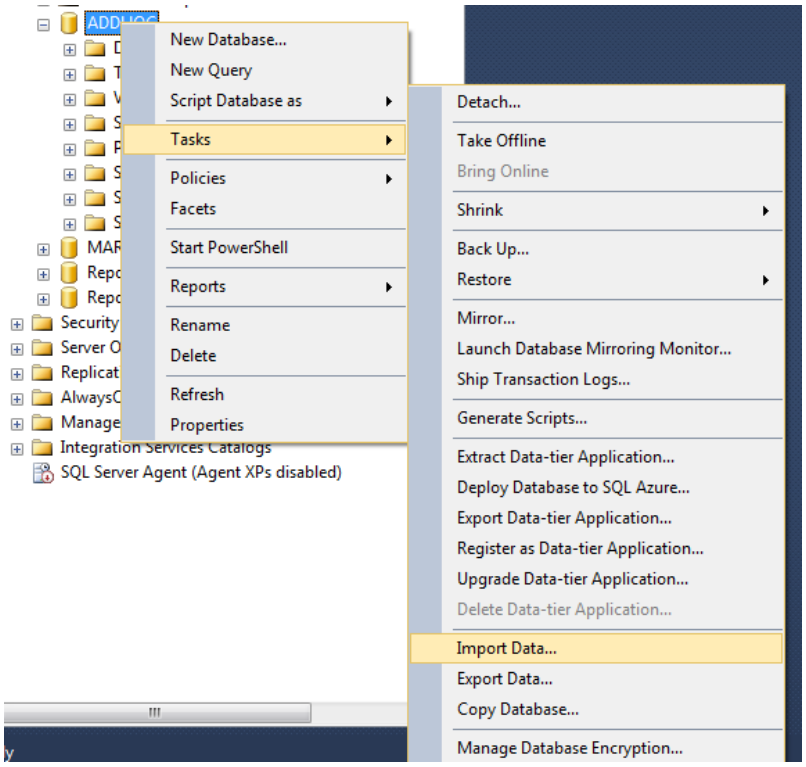
Εν συνεχεία, το μετασχηματισμένο αρχείο .txt το οποίο προέκυψε από την εφαρμογή του κώδικα Java, εισήχθηκε σε μια βάση δεδομένων μέσω της Microsoft Access καθώς, λόγω του πλήθους των εγγραφών, το λογισμικό Microsoft Excel δεν μπορούσε να υποστηρίξει την ενέργεια αυτή.



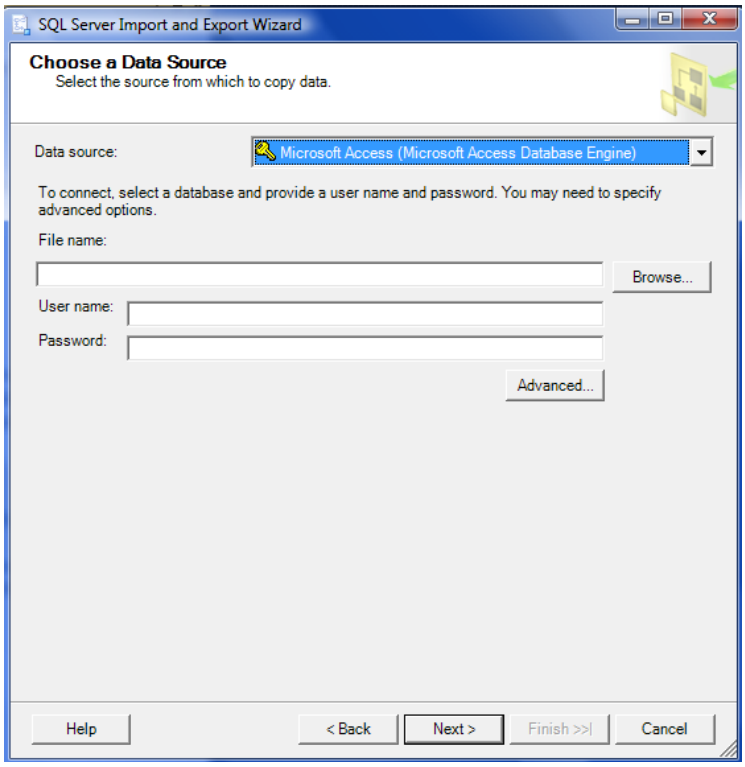
Εικόνα 1: Ενδεικτική οθόνη δεδομένων στην Microsoft Access

5.4 ΕΙΣΑΓΩΓΗ ΒΔ ΣΤΟΝ MICROSOFT SQL SERVER

Η χρήση μιας βάσης δεδομένων στην παρούσα πτυχιακή εργασία κρίθηκε σκόπιμη λόγω του μεγάλου μεγέθους της πληροφορίας η οποία υπήρχε προς διαχείριση. Ο Microsoft SQL Server επιλέχθηκε λαμβάνοντας υπόψη πως ως περιβάλλον διαχείρισης βάσεων δεδομένων δεν υστερεί σε τίποτα από άλλες γνωστές λύσεις, όπως ORACLE, MySQL κ.ά., και ο τρόπος διαχείρισης των εγγραφών μέσω αυτού κάλυπτε πλήρως τις ανάγκες που υπήρχαν.



Εικόνα 2: Ενδεικτική οθόνη εισαγωγής ΒΔ στον SQL Server



Εικόνα 3: Ενδεικτική οθόνη εισαγωγής ΒΔ στον SQL Server – Σύνδεση με Microsoft Access

5.5 ΔΙΑΧΕΙΡΙΣΗ ΒΔ ΜΕΣΩ MICROSOFT SQL MANAGEMENT STUDIO

5.5.1 SQL QUERY 1^{Ο}

Με το SQL query αυτό δημιουργείται ένας πίνακας στον οποίο μεταφέρονται όλες οι εγγραφές και κάθε εγγραφή παίρνει έναν μοναδικό αναγνωριστικό αριθμό, μέσω της εντολής IDENTITY.

```
CREATE TABLE [dbo].[delos_output2]
    ( ID INT IDENTITY(1,1) NOT NULL ,
     [LEXH] [nvarchar](50) NULL,
     [TAGGING] [nvarchar](50) NULL,
     [RIZA] [nvarchar](50) NULL,
     [MEROS_LOGOU] [nvarchar](10) NULL,
     [PROTASH] [int] NULL
) ON [PRIMARY]

GO


INSERT INTO [dbo].[delos_output2]
SELECT [LEXH]
       ,[TAGGING]
       ,[RIZA]
       ,[MEROS_LOGOU]
       ,[PROTASH]
  FROM [dbo].[delos_output]

GO
```

5.5.2 SQL QUERY 2^{Ο}

Με το SQL query αυτό δημιουργείται ένας πίνακας όπου κάθε χαρακτήρας, δηλαδή κάθε τιμή κάθε χαρακτηριστικού, των ετικετών επισημείωσης καταχωρείται σε διαφορετική στήλη. Ο διαχωρισμός αυτός και η καταχώρηση σε διαφορετικές στήλες γίνεται μέσω της εντολής SUBSTRING. Σκοπός είναι, στη συνέχεια της μελέτης, να έχουμε εύκολη πρόσβαση στις τιμές οποιονδήποτε χαρακτηριστικών κρίνουμε χρήσιμα για τη μελέτη μας.

```
SELECT
[ID]
,[LEXH]
```

```
,[TAGGING]
,LEFT(TAGGING,1) AS  TAG0
,SUBSTRING(TAGGING,2,1) AS TAG1
,SUBSTRING(TAGGING,3,1) AS TAG2
,SUBSTRING(TAGGING,4,1) AS TAG3
,SUBSTRING(TAGGING,5,1)AS TAG4
,SUBSTRING(TAGGING,6,1) AS TAG5
,SUBSTRING(TAGGING,7,1)  AS TAG6
,SUBSTRING(TAGGING,8,1)  AS TAG7
,SUBSTRING(TAGGING,9,1)  AS TAG8
,[RIZA]
,[XARAKTIRISMOS]
,[PROTASH]
INTO    [ADDHOC].[dbo].[delos_output_tags]

FROM [ADDHOC].[dbo].[delos_output2]

GO
```

5.5.3 SLQ QUERY 3^{O}

Με το SQL query αυτό πραγματοποιείται η δημιουργία τριών νέων στηλών στον αρχικό πίνακα ο οποίος περιέχει τις εγγραφές με τις λέξεις του σώματος κειμένου. Μέσα από το paper στο οποίο περιγράφεται το DELOS, προκύπτει πως τα χαρακτηριστικά case (πτώση), gender (γένος) και number (αριθμός) είναι τα μοναδικά κοινά χαρακτηριστικά για κάποια μέρη του λόγου (όχι όλα), λαμβάνουν κοινές τιμές (case: n/g/a, gender: m,/f/n και number: s/p) και είναι επισημειωμένα με τη μεγαλύτερη ακρίβεια σε σχέση με τα υπόλοιπα χαρακτηριστικά. Έτσι, έπειτα από προσεκτική παρατήρηση του τρόπου επισημείωσης κάθε λέξης βάσει του μέρους του λόγου που το χαρακτηρίζει, με το SQL query καταχωρείται η τιμή των case, gender και number για κάθε λέξη, εάν αυτή διαθέτει τις πληροφορίες αυτές.

```
SELECT *, '-' as CASE1, '-' AS GENDER , TAG5 AS NUMBER

INTO [dbo].[FINAL_DATASET2]

FROM [ADDHOC].[dbo].[delos_output_tags]
where tag0='V'
UNION ALL
SELECT *, [TAG4] as CASE1, [TAG2] AS GENDER ,TAG3
FROM [ADDHOC].[dbo].[delos_output_tags]
where tag0='N'
UNION ALL
SELECT *,  [TAG5] as CASE1, [TAG3] AS GENDER ,TAG4
FROM [ADDHOC].[dbo].[delos_output_tags]
where tag0='A'
UNION ALL
SELECT *,  [TAG5] as CASE1, [TAG3] AS GENDER,TAG4
```

```
FROM [ADDHOC].[dbo].[delos_output_tags]
where tag0='P'
UNION ALL
SELECT *, [TAG4] as CASE1 , [TAG2] AS GENDER ,TAG3
FROM [ADDHOC].[dbo].[delos_output_tags]
where tag0='T'
UNION ALL
SELECT *, [TAG4] as CASE1 , [TAG2] AS GENDER ,TAG3
FROM [ADDHOC].[dbo].[delos_output_tags]
where tag0='M'
UNION ALL
SELECT *, '-' as CASE1 ,'-'  AS GENDER ,'-'
FROM [ADDHOC].[dbo].[delos_output_tags]
WHERE tag0 IN ('C','R','S','U',' ')
order by [ID]

GO
```

5.5.4 SQL QUERY 4^{O}

Με το SQL query αυτό δημιουργείται ένας πίνακας στον οποίο τοποθετούνται όλες οι εμφανίσεις των λέξεων-στόχων (που, πού, πως, πώς). Έτσι, στον πίνακα αυτό έχουμε τις όλες τις επιμέρους θέσεις στις οποίες συναντώνται οι λέξεις-στόχοι.

```
SELECT *

  INTO ADDHOC.[dbo].[DELOS_4]
    FROM ADDHOC.[dbo].[FINAL_DATASET2]

where LEXH='που'

union all

SELECT *
  FROM ADDHOC.[dbo].[FINAL_DATASET2]
where LEXH='πού'

union all

SELECT *
  FROM ADDHOC.[dbo].[FINAL_DATASET2]
where LEXH='πως'

union all

SELECT *
  FROM ADDHOC.[dbo].[FINAL_DATASET2]
where LEXH='πώς'
```

5.5.5 SQL QUERY 5^{Ο}

Με το SQL query αυτό δημιουργείται ένας πίνακας ο οποίος, εκτός της λέξης-στόχου και της επισημείωσής της, περιέχει επίσης την προηγούμενη και την επόμενή της λέξη, με τις χρήσιμες πληροφορίες της επισημείωσης κάθε μιας. Η προηγούμενη και η επόμενη λέξη της λέξη-στόχου βρίσκονται με τη βοήθεια του αναγνωριστικού αριθμού που έχει ήδη αποδοθεί σε κάθε λέξη μέσω του 1^{ου} SQL query, που παρουσιάστηκε παραπάνω .

```
SELECT  A.[PROTASH],
       B.ID AS ID_PRIN,B.[TAG0] AS MEROS_LOGOU_PRIN,  B.[LEXH] AS
LEXH_PRIN ,B.[TAGGING] AS TAGGING_PRIN ,B.[RIZA] AS RIZA_PRIN
,B.[XARAKTIRISMOS] AS XARAKTIRISMOS_PRIN,
       B.CASE1 AS CASEPRIN, B.[GENDER] AS GENDERPRIN, B.[NUMBER] AS
NUMBERPRIN
       ,A.[ID] ,A.[TAG0] AS MEROS_LOGOU, A.[LEXH] , A.[TAGGING] ,
A.[RIZA], A.[XARAKTIRISMOS] ,  A.CASE1 AS CASE1, A.[GENDER] AS
GENDER, A.[NUMBER] AS NUMBER
       , C.ID As ID_META ,C.[TAG0] AS MEROS_LOGOU_META, C.[LEXH] As
LEXH_META ,C.[TAGGING] AS TAGGING_META ,C.[RIZA] as RIZA_META
,C.[XARAKTIRISMOS] AS XARAKTIRISMOS_META
        ,C.CASE1 AS CASEMETA, C.[GENDER] AS GENDERMETA, C.[NUMBER] AS
NUMBERMETA

INTO FINAL_DATASET3

FROM [ADDHOC].[dbo].[DELOS_4] A

INNER JOIN  [dbo].[FINAL_DATASET2]  B ON  A.[PROTASH]=B.[PROTASH] AND
A.ID= B.ID +1

INNER JOIN  [dbo].[FINAL_DATASET2] C ON  A.[PROTASH]=C.[PROTASH] AND
A.ID= C.ID -1

ORDER BY A.PROTASH
```

5.5.6 SQL QUERY 6^{Ο}

Με το SQL query αυτό δημιουργείται ένας πίνακας ο οποίος, εκτός της λέξης-στόχου και της επισημείωσής της, περιέχει επίσης τις δύο προηγούμενες και τις δύο επόμενές της λέξεις, με τις χρήσιμες πληροφορίες της επισημείωσης κάθε μιας. Οι προηγούμενες και οι επόμενες λέξεις της λέξη-στόχου βρίσκονται με τη βοήθεια του αναγνωριστικού αριθμού που έχει ήδη αποδοθεί μέσω του 1^{ου} SQL query, που παρουσιάστηκε παραπάνω.

```
SELECT A.[PROTASH],

B1.ID AS ID_2PRIN ,B1.[TAG0] AS MEROS_LOGOU_2PRIN,B1.[LEXH] AS
LEXH_2PRIN ,B1.[TAGGING] AS TAGGING_2PRIN  ,
B1.[RIZA] AS RIZA_2PRIN ,B1.[XARAKTIRISMOS] AS XARAKTIRISMOS_2PRIN
,B1.CASE1 AS CASE_2PRIN, B1.[GENDER] AS GENDER2PRIN, B1.[NUMBER] AS
NUMBER2PRIN,

B.ID AS ID_PRIN,B.[TAG0] AS MEROS_LOGOU_PRIN,  B.[LEXH] AS LEXH_PRIN
,B.[TAGGING] AS TAGGING_PRIN ,B.[RIZA] AS RIZA_PRIN
,B.[XARAKTIRISMOS] AS XARAKTIRISMOS_PRIN,
B.CASE1 AS CASEPRIN, B.[GENDER] AS GENDERPRIN, B.[NUMBER] AS
NUMBERPRIN
,A.[ID] ,A.[TAG0] AS MEROS_LOGOU, A.[LEXH] , A.[TAGGING] , A.[RIZA],
A.[XARAKTIRISMOS] ,  A.CASE1 AS CASE1, A.[GENDER] AS GENDER,
A.[NUMBER] AS NUMBER
, C.ID As ID_META ,C.[TAG0] AS MEROS_LOGOU_META, C.[LEXH] As
LEXH_META ,C.[TAGGING] AS TAGGING_META ,C.[RIZA] as RIZA_META
,C.[XARAKTIRISMOS] AS XARAKTIRISMOS_META
,C.CASE1 AS CASEMETA, C.[GENDER] AS GENDERMETA, C.[NUMBER] AS
NUMBERMETA,
C1.ID As ID_2META ,C1.[TAG0] AS MEROS_LOGOU_2META, C1.[LEXH] As
LEXH_2META ,C1.[TAGGING] AS TAGGING_2META ,C1.[RIZA] as RIZA_2META
,C1.[XARAKTIRISMOS] AS XARAKTIRISMOS_2META
,C1.CASE1 AS CASE2META, C1.[GENDER] AS GENDER2META, C1.[NUMBER] AS
NUMBER2META

INTO FINAL_DATASET4

FROM [ADDHOC].[dbo].[DELOS_4] A

INNER JOIN  [dbo].[FINAL_DATASET2]  B1 ON  A.[PROTASH]=B1.[PROTASH]
AND A.ID= B1.ID +2

INNER JOIN  [dbo].[FINAL_DATASET2]  B ON  A.[PROTASH]=B.[PROTASH] AND
A.ID= B.ID +1

INNER JOIN  [dbo].[FINAL_DATASET2] C ON  A.[PROTASH]=C.[PROTASH] AND
A.ID= C.ID -1

INNER JOIN  [dbo].[FINAL_DATASET2]  C1 ON  A.[PROTASH]=C1.[PROTASH]
AND A.ID= C1.ID -2

ORDER BY A.PROTASH
```

5.6 ΚΑΤΑΣΚΕΥΗ ΑΡΧΕΙΩΝ .ARFF

Στο σημείο αυτό, πρέπει να δημιουργηθούν τα .arff (attribute-relation file format) αρχεία τα οποία θα εισαχθούν στο λογισμικό WEKA ώστε να πραγματοποιηθεί η μηχανική μάθηση. Για το σκοπό αυτό, τα αποτελέσματα που επιστρέφονται από την εκτέλεση των κατάλληλων SQL queries, αντιγράφονται σε ένα αρχείο του Microsoft Excel και αποθηκεύονται σε ένα αρχείο της μορφής CSV (διαχωρισμένο με κόμματα - comma delimited). Εν συνεχεία, το .csv αρχείο αυτό επεξεργάζεται μέσω του λογισμικού Notepad++,

ώστε ο χαρακτήρας «;» ο οποίος διαχωρίζει τα δεδομένα των διαφορετικών στηλών, να αντικατασταθεί από τον χαρακτήρα «,». Τέλος, δημιουργούνται χειρωνακτικά τα .arff αρχεία, με την σωστή παραμετροποίηση των attributes τους και την αντιγραφή των διαχωρισμένων με «,» εγγραφών στο τμήμα «@data» των αρχείων.

```
4  @attribute MEROS_LOGOU_PRIN {A,C,N,p,R,S,T,V}
5  @attribute CASE_PRIN {-,a,g,n}
6  @attribute GENDER_PRIN {-,f,m,n}
7  @attribute NUMBER_PRIN {-,p,s}
8  %@attribute MEROS_LOGOU {A,N,R}
9  @attribute MEROS_LOGOU_META {A,C,M,N,p,R,S,T,U,V}
10 @attribute CASEMETA {-,a,g,n}
11 @attribute GENDERMETA {-,f,m,n}
12 @attribute NUMBER_META {-,p,s}
13 @attribute CLASS {Q_POS,POS}
14
15
16 @data
17 N,a,n,s,U,-,-,-,POS
18 V,-,-,p,T,n,n,p,POS
19 V,-,-,s,T,n,m,s,POS
20 V,-,-,s,V,-,-,p,POS
21 V,-,-,p,T,n,f,s,POS
22 N,a,f,s,U,-,-,-,POS
23 N,n,f,s,U,-,-,-,POS
24 A,n,f,s,R,-,-,-,POS
25 V,-,-,p,V,-,-,p,POS
26 V,-,-,s,S,-,-,-,POS
27 N,g,f,s,M,-,f,s,POS
28 N,a,m,s,S,-,-,-,POS
29 V,-,-,s,p,a,f,s,POS
30 R,-,-,-,T,n,f,s,POS
31 N,a,f,s,T,n,m,p,POS
32 N,n,f,s,R,-,-,-,POS
33 N,n,n,p,V,-,-,s,POS
34 N,a,f,s,V,-,-,s,POS
```

Εικόνα 4: Ενδεικτικό τμήμα .arff αρχείου

5.7 ΤΕΛΙΚΑ DATASETS ΜΕΛΕΤΗΣ

5.7.1 DATASETS ΠΟΥ ΑΦΟΡΟΥΝ ΤΟ ΣΕΤ ΛΕΞΕΩΝ «ΠΟΎ-ΠΟΥ»

5.7.1.1 DATASET 1ο

Το πρώτο σύνολο εκπαίδευσης το οποίο δημιουργήθηκε αφορά το σετ ομόηχων λέξεων «πού-που». Πιο συγκεκριμένα, το σύνολο εκπαίδευσης αυτό περιέχει, εκτός της λέξης-στόχου και της χρήσιμης πληροφορίας της επισημείωσής της, την προηγούμενη και την επόμενη λέξη της λέξης-στόχου, καθώς και τις πληροφορίες των επισημειώσεων τους που κρίθηκαν χρήσιμες για τη δημιουργία του μοντέλου. Τα χαρακτηριστικά του 1ου συνόλου εκπαίδευσης είναι επτά.

```
@attribute MEROS_LOGOU_PRIN {A,C,M,N,p,R,S,T,U,V}
@attribute CASE_PRIN {-,a,g,n}
@attribute GENDER_PRIN {-,f,m,n}
@attribute NUMBER_PRIN {-,p,s}
@attribute MEROS_LOGOU_META {A,C,M,N,p,R,S,T,U,V}
@attribute NUMBER_META {-,p,s}
@attribute CLASS {Q_POU,POU}
```

Εικόνα 5: Attributes 1ου dataset

Ο πίνακας συχνοτήτων για το σύνολο εκπαίδευσης αυτό είναι ο παρακάτω:

Κλάση	Πλήθος
πού (Q_POU)	34
που (POU)	670

5.7.1.2 DATASET 2ο

Το δεύτερο σύνολο εκπαίδευσης το οποίο δημιουργήθηκε αφορά επίσης το σετ ομόηχων λέξεων «πού-που». Πιο συγκεκριμένα, το σύνολο εκπαίδευσης αυτό περιέχει, εκτός της λέξης-στόχου και της χρήσιμης πληροφορίας της επισημείωσής της, τις δύο προηγούμενες και τις δύο επόμενες λέξεις της λέξης-στόχου, καθώς και τις πληροφορίες των επισημειώσεων τους που κρίθηκαν χρήσιμες για τη δημιουργία του μοντέλου. Τα χαρακτηριστικά του 2ου συνόλου εκπαίδευσης είναι είκοσι.

```
@attribute MEROS_LOGOU_2PRIN {A,C,M,N,p,R,S,T,U,V,}
@attribute CASE_2PRIN {-,a,g,n}
@attribute GENDER2PRIN{-,f,m,n}
@attribute NUMBER2PRIN {-,p,s}
@attribute MEROS_LOGOU_PRIN {A,C,M,N,p,R,S,T,U,V}
@attribute  CASEPRIN {-,a,g,n}
@attribute GENDERPRIN{-,f,m,n}
@attribute NUMBERPRIN{-,p,s}
@attribute CASE1 {-,a,g,n}
@attribute GENDER {-,m,n}
@attribute NUMBER {-,s}
@attribute MEROS_LOGOU_META {A,C,M,N,p,R,S,T,U,V}
@attribute CASEMETA {-,a,g,n}
@attribute GENDERMETA {-,f,m,n}
@attribute NUMBERMETA {-,p,s}
@attribute MEROS_LOGOU_2META {A,C,M,N,p,R,S,T,U,V }
@attribute CASE2META {-,a,g,n}
@attribute GENDER2META {-,f,m,n}
@attribute NUMBER2META {-,g,p,s}
@attribute CLASS {Q_POU,POU}
```

Εικόνα 6: Attributes 2ου dataset

Ο πίνακας συχνοτήτων για το σύνολο εκπαίδευσης αυτό είναι ο παρακάτω:

Κλάση	Πλήθος
πού (Q_POU)	46
που (POU)	9478

5.7.2 DATASETS ΠΟΥ ΑΦΟΡΟΥΝ ΤΟ ΣΕΤ ΛΕΞΕΩΝ «ΠΏΣ-ΠΩΣ»

5.7.2.1 DATASET 3^{ο}

Το τρίτο σύνολο εκπαίδευσης το οποίο δημιουργήθηκε αφορά το σετ ομόηχων λέξεων «πώς-πως». Πιο συγκεκριμένα, το σύνολο εκπαίδευσης αυτό περιέχει, εκτός της λέξης-στόχου και της χρήσιμης πληροφορίας της επισημείωσής της, την προηγούμενη και την επόμενη λέξη της λέξης-στόχου, καθώς και τις πληροφορίες των επισημειώσεων τους που κρίθηκαν χρήσιμες για τη δημιουργία του μοντέλου. Τα χαρακτηριστικά του 3^{ου} συνόλου εκπαίδευσης είναι εννέα.

```
@attribute MEROS_LOGOU_PRIN {A,C,N,p,R,S,T,V}
@attribute CASE_PRIN {-,a,g,n}
@attribute GENDER_PRIN {-,f,m,n}
@attribute NUMBER_PRIN {-,p,s}
%@attribute MEROS_LOGOU {A,N,R}
@attribute MEROS_LOGOU_META {A,C,M,N,p,R,S,T,U,V}
@attribute CASEMETA {-,a,g,n}
@attribute GENDERMETA {-,f,m,n}
@attribute NUMBER_META {-,p,s}
@attribute CLASS {Q_POS,POS}
```

Εικόνα 7: Attributes 3ου dataset

Ο πίνακας συχνοτήτων για το σύνολο εκπαίδευσης αυτό είναι ο παρακάτω:

Κλάση	Πλήθος
πώς (Q_POS)	83
πως (POS)	265

5.7.2.2 DATASET 4ο

Το τέταρτο σύνολο εκπαίδευσης το οποίο δημιουργήθηκε αφορά επίσης το σετ ομόηχων λέξεων «πώς-πως». Πιο συγκεκριμένα, το σύνολο εκπαίδευσης αυτό περιέχει, εκτός της λέξης-στόχου και της χρήσιμης πληροφορίας της επισημείωσής της, τις δύο προηγούμενες και τις δύο επόμενες λέξεις της λέξης-στόχου, καθώς και τις πληροφορίες των επισημειώσεων τους που κρίθηκαν χρήσιμες για τη δημιουργία του μοντέλου. Τα χαρακτηριστικά του 2^{ου} συνόλου εκπαίδευσης είναι είκοσι.

```
@attribute MEROS_LOGOU_2PRIN {A,C,M,N,p,R,S,T,U,V,}
@attribute CASE_2PRIN {-,a,g,n}
@attribute GENDER2PRIN{-,f,m,n}
@attribute NUMBER2PRIN {-,p,s}
@attribute MEROS_LOGOU_PRIN {A,C,M,N,p,R,S,T,U,V}
@attribute  CASEPRIN {-,a,g,n}
@attribute GENDERPRIN{-,f,m,n}
@attribute NUMBERPRIN{-,p,s}
@attribute CASE1 {-,a,g,n}
@attribute GENDER {-,f,m,n}
@attribute NUMBER {-,s}
@attribute MEROS_LOGOU_META {A,C,M,N,p,R,S,T,U,V}
@attribute CASEMETA {-,a,g,n}
@attribute GENDERMETA {-,f,m,n}
@attribute NUMBERMETA {-,p,s}
@attribute MEROS_LOGOU_2META {A,C,M,N,p,R,S,T,U,V }
@attribute CASE2META {-,a,g,n}
@attribute GENDER2META {-,f,m,n}
@attribute NUMBER2META {-,g,p,s}
@attribute CLASS {Q_POS,POS}
```

Εικόνα 8: Attributes 4ου dataset

Ο πίνακας συχνοτήτων για το σύνολο εκπαίδευσης αυτό είναι ο παρακάτω:

Κλάση	Πλήθος
πού (Q_POU)	134
που (POU)	879

6 ΣΤΑΔΙΟ ΕΠΙΛΥΣΗΣ 3^{Ο}: ΠΕΙΡΑΜΑ

6.1 ΕΡΓΑΛΕΙΟ ΠΕΙΡΑΜΑΤΟΣ: ΛΟΓΙΣΜΙΚΟ WEKA

Η σουίτα λογισμικού WEKA (Waikato Environment for Knowledge Analysis) είναι μια ολοκληρωμένη σουίτα λογισμικού για όσους ασχολούνται με τους αλγορίθμους μηχανικής μάθησης. Αναπτύχθηκε στο Πανεπιστήμιο του Waikato της Νέας Ζηλανδίας, από την Ομάδα Μηχανικής Μάθησης του πανεπιστημίου, το 1993. Χαρακτηρίζεται ως ολοκληρωμένη λύση για τους χρήστες οι οποίοι πειραματίζονται με τη μηχανική μάθηση καθώς τους προσφέρει δυνατότητα για προεπεξεργασία των δεδομένων, συσταδοποίηση, παλινδρόμηση, κατηγοριοποίηση, κανόνες συσχέτισης, οπτικοποίηση των δεδομένων, ακόμα και ανάπτυξη αλγορίθμων.

Το WEKA αρχικά κατασκευάστηκε στην γλώσσα προγραμματισμού C όμως, λόγω προβλημάτων που αντιμετώπισε η ομάδα ανάπτυξης και συντήρησής του, το 1997, 4 μόλις χρόνια μετά την κατασκευή του, αποφασίστηκε η αναδημιουργία του. Έτσι, δομήθηκε εξ αρχής και εξ ολοκλήρου στην γλώσσα προγραμματισμού Java, στην οποία βρίσκεται έως και σήμερα. Το γεγονός αυτό του έδωσε αφενός πολλές περισσότερες δυνατότητες από όσες είχε στην αρχική του μορφή και αφετέρου του εξασφάλισε φορητότητα σε σχεδόν κάθε σύγχρονη πλατφόρμα. Έτσι, σήμερα είναι συμβατό με Windows, Mac OS και Linux. Η τρέχουσα σταθερή έκδοση (stable version) που κυκλοφορεί είναι η 3.8 ενώ η έκδοση για προγραμματιστές (development version) είναι η 3.9, χωρίς αυτό να σημαίνει πως δεν μπορεί κάποιος να βρει στο διαδίκτυο και τις προηγούμενες εκδόσεις του λογισμικού.

Επιπρόσθετα, το WEKA είναι ένα ελεύθερο λογισμικό, ή εναλλακτικά «λογισμικό ανοιχτού κώδικα», ένα λογισμικό, δηλαδή, που μπορεί να χρησιμοποιηθεί, να μελετηθεί, να τροποποιηθεί και να διανεμηθεί χωρίς περιορισμό από οποιονδήποτε χρήστη. Η διάθεσή του γίνεται δωρεάν και υπό τη Γενική Άδεια Δημόσιας Χρήσης (General Public License) GNU, υπό την προστασία της οποίας κυκλοφορούν τα περισσότερα ελεύθερα λογισμικά σήμερα. Το γεγονός πως πρόκειται για ένα λογισμικό ανοιχτού κώδικα κάνει το WEKA ευρέως χρησιμοποιήσιμο από χρήστες και προγραμματιστές που ενδιαφέρονται για τους αλγορίθμους μηχανικής μάθησης και έτσι υπάρχουν πολλά αποθετήρια, με πολλές γραμμές κώδικα γραμμένες για το WEKA, στο GitHub, αλλά και πλήθος συζητήσεων μεταξύ χρηστών και προγραμματιστών, που θέλουν να αλληλοβοηθηθούν ή απλά να ανταλλάξουν απόψεις γύρω από λειτουργίες και δυνατότητες του WEKA, στο Stack Overflow.

Το 2006, η σουίτα λογισμικού WEKA ξεκινά να υποστηρίζεται οικονομικά από την εταιρία Pentaho, μια εταιρία με έδρα το Ορλάντο, η οποία διαθέτει το ομώνυμο λογισμικό εκμετάλλευσης της επιχειρηματικής ευφυΐας (business intelligence software) Pentaho, επίσης ανοιχτού κώδικα. Μετά την ανάληψη της οικονομικής υποστήριξης του WEKA από την εταιρία Pentaho, το WEKA συνδέθηκε με το λογισμικό Pentaho με στόχο την διεκπεραίωση εργασιών εξόρυξης δεδομένων.

6.2 ΤΕΛΙΚΑ ΕΞΙΣΟΡΡΟΠΗΜΕΝΑ DATASET ΜΕΛΕΤΗΣ

6.2.1 SMOTE

Το SMOTE (Synthetic Minority Oversampling TEchnique) δημιουργήθηκε το 2002 από τους Nitesh V. Chawla et al. [11] και είναι μια τεχνική υπερδειγματοληψίας, στην οποία η κλάση των μειονοτήτων είναι υπερδειγματολειπτική με τη δημιουργία «συνθετικών» παραδειγμάτων.

Στο λογισμικό WEKA, το SMOTE αποτελεί ένα από τα πιο γνωστά φίλτρα στην κατηγορία της επιβλεπόμενης μάθησης και, πιο συγκεκριμένα, στην κατηγορία των φίλτρων εκείνων της επιβλεπόμενης μάθησης που εφαρμόζονται στα παραδείγματα του συνόλου εκπαίδευσης. Εφαρμόζεται κατά τη διαδικασία της προεπεξεργασίας με σκοπό την αποφυγή της υπερεξειδίκευσης (overfitting) του μοντέλου και δημιουργεί νέα συνθετικά παραδείγματα στην κλάση η οποία έχει τα λιγότερα. Αποτέλεσμα είναι η διαφορά μεταξύ του πλήθους των παραδειγμάτων των κλάσεων να εξισορροπείται, έως έναν βαθμό.

6.2.2 ΕΞΙΣΟΡΡΟΠΗΜΕΝΑ DATASETS ΓΙΑ ΣΕΤ ΛΕΞΕΩΝ «ΠΟΥ-ΠΟΥ»

6.2.2.1 DATASET 5ο

Για τη δημιουργία του πέμπτου συνόλου εκπαίδευσης εφαρμόστηκε το φίλτρο SMOTE τέσσερις φορές. Το σύνολο εκπαίδευσης αυτό αποτελεί την εξισορροπημένη μορφή του πρώτου συνόλου εκπαίδευσης και ο πίνακας συχνοτήτων του είναι ο παρακάτω:

Κλάση	Πλήθος
πού (Q_POU)	544
που (POU)	670

6.2.2.1 DATASET 6ο

Για τη δημιουργία του έκτου συνόλου εκπαίδευσης εφαρμόστηκε το φίλτρο SMOTE επτά φορές. Το σύνολο εκπαίδευσης αυτό αποτελεί την εξισορροπημένη μορφή του δεύτερου συνόλου εκπαίδευσης και ο πίνακας συχνοτήτων του είναι ο παρακάτω:

Κλάση	Πλήθος
πού (Q_POU)	5888
που (POU)	9478

5.8.3 ΕΞΙΣΟΡΡΟΠΗΜΕΝΑ DATASETS ΓΙΑ ΣΕΤ ΛΕΞΕΩΝ «ΠΏΣ-ΠΩΣ»

6.2.3.1 DATASET 7ο

Για τη δημιουργία του έβδομου συνόλου εκπαίδευσης εφαρμόστηκε το φίλτρο SMOTE μία φορά. Το σύνολο εκπαίδευσης αυτό αποτελεί την εξισορροπημένη μορφή του τρίτου συνόλου εκπαίδευσης και ο πίνακας συχνοτήτων του είναι ο παρακάτω:

Κλάση	Πλήθος
πώς (Q_POS)	166
πως (POS)	265

6.2.3.2 DATASET 8ο

Για τη δημιουργία του όγδοου και τελευταίου συνόλου εκπαίδευσης εφαρμόστηκε το φίλτρο SMOTE δύο φορές. Το σύνολο εκπαίδευσης αυτό αποτελεί την εξισορροπημένη μορφή του τέταρτου συνόλου εκπαίδευσης και ο πίνακας συχνοτήτων του είναι ο παρακάτω:

Κλάση	Πλήθος
πώς (Q_POS)	536
πως (POS)	879

6.3 ΚΑΤΗΓΟΡΙΕΣ ΤΑΞΙΝΟΜΗΤΩΝ WEKA

Το WEKA διαθέτει πλήθος διαθέσιμων ταξινομητών τους οποίους και έχει ομαδοποιήσει στις παρακάτω κατηγορίες:

- **bayes**

  Στην κατηγορία αυτή συμπεριλαμβάνεται ένα σετ αλγορίθμων κατηγοριοποίησης με κοινό τους χαρακτηριστικό το ότι βασίζονται στο θεώρημα του Bayes, όπως ο αλγόριθμος Naive Bayes κ.ά.
- **function**

  Στην κατηγορία αυτή συγκαταλέγεται ένα σετ συναρτήσεων παλινδρόμησης, όπως η γραμμική παλινδρόμηση (linear regression) κ.ά.
- **lazy**

  Στην κατηγορία αυτή συμπεριλαμβάνονται οι λεγόμενοι «τεμπέληδες» αλγόριθμοι μάθησης, όπως ο αλγόριθμος των k-Πλησιέστερων Γειτόνων (k-Nearest Neighbors) κ.ά.
- **meta**

  Στην κατηγορία αυτή συμπεριλαμβάνονται συνδυαστικές μέθοδοι αλλά και αλγόριθμοι μείωσης των διαστάσεων, όπως ο AdaBoost και ο Bagging, οι οποίοι χρησιμοποιούν τις μεθόδους boosting και bagging αντίστοιχα, κ.ά.
- **misc**

  Στην κατηγορία αυτή συμπεριλαμβάνονται ταξινομητές οι οποίοι μπορούν να χρησιμοποιηθούν ώστε να γίνουν προβλέψεις, έχοντας ως βάση ένα προ-εκπαιδευμένο μοντέλο. Τέτοιοι ταξινομητές είναι ο SerializedClassifier κ.ά.
- **rules**

Στην κατηγορία αυτή απαντώνται αλγόριθμοι οι οποίοι βασίζονται σε κανόνες συσχέτισης, όπως ο ZeroR κ.ά.

- **trees**
  Στην κατηγορία αυτή συγκαταλέγονται οι αλγόριθμοι οι οποίοι βασίζονται στα δέντρα αποφάσεων, όπως ο Random Forest κ.ά.

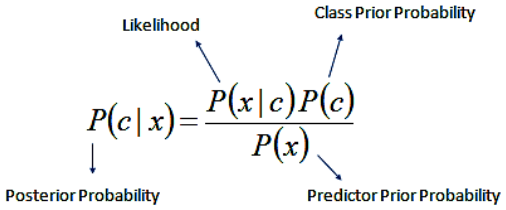
6.4 ΤΑΞΙΝΟΜΗΤΕΣ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΗΘΗΚΑΝ

6.4.1 ZERO-R

Ο αλγόριθμος ZeroR χρησιμοποιείται ώστε να βρεθεί η βασική γραμμή (baseline) – σημείο αναφοράς ώστε να υπάρχει ένα κριτήριο για την αξιολόγηση της επιτυχίας ή όχι μιας μεθόδου ταξινόμησης για το μοντέλο. Και αυτό διότι, για να μπορούν να συγκριθούν τα ποσοστά επιτυχίας διαφόρων ταξινομητών, θα πρέπει να υπάρχει κάποιο σημείο αναφοράς που να προκύπτει από μια απλή διαδικασία. Η απλή αυτή διαδικασία που ακολουθεί ο ZeroR είναι πως αγνοεί όλα τα χαρακτηριστικά των στιγμιοτύπων και, βασιζόμενος στην μεταβλητή εξόδου και μόνο, προβλέπει τελικά την κλάση στην οποία ανήκει η πλειοψηφία των στιγμιοτύπων. Έτσι, αφού κατασκευάζει έναν πίνακα συχνοτήτων για τη μεταβλητή εξόδου, επιλέγει τη τιμή που πλεονεκτεί και κατασκευάζει έναν πίνακα σύγχυσης ο οποίος αποδεικνύει πως ο ZeroR απλώς προβλέπει σωστά την κλάση με τα περισσότερα στιγμιότυπα. Ο ZeroR ανήκει στην κατηγορία ταξινομητών rules του WEKA.

6.4.2 NAIVE BAYES

Ο ταξινομητής Naive Bayes (Αφελής Ταξινομητής Bayes) είναι ένας από τους διαφόρους ταξινομητές της κατηγορίας bayes του WEKA αφού βασίζεται στο θεώρημα του Bayes, το οποίο περιγράφεται από τη σχέση:



$$P(c \mid \mathrm{X}) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

όπου:

- **P(c|x)**: ονομάζεται «εκ των υστέρων πιθανότητα» και εκφράζει τη δεσμευμένη πιθανότητα να ισχύει το c, δεδομένου ότι ισχύει το x,
- **P(x|c)**: ονομάζεται «πιθανοφάνεια» και είναι η πιθανότητα να ισχύει το x, δεδομένου ότι ισχύει το c,
- **P(c)**: ονομάζεται «εκ των προτέρων πιθανότητα» και εκφράζει την πιθανότητα να είναι αληθές το c και, τέλος,
- **P(x)**: ονομάζεται «σταθερά κανονικοποίησης» και εκφράζει την πιθανότητα να είναι αληθές το x.

Ο Naive Bayes βασίζεται σε δύο απλές υποθέσεις, γι' αυτό και αποκαλείται αφελής, χωρίς αυτό να σημαίνει πως δεν είναι αποτελεσματικός. Η πρώτη υπόθεση είναι πως η παρουσία ενός χαρακτηριστικού σε μία τάξη δε σχετίζεται με την παρουσία οποιουδήποτε άλλου χαρακτηριστικού. Πιο συγκεκριμένα, ο ταξινομητής υποθέτει πως η επίδραση της αξίας του προγνωστικού (x) σε μια δεδομένη κατηγορία (c) είναι ανεξάρτητη από τις τιμές των άλλων προγνωστικών και η υπόθεση αυτή ονομάζεται «τάξη ανεξαρτησίας υπό όρους». Η δεύτερη υπόθεση είναι πως δεν υπάρχουν άλλα κρυφά χαρακτηριστικά που να επηρεάζουν τη διαδικασία της πρόβλεψης.

6.4.3 ID3

Ο ID3 (Iterative Dichotomizer 3) είναι ένας από τους δημοφιλέστερους αλγόριθμους κατασκευής δέντρων απόφασης, γι' αυτό και ανήκει στην κατηγορία ταξινομητών trees του WEKA. Πρόκειται για έναν επαγωγικό αλγόριθμο ο οποίος πραγματοποιεί αναδρομικά μια άπληστη αναζήτηση, αφού, σε κάθε κόμβο του δέντρου που παράγει, αναζητά το χαρακτηριστικό εκείνο το οποίο διαχωρίζει καλύτερα τα δεδομένα δείγματα, χωρίς ωστόσο να ανατρέχει σε προηγούμενα χαρακτηριστικά που έχει χρησιμοποιήσει κατά τη δεδομένη πορεία του ώστε να τα επανεξετάσει.

Ένα από τα κυριότερα χαρακτηριστικά του ID3 είναι πως χρησιμοποιεί στατιστικές μεθόδους, όπως αυτές της εντροπίας πληροφορίας (information entropy) και του κέρδους πληροφορίας (information gain), ώστε να καθορίσει βάσει ποιού χαρακτηριστικού θα πραγματοποιηθεί ο διαχωρισμός των δειγμάτων.

Όσον αφορά την εντροπία πληροφορίας, αυτή εκφράζει το βαθμό βεβαιότητας ενός συνόλου δεδομένων S και δίνεται από τον παρακάτω τύπο:

$$Entropy\ (S)\ = \sum_{1=n}^{n} -p_i\ log p_i$$

όπου p1, p2, …, pi οι πιθανότητες του κάθε ενδεχομένου που περιλαμβάνει το σύνολο.

Εάν τα δείγματα ενός συνόλου εκπαίδευσης είναι ομοιογενή ως προς μία κατηγορία, η εντροπία ισούται με μηδέν ενώ, εάν οι κατηγορίες είναι διαφορετικές και τα δείγματα που ανήκουν σε καθεμία από αυτές είναι ισοπληθή, η εντροπία ισούται με τη μονάδα.

Από την άλλη πλευρά, όσον αφορά το κέρδος πληροφορίας, αυτό εκφράζει το πόση πληροφορία «φέρει» το χαρακτηριστικό A ενός συνόλου δειγμάτων S, το οποίο S αποτελεί τμήμα του συνόλου εκπαίδευσης. Το κέρδος πληροφορίας εκφράζεται από τον τύπο:

$$G(S,A) = E(S) - \sum_{i=1}^{m} f_s\,(A_i) * E\left(S_{A_i}\right)$$

όπου:

- **E(…)** η συνάρτηση εντροπίας,
- **m** το πλήθος των τιμών A_i που παίρνει το A στο S,
- **$f_s(A_i)$** το ποσοστό των δειγμάτων στο S που παίρνουν τη τιμή A_i και
- **S_{Ai}** το υποσύνολο του S όπου η τιμή του A είναι A_i.

Έτσι, συγκεντρωτικά, η βασική διαδικασία με την οποία ο ID3 επιχειρεί να διαχωρίσει ένα μη ομοιογενές σύνολο δειγμάτων σε έναν κόμβο που ανήκει σε ένα συγκεκριμένο κλαδί του δέντρου απόφασης, είναι η εξής:

1. Υπολογισμός της εντροπίας για όλα τα αχρησιμοποίητα χαρακτηριστικά του συγκεκριμένου κλαδιού σε σχέση με τα δείγματα.
2. Επιλογή του χαρακτηριστικού με το μεγαλύτερο κέρδος πληροφορίας.
3. Κατασκευή κόμβο για το χαρακτηριστικό αυτό.

Η διαδικασία αυτή επαναλαμβάνεται αναδρομικά και ο ID3 σταματά όταν κάποιο χαρακτηριστικό διαχωρίζει πλήρως το σύνολο εκπαίδευσης.

6.4.4 J48

Ο αλγόριθμος J48 είναι μια υλοποίηση σε Java του αλγορίθμου C4.5, φτιαγμένη για το λογισμικό WEKA. Ο αλγόριθμος C4.5, με τη σειρά του, αποτελεί εξέλιξη του ID3. Πιο συγκεκριμένα, ενώ ο ID3 αντιμετωπίζει μόνο ονοματικές μεταβλητές, ο C4.5 αντιμετωπίζει τόσο ονοματικές όσο και αριθμητικές μεταβλητές.

Τα βασικά βήματα τα οποία ακολουθεί ο C4.5 είναι τα εξής:

1. Επιλογή του χαρακτηριστικού εκείνου που διαχωρίζει καλύτερα τις κατηγορίες βάσει του μέτρου του λόγου του κέρδους πληροφορίας.
2. Διαχωρισμός των δεδομένων σε υποσύνολα, βάσει των τιμών του επιλεγμένου χαρακτηριστικού.
3. Επανάληψη της διαδικασίας αυτής για τα υποσύνολα που περιέχουν περισσότερες από μία κατηγορίες.
4. Ο αλγόριθμος τερματίζει είτε όταν δεν υπάρχουν άλλα υποσύνολα τα οποία να περιέχουν περισσότερες από μία κατηγορίες είτε όταν έχουν εξεταστεί όλα τα χαρακτηριστικά.

6.4.5 RANDOM FOREST

Η μέθοδος Random Forest είναι αλληλένδετη με τη μέθοδο των δέντρων απόφασης. Τα βήματα τα οποία ακολουθεί ο Random Forest είναι, σε γενικά πλαίσια, τα εξής:

1. Ανάπτυξη πολλών δέντρων απόφασης.
2. Κάθε δέντρο απόφασης δίνει μια ταξινόμηση και έτσι, μπορούμε να πούμε, πως κάθε δέντρο «ψηφίζει» μια κλάση.
3. Έτσι, ένας αριθμός «ψήφων» συγκεντρώνεται για κάθε κλάση.
4. Για τη τελική ταξινόμηση, ο Random Forest επιλέγει την κλάση με τις περισσότερες ψήφους.

6.5 ΑΠΟΤΕΛΕΣΜΑΤΑ ΤΑΞΙΝΟΜΗΤΩΝ

6.5.1 ΑΠΟΤΕΛΕΣΜΑΤΑ DATASETS ΓΙΑ ΣΕΤ ΛΕΞΕΩΝ «ΠΟΥ-ΠΟΥ»

6.5.1.1 ΑΠΟΤΕΛΕΣΜΑΤΑ 1ου DATASET

Ταξινομητής	Ποσοστό επιτυχίας
ZeroR	95,1705%
J48	95,1705%
Random Forest	90,7670%
Naive Bayes	89,3466%
Id3	84,0909%

6.5.1.2 ΑΠΟΤΕΛΕΣΜΑΤΑ 2ου DATASET

Ταξινομητής	Ποσοστό επιτυχίας
ZeroR	99,5170%
J48	99,5170%
Random Forest	99,98%
Id3	99,98%
Naive Bayes	99,97%

6.5.1.3 ΑΠΟΤΕΛΕΣΜΑΤΑ 5ου DATASET

Ταξινομητής	Ποσοστό επιτυχίας
J48	94,6458%
Random Forest	93,6573%
Id3	91,1038%
Naive Bayes	85,5025%
ZeroR	55,1895%

6.5.1.4 ΑΠΟΤΕΛΕΣΜΑΤΑ 6ου DATASET

Ταξινομητής	Ποσοστό επιτυχίας
Random Forest	99,4859%
J48	99,3362%
Id3	98,8676%
Naive Bayes	89,3857%
ZeroR	61,6816%

6.5.2 ΑΠΟΤΕΛΕΣΜΑΤΑ DATASETS ΓΙΑ ΣΕΤ ΛΕΞΕΩΝ «ΠΩΣ-ΠΩΣ»

6.5.2.1 ΑΠΟΤΕΛΕΣΜΑΤΑ 3ου DATASET

Ταξινομητής	Ποσοστό επιτυχίας
ZeroR	76,1494%
J48	75,2874%
Naive Bayes	69,8276%
Random Forest	67,2414%
Id3	58,0460%

6.5.2.2 ΑΠΟΤΕΛΕΣΜΑΤΑ 4ου DATASET

Ταξινομητής	Ποσοστό επιτυχίας
J48	88,6476%
ZeroR	86,7720%
Random Forest	86,6732%
Naive Bayes	81,1451%
Id3	79,0721%

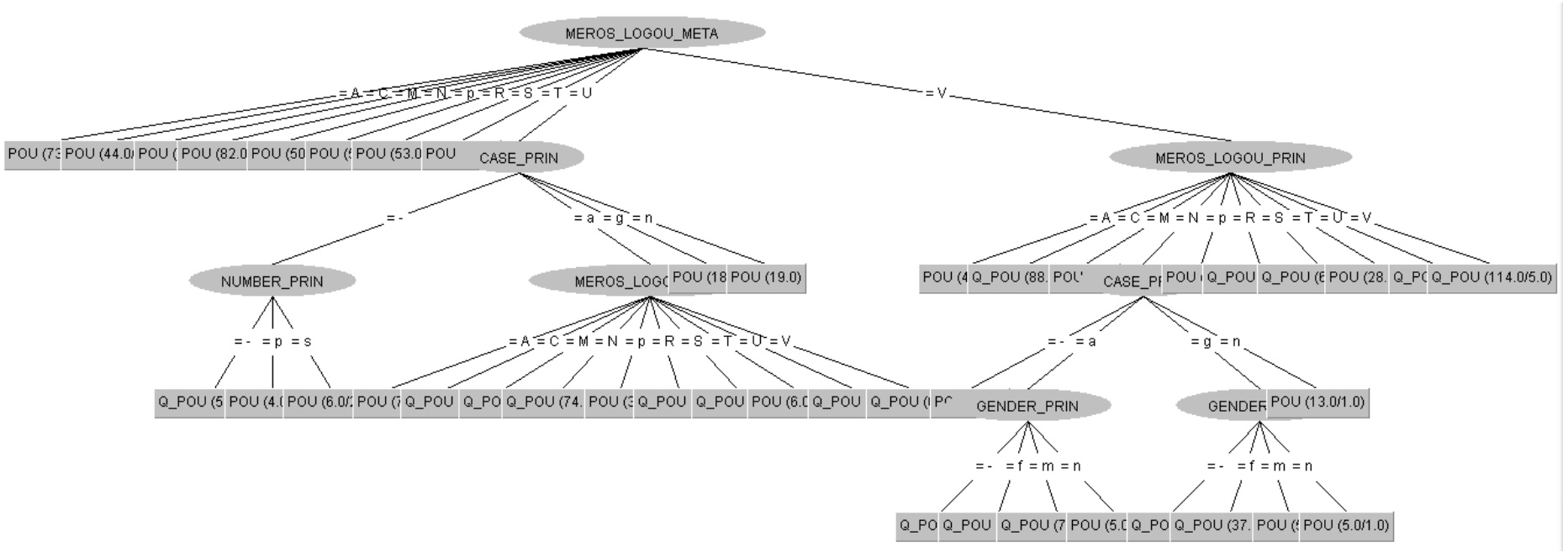
6.5.2.3 ΑΠΟΤΕΛΕΣΜΑΤΑ 7ου DATASET

Ταξινομητής	Ποσοστό επιτυχίας
J48	71,9258%
Random Forest	70,9977%
Naive Bayes	70,9977%
Id3	69,1415%
ZeroR	61,4849%

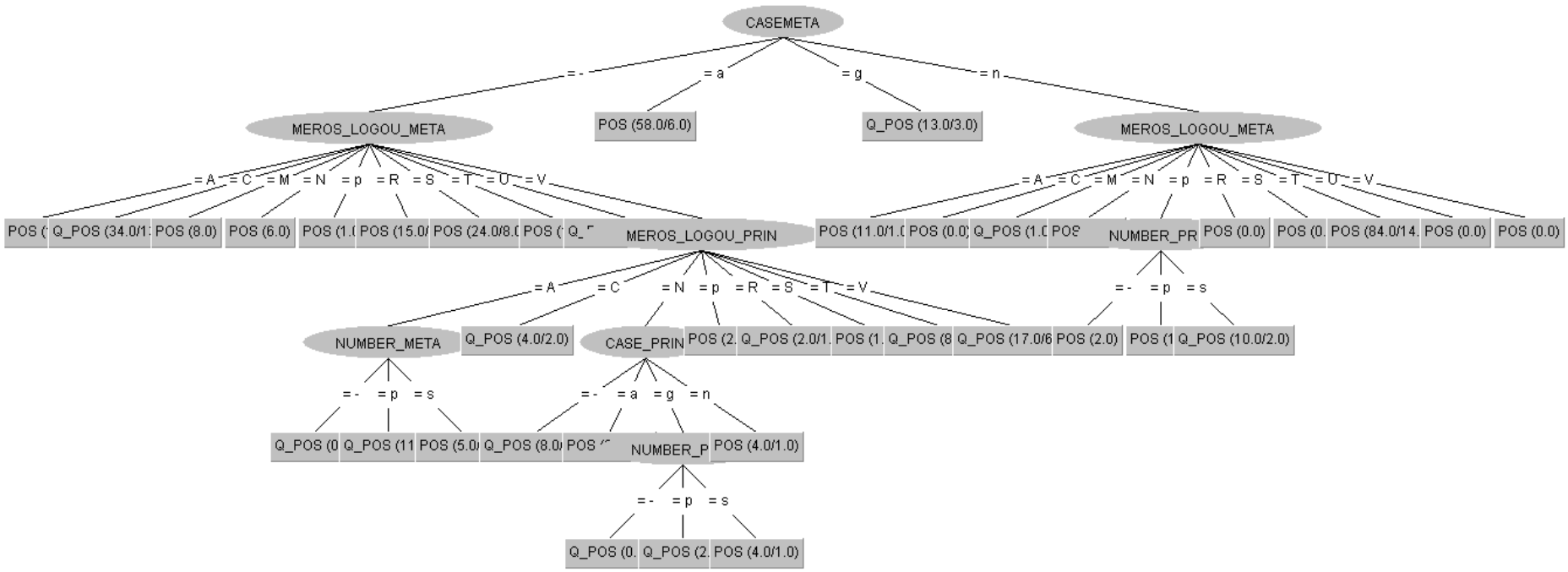
6.5.2.4 ΑΠΟΤΕΛΕΣΜΑΤΑ 8ου DATASET

Ταξινομητής	Ποσοστό επιτυχίας
Random Forest	90,8127%
J48	89,1873%
Id3	85,4417%
Naive Bayes	80,4947%
ZeroR	62,1201%

6.7 ΕΝΔΕΙΚΤΙΚΕΣ ΑΠΕΙΚΟΝΙΣΕΙΣ ΔΕΝΤΡΩΝ ΠΑΡΑΓΟΜΕΝΩΝ ΑΠΟ ΤΟΝ J48



Εικόνα 9: Απεικόνιση δέντρου J48 για το εξισορροπημένο dataset του σετ λέξεων "πού-που", με παράθυρο ±1 λέξης



Εικόνα 10: Απεικόνιση δέντρου J48 για το εξισορροπημένο dataset του σετ λέξεων "πώς-πως", με παράθυρο ±1 λέξης

7 ΣΤΑΔΙΟ ΕΠΙΛΥΣΗΣ 4^{Ο}: ΕΡΜΗΝΕΙΑ / ΑΞΙΟΛΟΓΗΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ

7.1 ΣΧΟΛΙΑΣΜΟΣ ΕΦΑΡΜΟΓΗΣ ΦΙΛΤΡΟΥ SMOTE

7.1.1 ΣΥΓΚΡΙΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ ΑΡΧΙΚΩΝ ΚΑΙ ΕΞΙΣΟΡΡΟΠΗΜΕΝΩΝ DATASETS ΓΙΑ ΣΕΤ ΛΕΞΕΩΝ «ΠΟΥ-ΠΟΥ»

Στον παρακάτω πίνακα παρουσιάζεται μια σύγκριση των ποσοστών επιτυχίας κάθε ταξινομητή για τα σύνολα εκπαίδευσης που αφορούν το σετ λέξεων «πού-που» και περιλαμβάνουν το παράθυρο ±1 λέξεων από τη λέξη στόχο. Φαίνεται, λοιπόν, πως για τους περισσότερους ταξινομητές το ποσοστό επιτυχίας δε βελτιώνεται όταν το σύνολο εκπαίδευσης είναι εξισορροπημένο, καθώς μόνο ο Id3 και ο Random Forest βελτιώνουν την επιτυχία τους στο εξισορροπημένο σύνολο εκπαίδευσης. Επιπρόσθετα, αξίζει να παρατηρηθεί πως το baseline που μας δίνει ο ZeroR μειώνεται σημαντικά.

Ταξινομητής	Αρχικό dataset (Dataset 1^{Ο})	Εξισορροπημένο dataset (Dataset 5^{Ο})
ZeroR	95,1705%	55,1895%
Naïve Bayes	89,3466%	85,5025%
Id3	84,0909%	91,1038%
J48	95,1705%	94,6458%
Random Forest	90,7670%	93,6573%

Όσον αφορά τη σύγκριση των ποσοστών επιτυχίας των ταξινομητών για το ίδιο σετ λέξεων, με παράθυρο όμως ±2 λέξεων αυτή τη φορά, παρατηρείται ξανά σημαντική πτώση του baseline που μας δίνεται από τον ZeroR, ωστόσο, γενικότερα, τα ποσοστά επιτυχίας των ταξινομητών στο εξισορροπημένο μοντέλο είναι ιδιαίτερα ικανοποιητικά.

Ταξινομητής	Αρχικό dataset (Dataset 2^{Ο})	Εξισορροπημένο dataset (Dataset 6^{Ο})
ZeroR	99,5170%	61,6816%
Naïve Bayes	99,97%	89,3857%
Id3	99,98%	98,8676%
J48	99,5170%	99,3362%
Random Forest	99,98%	99,4859%

7.1.2 ΣΥΓΚΡΙΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ ΑΡΧΙΚΩΝ ΚΑΙ ΕΞΙΣΟΡΡΟΠΗΜΕΝΩΝ DATASETS ΓΙΑ ΣΕΤ ΛΕΞΕΩΝ «ΠΩΣ-ΠΩΣ»

Στον παρακάτω πίνακα παρουσιάζεται μια σύγκριση των ποσοστών επιτυχίας των ταξινομητών όταν αυτοί εφαρμόστηκαν στα σύνολα εκπαίδευσης που αφορούν το σετ λέξεων «πώς-πως» με παράθυρο ±1 λέξεων από τη λέξη στόχο. Παρατηρείται, λοιπόν, πως οι περισσότεροι ταξινομητές βελτιώνουν τα ποσοστά τους στο εξισορροπημένο μοντέλο, με μοναδικό ταξινομητή που μειώνει σημαντικά το ποσοστό επιτυχίας του να είναι ο ZeroR.

Ταξινομητής	Αρχικό dataset (Dataset 3^{Ο})	Εξισορροπημένο dataset (Dataset 7^{Ο})
ZeroR	76,1494%	61,4849%
Naïve Bayes	69,8276%	70,9977%
Id3	58,0460%	69,1415%
J48	75,2874%	71,9258%
Random Forest	67,2414%	70,9977%

Όσον αφορά το ίδιο σετ λέξεων, με παράθυρο ±2 λέξεων αυτή τη φορά, οι ταξινομητές επιτυγχάνουν πάλι βελτιωμένα ποσοστά επιτυχίας στο εξισορροπημένο μοντέλο. Ωστόσο, και πάλι, η απόδοση του ZeroR είναι αισθητά μειωμένη στο εξισορροπημένο σύνολο εκπαίδευσης σε σχέση με την απόδοσή του στο αρχικό.

Ταξινομητής	Αρχικό dataset (Dataset 4^{Ο})	Εξισορροπημένο dataset (Dataset 8^{Ο})
ZeroR	86,7720%	62,1201%
Naïve Bayes	81,1451%	80,4947%
Id3	79,0721%	85,4417%
J48	88,6476%	89,1873%
Random Forest	86,6732%	90,8127%

7.2 ΣΧΟΛΙΑΣΜΟΣ ΕΦΑΡΜΟΓΗΣ ΠΑΡΑΘΥΡΟΥ ΛΕΞΕΩΝ

7.2.1 ΣΥΓΚΡΙΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ DATASETS ΓΙΑ ΣΕΤ ΛΕΞΕΩΝ «ΠΟΥ-ΠΟΥ»

Ο παρακάτω πίνακας παρουσιάζει μια συνολική εικόνα όσον αφορά το σετ ομόηχων λέξεων «πού-που» και δίνει μια εικόνα σχετικά με το εάν η χρήση της μίας ή της άλλης λέξης εξαρτάται περισσότερο από το παράθυρο της ±1 λέξης ή από το παράθυρο των ±2 λέξεων.

Έτσι, παρατηρείται πως όλοι οι ταξινομητές αποδίδουν καλύτερα όταν χρησιμοποιούν πληροφορίες από το παράθυρο των ±2 λέξεων από τη λέξη-στόχο. Επίσης, φαίνεται πως τα ποσοστά επιτυχίας των ταξινομητών όταν εφαρμόζονται στο παράθυρο των ±2 λέξεων, είναι εξίσου καλά τόσο στο αρχικό όσο και στο εξισορροπημένο dataset.

Ταξινομητής	**Αρχικά datasets**		**Εξισορροπημένα datasets**	
	±1 λέξη (Dataset 1^{O})	**±2 λέξεις (Dataset 2^{O})**	**±1 λέξη (Dataset 5^{O})**	**±2 λέξεις (Dataset 6^{O})**
ZeroR	95,1705%	99,5170%	55,1895%	61,6816%
Naïve Bayes	89,3466%	99,97%	85,5025%	89,3857%
Id3	84,0909%	99,98%	91,1038%	98,8676%
J48	95,1705%	99,5170%	94,6458%	99,3362%
Random Forest	90,7670%	99,98%	93,6573%	99,4859%

7.2.2 ΣΥΓΚΡΙΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ DATASETS ΓΙΑ ΣΕΤ ΛΕΞΕΩΝ «ΠΩΣ-ΠΩΣ»

Ο παρακάτω πίνακας παρουσιάζει μια συνολική εικόνα όσον αφορά το σετ ομόηχων λέξεων «πώς-πως» και δίνει μια εικόνα σχετικά με το εάν η χρήση της μίας ή της άλλης λέξης εξαρτάται περισσότερο από το παράθυρο της ±1 λέξης ή από το παράθυρο των ±2 λέξεων.

Έτσι, παρατηρείται, τόσο στα αρχικά όσο και στα εξισορροπημένα datasets που προέκυψαν από την εφαρμογή του φίλτρου SMOTE, πως το παράθυρο των ±2 λέξεων δίνει μεγαλύτερα ποσοστά επιτυχίας από το παράθυρο της ±1 λέξης και, άρα, λειτουργεί καλύτερα με όλους τους ταξινομητές. Αυτό σημαίνει, όπως είχε προκύψει και από τη μελέτη άλλων βιβλιογραφικών πηγών, πως και στην περίπτωση της μελέτης του σετ λέξεων αυτού, όσο

μεγαλύτερο είναι το παράθυρο των λέξεων, οι πληροφορίες των οποίων εξετάζονται, τόσο ορθότερα ο ταξινομητής ταξινομεί τα παραδείγματα ελέγχου.

Επίσης, παρατηρείται πως στο αρχικό dataset με παράθυρο ±2 λέξεων, την καλύτερη απόδοση παρουσιάζει ο J48 ενώ, στην εξισορροπημένη μορφή του, την καλύτερη επίδοση έχει ο Random Forest.

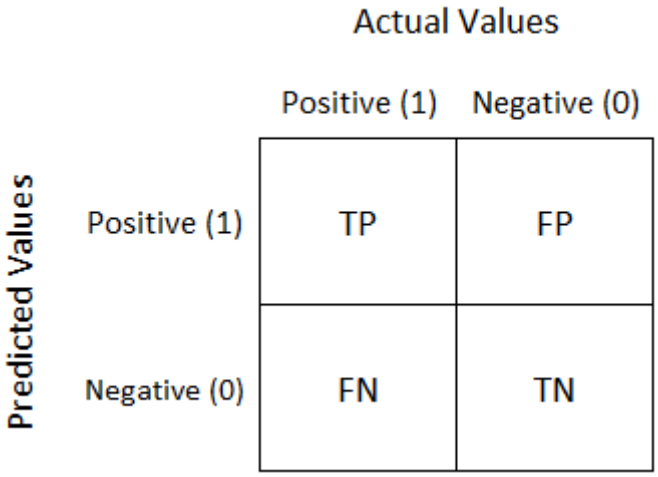
Τέλος, όσον αφορά το παράθυρο της ±1 λέξης, η καλύτερη απόδοση σημειώνεται στο αρχικό dataset με τον ZeroR ενώ, όσον αφορά το παράθυρο των ±2 λέξεων, η καλύτερη απόδοση σημειώνεται στο εξισορροπημένο dataset με τον Random Forest.

Ταξινομητής	Αρχικά datasets		Εξισορροπημένα datasets	
	±1 λέξη (Dataset 3^O)	**±2 λέξεις (Dataset 4^O)**	**±1 λέξη (Dataset 7^O)**	**±2 λέξεις (Dataset 8^O)**
ZeroR	76,1494%	86,7720%	61,4849%	62,1201%
Naïve Bayes	69,8276%	81,1451%	70,9977%	80,4947%
Id3	58,0460%	79,0721%	69,1415%	85,4417%
J48	75,2874%	88,6476%	71,9258%	89,1873%
Random Forest	67,2414%	86,6732%	70,9977%	90,8127%

7.3 ΑΛΛΑ ΜΕΤΡΑ ΑΞΙΟΛΟΓΗΣΗΣ ΤΑΞΙΝΟΜΗΤΩΝ

7.3.1 ΠΙΝΑΚΑΣ ΣΥΓΧΥΣΗΣ

Ο πίνακας σύγχυσης (confusion matrix) είναι ένας δισδιάστατος πίνακας μέσω του οποίου μπορούν να παρουσιαστούν οι επιδόσεις ανά κλάση ενός ταξινομητή. Πιο συγκεκριμένα, οι στήλες του πίνακα αναπαριστούν τις πραγματικές τιμές της κλάσης ενώ οι γραμμές τις προβλέψεις του ταξινομητή (όχι υποχρεωτικά, μπορεί να συμβεί και το αντίθετο και ανάλογα να προσαρμοστεί ο πίνακας).



Εικόνα 11: Υπόδειγμα πίνακα σύγχυσης (πηγή: Towards Data Science)

Στα κελιά ενός πίνακα σύγχυσης, λοιπόν, όπως ο παραπάνω, συναντώνται οι παρακάτω τιμές:

- TP (True Positive-Αληθινές Θετικές): το πλήθος των θετικών παρατηρήσεων που πράγματι ταξινομήθηκαν ως θετικές,
- FP (False Positive-Ψευδείς Θετικές): το πλήθος των αρνητικών παρατηρήσεων που λανθασμένα ταξινομήθηκαν ως θετικές,
- FN(False Negative-Ψευδείς Αρνητικές): το πλήθος των θετικών παρατηρήσεων που λανθασμένα ταξινομήθηκαν ως αρνητικές και, τέλος,
- TN(True Negative-Αληθείς Αρνητικές): το πλήθος των αρνητικών παρατηρήσεων που ορθά ταξινομήθηκαν ως αρνητικές από τον ταξινομητή.

Εν συνεχεία, βάσει των τιμών TP, FP, FN και TN υπάρχει δυνατότητα για τον υπολογισμό κάποιων ακόμη μέτρων.

7.3.2 ΜΕΤΡΑ ΠΟΥ ΠΡΟΚΥΠΤΟΥΝ ΑΠΟ ΠΙΝΑΚΑ ΣΥΓΧΥΣΗΣ

Το πρώτο μέτρο είναι η ακρίβεια (precision) και προκύπτει από το πλήθος των TP παρατηρήσεων προς το πλήθος των συνολικά ταξινομημένων ως θετικών παρατηρήσεων. Η ακρίβεια ουσιαστικά απαντά στο ερώτημα «ποιο ποσοστό των ταξινομημένων ως θετικών παρατηρήσεων είναι πράγματι θετικές;». Έτσι,

Precision (ακρίβεια) = TP / (TP + FP)

Το δεύτερο μέτρο είναι η ανάκληση (recall) και προκύπτει από το πλήθος των TP παρατηρήσεων προς το πλήθος των FN και TP παρατηρήσεων. Η ανάκληση ουσιαστικά απαντά στο ερώτημα «ποιο ποσοστό των πραγματικά θετικών παρατηρήσεων ταξινομήθηκαν τελικά ορθά;». Έτσι,

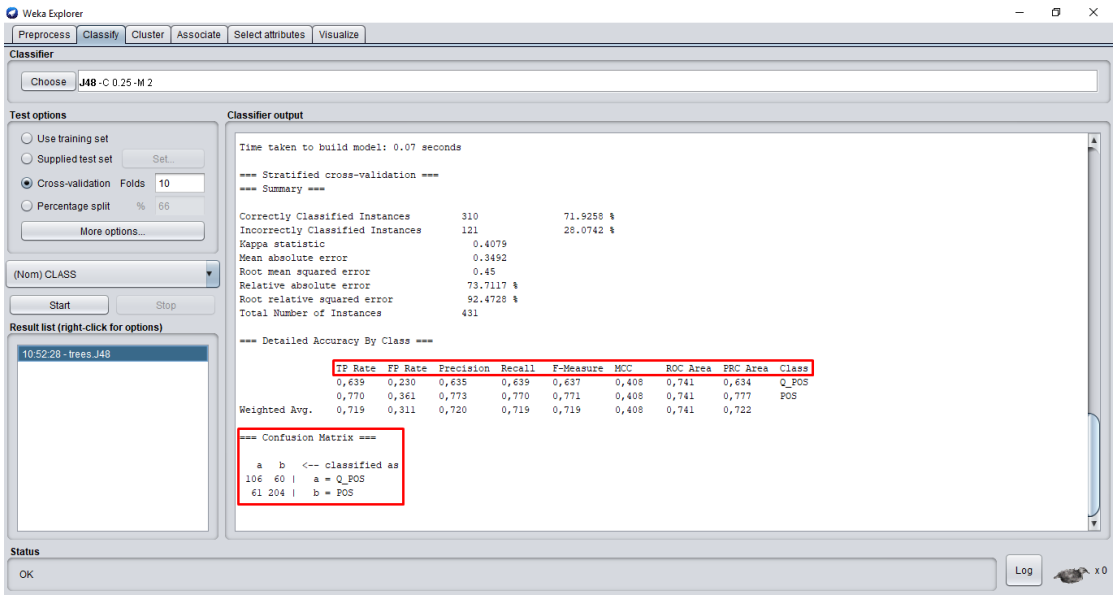
Recall (ανάκληση) = TP / (FN + TP)

Το τρίτο μέτρο είναι η ορθότητα (accuracy) και προκύπτει από το πλήθος των ορθά ταξινομημένων παρατηρήσεων προς το πλήθος των συνολικών παρατηρήσεων. Έτσι,

Accuracy = (TP + TF) / (TP + FP + FN + TN)

7.3.3 ΠΙΝΑΚΑΣ ΣΥΓΧΥΣΗΣ ΚΑΙ ΑΛΛΑ ΜΕΤΡΑ ΣΤΟ WEKA

Το περιβάλλον WEKA παρέχει στο χρήστη τον πίνακα σύγχυσης και κάποια άλλα μέτρα που προκύπτουν από αυτό υπολογισμένα, όταν ο χρήστης χρησιμοποιήσει κάποιον κατηγοριοποιητή.



Εικόνα 12: Θέση πίνακα σύγχυσης και μέτρων σε ενδεικτική οθόνη του WEKA

Όλα τα αποτελέσματα αυτά των πειραμάτων της παρούσας πτυχιακής εργασίας παρουσιάζονται συγκεντρωτικά σε αρχεία του Microsoft Excel, που δημιουργήθηκαν για την εύκολη πρόσβαση του χρήστη σε αυτά. Επίσης, δημιουργήθηκαν .txt αρχεία με το output της οθόνης αποτελεσμάτων του WEKA για όλα τα πειράματα που πραγματοποιήθηκαν.

7.4 ΜΕΛΛΟΝΤΙΚΕΣ ΒΕΛΤΙΩΣΕΙΣ

Στο μέλλον η παρούσα πτυχιακή εργασία θα μπορούσε να βελτιωθεί με ποικίλους τρόπους. Για παράδειγμα, θα μπορούσαν να εξεταστούν τα ίδια σύνολα εκπαίδευσης με άλλους ταξινομητές αλλά και να δημιουργηθούν σύνολα εκπαίδευσης με μεγαλύτερα ή διαφοροποιημένα παράθυρα λέξεων.

Ωστόσο, ο κώδικας Java που παρουσιάστηκε στην ενότητα 5.2.1 καθώς και η ακριβής περιγραφή που ακολούθησε για την εύκολη δημιουργία της βάσης δεδομένων με τις εγγραφές του DELOS, βοηθά καταλυτικά στη δημιουργία οποιουδήποτε dataset επιθυμεί ένας ερευνητής, καθώς υπάρχει δυνατότητα για απομόνωση οποιασδήποτε πληροφορίας μέσα από το DELOS απλά και μόνο με τη δημιουργία του κατάλληλου SQL query. Έτσι, στο μέλλον θα μπορούσαν να εξεταστούν και άλλα λάθη της ελληνικής γλώσσας, εκτός των σετ λέξεων που μελετήθηκαν στην παρούσα πτυχιακή.

ΒΙΒΛΙΟΓΡΑΦΙΑ

ΕΝΤΥΠΗ ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Fayyad U., Piatesky-Shaprio G., Smyth P. and Uthurusamy R., (1996): *Advances in Knowledge Discovery and Data Mining*, MIT Press, Cambridge
2. Andrew R. Golding, (1995): *A Bayesian Hydrid Method for Context-Sensitive Spelling Correction*
3. Andrew R. Golding and Yves Schabes, (1996): *Combining Trigram-Based and Feature-Based Methods for Context-Sensitive Spelling Correction*
4. Andrew R. Golding and Dan Roth, (1999): *A Winnow-Based Approach to Context-Sensitive Spelling Correction*
5. Andrew Carlson and Ian Fette, (2007): *Memory-Based Context-Sensitive Spelling Correction at Web Scale*
6. Johannes Schaback and Fang Li, (2007): *Multi-Level Feature Extraction for Spelling Correction*
7. Gasteratos P., Dragonas V., Kalamara A., Sagiadinos S., Spyridonidou A. and Kermanidis K., (2014): *Knowledge-Poor Context-Sensitive Spelling Correction for Modern Greek*
8. Katia Lida Kermanidis, Nikos Fakotakis, George Kokkinakis: *DELOS: An Automatically Tagged Economic Corpus for Modern Greek*
9. Arthur L. Samuel, (1959): Some *Studies in Machine Learning Using the Game of Checkers*, IBM Journal of Research and Development
10. Michell T., (1997): *Machine Learning*, Maidenhead, H.B.: McGraw-Hill International Editions
11. Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, W. Philip Kegelmeyer, (2002): *Smote: Synthetic Minority Over-Sampling Technique*, Journal of Artificial Intelligence Research 16

ΗΛΕΚΤΡΟΝΙΚΗ ΒΙΒΛΙΟΓΡΑΦΙΑ

12. Bernard Marr, (2016): *A Short History of Machine Learning -- Every Manager Should Read*, Forbes. Ανακτήθηκε από https://www.forbes.com/sites/bernardmarr/2016/02/19/a-short-history-of-machine-learning-every-manager-should-read/#3c7c6d8715e7