Ανοικτό Πανεπιστήμιο Κύπρου

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

Μεταπτυχιακή Διατριβή
στα Πληροφοριακά και Επικοινωνιακά Συστήματα

Συγκριτική Μελέτη Εργαλείων Μηχανικής Μάθησης στην Εξόρυξη Κειμένων

Βασιλική Ε. Μαρίνου

Επιβλέπων Καθηγητής
Ιωάννης Κατάκης

Σεπτέμβριος 2015

Ανοικτό Πανεπιστήμιο Κύπρου

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

Συγκριτική Μελέτη Εργαλείων Μηχανικής Μάθησης στην Εξόρυξη Κειμένων

Βασιλική Μαρίνου

Επιβλέπων Καθηγητής
Ιωάννης Κατάκης

Η παρούσα μεταπτυχιακή διατριβή υποβλήθηκε
προς μερική εκπλήρωση των απαιτήσεων για απόκτηση

μεταπτυχιακού τίτλου σπουδών
στα Πληροφοριακά Συστήματα

από τη Σχολή Θετικών και Εφαρμοσμένων Επιστημών
του Ανοικτού Πανεπιστημίου Κύπρου

Σεπτέμβριος 2015

Περίληψη

Το διαδίκτυο συγκεντρώνει μεγάλες ποσότητες πληροφοριών, οι οποίες λόγω του όγκου τους πολλές φορές μένουν αναξιοποίητες. Οι τεχνικές εξόρυξης γνώσης μπορούν να αυτοματοποιήσουν τη διαδικασία της ανάκτησης χρήσιμων πληροφοριών από τον ιστό και να συνδυαστούν με συστήματα που μπορούν να αξιοποιήσουν τη γνώση αυτή. Αναδεικνύεται επομένως η επιτακτική ανάγκη σύγκρισης των εργαλείων που θα διευκολύνουν την προσπέλαση και τη διαχείριση της διαθέσιμης πληροφορίας ανάλογα με τις ανάγκες των χρηστών.

Αντικείμενο της παρούσας μεταπτυχιακής διατριβής είναι μια σύγκριση σε βάθος των κορυφαίων λογισμικών ανοιχτού κώδικα, σε πολλά επίπεδα όπως ποικιλία αλγορίθμων που υλοποιούνται, ποικιλία εργαλείων προ-επεξεργασίας κειμένων, υπολογιστικοί πόροι που καταλαμβάνουν, παρουσίαση αποτελεσμάτων, κοινότητα χρηστών, ευελιξία επέκτασης, ευχρηστία κ.λ.π., με έμφαση στις διεργασίες αυτόματης ανάλυσης κειμένων όπως κατηγοριοποίηση (classification) και συσταδοποίηση (clustering), στο πεδίο της εξόρυξης κειμένου. Τα υπό εξέταση λογισμικά είναι το R, το Weka, το Rapid Miner και scikit learn (Python).

Πιο συγκεκριμένα, αρχικά παρουσιάζεται μια εισαγωγή στην έννοια της εξόρυξης κειμένου και των λογισμικών ανοιχτού κώδικα καθώς εξετάζεται η δομή, η λειτουργικότητα και οι επιμέρους εφαρμογές των υπό εξέταση λογισμικών, αλλά και η άποψη του συγγραφέα αναφορικά με τα κριτήρια ευχρηστίας προκειμένου να δημιουργηθεί μία πρώτη ιδέα ως προς την ποιότητα, τη χρησιμότητα και το περιεχόμενο του καθενός από αυτά. Στη συνέχεια, γίνεται μια αναφορά στους πιο βασικούς αλγορίθμους μηχανικής μάθησης και στις μετρικές αξιολόγησης τους, προκειμένου να πραγματοποιηθεί η συγκριτική μελέτη των πλεονεκτημάτων και των μειονεκτημάτων των υπό εξέταση λογισμικών μέσω των αποτελεσμάτων της πειραματικής ανάλυσης των παραπάνω μεθόδων.

Συνοψίζοντας, ο στόχος της παρούσας μεταπτυχιακής διατριβής είναι η πραγματοποίηση μιας συγκριτικής μελέτης των υπό εξέταση λογισμικών, ώστε ο χρήστης δεδομένου ενός προβλήματος εφαρμογής να είναι σε θέση να επιλέξει την βέλτιστη τεχνική (εργαλείο) βάσει των προτεραιοτήτων του.

Summary

Internet brings together large amounts of information, which due to their volume are often left unexploited. The data mining techniques can automate the process of recovering useful information from the web and combine them with systems that can leverage this knowledge.

This highlights an urgent need to compare the tools which facilitate the access and management of the available information depending on the user needs. Objective of this master thesis is an in-depth comparison between the leading open source software.

The comparisons are multi-leveled and include a variety of implemented algorithms, a variety of pre-processing text tools, computational resources they occupy, results presentation, user community, expansion flexibility, usability, etc. emphasizing on automated analysis processes such as text classification and text clustering in the field of text mining. The software tools under consideration is R, Weka, Rapid Miner and scikit learn (Python).

Specifically, an introduction to the concept of text mining and open source software is presented while examining structure, functionality and additional applications of each of the software tools, but also the view of the author regarding the usability criteria in order to provide an overview as to the quality, usefulness and content of each one of them.

Subsequently there is a reference on the most basic machine learning algorithms and their test metrics in order to perform a comparative study on the advantages and the disadvantages of the test software through the results of the experimental analysis of the above methods.

Summarizing, the goal of this master thesis is a comparative study between the tested software, so that users with an application problem will be able to choose the best technique (tool) based on their priorities.

Ευχαριστίες

Θα ήθελα σε αυτό το σημείο να εκφράσω τις θερμότατες ευχαριστίες μου στον επιβλέποντα καθηγητή της διπλωματικής εργασίας κύριο Ιωάννη Κατάκη, για την ευκαιρία που μου έδωσε να ασχοληθώ με ένα τόσο ενδιαφέρον και σύγχρονο θέμα, καθώς και για την καθοδήγηση, την υποστήριξη και τη βοήθεια που μου παρείχε καθ' όλη τη διάρκεια εκπόνησης της μεταπτυχιακής διατριβής.

Τέλος, θα ήθελα να ευχαριστήσω την οικογένειά μου για το ενδιαφέρον και τη στήριξη που μου έδειξε κατά τη διάρκεια των μεταπτυχιακών μου σπουδών.

Περιεχόμενα

Περιεχόμενα Εικόνων

Περιεχόμενα Πινάκων

Περιεχόμενα Γραφημάτων

Κεφάλαιο 1

Εισαγωγή

Τα τελευταία χρόνια η αλματώδης ανάπτυξη της πληροφορικής έχει διευρύνει σε σημαντικό βαθμό τον όγκο και τη διακίνηση της πληροφορίας, η οποία καθίσταται προσβάσιμη σε ολοένα και μεγαλύτερο αριθμό χρηστών. Μέσα σε αυτόν τον κυκεώνα διακινούμενων δεδομένων σε ηλεκτρονική μορφή, ο χρήστης συχνά δυσκολεύεται να εντοπίσει και να αντλήσει την πληροφορία που τον ενδιαφέρει.

Με περισσότερα από δύο δισεκατομμύρια σελίδες που δημιουργούνται από τα εκατομμύρια των δημιουργών ιστοσελίδας, εταιριών και οργανισμών, το World Wide Web είναι μια εξαιρετικά πλούσια βάση γνώσεων. Συνεπώς, παρατηρούμε να αυξάνεται με αλματώδη τρόπο, το σύνολο των κειμένων στο διαδίκτυο, τα οποία τις περισσότερες φορές δεν είναι δομημένα, και μπορεί να είναι γραμμένα σε διάφορους τύπους κειμένων (άρθρα, e-mail, δημοσιεύσεις, HTML κείμενα ιστοσελίδων, δεδομένα ηλεκτρονικού εμπορίου, κα) αλλά και σε διαφορετικές γλώσσες.

Ο κυριότερος τρόπος πρόσβασης στα έγγραφα που υπάρχουν στον Παγκόσμιο Ιστό είναι μέσω της διαδικασίας εξόρυξης κειμένου, δηλαδή της ανάκτησης της πληροφορίας υψηλής ποιότητας

από αυτά τα έγγραφα με διάφορες τεχνικές όπως στατιστικές, μηχανικής μάθησης, βάσεων δεδομένων κ.α.. Η εξόρυξη κειμένου είναι ένας νέος τομέας της επιστήμης των υπολογιστών που ευνοεί ισχυρές συνδέσεις με την επεξεργασία φυσικής γλώσσας, εξόρυξη δεδομένων, μηχανική μάθηση, ενημέρωση ανάκτησης και διαχείρισης της γνώσης. Ο τομέας της εξόρυξης κειμένου επιδιώκει να εξάγει χρήσιμες πληροφορίες από αδόμητα δεδομένα κειμένου, μέσα από τον προσδιορισμό και την εξερεύνηση χρήσιμων προτύπων.

1.1 Η Αναγκαιότητα της Εξόρυξης Γνώσης

Το μέγεθος του Παγκόσμιου Ιστού και το αδόμητο και δυναμικό περιεχόμενο του, καθώς και ο πολυγλωσσικός χαρακτήρας του, κάνει την εξαγωγή χρήσιμων γνώσεων ένα προκλητικό και απαιτητικό πρόβλημα. Το διαδίκτυο συγκεντρώνει μεγάλες ποσότητες πληροφοριών, οι οποίες λόγω του όγκου τους πολλές φορές μένουν αναξιοποίητες. Λαμβάνοντας υπόψη ότι περίπου 90% των παγκόσμιων δεδομένων διατηρείται σε αδόμητους τύπους γίνεται αντιληπτό πως η σημασία της εξόρυξης κειμένου είναι μεγάλη, καθότι ο συνήθης βασισμένος στη λογική προγραμματισμός, αντιμετωπίζει μεγάλες στη σύλληψη των ασαφών και πολλές φορές αμφίσημων σχέσεων που περιέχονται σε έγγραφα κειμένου [10].

Η συνεχώς αυξανόμενη ποσότητα των κειμενικών εγγράφων που διατίθενται στο διαδίκτυο, π.χ. οι ειδήσεις, τα φόρουμ, τα chat lines , τα intranets των επιχειρήσεων, οι προσωπικοί υπολογιστές, τα e-mails κ.α., είναι συντριπτική. Ενώ είναι κοινή άποψη ότι η υπεραφθονία των πληροφοριών έχει τεράστια οφέλη, ενώ γίνεται παράλληλα σαφές ότι δημιουργεί νέες προκλήσεις. Η γνώση που κρύβεται σε τέτοιο τεράστιο όγκο δεδομένων, έχει σοβαρή επιρροή στην κοινωνική συμπεριφορά, στις πολιτικές αποφάσεις, στην ιατρική έρευνα και στην υγειονομική περίθαλψη, στα επιχειρηματικά μοντέλα και στις εταιρικές στρατηγικές καθώς και στις οικονομικές επενδυτικές ευκαιρίες.

Ο συντριπτικός αριθμός των διαθέσιμων μη-δομημένων δεδομένων, έχουν μεταμορφώσει τις πληροφορίες από χρήσιμες σε ενοχλητικές. Οι μηχανές αναζήτησης επιδεινώνουν το πρόβλημα και συνήθως παρέχεται στους χρήστες ένας τεράστιος όγκος των μη δομημένων αποτελεσμάτων, τα οποία πρέπει να ελαχιστοποιηθούν και να οργανωθούν ώστε η πληροφορία να γίνει χρήσιμη και πολύτιμη.

Παράλληλα, η υπερφόρτωση της πληροφορίας κάνει όλο και πιο εμφανές το πρόβλημα της κατανόησης και του χειρισμού της (π.χ. αναζήτηση κάποιου στοιχείου, ανάλυση κειμένου, ερώτηση, κ.α.) από τους ανθρώπους ενώ τα περισσότερα κείμενα δεν μπορούν να υποβληθούν σε αυτόματη επεξεργασία με κάποιο τυποποιημένο τρόπο, εξαιτίας της μη σύνδεσής τους με μεταδεδομένα (metadata: δεδομένα τα οποία χρησιμοποιούνται για την περιγραφή και αναφορά σε άλλα δεδομένα).

Έτσι λοιπόν, οι τεχνικές εξόρυξης γνώσης μπορούν να αυτοματοποιήσουν τη διαδικασία της ανάκτησης χρήσιμων πληροφοριών από τον ιστό και να συνδυαστούν με συστήματα που μπορούν να αξιοποιήσουν τη γνώση αυτή. Αναδεικνύεται επομένως το αίτημα σύγκρισης των τεχνικών-εργαλείων που θα διευκολύνουν την προσπέλαση και τη διαχείριση της διαθέσιμης πληροφορίας ανάλογα με τις ανάγκες των χρηστών.

1.2 Συνεισφορά της μεταπτυχιακής διατριβής

Η εξόρυξη κειμένου είναι ένας ευρύς ορισμός ενός τεράστιου συνόλου μοντέλων, μεθόδων και αλγορίθμων, που έχουν στόχο την εξόρυξη πληροφοριών από τα στοιχεία του κειμένου για την ανακάλυψη πολύτιμης γνώσης η οποία παραμένει ανεκμετάλλευτη.

Η παρούσα μεταπτυχιακή διατριβή θα δώσει μία εποπτική εικόνα των επιλεγμένων λογισμικών εξόρυξης κειμένου βάσει δημοτικότητας R, RapidMiner, Weka, Python, ενώ ταυτόχρονα θα παρουσιάσει μία σύγκριση των χαρακτηριστικών τους σε βάθος. Έτσι λοιπόν, θα πραγματοποιηθεί μια κριτική μελέτη των πλεονεκτημάτων και των μειονεκτημάτων των πιο διαδεδομένων εργαλείων ανοιχτού κώδικα, της ερευνητικής περιοχής της εξόρυξης κειμένου. Με τη συγκριτική μελέτη που θα εκπονηθεί ο εκάστοτε χρήστης δεδομένου ενός προβλήματος εφαρμογής, θα είναι σε θέση να επιλέξει την βέλτιστη τεχνική (εργαλείο) βάσει των προτεραιοτήτων του.

Είναι κοινή διαπίστωση ότι οι εταιρείες δαπανούν ένα μεγάλο μέρος των προσπαθειών τους στον τομέα της διαχείρισης και οργάνωσης εγγράφων με ελαφρώς ικανοποιητικά αποτελέσματα. Στην παρούσα μεταπτυχιακή διατριβή, ενδιαφερόμαστε για νέα, αποτελεσματικά και υγιή εργαλεία που χρησιμοποιούνται για τη διαχείριση των αποθετηρίων εγγράφων, προσφέροντας στους χρήστες ψήγματα πληροφοριών που είναι απαραίτητα για να αποκτήσουν ανταγωνιστικό πλεονέκτημα για την ορθολογική λήψη αποφάσεων .

Έτσι λοιπόν, η συνεισφορά της παρούσας διατριβής έγκειται σε ένα πρωτόκολλο για την αξιολόγηση των επιδόσεων των εργαλείων εξόρυξης κειμένου, και ένα συνδυασμένο μοντέλο που εκμεταλλεύεται σημασιολογικές πληροφορίες, μαζί με γραπτές πηγές για να προσφέρουν στο χρήστη αποτελεσματικότητα στην ανάκτηση πληροφοριών .

Συμπερασματικά, αντικείμενο της παρούσας μεταπτυχιακής διατριβής είναι η μελέτη και η εφαρμογή τεχνικών εξόρυξης κειμένου μέσω της πειραματικής διερεύνησης των εργαλείων εξόρυξης γνώσης ανοιχτού κώδικα που υπάρχουν στο διαδίκτυο, με σκοπό να βελτιωθεί η διαδικασία της εξόρυξης κειμένων. Τα αποτελέσματα θα είναι χρήσιμα για νέους αλλά και έμπειρους ερευνητές στην περιοχή της εξόρυξης κειμένων, σε εταιρίες κ.λ.π..

Κεφάλαιο 2

Εξόρυξη Κειμένου

Η γλώσσα αποτελεί το πιο κοινό μέσο για την επίσημη ανταλλαγή πληροφοριών. Η εξόρυξη κειμένου είναι ένα νεοσύστατο πεδίο που προσπαθεί να εξάγει χρήσιμες πληροφορίες από το φυσικό γλωσσικό κείμενο. Γενικότερα μπορεί να θεωρηθεί ως διαδικασία ανάλυσης κειμένου με σκοπό την εξαγωγή πληροφοριών που είναι χρήσιμες για συγκεκριμένους σκοπούς. Παρακάτω θα αναφερθούμε στη διαδικασία, στα μοντέλα και τις κατηγορίες αλγορίθμων εξόρυξης δεδομένων από κείμενο ώστε να εμβαθύνουμε στο πεδίο της εξόρυξης κειμένου.

2.1 Εξόρυξη Γνώσης από κείμενο

Η εξόρυξη γνώσης από κείμενο " Knowledge Discovery in Text – KDT " καθώς και η εξόρυξη κειμένου " Text Mining- TM " περιλαμβάνουν αυτοματοποιημένες τεχνικές για την ανάλυση μεγάλων συλλογών από δεδομένα αλλά και την εξαγωγή χρήσιμων πληροφοριών από αυτά, οι οποίες βρίσκονται σήμερα στο επίκεντρο του ενδιαφέροντος τόσο από εμπορική όσο και από επιστημονική πλευρά. Χρησιμοποιώντας τεχνικές από την εξόρυξη δεδομένων " Data Mining ", τη μηχανική

μάθηση " Machine Learning ", τη στατιστική " Statistics " την επεξεργασία φυσικής γλώσσας " Natural Language Processing", την ανάκτηση πληροφορίας " Information Retrieval ", την εξαγωγή πληροφορίας " Information Extraction " και τη διαχείριση πληροφορίας " Knowledge Management", οι τεχνικές αυτές προσπαθούν να επιλύσουν το πρόβλημα της μετατροπής των τεραστίων ποσοτήτων από δεδομένα, σε χρήσιμη γνώση.

Αν επεκτείνουμε τον ορισμό για την εξόρυξη πληροφορίας από δεδομένα των Fayyad, Piatetsky – Shapiro και Smyth [06] μπορούμε να δώσουμε τον εξής απλό ορισμό:

«Η εξόρυξη γνώσης από Κείμενο (KDT) είναι μια μη τετριμμένη διαδικασία ανακάλυψης έγκυρων, καινούργιων, δυνητικά χρήσιμων και τελικά κατανοητών προτύπων σε δεδομένα κειμένου. »

Ας δούμε από λίγο πιο κοντά τις βασικές έννοιες που εμφανίζονται στον παραπάνω ορισμό: Μη δομημένα δεδομένα κειμένου "Unstructured Textual Data" είναι μια συλλογή κειμένων. Χρησιμοποιούμε τον όρο έγγραφο "Document" για να αναφερθούμε σε μια λογική μονάδα κειμένου. Αυτό θα μπορούσε να είναι μια σελίδα στο παγκόσμιο ιστό, ένα τιμολόγιο, ένα ηλεκτρονικό ταχυδρομείο κ.α.. Μια μονάδα κείμενου θα μπορούσε να είναι μακριά και πολύπλοκη [01] ακόμα και να περιέχει κάτι παραπάνω από απλώς κείμενο, όπως γραφικά. Σε αυτή τη μεταπτυχιακή διατριβή, θα ασχοληθούμε με έγγραφα που περιέχουν μόνο κείμενο.

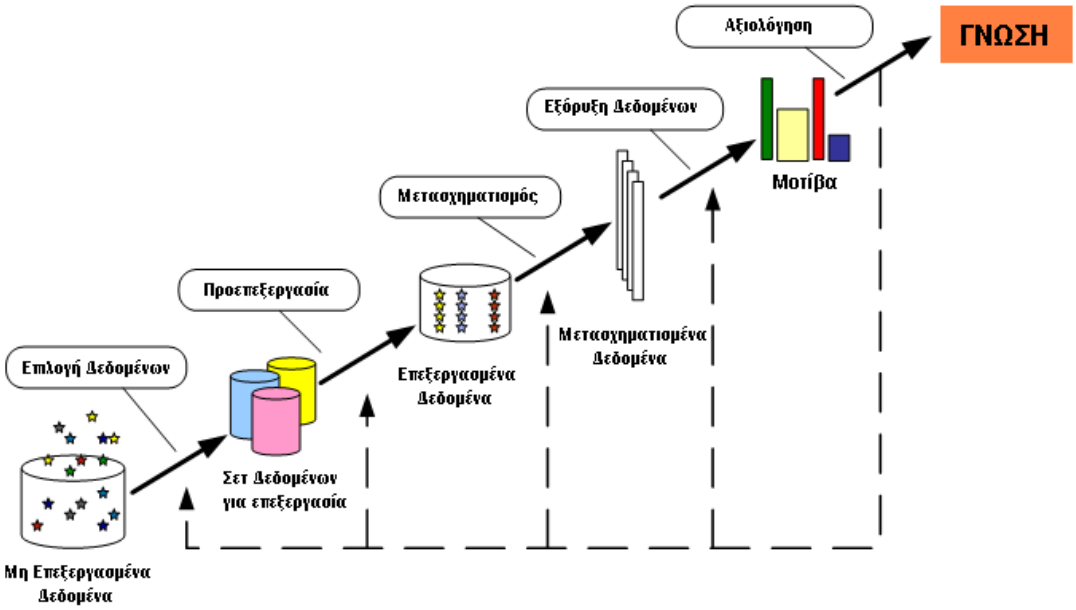
Εάν θεωρήσουμε τα δεδομένα μας ως ένα σύνολο γεγονότων (π.χ. περιπτώσεις σε μια βάση δεδομένων), ένα πρότυπο "pattern" είναι ένας κανόνας, ο οποίος περιγράφει τα γεγονότα σε ένα υποσύνολο του συνόλου των γεγονότων. Γενικότερα θα μπορούσαμε να πούμε ότι υπάρχουν δύο τύποι προτύπων [06, 19]:

1. Τα πρότυπα πρόβλεψης "predictive pattern" χρησιμοποιούνται για να προβλέψουν ένα ή περισσότερα γνωρίσματα "attributes" από αυτά που υπάρχουν στη βάση. Αυτό το είδος των προτύπων εικάζουν την τιμή ενός άγνωστου γνωρίσματος, δεδομένου των τιμών των γνωρισμάτων των άλλων δεδομένων.

2. Τα πληροφοριακά πρότυπα “informative pattern” δεν επιλύουν ένα συγκεκριμένο πρόβλημα αλλά παρουσιάζουν στον χρήστη πρότυπα που θα έπρεπε να γνωρίζει.

Η εξόρυξη γνώσης από κείμενο είναι μια διαδικασία πολλών βημάτων, που περιλαμβάνει όλες τις διαδικασίες από την συλλογή των εγγράφων μέχρι την οπτικοποίηση της γνώσης που έχει προκύψει. Η διαδικασία θα πρέπει να είναι μη τετριμμένη, δηλαδή το αποτέλεσμα θα πρέπει μπορεί να αξιολογηθεί ως ανακάλυψη. Τα πρότυπα που ανακαλύφθηκαν θα πρέπει να είναι έγκυρα σε καινούργια δεδομένα με κάποιον βαθμό βεβαιότητας. Επίσης, θα πρέπει να είναι καινούργια, τουλάχιστον για το σύστημα, και θα πρέπει να οδηγούν σε χρήσιμες δράσεις, όπως προσδιορίζονται από μια συνάρτηση χρησιμότητας. Ο κύριος σκοπός της εξόρυξης γνώσης από κείμενο, είναι να κάνει τα πρότυπα κατανοητά στους ανθρώπους για τη διευκόλυνση της κατανόησης των παρόντων δεδομένων [06].

Η διεργασία της εξόρυξης της γνώσης από κείμενο περιλαμβάνει τέσσερα βασικά στάδια [01, 06, 13, 14, 17, 32], όπως φαίνονται στο Γράφημα 1. Διαχωρίζοντας τον όρο εξόρυξη γνώσης από κείμενο από τον όρο εξόρυξη κειμένου, μπορούμε να πούμε ότι η εξόρυξη κειμένου αποτελεί ένα στάδιο της εξόρυξης γνώσης από κείμενο, η οποία είναι μια διαδικασία που περιλαμβάνει πολλά βήματα για την ανεύρεση χρήσιμης πληροφορίας από κείμενα, από την συλλογή των εγγράφων, την προ-επεξεργασία τους (ώστε να μετατραπούν σε κάποια επιθυμητή αναπαράσταση όπως XML, SGML κλπ), την εξαγωγή λεκτικών πληροφοριών σχετικών με το περιεχόμενο κάθε εγγράφου, την εξόρυξη κειμένου μέσω της δημιουργίας μεταδεδομένων και της αναγνώρισης προτύπων και συσχετίσεων μεταξύ των δεδομένων, μέχρι και την απεικόνιση της γνώσης που προκύπτει.



Γράφημα 1. Διαδικασία Εξόρυξης Κειμένου.

Πιο αναλυτικά, η διαδικασία της εξόρυξης κειμένου περιλαμβάνει τα παρακάτω βασικά στάδια :

1. Συλλογή των σχετικών εγγράφων: Στο πρώτο στάδιο προσπαθούμε να αναγνωρίσουμε ποια κείμενα θα χρησιμοποιήσουμε. Αφού πρώτα βρούμε την πηγή από όπου θα ανακτήσουμε τα κείμενά μας (διαδίκτυο ή άλλες συμβατικές πηγές), θα πρέπει στη συνέχεια να τα συλλέξουμε. Σε αυτό το στάδιο, μπορεί να χρησιμεύσει η τεχνική που βασίζεται στη συσταδοποίηση με βάση τα χαρακτηριστικά του κειμένου.

2. Προ – επεξεργασία των εγγράφων “pre-processing”: Αυτό το στάδιο περιλαμβάνει οποιοδήποτε είδος διαδικασιών μετασχηματισμού των αρχικών εγγράφων που ανακτώνται. Στο σύνολο των εγγράφων που ανακτήθηκαν, μπορεί να εμπεριέχονται έγγραφα με διαφορετικούς τύπους (π.χ. PDF, DOC, RTF, HTML), έτσι λοιπόν η πρώτη ενέργεια που πρέπει να γίνει είναι η τυποποίηση. Το στάδιο της τυποποίησης συνίσταται στη μετατροπή των δεδομένων σε μια κοινή και αναγνώσιμη μορφή. Οι κοινές μορφές εγγράφων ταυτίζονται με Extensible Markup Language (XML), όπου επιτρέπεται μια δομημένη απεικόνιση των εγγράφων, δηλαδή είναι δυνατόν να προσδιοριστούν διάφορα τμήματα σε ένα έγγραφο, όπως τίτλος, περίληψη, κεφάλαια κ.λπ., ή TXT (ASCII ή Unicode) που προτιμάται κάθε φορά που το

σύστημα χρειάζεται ένα απλό και ευθύ format. Ο σκοπός είναι να δημιουργηθεί ένα σύνολο TXT ή XML εγγράφων.

Μόλις το σύνολο των εγγράφων παραχθεί, το επόμενο βήμα είναι η προ-επεξεργασία του συνόλου, στην οποία εφαρμόζονται διαφορετικά φίλτρα για την απομάκρυνση όλων των δεδομένων που στη συγκεκριμένη εφαρμογή θεωρούνται μη κατατοπιστικά. Τα έγγραφα προς επεξεργασία λοιπόν, μπορεί να περιέχουν ένα πολύ μεγάλο αριθμό λέξεων. Χρησιμοποιώντας μεθόδους προ-επεξεργασίας κειμένων, οι οποίοι αποσκοπούν στη μείωση του όγκου των λέξεων, οι οποίες θα αποτελέσουν την είσοδο του αλγορίθμου εξόρυξης κειμένου που πρόκειται να εφαρμοστεί. Επίσης, η μέθοδος προ-επεξεργασίας θα πρέπει να μπορεί να επεξεργαστεί κείμενα με "θόρυβο", δηλαδή κείμενα με τυχόν τυπογραφικά και ορθογραφικά λάθη. Από τις κυριότερες μεθόδους προ-επεξεργασίας είναι [09, 17]:

- Ετικετοποίηση των μερών του λόγου (tokenization), είναι η μέθοδος που καθορίζει τα μέρη του λόγου στο κείμενο π.χ. ρήμα, ουσιαστικό, επίθετο, μετοχή κλπ, ώστε να χρησιμοποιηθούν αργότερα από τη λημματοποίηση. Η μέθοδος αυτή ενσωματώνει γλωσσολογική γνώση, και εξαρτάται από τη γλώσσα του κειμένου.

- Λημματοποίηση (lemmatization), είναι η μέθοδος που προσπαθεί να χαρτογραφήσει ρηματικούς τύπους στην ενεργητική φωνή και ουσιαστικά στον ενικό αριθμό. Ωστόσο, προκειμένου να επιτευχθεί αυτό, η μορφή των λέξεων πρέπει να είναι γνωστή, δηλαδή το μέρος του λόγου της κάθε λέξης στο έγγραφο του κειμένου πρέπει να έχει εκχωρηθεί. Δεδομένου ότι αυτή η διαδικασία εκχώρησης των μέρων του λόγου είναι συνήθως χρονοβόρα και ακόμα επιρρεπής σε λάθη, στην πράξη συχνά εφαρμόζεται η μέθοδος της εύρεσης της ρίζας των λέξεων "stemming". Η μέθοδος αυτή ενσωματώνει γλωσσολογική γνώση, και εξαρτάται από τη γλώσσα του κειμένου.

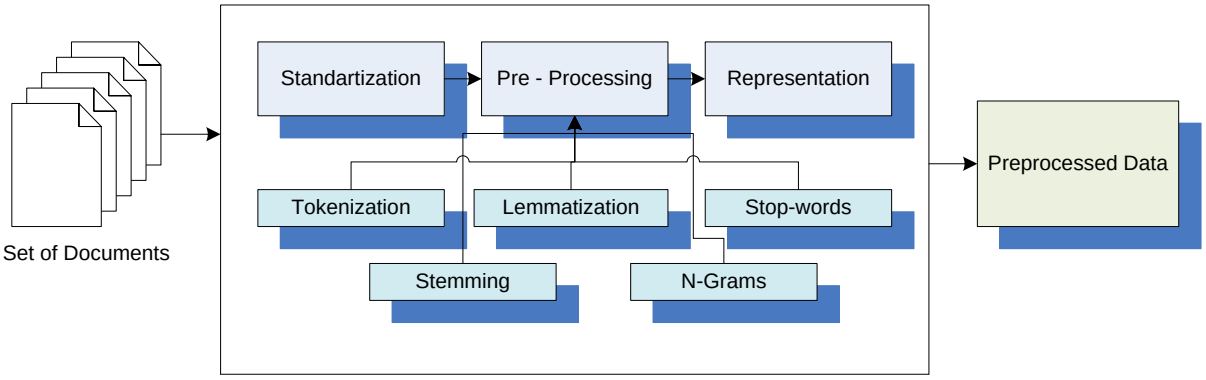
- Εύρεση της ρίζας των λέξεων (stemming), είναι η μέθοδος που μετατρέπει κάθε λέξη που πρόκειται να επεξεργαστεί, στην αντίστοιχη ρίζα της. Επιπλέον, καθορίζει τις παραγωγικές καταλήξεις και/ή τις

ρηματικές πληθυντικές κλίσεις της κάθε λέξης. Η μέθοδος αυτή ενσωματώνει γλωσσολογική γνώση, και εξαρτάται από τη γλώσσα του κειμένου.

- Αφαίρεση των κοινών – μη σημαντικών λέξεων, είναι η μέθοδος που αναγνωρίζει και απομακρύνει τις λέξεις του κειμένου που είναι κοινές – μη σημαντικές, εφόσον αποτελούν μη χρήσιμα στοιχεία και δεν πρέπει να χρησιμοποιηθούν κατά την επεξεργασία του κειμένου. Οι κοινές – μη σημαντικές λέξεις, μπορούν να αναγνωριστούν με τη χρήση πληροφοριών που έχουν προκύψει από γλωσσολογικές μελέτες.

- Η αναπαράσταση Ν χαρακτήρων (N-Grams), είναι μια εναλλακτική μέθοδος προ-επεξεργασίας του κειμένου έναντι του "stemming" και της απομάκρυνσης των κοινών μη σημαντικών λέξεων. Η αναπαράσταση αυτή δημιουργεί συμβολοσειρές Ν χαρακτήρων με βάση τη μελέτη κάθε λέξης του κειμένου. Για παράδειγμα, η λέξη ΛΕΞΗ, μπορεί να αναπαρασταθεί με συμβολοσειρές 3 χαρακτήρων, δηλαδή _ΛΕ, ΛΕΞ, ΕΞΗ, ΞΗ_. Είναι πιο εύρωστη μέθοδος από το "stemming" και την απομάκρυνση κοινών μη σημαντικών λέξεων, διότι είναι λιγότερο ευαίσθητη σε τυπογραφικά και ορθογραφικά λάθη, και επιπλέον δεν απαιτεί κάποια γλωσσολογική επεξεργασία του κειμένου που πρόκειται να προ-επεξεργαστεί. Έτσι λοιπόν είναι ανεξάρτητη της γλώσσας του κειμένου. Παρόλα αυτά δεν είναι αποτελεσματική μέθοδος για τη διαδικασία της μείωσης των λέξεων, ενώ η μείωση του πλήθους των λέξεων επιτυγχάνεται καλύτερα με τις μεθόδους απομάκρυνσης κοινών μη σημαντικών λέξεων και του "stemming".

- Το τελευταίο στοιχείο της προ-επεξεργασίας του κειμένου είναι η αναπαράσταση του κειμένου, όπου αναλύεται στο παρακάτω υποκεφάλαιο. Μόλις τα έγγραφα μεταμορφωθούν και όλα τα άχρηστα στοιχεία φιλτραριστούν, το σύστημα θα μετατρέψει το αδόμητο κείμενο σε μια δομημένη μορφή.



Γράφημα 2. Διαδικασία προ-επεξεργασίας εγγράφων.

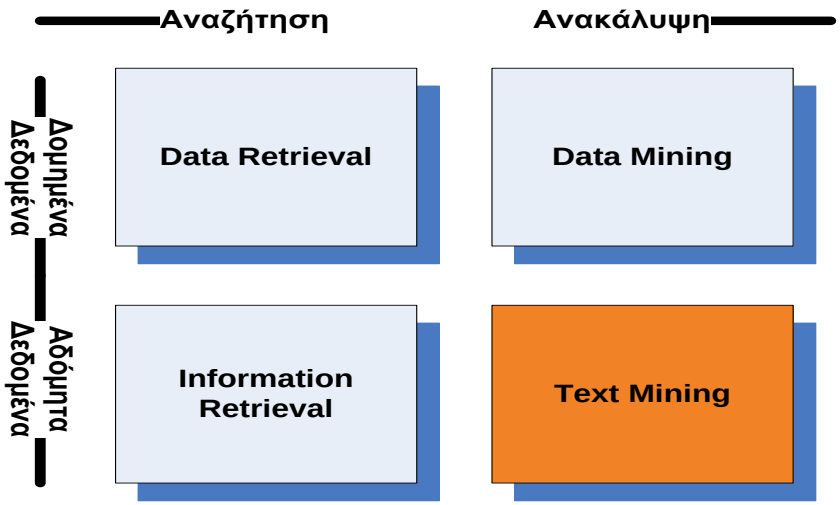
3. Διαδικασίες εξόρυξης κειμένου: Το αρχείο που παράγεται κατά το στάδιο της προ-επεξεργασίας δεδομένων, διοχετεύεται σαν είσοδος σε μια ποικιλία αλγορίθμων και τεχνικών ανάκτησης δεδομένων, που προέρχονται είτε από το χώρο της στατιστικής, είτε από το χώρο της μηχανικής μάθησης, είτε της εξόρυξης δεδομένων. Σε αυτό το στάδιο οι υψηλού επιπέδου πληροφορίες εξάγονται, (δημιουργία μετα-δεδομένων "metadata creation"), αναγνωρίζονται τα πρότυπα και οι συσχετίσεις των δεδομένων. Σε αυτό το στάδιο χρησιμοποιούνται τεχνικές εξόρυξης κειμένου, κάποιες από τις κυριότερες θα αναλυθούν παρακάτω.

4. Αξιολόγηση. Στο τελικό στάδιο παρέχεται η αξιολόγηση και η ερμηνεία των αποτελεσμάτων. Αναφέρουμε ότι δεν είναι όλοι οι συσχετισμοί που βρέθηκαν από τους αλγορίθμους εξόρυξης δεδομένων κειμένου απαραίτητα έγκυροι. Είναι κοινό για τους αλγόριθμους εξόρυξης δεδομένων κειμένου να βρουν συσχετισμούς στο σύνολο δεδομένων εκμάθησης, που δεν είναι παρόντα στο γενικό σύνολο δεδομένων. Για να ξεπεραστεί αυτό, η αξιολόγηση των αλγορίθμων χρησιμοποιεί ένα σύνολο δεδομένων για δοκιμή κατά την οποία ο αλγόριθμος εξόρυξης δεδομένων δεν είναι εκπαιδευμένος. Οι συσχετισμοί που έχουν βρεθεί εφαρμόζονται σε αυτό το σύνολο δεδομένων και τα αποτελέσματα συγκρίνονται με το επιθυμητό αποτέλεσμα. Εάν οι συσχετίσεις που θα βρεθούν δεν πληρούν τις επιθυμητές προδιαγραφές, τότε θα πρέπει να αξιολογηθεί εκ νέου το στάδιο προ-επεξεργασίας και εξόρυξης κειμένου. Στην άλλη περίπτωση οι συσχετίσεις μετατρέπονται σε γνώση και παρουσιάζονται σε μια χρήσιμη μορφή, όπως ένα γράφημα, έναν πίνακα ή μια αναφορά.

2.2 Ορισμός της Εξόρυξης Κειμένου

Η εξόρυξη κειμένου είναι ένα βήμα στην διαδικασία εξόρυξης γνώσης από κείμενο, που αποτελείται από συγκεκριμένους αλγορίθμους επεξεργασίας φυσικής γλώσσας της εξόρυξης δεδομένων, οι οποίοι κάτω από υπολογιστικά παραδεκτούς περιορισμούς παράγουν μια συγκεκριμένη ποσότητα προτύπων από ένα σύνολο μη δομημένων δεδομένων κειμένου [10].

Έτσι λοιπόν, η εξόρυξη κειμένου χρησιμοποιεί μη δομημένα κείμενα που τα εξετάζει με σκοπό να ανακαλύψει δομή και αυτονόητα «νοήματα» που βρίσκονται κρυμμένα μες στο κείμενο [12]. Ο κύριος στόχος της εξόρυξης κειμένου είναι η υποστήριξη της εξόρυξης πληροφορίας σε τεράστιες συλλογές κειμένων, που αποθηκεύονται είτε στο διαδίκτυο είτε συμβατικά. Η εξόρυξη κειμένου χρησιμοποιεί ειδικές τεχνικές επεξεργασίας φυσικής γλώσσας και εξόρυξης δεδομένων που εφαρμόζονται σε δεδομένα κειμένου με σκοπό την εξαγωγή χρήσιμων πληροφοριών. Οι εφαρμογές εξόρυξης κειμένου επιβάλουν αυστηρούς περιορισμούς στις συνήθεις τεχνικές επεξεργασίας φυσικής γλώσσας [10]. Είναι αυτή η κοντινή συγγένεια που κάνει την εξόρυξη κειμένου έναν καινούργιο τομέα της έρευνας που προέρχεται από την εξόρυξη δεδομένων και από την επεξεργασία φυσικής γλώσσας. Η εξόρυξη κειμένου διαχωρίζεται από την εξόρυξη δεδομένων, όπως φαίνεται στο παρακάτω γράφημα.



Γράφημα 3. Κατάταξη της εξόρυξης κειμένου.

Για αυτό το λόγο υπάρχουν διάφορες προσεγγίσεις οι οποίες προσπαθούν να ορίσουν την εξόρυξη κειμένου. Οι Καρανίκας και Θεοδουλίδης [11] ορίζουν την εξόρυξη κειμένου ως:

« Η διαδικασία της εξόρυξης γνώσης από κείμενο (K.D.T.) αποτελείται από αλγορίθμους της εξόρυξης δεδομένων και της επεξεργασίας της Φυσικής γλώσσας, που κάτω από μερικούς αποδεκτούς υπολογιστικούς περιορισμούς αποδοτικότητας, παράγουν έναν ιδιαίτερο αριθμό από υποδείγματα μέσα από ένα σύνολο μη δομημένων κειμενικών δεδομένων».

Ενώ, οι Nahm και Mooney [15] περιγράφουν την εξόρυξη κειμένου ως «την αναζήτηση των προτύπων σε μη δομημένο κείμενο», ενώ οι Besancon και Rajman [17] το θεωρούν ως μια επέκταση της εξόρυξης δεδομένων σε μη δομημένα κείμενα όπου η οποία περιλαμβάνει διάφορους στόχους όπως εξόρυξη πληροφορίας και δημιουργία δομής βασισμένη στην ομοιότητα.

Συνοψίζοντας τα παραπάνω η εξόρυξη κειμένου μπορούμε να πούμε ότι είναι μια διαδικασία εξαγωγής νέας, έγκυρης και αγώγιμης γνώσης από διαφορετικούς γραπτούς πόρους, καθώς επίσης και όσο το δυνατόν καλύτερης οργάνωσης αυτής της νέας γνώσης της πληροφορίας για μελλοντική αναφορά.

2.3 Εφαρμογές Εξόρυξης Κειμένου

Εφαρμόζοντας τις τεχνικές της εξόρυξης κειμένου συνήθως θέλουμε να επιτύχουμε:

- Κατηγοριοποίηση κειμένων: Την Κατηγοριοποίηση ενός κειμένου σε μια κατηγορία (τάξη). Γνωστές εφαρμογές αποτελούν η διήθηση ανεπιθύμητης αλληλογραφίας και η Κατηγοριοποίηση ιστοσελίδων.

- Συσταδοποίηση κειμένων: Αυτόματη οργάνωση κειμένων σε ομάδες που θα έχουν κάποια κοινά χαρακτηριστικά. Γνωστές εφαρμογές αποτελούν η ιεραρχική οργάνωση ιστοσελίδων και η συσταδοποίηση ειδησεογραφικών άρθρων.

- Εξαγωγή Πληροφορίας: Είναι η προσπάθεια εξαγωγής συγκεκριμένης πληροφορίας από μεγάλο αριθμό κειμένων. Εφαρμογή του αποτελεί η αυτόματη εξαγωγή πληροφοριών όπως η διαθεσιμότητα σε αεροπορικές πτήσεις.

- Περίληψη κειμένου: Πρόκειται για την δημιουργία ενός κειμένου, που να δίνει επιγραμματικά το περιεχόμενο του αρχικού και να είναι μικρότερο σε μέγεθος.

- Ανάκτηση κειμένου: Πρόκειται για την εύρεση νέων κειμένων με βάση κάποια λέξη κλειδί ή κάποιο σύντομο κείμενο.

Τα αδόμητα κείμενα είναι πολύ κοινά στον Παγκόσμιο Ιστό, και στην πραγματικότητα μπορεί να αντιπροσωπεύουν την πλειοψηφία των διαθέσιμων πληροφοριών σε μια συγκεκριμένη έρευνα ή σε μια εργασία εξόρυξης δεδομένων. Οι βασικότεροι τομείς όπου εφαρμόζεται η εξόρυξη κειμένου είναι σε επιχειρηματικές εφαρμογές, στα Μ.Μ.Ε., σε θέματα ασφαλείας, σε ακαδημαϊκές εφαρμογές κ.α., και μπορούν να συνοψιστούν στις παρακάτω [09, 37]:

- Αυτόματη επισήμανση των εγγράφων σε βιβλιοθήκες των επιχειρήσεων.

- Μέτρηση προτιμήσεων των πελατών με την ανάλυση ποιοτικών συνεντεύξεων, ή αναρτήσεων σε blogs.

- Ανάλυση προφίλ πελατών, εξορύσσοντας εισερχόμενα emails μέσω των ανατροφοδοτήσεων από τους πελάτες.

- Ανάλυση ενός συγκεκριμένου συνόλου λέξεων ή όρων που χρησιμοποιούνται συνήθως από τους ερωτηθέντες για να περιγράψουν τα υπέρ και τα κατά ενός προϊόντος ή μιας υπηρεσίας.

- Ανταγωνιστική κατασκοπεία, επιτρέποντας στις επιχειρήσεις να οργανώνουν και να τροποποιούν τις στρατηγικές τους σύμφωνα με τις σημερινές απαιτήσεις της αγοράς και τις ευκαιρίες, με βάση τις πληροφορίες που συλλέγονται από την ίδια την εταιρία, την αγορά και τους ανταγωνιστές της.

- Συγκέντρωση και σύγκριση των πληροφοριών που προέρχονται αυτόματα από έγγραφα συγκεκριμένου τύπου, όπως τα εισερχόμενα μηνύματα πελατών και λοιπά.

- CRM διαχείριση των σχέσεων με τους πελάτες, αποστέλλοντας συγκεκριμένα αιτήματα αυτόματα στην κατάλληλη υπηρεσία ή παροχή άμεσων απαντήσεων στις πιο συχνές ερωτήσεις.

- Προσωποποίηση, Κατηγοριοποίηση και συσταδοποίηση εγγράφων με βάση εξατομικευμένα ερωτήματα.

- Συσταδοποίηση ειδήσεων.

- Δημιουργία προτάσεων και συστάσεων (όπως η Amazon).

- Παρακολούθηση κοινής γνώμης (για παράδειγμα, σε blogs ή ιστοσελίδες), ώστε να αποκτήσουν γνώσεις σχετικά με τις τάσεις, τις σχέσεις μεταξύ των ανθρώπων / οργανώσεων/ μέρη.

- Ανάλυση συναισθήματος, η οποία αναφέρεται στη χρήση της επεξεργασίας της φυσικής γλώσσας (NLP), με την ανάλυση κειμένου και την υπολογιστική γλωσσολογία για να εντοπίσει και να εξαγάγει υποκειμενικές πληροφορίες. Γενικά μιλώντας, η ανάλυση συναισθήματος έχει ως στόχο να καθορίσει τη

στάση του ομιλητή ή ενός συγγραφέα σε κάποιο θέμα ή τη συνολική πολικότητα ενός εγγράφου.

- Καταπολέμηση του φαινομένου της ηλεκτρονικής παρενόχλησης ή του ηλεκτρονικού εγκλήματος στον κυβερνοχώρο IM και IRC Chat.

- Συμβάλει στη μελέτη της κρυπτογράφησης κειμένου, αναλύοντας πηγές απλού κειμένου όπως οι ειδήσεις στο διαδίκτυο.

- Κατηγοριοποίηση κειμένου, με το φιλτράρισμα των spam emails με βάση ορισμένες λέξεις ή όρους που δεν είναι πιθανό να εμφανίζονται στα νόμιμα μηνύματα.

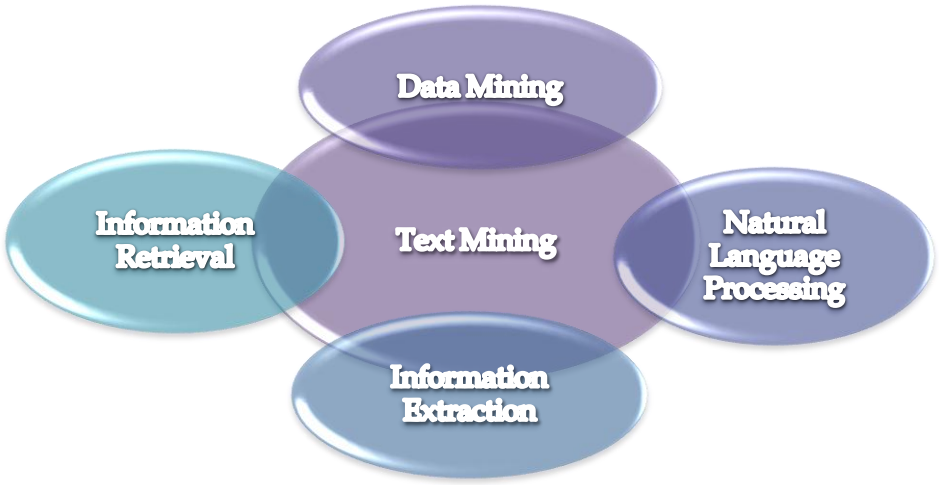
- Εντοπισμός και ανάλυση των ιστοσελίδων που δημοσιεύθηκαν σε διάφορες γλώσσες, με πολύγλωσσες εφαρμογές της επεξεργασίας της φυσικής γλώσσας NLP.

- Δημιουργία μιας συμπυκνωμένης έκδοσης ενός εγγράφου ή μια συλλογή εγγράφων που θα πρέπει να περιέχει τα σημαντικότερα θέματα και έννοιες.

- Εντοπισμός και Κατηγοριοποίηση των τεχνικών όρων στον τομέα της μοριακής βιολογίας που αντιστοιχεί σε περιπτώσεις όπως τα ονόματα των πρωτεϊνών, τα γονίδια και η θέση δραστηριότητας τους και λοιπά.

2.4 Ερευνητικές περιοχές που σχετίζονται με την Εξόρυξη Κειμένου

Η εξόρυξη κειμένου περιλαμβάνει την εφαρμογή των τεχνικών που προέρχονται από περιοχές όπως η ανάκτηση πληροφοριών, η επεξεργασία φυσικής γλώσσας, η άντληση πληροφοριών και η εξόρυξη δεδομένων, όπως φαίνεται στο παρακάτω σχήμα. Αυτά τα διάφορα στάδια της διαδικασίας εξόρυξης κειμένου μπορούν να συνδυαστούν σε μια ενιαία ροή εργασίας. Θα εξετάσουμε τώρα πιο αναλυτικά κάθε μία από αυτές τις περιοχές και πώς, μαζί, αποτελούν έναν αγωγό εξόρυξης κειμένου [09].



Γράφημα 4. Περιοχές που αποτελούν αγωγό της εξόρυξης κειμένου.

2.4.1 Εξόρυξη Δεδομένων "Data Mining"

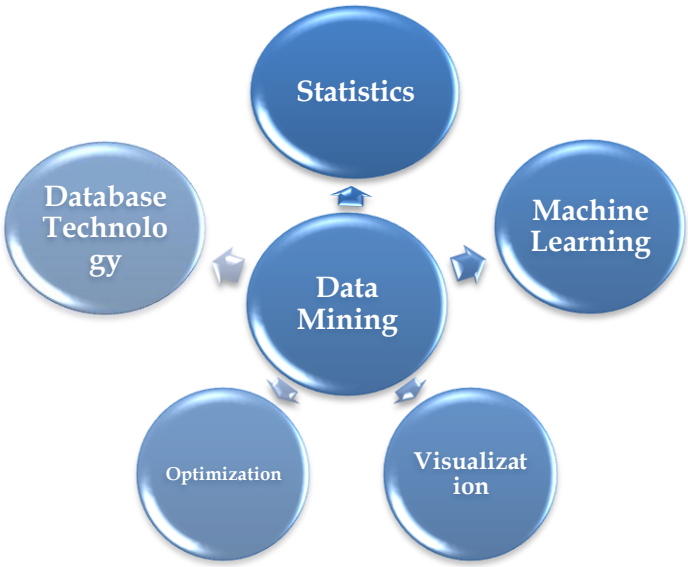
Η εξόρυξη δεδομένων "Data Mining" είναι η διαδικασία της αναγνώρισης προτύπων σε μεγάλα σύνολα δεδομένων. Η εξόρυξη δεδομένων είναι η διαδικασία εξόρυξης πληροφορίας από μεγάλες ποσότητες δεδομένων που είναι αποθηκευμένα σε βάσεις δεδομένων. Η εξόρυξη δεδομένων περιλαμβάνει την ενσωμάτωση των τεχνικών από πολλούς κλάδους όπως η τεχνολογία των βάσεων δεδομένων και των στατιστικών δεδομένων αποθήκευσης, της μηχανικής μάθησης, της αναγνώριση προτύπων, των νευρωνικών δικτύων, της οπτικοποίησης δεδομένων, της ανάκτησης πληροφοριών, της εικόνας και επεξεργασίας σήματος, και της χωρικής ή χρονικής ανάλυσης δεδομένων [09, 23]. Πιο συγκεκριμένα, η εξόρυξη δεδομένων είναι η εφαρμογή αλγορίθμων σε ένα σύνολο δεδομένων για να αποκαλυφθούν συνδέσεις και συσχετίσεις που δεν ήταν από πριν προφανείς.

Κατά τη φάση της εξόρυξης δεδομένων, χρησιμοποιούνται οι υπάρχουσες τεχνικές εξόρυξης δεδομένων, όπως η Κατηγοριοποίηση, η συσταδοποίηση, η συσχέτιση, τα πρότυπα ακολουθιών κ.α.., ώστε να ανιχνευθούν τα πρότυπα που θα μας βοηθήσουν στην εξόρυξη της πληροφορίας.

Η ερμηνεία της ανίχνευσης προτύπου αποκαλύπτει κατά πόσον ή όχι το πρότυπο περιέχει πληροφορία. Αυτό είναι ο λόγος που αυτό το βήμα, ονομάζεται βήμα αξιολόγησης. Το καθήκον είναι να εκπροσωπεί το αποτέλεσμα με τον κατάλληλο τρόπο, ώστε να μπορεί να εξεταστεί διεξοδικά. Εάν το πρότυπο δεν είναι ενδιαφέρον, κατά

πάσα πιθανότητα θα χρειαστεί να καταφύγουμε σε ένα προηγούμενο βήμα για να επαναλάβουμε την προσπάθεια

Η εξόρυξη δεδομένων λειτουργεί με δομημένα δεδομένα, τα οποία τις πιο πολλές φορές είναι αριθμητικά. Η εξόρυξη κειμένου είναι ανάλογη με την εξόρυξη δεδομένων στο γεγονός ότι αποκαλύπτει σχέσεις που ενυπάρχουν στην πληροφορία μας. Από την άλλη πλευρά όμως η εξόρυξη κειμένου, διαφέρει από την εξόρυξη δεδομένων, στο ότι αυτή δουλεύει με πληροφορίες που βρίσκονται αποθηκευμένες ως μια μη δομημένη συλλογή εγγράφων κειμένου.



Γράφημα 5. Τεχνικές εξόρυξης δεδομένων.

- **Μηχανική Μάθηση**

  Η εξόρυξη δεδομένων βασίζεται σε μια ποικιλία από υπολογιστικές τεχνικές, μερικές από τις οποίες εμπίπτουν στην επικεφαλίδα της μηχανικής μάθησης. Παραδείγματα είναι τα δέντρα απόφασης, νευρωνικά δίκτυα, και κανόνων συσχέτισης. Σε αυτό το πλαίσιο, η μηχανική μάθηση περιλαμβάνει «την εξαγορά των διαρθρωτικών περιγραφών από παραδείγματα που μπορούν να χρησιμοποιηθούν για την πρόβλεψη, εξήγηση και κατανόηση». Από μια ευρύτερη προοπτική της τεχνητής νοημοσύνης, η μηχανική μάθηση είναι μια από τις τέσσερις ικανότητες που απαιτούνται για ένα σύστημα τεχνητής νοημοσύνης. Από την έρευνα του Mjolsness και DeCoste [14] μηχανική μάθηση ορίζεται « η μελέτη των αλγορίθμων που είναι σε θέση να μάθουν να

βελτιώνουν τις επιδόσεις τους από μια εργασία με βάση την προηγούμενη εμπειρία τους », κυρίως μέσω της αναγνώρισης προτύπων και της στατιστικής συμπερασματολογίας. Η εξόρυξη κειμένου θα μπορούσε να βοηθήσει με τους μεγάλους όγκους δεδομένων που εμπλέκονται στην αναζήτηση βιβλιογραφίας.

- **Στατιστική ανάλυση**

Η στατιστική ανάλυση αποσκοπεί στην αναζήτηση χρήσιμων πληροφοριών και προτύπων στα δεδομένα, έτσι λοιπόν ένα πολύ μεγάλο μέρος της εξόρυξης δεδομένων βασίζεται στη στατιστική ανάλυση. Μέρος των διαδικασιών σε ένα μοντέλο εξόρυξης δεδομένων μπορεί να αποτελεί η αναζήτηση δεδομένων με την παράλληλη εξαγωγή συμπερασμάτων από τα αποτελέσματα μιας αναζήτησης. Μια τέτοια τεχνική στην εξόρυξη δεδομένων είναι η δειγματοληψία, όπου στη στατιστική λέγεται στατιστική εξαγωγή συμπεράσματος. Ένα σημαντικό τμήμα των αλγορίθμων εξόρυξης δεδομένων αποτελούν στατιστικές τεχνικές, όπως για παράδειγμα. ανάλυση παλινδρόμησης, ανάλυση συστάδων και άλλα.

- **Βάσεις Δεδομένων**

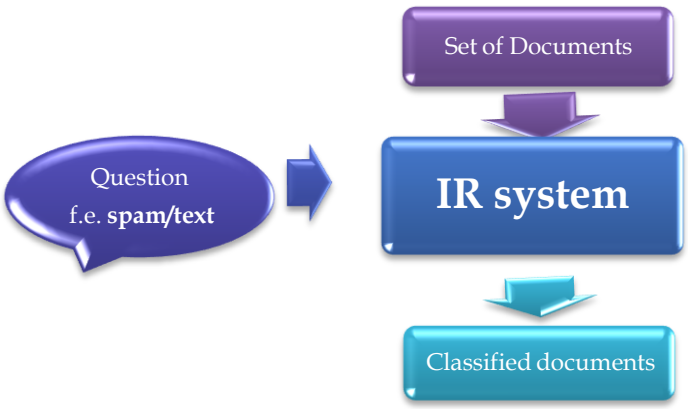
Μια βάση δεδομένων είναι μια συλλογή από δεδομένα που έχουν μια ορισμένη δομή με την οποία σχετίζονται. Ο τομέας της εξόρυξης δεδομένων με τον τομέα βάσης δεδομένων είναι αλληλένδετοι, διότι χωρίς συστήματα διαχείρισης δεδομένων δεν μπορούμε να εφαρμόσουμε αλγορίθμους εξόρυξης δεδομένων. Ένα πετυχημένο παράδειγμα συνδυασμού εξόρυξης δεδομένων και βάσεων δεδομένων είναι η μηχανή αναζήτησης Google, η οποία εκτελεί εργασίες γρήγορα, αποδοτικά και με ακριβή αποτελέσματα.

2.4.2 Ανάκτηση Πληροφοριών "Information Retrieval"

Τα συστήματα ανάκτησης πληροφοριών, προσδιορίζουν τα έγγραφα σε μια συλλογή που ταιριάζουν με το ερώτημα του χρήστη. Τα πιο γνωστά συστήματα ανάκτησης πληροφοριών είναι οι μηχανές αναζήτησης όπως το Google ™, τα οποία προσδιορίζουν τα εν λόγω έγγραφα στον Παγκόσμιο Ιστό που είναι σχετικά με μια σειρά από λέξεις. Τα

συστήματα ανάκτησης πληροφοριών, επιτρέπουν να περιοριστεί το σύνολο των εγγράφων που σχετίζονται με ένα συγκεκριμένο πρόβλημα. Όπως η εξόρυξη κείμενου περιλαμβάνει την εφαρμογή υπολογιστικά απαιτητικών αλγορίθμων σε μεγάλες συλλογές εγγράφων. Η ανάκτηση πληροφοριών μπορεί να επιταχύνει σημαντικά την ανάλυση, με τη μείωση του αριθμού των εγγράφων προς ανάλυση. Γενικά, στην ανάκτηση πληροφορίας γίνεται μια ερώτηση και στόχος είναι να εξαχθούν όλα τα έγγραφα που είναι πιο κοντά στην ερώτηση, και όχι εύρεση νέας γνώσης [07, 09].

Ως μέθοδος της εξόρυξης κειμένου η ανάκτηση πληροφορίας χρησιμοποιείται ως εξής [21] : Σε ένα σύστημα ανάκτησης πληροφορίας δίνονται ως είσοδοι ένα σύνολο εγγράφων κειμένου και ένα ερώτημα. Ενώ το σύστημα ως έξοδο δίνει ένα σύνολο ταξινομημένων εγγράφων σχετικά με το ερώτημα. Σχηματικά η διαδικασία παρουσιάζεται παρακάτω:



Γράφημα 6. Μέθοδος Ανάκτησης Πληροφορίας.

Ένα ευφυές σύστημα ανάκτησης πληροφορίας, θα πρέπει να λαμβάνει υπόψη τα παρακάτω θέματα:

- Σημασία των λέξεων
    - Συνώνυμα
    - Διφορούμενες έννοιες
- Σειρά των λέξεων στο ερώτημα (σύνταξη)
- Αξιοπιστία της πηγής

2.4.3 Επεξεργασία Φυσικής Γλώσσας "Natural Language Process"

Η επεξεργασία φυσικής γλώσσας, είναι ένα από τα παλαιότερα και πιο δύσκολα προβλήματα στον τομέα της τεχνητής νοημοσύνης. Πιο συγκεκριμένα, έγκειται στην ανάλυση της ανθρώπινης γλώσσας, έτσι ώστε οι υπολογιστές να μπορούν να καταλάβουν φυσικές γλώσσες, όπως κάνουν οι άνθρωποι [09]. Παρά το γεγονός ότι αυτός ο στόχος είναι ακόμα αρκετά μακριά, η επεξεργασία φυσικής γλώσσας μπορεί να εκτελέσει ορισμένα είδη ανάλυσης με υψηλό βαθμό επιτυχίας. Ο ρόλος της επεξεργασίας φυσικής γλώσσας στην εξόρυξη κειμένου είναι να παρέχει στα συστήματα κατά τη φάση της εξαγωγής πληροφορίας, με γλωσσικά δεδομένα που χρειάζονται για να εκπληρώσουν την αποστολή τους [21]. Συχνά αυτό γίνεται με σχολιασμό εγγράφων με πληροφορίες όπως όρια πρότασης, με ετικέτες των μερών του λόγου και με τα αποτελέσματα ανάλυσης της πρότασης, η οποία μπορεί στη συνέχεια να διαβαστεί από τα εργαλεία ανάκτησης πληροφοριών .

Πιο αναλυτικά, το αδόμητο ή ημι-δομημένο σύνολο δεδομένων μέσα σε ένα κείμενο μετατρέπεται σε ένα δομημένο σύνολο αντιπροσωπευτικών όρων "feature extraction", με τη βοήθεια μιας σειράς τεχνικών και μεθόδων επεξεργασίας κειμένου σε φυσική γλώσσα [09]. Οι όροι που χαρακτηρίζουν το κείμενο είναι συνήθως λέξεις με υψηλής ποιότητας πληροφορίες.

Οι γενικές τεχνικές επεξεργασίας σε φυσική γλώσσα χρησιμοποιούν ορισμένους απλούς ευρετικούς κανόνες, οι οποίοι στηρίζονται στη συντακτική και σημασιολογική προσέγγιση και ανάλυση του κειμένου. Επιγραμματικά, αυτές οι μέθοδοι υλοποιούν []:

- Διαμερισμό στα συστατικά στοιχεία του κειμένου "tokenization",
- Τη χρήση διάταξης του κειμένου "structural data mining",
- Την απαλοιφή λέξεων που δε φέρουν ουσιαστική πληροφορία, όπως άρθρα και αντωνυμίες "Elimination of insignificant words",
- Τη γραμματική δεικτοδότηση "PoS tagging",
- Τη μορφολογική ανάλυση "morphological analysis",

- Τη συντακτική ανάλυση "syntactic analysis",

Γράφημα 7. Μέθοδοι επεξεργασίας φυσικής γλώσσας.

2.4.4 Εξαγωγή Πληροφοριών "Information Extraction"

Η εξαγωγή πληροφοριών, είναι η διαδικασία της αυτόματης απόκτησης δομημένων δεδομένων από ένα μη δομημένο έγγραφο φυσικής γλώσσας. Συχνά πρόκειται για τον καθορισμό της γενικής μορφής των στοιχείων που μας ενδιαφέρουν όπως ένα ή περισσότερα πρότυπα, τα οποία στη συνέχεια χρησιμοποιούνται για να κατευθύνουν τη διαδικασία εξαγωγής [09].

Τα συστήματα εξαγωγής πληροφορίας εξαρτώνται σε μεγάλο βαθμό από τα δεδομένα που προκύπτουν από τα συστήματα επεξεργασίας φυσικής γλώσσας. Τα συστήματα εξαγωγής πληροφορίας μπορούν να εκτελέσουν, βραχυπρόθεσμη ανάλυση, η οποία προσδιορίζει τους όρους σε ένα έγγραφο, όπου ένας όρος μπορεί να αποτελείται από μία ή περισσότερες λέξεις [21]. Αυτό είναι ιδιαίτερα χρήσιμο για τα έγγραφα που περιέχουν σύνθετους πολυγλωσσικούς όρους, όπως είναι οι επιστημονικές ερευνητικές εργασίες. Επίσης, αναγνωρίζει τις οντότητες- ονόματα, όπου προσδιορίζονται τα ονόματα σε ένα έγγραφο, όπως τα ονόματα των ανθρώπων ή των οργανισμών. Ορισμένα συστήματα είναι επίσης σε θέση να αναγνωρίζουν τις ημερομηνίες και εκφράσεις του χρόνου, τις ποσότητες και συναφείς μονάδες, ποσοστά, και ούτω καθεξής.

Τα δεδομένα που παράγονται κατά τη διάρκεια της διαδικασίας εξαγωγής πληροφορίας [22] συνήθως αποθηκεύονται σε μια βάση δεδομένων έτοιμα για ανάλυση στο τελικό στάδιο, από την εξόρυξη δεδομένων. Η εξαγωγή χαρακτηριστικών γνωρισμάτων πληροφορίας δεν μπορεί να ταξινομηθεί ως εξόρυξη κειμένου και αυτό διότι δεν περιλαμβάνει την έννοια της «καινούριας» πληροφορίας. Τα χαρακτηριστικά που εξάγονται είναι γνώση που είναι ήδη γνωστή.

Γράφημα 8. Διαδικασία εξαγωγής πληροφορίας.

Πιο αναλυτικά, σε ένα σύστημα εξαγωγής πληροφορίας δίνονται ως είσοδοι ένα σύνολο εγγράφων κειμένου και ένα καλά διατυπωμένο περιορισμένο ερώτημα. το ζητούμενο από το σύστημα αυτό είναι:

i. η εύρεση προτάσεων με σχετική πληροφορία,

ii. η εξαγωγή της σχετικής πληροφορίας και η απόρριψη της άσχετης,

iii. η σύνδεση της πληροφορίας και,

iv. η έξοδος της σε ένα προκαθορισμένο υπόδειγμα.

2.5 Μοντέλα Εξόρυξης Δεδομένων από Κείμενο

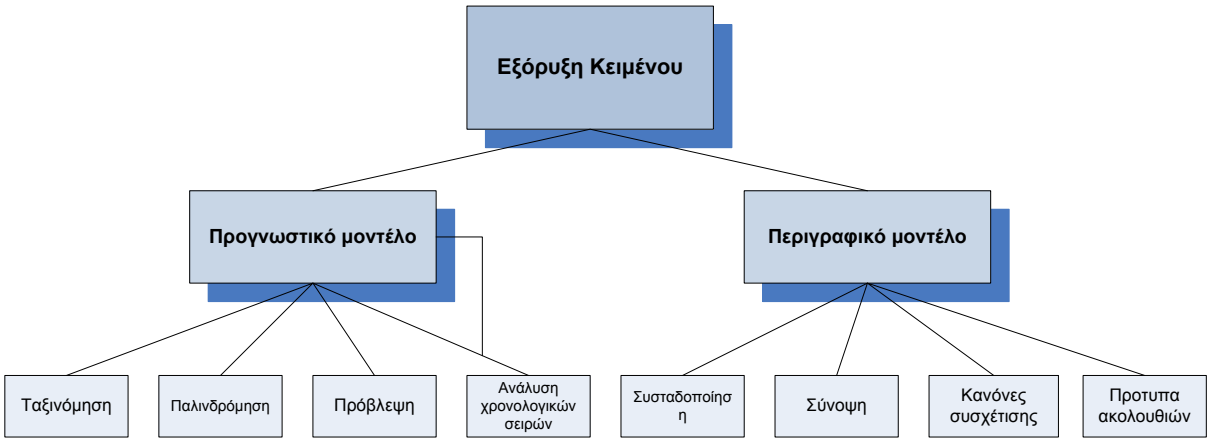
Στη συγκεκριμένη παράγραφο γίνεται μια σύντομη ανασκόπηση των μοντέλων και των βασικότερων κατηγοριών αλγορίθμων εξόρυξης κειμένου. Όλοι αυτοί οι αλγόριθμοι προσαρμόζουν ένα μοντέλο στα δεδομένα. Εξετάζουν τα δεδομένα και βρίσκουν ένα μοντέλο με τα χαρακτηριστικά των δεδομένων προς εξέταση. Οι αλγόριθμοι εξόρυξης κειμένου αποτελούνται από τρία μέρη:

i. Μοντέλο: Ο αλγόριθμος προσαρμόζει ένα μοντέλο στα δεδομένα.

ii. Προτίμηση: Υπάρχουν κριτήρια για την επιλογή ενός μοντέλου από ένα άλλο.

iii. Αναζήτηση: Ο αλγόριθμος απαιτεί μια τεχνική για την αναζήτηση στα δεδομένα.

Όπως φαίνεται στο Σχήμα 9, το μοντέλο εξόρυξης δεδομένων από κείμενο μπορεί να είναι είτε προγνωστικό είτε περιγραφικό [08].



Γράφημα 9. Μοντέλο Εξόρυξης Δεδομένων.

- **Προγνωστικό μοντέλο.** Το μοντέλο αυτό κάνει μία πρόβλεψη για τις τιμές των δεδομένων, χρησιμοποιώντας γνωστά αποτελέσματα που έχει βρει από άλλα δεδομένα. Η μοντελοποίηση της πρόβλεψης μπορεί να γίνει με βάση τη χρήση ιστορικών δεδομένων. Η πρόβλεψη μπορεί να χρησιμοποιηθεί επίσης για να υποδηλώσει ένα συγκεκριμένο τύπο λειτουργίας εξόρυξης γνώσης από δεδομένα.

  Προκειμένου να καταστεί σαφής η έννοια του προβλεπτικού μοντέλου, παρατίθεται το εξής παράδειγμα: Η πρόγνωση του καιρού είναι μία προσέγγιση που περιλαμβάνει την χρήση μετεωρολογικών οργάνων που έχουν τοποθετηθεί σε διάφορα γεωγραφικά ύψη και μέρη ανά τη Γη, αλλά και σε δορυφόρους. Αυτά τα όργανα συλλέγουν δεδομένα σχετικά με την πρόγνωση του καιρού: ατμοσφαιρική πίεση, θερμοκρασία αέρα ή ατμόσφαιρας, υγρασία αέρα ή ατμόσφαιρας αλλά και η κατάσταση της θάλασσας, οι τύποι νεφών και η ηλιοφάνεια κ.ά.. Στην συνέχεια μπορεί να προβλεφθεί ο καιρός σε ένα γεωγραφικό σημείο της Γης, βάσει των δεδομένων που συλλέχθηκαν από αισθητήρες και όργανα που βρίσκονται στο

συγκεκριμένο γεωγραφικό σημείο. Η πρόβλεψη πρέπει να γίνει σε σχέση με το χρόνο που συλλέχθηκαν τα δεδομένα.

Οι πιο συνηθισμένες εργασίες εξόρυξης γνώσης από δεδομένα που χρησιμοποιούν αυτό το είδος μοντέλου, είναι:

i. Η κατηγοριοποίηση, όπου απεικονίζει τα δεδομένα σε προκαθορισμένες ομάδες ή κατηγορίες – κλάσεις Αναφέρεται συχνά σαν εποπτευόμενη μάθηση, επειδή οι κατηγορίες – κλάσεις καθορίζονται πριν ακόμη εξεταστούν τα δεδομένα. Η αναγνώριση προτύπου "pattern recognition" αποτελεί ένα είδος κατηγοριοποίησης, όπου ένα πρότυπο εισόδου κατηγοριοποιείται σε μία από διάφορες κατηγορίες, με βάση την εγγύτητά του ως προς αυτές τις προκαθορισμένες κατηγορίες.

ii. Αντίστοιχα, η παλινδρόμηση "Regression": χρησιμοποιείται για να απεικονιστεί ένα στοιχειώδες δεδομένο σε μία πραγματική μεταβλητή πρόβλεψης. Περιλαμβάνει την εκμάθηση της συνάρτησης που κάνει αυτή την απεικόνιση. Η παλινδρόμηση προϋποθέτει ότι τα σχετικά δεδομένα ταιριάζουν με μερικά γνωστά είδη συνάρτησης και μετά καθορίζει την καλύτερη συνάρτηση αυτού του είδους που μοντελοποιεί τα δεδομένα που έχουν δοθεί. Ένα είδος ανάλυσης σφάλματος χρησιμοποιείται για να καθορίσει ποια συνάρτηση είναι η «καλύτερη».

iii. Η τρίτη εργασία, η ανάλυση χρονοσειρών "Time Series Analysis", περιλαμβάνει την διαδικασία μελέτης της τιμής ενός γνωρίσματος καθώς μεταβάλλεται στο χρόνο. Οι τιμές συνήθως λαμβάνονται σε ίσα χρονικά διαστήματα (ημερήσια, εβδομαδιαία, ωριαία, κ.ο.κ.). Για να παρασταθούν οπτικά οι χρονοσειρές, χρησιμοποιείται ένα διάγραμμα χρονοσειρών. Τρεις βασικές λειτουργίες πραγματοποιούνται στην ανάλυση χρονοσειρών: στη μία περίπτωση χρησιμοποιούνται μονάδες μέτρησης απόστασης για να καθορίσουν την ομοιότητα ανάμεσα σε διαφορετικές χρονοσειρές, στη δεύτερη εξετάζεται η δομή της χρονοσειράς για να καθορίσει (και ίσως να κατηγοριοποιήσει) την συμπεριφορά της και στην τρίτη χρησιμοποιούνται διαγράμματα χρονοσειρών για την πρόβλεψη μελλοντικών τιμών.

iv. Τέλος, η εργασία της πρόβλεψης μπορεί να θεωρηθεί σαν ένα είδος κατηγοριοποίησης. Αυτή η εργασία εξόρυξης γνώσης είναι διαφορετική από το μοντέλο πρόβλεψης, παρόλο που η διαδικασία πρόβλεψης αποτελεί έναν τύπο μοντέλου πρόβλεψης. Η διαφορά είναι ότι ως πρόβλεψη θεωρείται περισσότερο το να δίνεται τιμή σε μία μελλοντική κατάσταση, παρά σε μία τρέχουσα. Εδώ, γίνεται αναφορά σε ένα είδος εφαρμογής, παρά σε μία προσέγγιση μοντελοποίησης. Οι εφαρμογές πρόβλεψης περιλαμβάνουν πρόγνωση καιρού, αναγνώριση ομιλίας, μηχανική μάθηση και αναγνώριση προτύπου. Αν και μπορούν να προβλεφθούν οι μελλοντικές τιμές με τεχνικές ανάλυσης χρονοσειρών ή παλινδρόμησης, μπορούν να χρησιμοποιηθούν επίσης και άλλες προσεγγίσεις.

- **Περιγραφικό μοντέλο.** Το μοντέλο αυτό αναγνωρίζει πρότυπα ή συσχετίσεις στα δεδομένα. Αντίθετα με το προγνωστικό μοντέλο, το περιγραφικό εξυπηρετεί στην εξέταση των ιδιοτήτων των δεδομένων προς εξέταση και δεν προβλέπει νέες ιδιότητες.

  Ένα παράδειγμα περιγραφικού μοντέλου είναι το εξής: Μια αλυσίδα πολυκαταστημάτων δημιουργεί ειδικούς καταλόγους, που στοχεύουν σε διάφορες δημογραφικές ομάδες, με βάση γνωρίσματα όπως το εισόδημα, ο τόπος διαμονής και τα φυσικά χαρακτηριστικά των δυνητικών πελατών (ηλικία, ύψος, βάρος κλπ). Προκειμένου να καθορίσει σε ποιους από τους πελάτες των διαφόρων καταλόγων θα σταλεί ταχυδρομικά διαφημιστικό υλικό και προκειμένου να δημιουργηθούν καινούργιοι και πιο συγκεκριμένοι κατάλογοι, η εταιρεία κάνει συσταδοποίηση των πιθανών πελατών βασιζόμενη στις προκαθορισμένες τιμές γνωρισμάτων. Τα αποτελέσματα της συσταδοποίησης χρησιμοποιούνται στη συνέχεια από τη διεύθυνση προκειμένου να δημιουργηθούν ειδικοί κατάλογοι που θα διανεμηθούν στο πιο κατάλληλο τμήμα του πληθυσμού, βάσει της ομάδας που αντιστοιχεί σε αυτόν τον κατάλογο.

  Οι πιο συνηθισμένες εργασίες εξόρυξης γνώσης από δεδομένα που χρησιμοποιούν το περιγραφικό μοντέλο, είναι οι ακόλουθες:

i. Η συσταδοποίηση είναι κάτι αντίστοιχο με την κατηγοριοποίηση που παρουσιάζεται στην επόμενη ενότητα, εκτός από το ότι οι συστάδες – ομάδες δεδομένων δεν είναι προκαθορισμένες, αλλά ορίζονται κυρίως από τα ίδια δεδομένα. Η συσταδοποίηση αναφέρεται εναλλακτικά και σαν μη εποπτευόμενη μάθηση, ή τμηματοποίηση. Μπορεί να θεωρηθεί σαν μια διαμέριση ή τμηματοποίηση των δεδομένων σε ομάδες, που μπορεί να είναι ή να μην είναι διακριτές μεταξύ τους. Συνήθως επιτυγχάνεται με τον καθορισμό της ομοιότητας, ως προς τα προκαθορισμένα γνωρίσματα, ανάμεσα στα δεδομένα. Τα πιο σχετικά δεδομένα κατατάσσονται στις ίδιες ομάδες. Εάν οι ομάδες δεν είναι προκαθορισμένες, χρειάζεται ένας ειδικός του πεδίου για να ερμηνεύσει τη σημασία των συστάδων που δημιουργούνται. Μια ειδική κατηγορία συσταδοποίησης ονομάζεται κατάτμηση “segmentation”. Με την κατάτμηση, μια βάση δεδομένων χωρίζεται σε διακριτές ομάδες παρόμοιων εγγραφών που ονομάζονται τμήματα “segments”. Η κατάτμηση συχνά θεωρείται πανομοιότυπη με την συσταδοποίηση. Κατά άλλους, η κατάτμηση θεωρείται σαν ειδικός τύπος συσταδοποίησης που εφαρμόζεται στην ίδια βάση δεδομένων.

ii. Η παρουσίαση συνόψεων απεικονίζει τα δεδομένα σε υποσύνολά τους με συνοδευτικές απλές περιγραφές. Η σύνοψη των δεδομένων ονομάζεται επίσης και χαρακτηρισμός “characterization” ή γενίκευση “generalization”. Εξάγει ή παράγει αντιπροσωπευτικές πληροφορίες σχετικά με τις βάσεις δεδομένων. Αυτό γίνεται ανακτώντας, στην πραγματικότητα, τμήματα από τα δεδομένα. Εναλλακτικά, μπορούν να εξαχθούν από τα δεδομένα συνοπτικές πληροφορίες (όπως είναι ο μέσος όρος κάποιου αριθμητικού γνωρίσματος). Εν ολίγοις, η παρουσίαση συνόψεων χαρακτηρίζει τα περιεχόμενα της βάσης δεδομένων.

iii. Αντίστοιχα, οι κανόνες συσχετίσεων περιλαμβάνουν την ανάλυση συνδέσμων “link analysis”, που εναλλακτικά αναφέρεται και σαν ανάλυση συγγένειας “affinity analysis” ή συσχέτιση “association”. Πρόκειται ουσιαστικά για τη διαδικασία εκείνη της εξόρυξης γνώσης που αποκαλύπτει συσχετίσεις μεταξύ των δεδομένων. Ένας κανόνας συσχέτισης “association rule”, είναι ένα μοντέλο που αναγνωρίζει ειδικούς

τύπους συσχέτισης μεταξύ των δεδομένων. Η πιο διαδεδομένη προσέγγιση για την εύρεση κανόνων συσχετίσεων χρησιμοποιεί τα συχνά στοιχειοσύνολα "frequent itemsets", τα οποία ορίζονται ως τα στοιχειοσύνολα εκείνα των οποίων ο αριθμός των εμφανίσεων είναι πάνω από ένα σημείο s. Η προσέγγιση των συχνών στοιχειοσυνόλων εντοπίζει τα συχνά στοιχειοσύνολα βάσει του ορισμού τους και δημιουργεί κανόνες από τα συχνά στοιχειοσύνολα.

Οι συσχετίσεις συχνά χρησιμοποιούνται στις λιανικές πωλήσεις για να αναγνωρισθούν προϊόντα που συχνά αγοράζονται μαζί. Συσχετίσεις χρησιμοποιούνται επίσης σε πολλές άλλες εφαρμογές, όπως είναι η πρόβλεψη της αποτυχίας λειτουργίας των τηλεπικοινωνιακών διακοπτών. Οι συσχετίσεις αυτές μπορεί να μην αντιπροσωπεύουν καμία έμφυτη σχέση ανάμεσα στα δεδομένα, κάτι που ισχύει για παράδειγμα στις συναρτησιακές εξαρτήσεις.

iv. Τέλος, η ανακάλυψη ακολουθιών αφορά στην ακολουθιακή ανάλυση "sequential analysis" ή αλλιώς ανακάλυψη ακολουθιών "sequence discovery", η οποία χρησιμοποιείται για να καθορισθούν σειριακά πρότυπα στα δεδομένα. Αυτά τα πρότυπα βασίζονται σε μία χρονική ακολουθία ενεργειών και είναι παρόμοια με τις συσχετίσεις στο ότι τα δεδομένα που εξάγονται συσχετίζονται, με τη διαφορά ότι η συσχέτισή τους αυτή βασίζεται στο χρόνο.

2.6 Τεχνικές Εξόρυξης Δεδομένων από Κείμενο

Η εξόρυξη κειμένου στοχεύει στην εξαγωγή πληροφοριών από μεγάλο όγκο κειμένων οι οποίες μπορεί να φανούν χρήσιμες προς το χρήστη, μέσω της ανακάλυψης προτύπων ανάμεσα στις πληροφορίες και τα μεταδεδομένα που έχουν προκύψει από αυτά ύστερα από επεξεργασία των μη δομημένων δεδομένων τους. Οι κυριότερες τεχνικές στην εξόρυξη κειμένου συνοψίζονται στις παρακάτω:

- Εξαγωγή χαρακτηριστικών γνωρισμάτων "Feature Extraction": Έχει ως στόχο τον προσδιορισμό γεγονότων και σχέσεων στο κείμενο, διακρίνοντας συχνά

εάν κάποια ονομαστική φράση είναι πρόσωπο, θέση, οργανισμός ή άλλο διακριτό αντικείμενο. Οι αλγόριθμοι εξαγωγής χαρακτηριστικών περιλαμβάνουν [09]:

i. την εξαγωγή ονόματος, εντοπίζοντας εμφανίσεις ονομάτων στο κείμενο και καθορίζοντας σε ποιο τύπο οντότητας αναφέρεται το όνομα,

ii. την εξαγωγή όρου μιας περιοχής, προσδιορίζοντας τεχνικούς όρους σε ένα κείμενο και,

iii. την αναγνώριση συντμήσεων, προσδιορίζοντας συντμήσεις και αρκτικόλεξα και αντιστοιχίζοντας στην πλήρη μορφή τους.

Οι αλγόριθμοι εξαγωγής χαρακτηριστικών μερικές φορές χρησιμοποιούν λεξικά για να αναγνωρίσουν κάποιους όρους, άλλες πάλι φορές χρησιμοποιούν λεκτικά πρότυπα "linguistic patterns" για να αναγνωρίσουν άλλους όρους. Για παράδειγμα, το όνομα ενός οργανισμού όπως «ΑΠΚΥ» μπορεί να μην υπάρχει σε ένα λεξικό αλλά ένας αλγόριθμος εξαγωγής χαρακτηριστικών θα μπορούσε να το αναγνωρίσει ως ουσιαστικό και μάλιστα ως έναν σημαντικό όρο. Αλγόριθμοι αναγνώρισης προτύπων (όπως παραδείγματος χάριν Hidden Markov Models HMM), θα μπορούσαν να εκπαιδευτούν για να αναγνωρίζουν λεκτικά πρότυπα, όπως μια φράση ουσιαστικού ακολουθείται από μια φράση ρήματος και αυτή συνήθως ακολουθείται από μια φράση ουσιαστικού όπως παραδείγματος χάριν στο «το ΑΠΚΥ έχει σπουδαστές». Φυσικά σχεδόν πάντα απαιτείται μια προ – και μια μετά – επεξεργασία για να καθοριστούν οι σημαντικοί όροι και άλλοι που εξάχθηκαν ως σημαντικοί, και τελικά δεν ήταν.

Επιπλέον, οι όροι θα πρέπει να είναι βρίσκονται σε μια κανονική ή καθιερωμένη μορφή. Αυτό κάνει την αναζήτηση και ανάκτηση όπως και άλλες διεργασίες που θα ακολουθήσουν πιο ακριβείς. Για παράδειγμα οι λέξεις «γράφοντας» και «γράφω» θα πρέπει να ανιχνευθούν ως η ίδια λέξη (τεχνική stemming).

Τέλος, η διεργασία της εξαγωγής χαρακτηριστικών θα πρέπει επίσης να μας παρουσιάζει και τον αριθμό των εμφανίσεων κάθε όρου "word frequency". Αυτό κυρίως υποστηρίζει τις διεργασίες Κατηγοριοποίησης κειμένου.

- Αναζήτηση και Ανάκτηση "Search and Retrieval": Περιλαμβάνει την αναζήτηση σε εσωτερικές συλλογές εγγράφων ή σε συλλογές που βρίσκονται στον Παγκόσμιο Ιστό. Κύριο χαρακτηριστικό αποτελεί η δυνατότητα, αφού αρχικά συνταχθεί ένα ευρετήριο, να προσφέρεται ένα αρκετά ευρύ φάσμα επιλογών αναζήτησης κειμένου, στις οποίες συμπεριλαμβάνονται οι βασικές επιλογές αναζήτησης όπως η Boolean (and/or/not), η index-based, η βασισμένη σε οντολογίες, ή στην αριθμητική σειρά, η τμηματική αναζήτηση "segment", αλλά και πιο σύνθετες επιλογές αναζήτησης όπως η αναζήτηση έννοιας, η ασαφής αναζήτηση, και άλλα.

- Πλοήγηση με βάση το κείμενο "Text Based Navigation": Η πλοήγηση με βάση το κείμενο επιτρέπει τους χρήστες να πλοηγούνται μέσα σε μια συλλογή εγγράφων με βάση σχετικά θέματα ή σημαντικούς όρους. Αυτό βοηθάει στον να αναγνωριστούν έννοιες κλειδιά και επιπλέον να αναγνωριστούν και συσχετίσεις μεταξύ σημαντικών όρων. Για παράδειγμα, όταν αναζητούμε έγγραφα με τον όρο «ΑΠΚΥ» θα πρέπει να έχουμε γρήγορη πρόσβαση και να μετακινούμαστε σε έννοιες όπως «καθηγητές του ΑΠΚΥ» ή «σπουδαστές του ΑΠΚΥ», όπως και σε άλλους παρεμφερείς όρους που να περιέχουν το «ΑΠΚΥ». Τα σημαντικά σημεία σε αυτήν την διεργασία είναι δύο.

    1. Η ικανότητα να μπορούμε να βλέπουμε και άλλους σχετικούς όρους. Για παράδειγμα, αν αναγνωρίσουμε ότι δύο όροι εμφανίζονται συχνά μαζί, τότε θα μπορούμε να υποθέσουμε ότι αυτοί οι δυο όροι έχουν πιθανόν κάποια σχέση μαζί μεταξύ τους.

    2. Η ικανότητα να μπορούμε να μεταβούμε από ένα ζευγάρι εννοιών σε ένα άλλο ζευγάρι εννοιών, όπως παραδείγματος χάριν αν δεν μας ικανοποιεί το ζευγάρι «ΑΠΚΥ» και «καθηγητές» τότε ίσως να μας ικανοποιήσει το ζευγάρι «ΑΠΚΥ» και «σπουδαστές».

- Κατηγοριοποίηση, κατάταξη με επίβλεψη "Categorization, Supervised Classification" [09, 22]: Η κατηγοριοποίηση είναι η διαδικασία της κατάταξης εγγράφων σε προκαθορισμένες κατηγορίες, μας βοηθάει λοιπόν στο να προσδιορίσουμε ποια είναι τα κύρια θέματα μιας συλλογής εγγράφων. Οι κατηγορίες είτε έχουν διαμορφωθεί εξαρχής από τον προγραμματιστή είτε μπορούν να προσδιοριστούν από το χρήστη. Οι τεχνικές της κατηγοριοποίησης κειμένων, αποτελούν σημαντικό στοιχείο σε πολλά θέματα διαχείρισης πληροφορίας όπως:

    - Κατηγοριοποίηση αρχείων σε ιεραρχίες φακέλων.

    - Αναγνώριση θεμάτων για την υποστήριξη διαδικασιών επεξεργασίας κειμένων συγκεκριμένης θεματολογίας.

    - Κατηγοριοποίηση ηλεκτρονικού ταχυδρομείου σε πραγματικό χρόνο.

    - Εύρεση εγγράφων που ταιριάζουν με τα ενδιαφέροντα και τις προτιμήσεις των χρηστών.

    - Δομημένη έρευνα και έρευνα για πληροφορία.

Επειδή η διαδικασία της κατηγοριοποίησης είναι αρκετά απαιτητική αναφορικά με το χρόνο εκτέλεσης και έχει μεγάλο κόστος, περιορίζεται η εφαρμογή της. Με σκοπό την αυτόματη κατηγοριοποίηση κειμένων έχει εφαρμοστεί ένας μεγάλος αριθμός από τεχνικές στατιστικής Κατηγοριοποίησης και μηχανικής μάθησης που περιλαμβάνει τις εξής:

i. Τεχνητά νευρωνικά δίκτυα

ii. Επαγωγή κανόνων

iii. Αλγορίθμους μάθησης κανόνων

iv. Καταλληλότητα της ανατροφοδότησης

v. Ψηφιζόμενη Κατηγοριοποίηση

vi. Μηχανές διανυσμάτων υποστήριξης

vii. Δέντρα αποφάσεων

viii. Ταξινομητές Bayesian

ix. Ταξινομητές του πλησιέστερου γείτονα

x. Παλινδρομικά μοντέλα

Υπάρχουν δύο τρόποι για την κατηγοριοποίηση:

1. Αυτή η διαδικασία, περιλαμβάνει τη δημιουργία ενός θησαυρού, δηλαδή ενός συνόλου που περιλαμβάνει όρους σχετικούς με το θέμα κάθε κατηγορίας καθώς και συσχετίσεις μεταξύ αυτών των όρων (για παράδειγμα διευρυμένους όρους, κοντινότερους όρους, συνώνυμα, σχετικούς όρους) και τελικά τον ορισμό του θέματος του κειμένου με βάση τη συχνότητα των όρων σχετικών με το θέμα που υπάρχουν στο έγγραφο.

2. Ενώ η δεύτερη διαδικασία, περιλαμβάνει την εκπαίδευση "training" του εργαλείου κατηγοριοποίησης με κάποια δείγματα από τα έγγραφα, τη στατιστική ανάλυση λεκτικών προτύπων "linguistic patterns" όπως είναι οι λεξικολογικές συγγένειες, οι συχνότητες λέξεων των εγγράφων προς εκπαίδευση, το χωρισμό αυτών των προτύπων σε κατηγορίες (με στατιστικό τρόπο), και τέλος την Κατηγοριοποίηση των υπόλοιπων εγγράφων.

   Τέλος, η δεύτερη προσέγγιση είναι προτιμότερη όταν έχουμε να κάνουμε με μεγάλους τομείς, καθώς τότε είναι αρκετά δύσκολο να δημιουργηθεί κάποιος θησαυρός εννοιών. Για να αποφύγουμε την λάθος κατάταξη ενός εγγράφου είναι μερικές φορές απαραίτητο να εφοδιάζουμε με διάφορες κατηγορίες κάθε έγγραφο.

- Συσταδοποίηση, μη επιβλεπόμενη κατάταξη "Clustering, Unsupervised Classification" [09, 22]: Μία συστάδα είναι μια συλλογή από σχετικά έγγραφα,

και η συσταδοποίηση είναι η διαδικασία της δημιουργίας ομάδων εγγράφων βάσει κάποιου κριτηρίου ομοιότητας, αυτόματα χωρίς να έχουμε προσδιορίσει από πριν τις κατηγορίες. Η συσταδοποίηση κειμένων κρίνεται χρήσιμη:

a. για τον προσδιορισμό κρυμμένων ομοιοτήτων,

b. για να διευκολύνει τη διαδικασία του να βρούμε παρόμοιες ή σχετικές πληροφορίες και,

c. για την παροχή μιας επισκόπησης περιεχομένου μιας μεγάλης συλλογής εγγράφων.

Οι πιο γνωστοί αλγόριθμοι που χρησιμοποιούνται είναι οι ιεραρχικοί, διαχωριστικοί, οι δυαδικοί σχεσιακοί και οι ασαφείς. Αφού επιλεγούν τα γνωρίσματα της επεξεργασίας, τότε οι διαδικασίες της συσταδοποίησης τα επεξεργάζονται και δημιουργούν ένα σύνολο συστάδων. Στις συστάδες αυτές, θα ανατεθούν τα έγγραφα της επεξεργασίας, οι οποίες ανακαλύπτονται από την επεξεργασία των περιεχομένων των εγγράφων.

- Αυτόματη δημιουργία περίληψης "Summarization" [22]: Η διαδικασία της δημιουργίας περιλήψεων κειμένων, αποσκοπεί στην παρουσίαση των κύριων σημείων των κειμένων που επεξεργάζονται, σε μια περιεκτική και κατανοητή μορφή. Αποτελεί την εξαγωγή της περίληψης ενός κειμένου, δηλαδή τη μείωση του μεγέθους του κειμένου διατηρώντας όμως τα βασικά στοιχεία του περιεχομένου του. Σε αυτή τη λειτουργία ο χρήστης έχει συνήθως τη δυνατότητα να καθορίσει διάφορες παραμέτρους, όπως το πλήθος των λέξεων που θα εξαχθούν ή το ποσοστό επί του συνολικού κειμένου που θα αποτελεί την περίληψη.

- Γλωσσικός προσδιορισμός "Language Identification", και απόδοση κειμένου στο συγγραφέα : Ένα εργαλείο γλωσσικού προσδιορισμού μπορεί να προσδιορίσει σε ποια γλώσσα είναι γραμμένο ένα κείμενο, ή και τι ποσοστό του κειμένου είναι γραμμένο σε κάθε γλώσσα, εάν αυτό είναι γραμμένο σε περισσότερες. Ο προσδιορισμός είναι βασισμένος σε ένα σύνολο

εγγράφων κατάρτισης στις γλώσσες. Η ακρίβεια των γλωσσικών προσδιοριστικών είναι συνήθως απόλυτα επιτυχημένη.

Επιπλέον, υπάρχει η δυνατότητα προσδιορισμού του συγγραφέα στον οποίο ανήκει το κείμενο, χρησιμοποιώντας τεχνικές εξόρυξης δεδομένων. Για παράδειγμα, μπορεί να χρησιμοποιηθεί μια μεθοδολογία που βασίζεται περισσότερο στην ανάλυση του περιεχομένου από ότι στη σύνταξη με χρήση μια τεχνικής κανόνων συσχέτισης [07].

- Συσχετίσεις "Associations": Στην ανάλυση συσχετίσεων, αναγνωρίζονται σχέσεις μεταξύ χαρακτηριστικών γνωρισμάτων που έχουν εξαχθεί από τη συλλογή εγγράφων, και ορίζεται ένα πρότυπο με τη χρήση μιας αντικειμενικής συσχέτισης. Το πρότυπο αυτό, εκφράζει έναν κανόνα που αναφέρει ότι αν βρεθεί η υπολέξη που περιέχεται στο πρότυπο, ακολουθούμενη από μία άλλη δεδομένη υπολέξη, σε συγκεκριμένη απόσταση μεταξύ τους, τότε η αντικειμενική συνθήκη θα διατηρηθεί με μεγάλη πιθανότητα. Οι κανόνες αυτοί είναι πολύ ευέλικτοι για την περιγραφή των τοπικών ομοιοτήτων που περιέχονται στα δεδομένα του κειμένου.

- Απεικόνιση – Οπτικοποίηση "Visualization": Η απεικόνιση χρησιμοποιεί την εξαγωγή χαρακτηριστικών γνωρισμάτων και το ευρετήριο βασικών όρων για να κατασκευάσει μια γραφική αναπαράσταση μιας συλλογής εγγράφων. Η προσέγγιση αυτή βοηθάει το χρήστη να αναγνωρίζει πολύ γρήγορα τα κύρια θέματα και τις βασικές έννοιες των κειμένων, με βάση τη σπουδαιότητα (π.χ. μέγεθος) αυτών στην αναπαράσταση.

  Οι στόχοι της απεικόνισης κειμένων, είναι η δημιουργία υπολογιστικών μετασχηματισμών ώστε να μειώσουν τη γνωστική προσπάθεια της εξέτασης των μεγάλων σωμάτων κειμένων και να βοηθήσουν στην ανακάλυψη νέας γνώσης [22]. Η απεικόνιση κειμένων φαίνεται να ενδιαφέρεται για την «μεγάλη εικόνα». Τα περισσότερα συστήματα εξόρυξης κειμένου έχουν τμήματα απεικόνισης κειμένων.

2.7 Μεθοδολογία Εξόρυξης Δεδομένων από Κείμενο

Η εξόρυξη κειμένου περιέχει μεθόδους από διάφορα τεχνολογικά πεδία, εντούτοις όλοι αυτοί οι μέθοδοι μπορούν να ομαδοποιηθούν σε δύο κύριους τίτλους. Οι δύο ευρείες ομάδες για την ανάπτυξη των συστημάτων που στοχεύουν να εξαγάγουν τις πληροφορίες και τη γνώση από κείμενο είναι [11]:

1. **Οι βασισμένες στην απόδοση**

2. **Οι βασισμένες στη γνώση**

Η πρώτη κατηγορία περιλαμβάνει μεθόδους βασισμένες στην απόδοση. Ενδιαφέρει δηλαδή περισσότερο η αποτελεσματική συμπεριφορά του συστήματος και όχι απαραίτητα τα μέσα με τα οποία λαμβάνεται αυτή η συμπεριφορά. Στην κατηγορία αυτοί περιλαμβάνονται:

i. διάφορες στατιστικές μέθοδοι που στηρίζονται συνήθως σε ένα σαφές θεμελιώδες πρότυπο πιθανότητας, καθώς και,

ii. τα νευρωνικά δίκτυα: συστήματα στα οποία οι διασυνδέσεις διαμορφώνονται όπως οι νευρώνες του εγκεφάλου, και οι οποίες μπορούν να αλλάξουν δυναμικά.

Η δεύτερη κατηγορία περιλαμβάνει μεθόδους βασισμένες στη γνώση. Χρησιμοποιούνται δηλαδή σαφείς αντιπροσωπεύσεις της γνώσης όπως οι έννοιες των λέξεων, οι σχέσεις μεταξύ των γεγονότων και των κανόνων για τα συμπεράσματα στις ιδιαίτερες περιοχές. Τέτοια συστήματα περιλαμβάνουν τους κανόνες διεξαγωγής συμπεράσματος, τις λογικές προτάσεις, τα σημασιολογικά δίκτυα (για παράδειγμα, ταξινομήσεις, οντολογίες), κανόνες ταιριάσματος των προτύπων και άλλα.

Οι ταξινομήσεις δεν είναι το μόνο είδος αντιπροσώπευσης της γνώσης. Οι κανόνες συσχέτισης των προτύπων χρησιμοποιούνται εκτενώς. Αυτοί οι τύποι κανόνων μπορούν να αντιπροσωπευθούν χρησιμοποιώντας IF-THEN συνθήκες. Ενώ οι

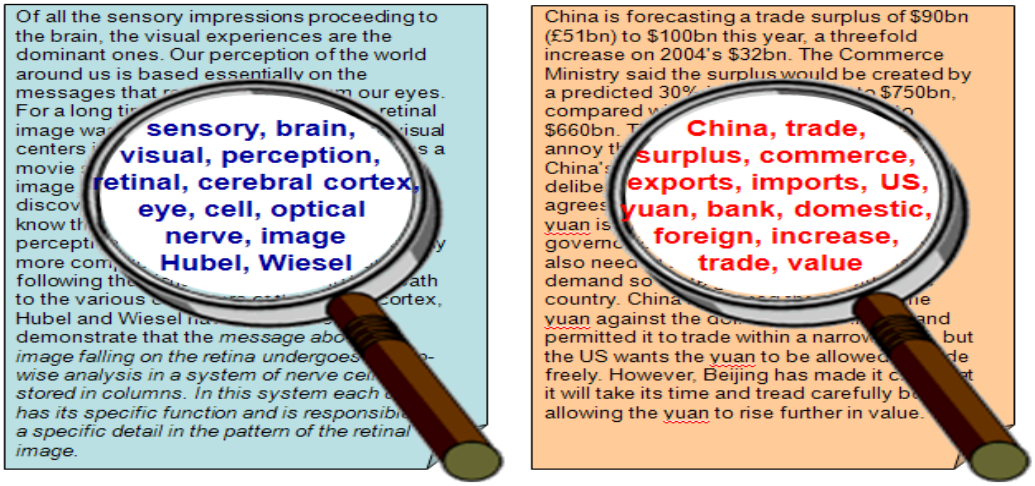
ταξινομήσεις πρέπει να είναι εξαρτώμενες από το πεδίο και η ευρετική ανάλυση πρέπει να είναι συγκεκριμένη όσον αφορά τη γλώσσα.

Τα περισσότερα βέβαια εργαλεία συνδυάζουν τις βασισμένες στην απόδοση με τις βασισμένες στη γνώση τεχνικές, προκειμένου να ισορροπηθούν η ευελιξία και η προσαρμοστικότητα των στατιστικών τεχνικών με την εκάστοτε περιοχή. Η επιλογή μεταξύ ενός στατιστικά προσανατολισμένου ή ενός βασισμένου στη γνώση εργαλείου εξαρτάται από την περιοχή στην οποία ενδιαφερόμαστε να κάνουμε εξόρυξη δεδομένων. Για παράδειγμα, για περιοχές που δεν αλλάζουν συχνά έννοιες και κανόνες, όπως είναι για παράδειγμα τα οικονομικά και η πολιτική, θα προτιμούσαμε κάποιον αλγόριθμο βασισμένο στη γνώση. Από την άλλη, σε μια περιοχή όπως η γενετική, η οποία αλλάζει συνεχώς έννοιες λόγω της ταχείας εξέλιξης του ερευνητικού αυτού τομέα, είναι προτιμότερο να χρησιμοποιηθεί κάποιο εργαλείο βασισμένο στην απόδοση.

2.7.1 Αναπαράσταση Κειμένου

Αφού μελετήσαμε τεχνικές και μεθόδους της εξόρυξης κειμένου, στη συνέχεια θα ασχοληθούμε με τον τρόπο αναπαράστασης κειμένου στη διαδικασία της εξόρυξης κειμένου. Λόγω της συχνής έλλειψης κάποιας δομής στα αρχεία κειμένων, είναι προφανής η ανάγκη εύρεσης μια αναπαράστασης για την αντιπροσώπευση των στοιχείων-όρων των κειμένων, έτσι ώστε να είναι δυνατή η μετέπειτα επεξεργασία τους.

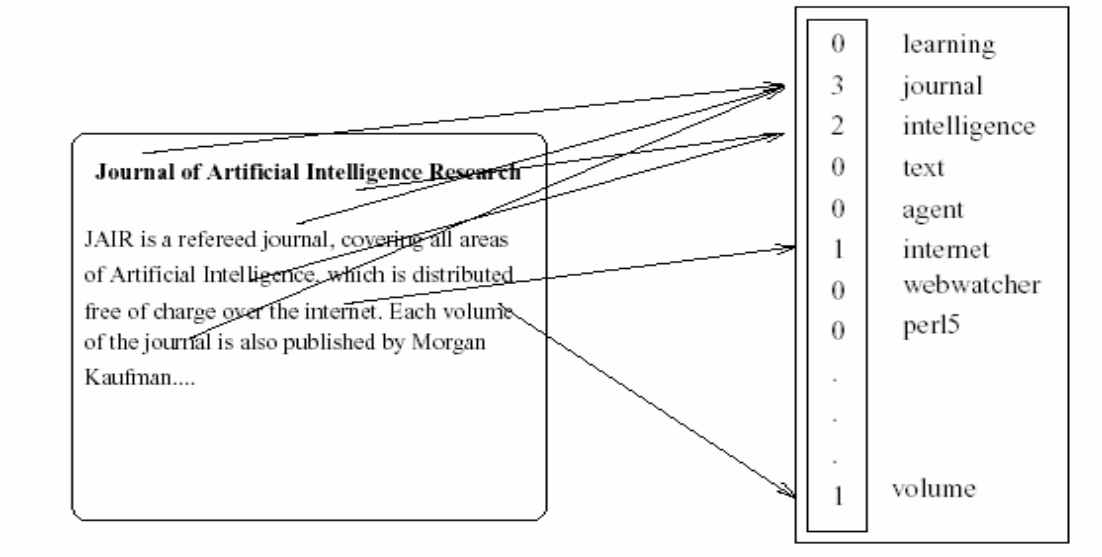
Όταν έχουμε μια συλλογή από αρχεία κειμένου, μπορούμε να θεωρήσουμε καθένα από αυτά ως ένα "bag-of-words", μια «σακούλα» η οποία περιλαμβάνει όλες τις λέξεις που βρίσκονται στο κείμενο [02].



Εικόνα 1. Κάθε αρχείο αποτελεί ένα bag-of-words.

Η πιο συχνά χρησιμοποιούμενη αναπαράσταση εγγράφων στην ανάκτηση πληροφοριών και στην μάθηση – κειμένου είναι η λεγόμενη αναπαράσταση διανύσματος "vector representation", η οποία προέρχεται από τα συστήματα ανάκτησης πληροφορίας "information retrieval". Έτσι, κάθε έγγραφο κειμένου από το σύνολο κειμένων, είναι ένα διάνυσμα όρων "term vector" στο οποίο κάθε όρος αποτελεί ένα μοναδικό ανεξάρτητο χαρακτηριστικό. Κάθε στοιχείο σε αυτό το διάνυσμα έχει και μια τιμή η οποία αντιστοιχεί στην εμφάνιση του όρου μέσα στο κείμενο [09].

Αυτή η αναπαράσταση δεν είναι τίποτα άλλο από μια "bag – of – words" «σακούλα λέξεων» αναπαράσταση: όλες οι λέξεις του κειμένου χρησιμοποιούνται, χωρίς να λαμβάνεται υπ' όψιν η σειρά των λέξεων ή κάποιου άλλου είδους δομής του κειμένου. Όταν έχουμε μια συλλογή κειμένων τότε κάθε κείμενο αναπαρίσταται με μια "bag – of – words", η οποία περιλαμβάνει όλες τις λέξεις που εμφανίζονται στο κείμενο [16].



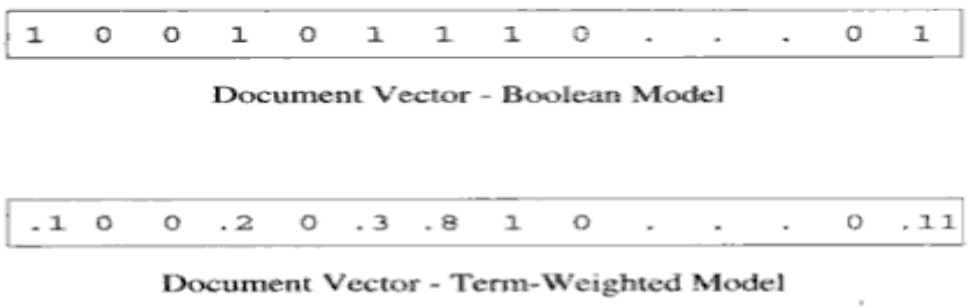
Εικόνα 2. Απεικόνιση της bag-of-words αναπαράστασης ενός κειμένου χρησιμοποιώντας ένα διάνυσμα συχνοτήτων.

Μερικές φορές μπορεί να χρησιμοποιηθούν και κάποιες επιπλέον πληροφορίες γύρω από το κείμενο που αναπαριστάται, όπως παραδείγματος χάριν, η δομή των προτάσεων, οι θέσεις των λέξεων ή οι γειτονικές λέξεις. Υπάρχουν κάποιες ενδείξεις στην ανάκτηση πληροφοριών που μας λένε ότι για μεγάλα κείμενα η επιπλέον θεώρηση άλλων πληροφοριών περάν της "bag –of –words" αναπαράστασης συχνοτήτων δεν αξίζει πραγματικά τον κόπο. Σ' αυτές τις εργασίες υποστηρίζεται ότι, η χρησιμοποίηση λέξεων κατά μονάδες και κατά ζευγάρια βελτιώνει την επίδοση της κατάταξης σχετικά μικρών κειμένων.

Πολλά από τα συστήματα που μαθαίνουν από κείμενο χρησιμοποιούν την "bag – of – words" αναπαράσταση χρησιμοποιώντας είτε λογικά (Boolean) χαρακτηριστικά για να αναπαραστήσουν το γεγονός, αν κάποια λέξη εμφανίζεται στο κείμενο ή όχι, είτε χρησιμοποιώντας την συχνότητα εμφάνισης κάθε λέξης. Υπάρχουν ακόμα και κάποιες εργασίες που χρησιμοποιούν κάποιες επιπλέον πληροφορίες όπως την θέση των λέξεων, ή χρησιμοποιούν n – άδες λέξεων (τα λεγόμενα n - grams). Πιο πρόσφατες δουλειές μας δείχνουν ότι η χρησιμοποίηση της δομής υπερκειμένου "hypertext structure" καθώς και η γραφική οργάνωση των ιστοσελίδων "graph organization of Web pages" βελτιώνει τα αποτελέσματα της κατάταξης.

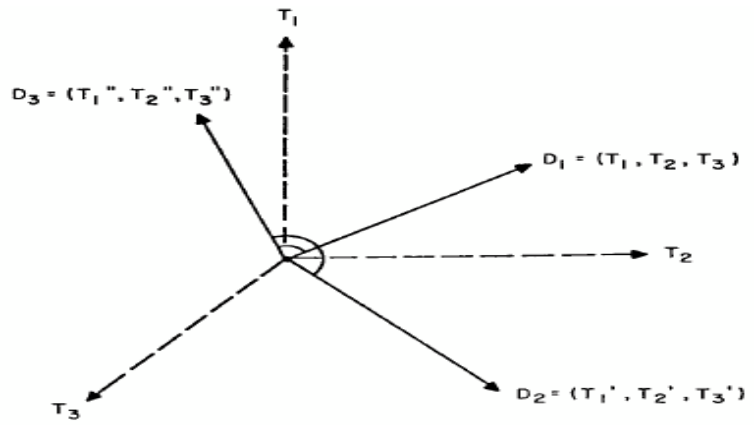
Με βάση αυτό μπορούμε να διακρίνουμε διάφορα μοντέλα διανυσματικής αναπαράστασης των κειμένων [02]:

Στο λογικό μοντέλο “Boolean model”, κάθε έγγραφο αναπαρίσταται από ένα σύνολο λογικών τιμών κάθε μία από τις οποίες δηλώνει εάν ένας συγκεκριμένος όρος εμφανίζεται στο έγγραφο: συνήθως η τιμή 1 σημαίνει ότι εμφανίζεται και η τιμή 0 σημαίνει απουσία του συγκεκριμένου όρου από το κείμενο. Τα πλεονεκτήματα του λογικού μοντέλου είναι η ευκολία και η ταχύτητα λειτουργιών ερώτησης, αναζήτησης, κα, εφόσον χρησιμοποιούνται λογικές πράξης AND, OR, NOT κλπ, και η δυνατότητα χρησιμοποίησης της Boolean άλγεβρας στο “Boolean Model”. Ωστόσο, το λογικό μοντέλο συνεπάγεται ότι η απάντηση στο κατά πόσον είναι σχετικό ένα κείμενο με ένα συγκεκριμένο όρο (και κατ’ επέκταση θέμα) είναι μια δυαδική απόφαση. Ενώ επιπλέον μία λογική τιμή για κάθε χαρακτηριστικό, δεν μπορεί να αποδώσει κατά πόσο σημαντική είναι η παρουσία μίας λέξης σε ένα κείμενο, γεγονός το οποίο συχνά μπορεί να οδηγήσει σε λάθος συμπεράσματα.



Εικόνα 3. Παράδειγμα ενός document vector σε boolean μορφή (πάνω), και σε term-weighted μορφή (κάτω).

Στο μοντέλο διανυσματικού χώρου “vector space model – VSM” τα αρχεία αναπαρίστανται ως διανύσματα σε ένα πολυδιάστατο Ευκλείδειο χώρο. Κάθε άξονας στο χώρο αντιστοιχεί σε ένα χαρακτηριστικό γνώρισμα “attribute”, δηλαδή σε έναν όρο/λέξη, με αποτέλεσμα η συντεταγμένη κάθε διανύσματος ως προς έναν άξονα να χαρακτηρίζει την εμφάνιση του όρου (στον οποίο αντιστοιχεί ο άξονας) στο συγκεκριμένο διάνυσμα-αρχείο κειμένου, και μάλιστα να αποτελεί ένα «βάρος» του όρου “term weight” ως προς το συγκεκριμένο κείμενο (πόσο σημαντικός θεωρείται δηλαδή ο όρος για το κείμενο). Τα βάρη που χρησιμοποιούνται για κάθε χαρακτηριστικό γνώρισμα, είναι πραγματικές τιμές και μπορεί να είναι είτε απλά η συχνότητα εμφάνισης της λέξης “word frequency”, είτε άλλες τιμές που θα μελετήσουμε ακολούθως. Τελικά, μια συλλογή εγγράφων αναπαρίσταται από ολόκληρο το διανυσματικό χώρο [09].



Εικόνα 4. Αναπαράσταση του Μοντέλου Διανυσματικού Χώρου.

Μια από τις πιο συχνά χρησιμοποιούμενες μεθόδους για την μείωση του αριθμού των διαφορετικών λέξεων που εμφανίζονται σε ένα κείμενο, είναι η διαγραφή των λέξεων που περιέχονται στην λεγόμενη "stop – list" λίστα, που περιέχει πάρα πολύ κοινές στην χρήση λέξεις μιας συγκεκριμένης γλώσσας, όπως παραδείγματος χάριν «και», «μια», «το», «με». Μια άλλη προσέγγιση είναι να παραλείπουμε τις λέξεις που δεν εμφανίζονται συχνά σε ένα κείμενο (συχνότητα εμφάνισης < min. συχνότητα). Μια άλλη πρακτική είναι και ο περιορισμός των λέξεων "word stemming", πράγμα που είναι συνδεδεμένο με την γλώσσα στο οποίο είναι γραμμένο το κείμενο, που χρησιμοποιείται και το οποίο συνίσταται στον περιορισμό των λέξεων χρησιμοποιώντας έναν αλγόριθμο περιορισμού ο οποίος παραδείγματος χάριν αντικαθιστά τις λέξεις «γράφω», «γράφοντας», με την λέξη «γραφή». Κάποιες προσεγγίσεις χρησιμοποιούν κάποιες ανεξάρτητες τεχνικές της γλώσσας και εισάγουν κάποιου είδους βαθμολογία των λέξεων "word scoring", με σκοπό να διαλέξουν μόνο τις καλύτερες λέξεις.

Ως μειονεκτήματα της μεθόδου του Μοντέλου Διανυσματικού Χώρου, είναι το γεγονός ότι είναι αρκετά αργή ως προς το χρόνο επεξεργασίας, λόγω της πληθώρας υπολογισμών που απαιτούνται, δεν εξυπηρετεί ιδιαίτερα την ενημέρωση αλλαγών στα κείμενα εφόσον για κάθε όρο προστίθεται ένας επιπλέον άξονας και πρέπει να γίνουν υπολογισμοί της συντεταγμένης για όλα τα διανύσματα στο χώρο, ενώ τέλος η πολυδιάστατη μορφή του απαιτεί κόστος μνήμης και χαμηλή ταχύτητα σε υπολογισμούς. Ως πιθανές λύσεις προτείνονται η χρήση συνόλων λέξεων-κλειδιά για την αναπαράσταση ενός αρχείου, η οποία θα μειώνει το πλήθος των διαστάσεων, καθώς και η χρήση n-άδων λέξεων (τα λεγόμενα n- grams), δηλαδή ακολουθιών από

n λέξεις (πχ το World Wide Web είναι ένα 3-gram), οι οποίες θα μπορούσαν να παρέχουν περισσότερη νοηματική πληροφορία για τα κείμενα από όσο μπορούν οι λέξεις μόνες τους.

Πειράματα με διαφορετικό πλήθος επιλεγμένων χαρακτηριστικών στην συσταδοποίηση κειμένων δείχνουν ότι καλύτερα αποτελέσματα επιτυγχάνονται όταν, είτε χρησιμοποιείται μόνο ένα μικρό μέρος από τα χαρακτηριστικά των κειμένων (10% περίπου) είτε όταν χρησιμοποιούνται όλα τα χαρακτηριστικά σε κάποιες από τις περιπτώσεις. Μια σύγκριση των διαφορετικών μέτρων βαθμολόγησης των λέξεων που χρησιμοποιούνται στην επιλογή των χαρακτηριστικών ενός υποσυνόλου από τα συνολικά χαρακτηριστικά ενός κειμένου, δείχνει ότι τα πιο υποσχόμενα χαρακτηριστικά λαμβάνουν υπ' όψιν τους την φύση του τομέα του προβλήματος και τον χρησιμοποιούμενο αλγόριθμο Κατηγοριοποίησης.

Υπενθυμίζεται ωστόσο και πάλι η ανάγκη για κανονικοποίηση της τιμής του βάρους, καθώς τα κείμενα μεγάλου μήκους τείνουν να έχουν μεγαλύτερες συχνότητες λέξεων καθώς και περισσότερους όρους. Υπάρχουν διάφοροι τρόποι κανονικοποίησης ως προς το μήκος των αρχείων "document length normalization", όπως για παράδειγμα, με πολλαπλασιασμό συχνότητας των όρων με κάποιο άλλο όρο, ή κανονικοποίηση της απόστασης μεταξύ των διανυσμάτων, την οποία και θα μελετήσουμε στην επόμενη ενότητα.

Συνοπτικά, μπορούμε να πούμε ότι για τον υπολογισμό του Μοντέλου Διανυσματικού Χώρου στο οποίο αντιστοιχεί μια συλλογή αρχείων, θα πρέπει αρχικά να γίνει μια προ-επεξεργασία των κειμένων: να αναγνωρισθούν δηλαδή οι λέξεις από τις οποίες αποτελείται κάθε κείμενο (bag- of-words) και μάλιστα, για να βελτιστοποιηθεί η διαδικασία εύρεσης των σημαντικών όρων κάθε κειμένου, οι λέξεις θα πρέπει να υπόκεινται σε κάποια επεξεργασία όπως αφαίρεση πολύ κοινών λέξεων οι οποίες δεν έχουν νοηματική αξία (άρθρα αντωνυμίες, κλπ), η εύρεση λέξεων αντιστοιχούν στο ίδιο θέμα αλλά έχουν διαφορετική μορφή (πχ παράγωγα της ίδιας λέξης, και η εύρεση τελικά όρων που είναι οι πιο αντιπροσωπευτικοί για κάθε κείμενο ξεχωριστά [16]. Τη διαδικασία αυτή της προ-επεξεργασίας θα την εξετάσουμε αναλυτικά σε επόμενη ενότητα. Μετά από αυτό το βήμα "document indexing", προχωράμε στο βήμα της ανάθεσης βάρους σε κάθε όρο για κάθε κείμενο

“term weighting” σε όλη τη συλλογή που έχουμε, ώστε κάθε βάρος να υποδηλώνει πόσο σημαντικός θεωρείται ο εκάστοτε όρος για το αντίστοιχο κείμενο.

Κεφάλαιο 3

Παρουσίαση των Συστημάτων

Στον χώρο της πληροφορικής και των ηλεκτρονικών υπολογιστών, με τον όρο λογισμικό ανοικτού κώδικα (Open Source Software, OSS), εννοείται το λογισμικό του οποίου ο πηγαίος κώδικας διατίθεται με κάποιον τρόπο ελεύθερα σε όσους ζητούν να τον εξετάσουν, να τον τροποποιήσουν ή να τον αξιοποιήσουν σε άλλες εφαρμογές [27, 43]. Κατά καιρούς έχουν εμφανιστεί αρκετές διαφορετικές άδειες χρήσης, οι οποίες είναι σχεδιασμένες να συνοδεύουν τα λογισμικά ανοικτού κώδικα.

Το ελεύθερο λογισμικό είναι ζήτημα ελευθερίας, όχι κόστους. Το ελεύθερο λογισμικό παρέχει στους χρήστες την ελευθερία να εκτελούν, να αντιγράφουν, να διανέμουν, να μελετούν, να τροποποιούν και να βελτιώνουν το ελεύθερο λογισμικό. Για την ακρίβεια, αναφέρεται σε τέσσερις βασικές ελευθερίες [41, 43]:

1. Την ελευθερία να εκτελείται το πρόγραμμα για οποιονδήποτε σκοπό (ελευθερία 0).

2. Την ελευθερία να μελετάται ο τρόπος λειτουργίας του προγράμματος και να προσαρμόζεται στις ανάγκες του χρήστη (ελευθερία 1). Η πρόσβαση στον πηγαίο κώδικα είναι προϋπόθεση για να ισχύει κάτι τέτοιο.

3. Την ελευθερία να αναδιανέμονται αντίγραφα του προγράμματος. (ελευθερία 2).

4. Την ελευθερία να βελτιώνεται το πρόγραμμα και να δημοσιεύονται οι βελτιώσεις στο ευρύ κοινό, ώστε να επωφελείται ολόκληρη η κοινότητα (ελευθερία 3). Η πρόσβαση στον πηγαίο κώδικα είναι προϋπόθεση για να ισχύει κάτι τέτοιο.

Το λογισμικό ανοικτού κώδικα δεν σημαίνει απαραιτήτως δωρεάν λογισμικό ούτε ελεύθερο λογισμικό, σύμφωνα με τον ευρύ ορισμό που δίνεται παραπάνω, αλλά αναφέρεται μόνο στο γεγονός πως επιτρέπεται σε κάθε χρήστη να εξετάσει και να χρησιμοποιήσει τη γνώση και τις δυνατότητες που προσφέρει ο παρεχόμενος πηγαίος κώδικας. Στην πράξη, τα περισσότερα προγράμματα ανοιχτού κώδικα παρέχονται δωρεάν και μπορούν να χαρακτηριστούν ελεύθερα.

Στην εξόρυξη δεδομένων υπάρχει μια κατηγορία εργαλείων που αναπτύχθηκαν από μια επιστημονική κοινότητα η οποία ασχολείται με την ανάλυση και την έρευνα δεδομένων, και κατ ‘επέκταση της εξόρυξης κειμένου. Τα εργαλεία εξόρυξης δεδομένων ανοιχτού κώδικα προσφέρονται στο ευρύτερο κοινό δωρεάν κάνοντας χρήση μιας άδειας ανοιχτού κώδικα, διαθέτοντας πολύ εξελιγμένες τεχνικές με την πορεία τους μέσα στα χρόνια, μεγάλη ευελιξία στη διαχείριση δεδομένων διαφόρων τύπων, δυνατότητα επεκτασιμότητας και ολοκληρωμένες και κατανοητές πλατφόρμες.

Στη συνέχεια παρατάσσονται τα χαρακτηριστικά, οι τεχνικές και τα εργαλεία που θα πρέπει να προσφέρονται στους αναλυτές από ένα σύστημα εξόρυξης κειμένου ανοιχτού κώδικα [43]:

1. Θα πρέπει να λειτουργεί στις μεγάλες συλλογές κειμένων φυσικής γλώσσας.

2. Θα πρέπει να χρησιμοποιεί περισσότερους αλγορίθμους από ότι τα ευρετικά και τα χειροκίνητα φιλτραρίσματα.

3. Θα πρέπει να εξάγει τις φαινομενολογικές μονάδες των πληροφοριών (π.χ. πρότυπα- υποδείγματα), αντί τα επιπρόσθετα έγγραφα.

4. Θα πρέπει να ανακαλύπτει νέες γνώσεις.

3.1 Επιλογή Συστημάτων και Γενικά Χαρακτηριστικά

Για τους σκοπούς της αξιολόγησης που θα ακολουθήσει, επιλέχθηκαν μόνο συστήματα ανοιχτού κώδικα, εφόσον, το κόστος διαδραματίζει ένα από τους ποιο σημαντικούς ρόλους στην έρευνα. Ενώ, παράλληλα στα εργαλεία ανοιχτού κώδικα μπορούν να ενσωματωθούν νέες, πειραματικές τεχνικές, συμπεριλαμβανομένων μερικών σε μορφή πρωτοτύπου και μπορούν να αντιμετωπιστούν τα αναδυόμενα προβλήματα νωρίτερα από ότι στα εμπορικά λογισμικά.

Επιπροσθέτως, τα εγχειρίδια χρήσης των λογισμικών ανοικτού κώδικα είναι διαθέσιμα σε πολλές μορφές και συχνά περιλαμβάνουν επιπλέον μαθήματα και περιπτώσεις χρήσης που γράφτηκαν από τους χρήστες των συστημάτων, έξω από την ομάδα ανάπτυξης του πυρήνα. Τέλος, υπάρχει υποστήριξη των χρηστών, η οποία είναι διαφορετική για τα λογισμικά ανοιχτού κώδικα απ' ότι για τα εμπορικά λογισμικά. Οι χρήστες των εμπορικών συστημάτων εξαρτώνται από το τμήμα υποστήριξης των χρηστών της εταιρείας, ενώ οι χρήστες συστημάτων ανοιχτού κώδικα είναι, ως θέμα αρχής, συνήθως πρόθυμοι να βοηθήσουν ο ένας τον άλλον. Γενικά, οι κοινότητες επικοινωνούν με ηλεκτρονικά φόρουμ, mailing lists και με συστήματα εντοπισμού σφαλμάτων για να παρέχεται ενθάρρυνση και ανατροφοδότηση για τους προγραμματιστές, ώστε να προτείνουν λύσεις και να δώσουν προτεραιότητα στη βελτίωση.

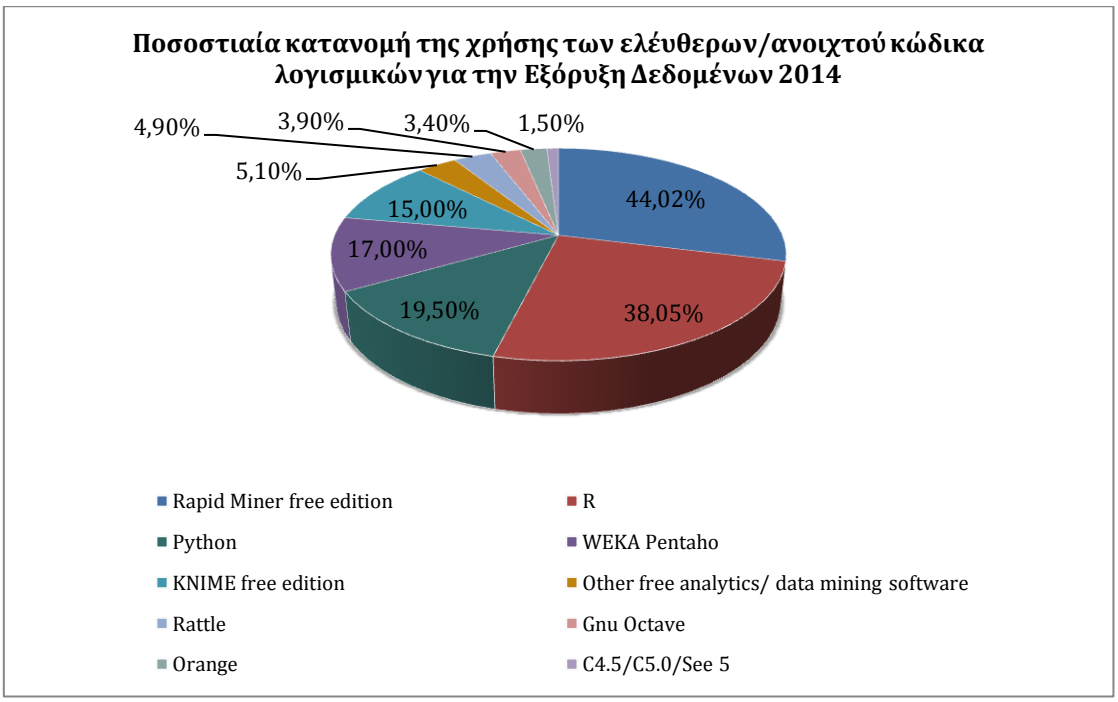
Δεδομένου ότι τα εργαλεία ανοικτού κώδικα ενσωματώνουν προόδους σε διεπαφές χρήστη και εργαλεία αναφοράς, με σκοπό να εφαρμόσουν τις πιο πρόσφατες μεθόδους

ανάλυσης, και να αναπτύξουν τις βάσεις τους, γίνονται χρήσιμες εναλλακτικές λύσεις για να συμπληρώσουν τα εμπορικά εργαλεία στην εξόρυξη δεδομένων.

Σε σύγκριση με τα εμπορικά λογισμικά εξόρυξης δεδομένων, συχνά τα εργαλεία ανοικτού κώδικα μπορεί να έχουν μειονεκτήματα. Έχουν αναπτυχθεί ως επί το πλείστον από ερευνητικές κοινότητες, που συχνά ενσωματώνουν πιο πρόσφατους αλγόριθμους ανάλυσης των δεδομένων τους, με αποτέλεσμα το λογισμικό να μην είναι τελείως σταθερό. Τα εμπορικά εργαλεία εξόρυξης δεδομένων είναι συχνά στενά συνδεδεμένα με ένα εμπορικό σύστημα διαχείρισης βάσεων δεδομένων, που συνήθως προσφέρεται από την ίδια την εταιρία. Οι ανοιχτού κώδικα σουίτες εξόρυξης δεδομένων, αντί να έρθουν με plug- ins που επιτρέπουν στο χρήστη να ζητήσουν τα δεδομένα από τυπικές βάσεις δεδομένων, έρχονται με την ενσωμάτωση αυτών που μπορεί να απαιτήσει περισσότερη προσπάθεια από ένα χρήστη.

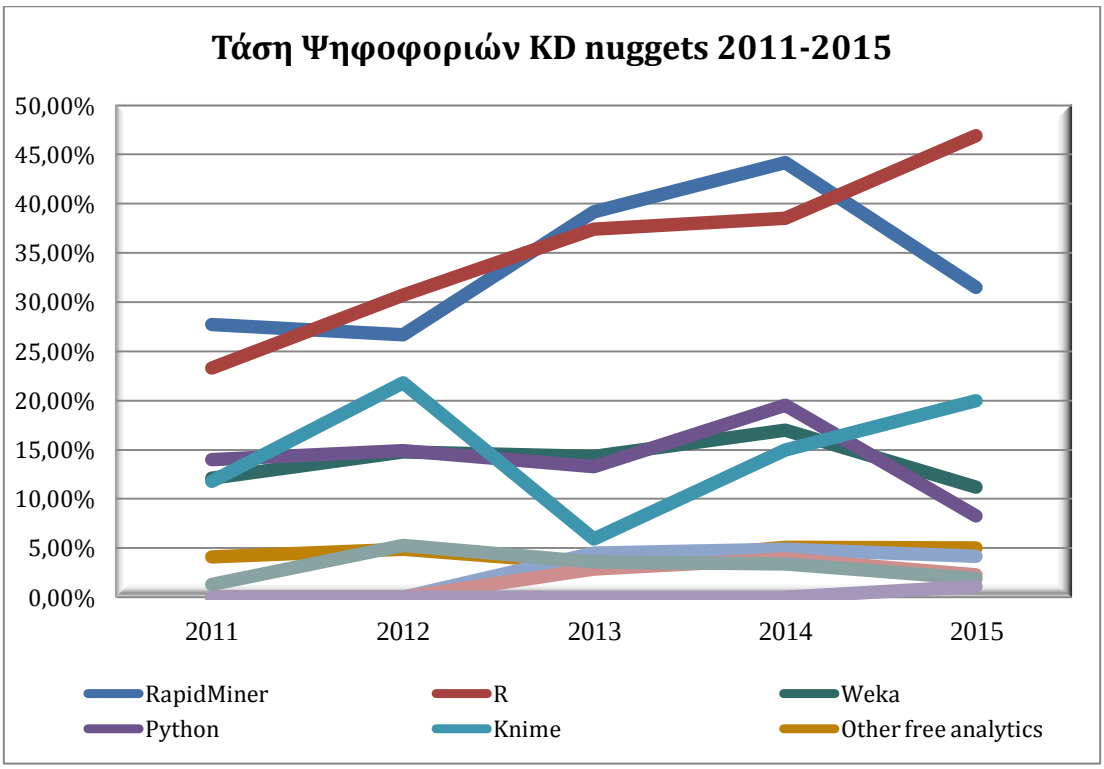
Εφόσον όπως αναλύθηκε παραπάνω, επιλέχθηκαν να συγκριθούν μόνο συστήματα ανοιχτού κώδικα, στη συνέχεια επιλέχθηκαν τα τέσσερα πιο δημοφιλή συστήματα σύμφωνα με τις ψηφοφορίες που διεξήχθησαν στο website KDnuggets [34] για τα έτη 2011 έως 2015 και πιο συγκεκριμένα εξετάστηκε η δημοσκόπηση που διεξήχθη τον Ιούνιο του 2014 και χαρακτηρίστηκε από μια μάχη μεταξύ των λογισμικών εξόρυξης δεδομένων.

Η εν λόγω δημοσκόπηση προσέλκυσε ρεκόρ συμμετοχής με 3.285 συμμετέχοντες. Αναφορικά με τη δημοσκόπηση του KDnuggets για το 2014 [34], παρατηρείται μια σταθερή ισορροπία μεταξύ των εμπορικών και των ελεύθερων λογισμικών, με περίπου το 71,00 % των συμμετεχόντων της δημοσκόπησης να επιλέγουν τα εμπορικά λογισμικά και το 78,00 % τα ελεύθερα λογισμικά. Ενώ, περίπου το 25,00 % των συμμετεχόντων χρησιμοποιεί εξολοκλήρου εμπορικά λογισμικά και το 30,00% εξολοκλήρου ελεύθερα λογισμικά, ενώ το 49,00 % χρησιμοποιεί και τα δύο παράλληλα. Τέλος, ο μέσος αριθμός των εργαλείων που χρησιμοποιήθηκαν ανά συμμετέχοντα ήταν 3,7, από 3 που ήταν το 2013. Αυτό είναι μια ένδειξη της αυξανόμενης ωριμότητας των εργαλείων, όμως συνεπάγεται μια αργή αύξηση της χρήσης του τομέα της εξόρυξης δεδομένων, και εξακολουθεί να είναι κατά κύριο λόγο ο τομέας που χρησιμοποιούν περισσότερο ομάδες αναλυτών σε κυβερνητικές υπηρεσίες, πολύ μεγάλες επιχειρήσεις κ.λ.π.. Κατά βάση η ανάλυση των δεδομένων εξακολουθεί να γίνεται σε μικρά δεδομένα.



Γράφημα 10. Ποσοστιαία κατανομή εφαρμογής των ελεύθερων/ανοιχτού κώδικα λογισμικών κατά την ανάλυση/ εξόρυξη δεδομένων για το έτος 2014. Πηγή: KD Nuggets Poll [34].

Ακολούθως, αναλύεται η τάση των ψηφοφοριών που διεξήχθησαν τα τελευταία πέντε έτη 2011-2015 μέσω του KDnuggets, για την χρήση των ελεύθερων/ανοιχτού κώδικα λογισμικών εξόρυξης δεδομένων. Όπως παρατηρείται από το παρακάτω γράφημα, το R έχει μια ανοδική πορεία και καταλήγει πρώτο στις προτιμήσεις των χρηστών για την εξόρυξη δεδομένων. Ενώ, το RapidMiner, το Weka και η Python απολαμβάνουν μια ανοδική πορεία κατά τα έτη 2011-2014, σε αντιδιαστολή παρατηρείται μείωση κατά 14, 5,8 και 11,2 ποσοστιαίες μονάδες αντίστοιχα για το κάθε σύστημα για το έτος 2015, με αποτέλεσμα συνολικά το RapidMiner να καταλήγει δεύτερο, το Weka τρίτο και η Python τέταρτη στις προτιμήσεις των χρηστών κατά τα έτη 2011-2015. Στην παρούσα μεταπτυχιακή διατριβή εξετάζονται μόνο τα τέσσερα πρώτα εργαλεία ανοιχτού κώδικα εξόρυξης δεδομένων, για τα έτη 2011-2015. Σε επόμενη προέκταση της παρούσας μεταπτυχιακής διατριβής θα μπορούσε να εξεταστεί το KNIME αλλά και το πρωτοεμφανιζόμενο Julia ως ένα σύστημα που σταθερά ανεβαίνει στις προτιμήσεις των χρηστών.



Γράφημα11. Τάση Ψηφοφοριών χρηστών συστημάτων εξόρυξης δεδομένων 2011-2015. Πηγή: [34].

Στη συνέχεια στον παρακάτω πίνακα συνοψίζονται τα γενικά χαρακτηριστικά των τεσσάρων επιλεγόμενων συστημάτων εξόρυξης δεδομένων, βάση του πλαισίου που αναλύθηκε παραπάνω. Ο πίνακας περιλαμβάνει χαρακτηριστικά όπως η εταιρία κάτω από την οποία διατίθεται το λογισμικό, οι άδειες χρήσης, η πηγαία γλώσσα προγραμματισμού, το λειτουργικό σύστημα που υποστηρίζεται, η διεπαφή χρήστη, ο τύπος των αρχείων που υποστηρίζονται κατά την εισαγωγή των δεδομένων, η πρόσβαση σε βάσεις δεδομένων και τέλος το εύρος της κοινότητας.

Χαρακτηριστικά	Rapid Miner	Weka	R	Python (scikit – learn)
Έκδοση	Community Edition, 5.3	3.6.2	3.2.2	3.4.3
Εταιρία / Εκπαιδευτικό Ίδρυμα	Rapid-I	University of Waikato	R Foundation	Python Software Foundation
Άδειες Χρήσης	a. GNU Affero GPL v 3.0	a. GNU GPL v3.0	a. GNU Affero GPL v3.0, b. Artistic License v2.0, c. GNU GPL v2.0 & v3.0, d. GNU Lesser GPL v 2.1 & v3.0 e. MIT License	a. GNU GPL v3.0 b. Python Software Foundation License v2.0
Πηγαία Γλώσσα Προγραμματισμού	Java	Java	R, C++, Fortran	Python
Λειτουργικό Σύστημα	a. Windows b. Mac OS X c. Linux	a. Windows b. Mac OS X c. Linux	a. Windows b. Mac OS X c. Linux	a. Windows b. Mac OS X c. Linux
Εισαγωγή Δεδομένων	a. TXT (Unicode or ASCII) αρχεία b. CSV αρχεία c. EXCEL αρχεία d. XML αρχεία e. ARFF αρχεία f. Βάσεις Δεδομένων	a. TXT (Unicode or ASCII) αρχεία b. CSV αρχεία c. ARFF αρχεία d. Βάσεις Δεδομένων e. URL	a. TXT (Unicode or ASCII) αρχεία b. CSV αρχεία c. EXCEL αρχεία d. XML αρχεία e. ARFF αρχεία f. Βάσεις Δεδομένων g. URL	a. TXT (Unicode or ASCII) αρχεία b. CSV αρχεία c. EXCEL αρχεία d. XML αρχεία e. ARFF αρχεία f. Βάσεις Δεδομένων

	g. URL			g. URL
Πρόσβαση σε Βάσεις Δεδομένων	a. ODBC b. JDBC c. MySQL d. PostgreSQL e. Sybase	a. ODBC b. JDBC c. MySQL d. PostgreSQL	a. ODBC b. JDBC c. MySQL d. PostreSQL	a. ODBC b. JDBC c. MySQL d. PostreSQL
Διεπαφή Χρήστη	Γραφική	Γραφική και Γραμμή Εντολών	Γραμμή Εντολών	Γραμμή Εντολών
Κοινότητα (εκτίμηση)	Μεγάλη (~ 250.000 χρήστες)	Μεγάλη (~ 200.000 χρήστες)	Πολύ μεγάλη (~2.000.000 χρήστες)	Μέτρια
Κόστος	Δωρεάν	Δωρεάν	Δωρεάν	Δωρεάν

Πίνακας 1.Γενικά Χαρακτηριστικά Συστημάτων.

Όπως παρατηρείται και στον Πίνακα η άδεια χρήσης που είναι κοινή και στα τέσσερα συστήματα είναι η GNU GPL [28, 32, 28, 39, 43], με το R να κατέχει τις περισσότερες άδειες χρήσης κάτω από το Ίδρυμα Ελεύθερου Λογισμικού και άλλα ιδρύματα. Το RapidMiner και το R κατέχουν την GNU Affero GPL, η οποία βασίζεται στη GNU GPL έχοντας έναν πρόσθετο όρο για να επιτρέπει στους χρήστες που αλληλεπιδρούν με το αδειοδοτημένο λογισμικό διαμέσου ενός δικτύου, να λαμβάνουν τον πηγαίο κώδικα για το εκάστοτε εργαλείο. Το R επιτρέπει επίσης, μέσω της GNU Lesser GPL, τη χρήση των βιβλιοθηκών και σε ιδιόκτητα προγράμματα. Επιπλέον, το R κατέχει την MIT license, όπου η άδεια αυτή χορηγείται σε οποιοδήποτε άτομο αποκτά αντίγραφο του λογισμικού και των σχετικών αρχείων τεκμηρίωσης, συμπεριλαμβανομένων χωρίς περιορισμό των δικαιωμάτων χρήσης, αντιγραφής, τροποποίησης, συγχώνευσης, δημοσίευσης, διανομής, παραχώρησης του εν λόγω λογισμικού. Αυτή η άδεια καθορίζει τους όρους υπό τους οποίους μπορεί να αντιγραφεί ένα συγκεκριμένο ελεύθερο πακέτο λογισμικού, η τροποποίηση, διανομή ή / και να αναδιανομή. Τέλος, το R κατέχει την Artistic License, κατά την οποία ο κάτοχος των πνευματικών δικαιωμάτων διατηρεί

κάποιο καλλιτεχνικό έλεγχο της ανάπτυξης του εν λόγω πακέτου, ενώ εξακολουθεί να διατηρεί το πακέτο διαθέσιμο ως ανοιχτού κώδικα δωρεάν λογισμικό.

Επιπροσθέτως, εξετάζοντας τις γλώσσες προγραμματισμού το RapidMiner και το Weka αναπτύσσονται σε Java η οποία είναι ευρέως διαδεδομένη στην κοινότητα των προγραμματιστών, ενώ το R σε R, C++ και Fortran, όπου η R είναι η βασική γλώσσα, η οποία είναι επεκτάσιμη, αλλά σχετικά δύσκολη για να τη μάθει ο χρήστης. Από την άλλη πλευρά, το scikit –learn πακέτο χρησιμοποιεί εξ ολοκλήρου την Python γλώσσα προγραμματισμού, όπου λόγω της διεπαφής γραμμής εντολών του συστήματος προϋποθέτει εξειδικευμένους προγραμματιστές στην Python. Έτσι λοιπόν, ένα από τα πιο σημαντικά χαρακτηριστικά ενός εργαλείου είναι η διεπαφή χρήστη, διότι διαδραματίζει σημαντικό ρόλο για την αξιολόγηση της ευχρηστίας του από τους εκάστοτε χρήστες. Το Rapidminer αναπτύσσεται κατά βάση σε γραφικό περιβάλλον, ενώ το Weka μπορεί να αναπτυχθεί είτε σε γραφικό περιβάλλον ή σε γραμμή εντολών, όπου εξαρτάται από την εμπειρία του χρήστη με αποτέλεσμα να προσδίδει ευελιξία στο εργαλείο. Τέλος, το R και το scikit –learn αναπτύσσονται εξ ολοκλήρου σε γραμμή εντολών με αποτέλεσμα να απευθύνονται σε γνώστες των αντιστοίχων γλωσσών προγραμματισμού.

Ακολούθως, παρατηρείται ότι όλα τα εργαλεία δύναται να εγκατασταθούν σε όλα τα λειτουργικά συστήματα [28, 32, 28, 39], έτσι παρέχεται υψηλός δείκτης ευκολίας εγκατάστασης των εργαλείων. Επίσης, τα εργαλεία R, scikit – learn και Rapid Miner υποστηρίζουν πολλούς τύπους αρχείων κατά την εισαγωγή των συνόλων των δεδομένων και με τη βοήθεια προσθήκης βιβλιοθηκών π.χ. για arff αρχεία, ενώ το Weka δεν είναι τόσο ευέλικτο και δεν υποστηρίζει αρχεία όπως EXCEL και XML. Παράλληλα, επιτυγχάνεται σε όλα τα εξεταζόμενα εργαλεία η πρόσβαση στις πιο βασικές βάσεις δεδομένων.

Όπως αναφέρθηκε παραπάνω, τα ανοιχτού κώδικα λογισμικά έχουν αναπτύξει ερευνητικές κοινότητες μέσω φόρουμ, blogs, mailing lists όπου μπορεί κάποιος να ζητήσει βοήθεια και να κοινωνικοποιηθεί με ανθρώπους που μοιράζονται το ίδιο ενδιαφέρον. Οι κοινότητες είναι ένα από τα πιο σημαντικά χαρακτηριστικά των λογισμικών ανοιχτού κώδικα, γιατί βοηθούν στην ανάπτυξη και τη βελτίωση τους.

Έτσι λοιπόν, αναφορικά με το μέγεθος των κοινοτήτων του κάθε συστήματος, το R κατέχει την υψηλότερη θέση με περίπου δύο εκατομμύρια χρήστες όπως αναφέρεται στο web site Revolution Analytics [35]. Υπάρχουν πολλές ομάδες ανά τον κόσμο (R user groups) οι οποίες έχουν ως σκοπό την αλληλεπίδραση μεταξύ των χρηστών του R συστήματος κάτω από τη χορηγία και επίβλεψη του Revolution Analytics. Επίσης, η εκτεταμένη κοινότητα των χρηστών φαίνεται και από τα αναρίθμητα θέματα που έχουν ανοιχτεί και επεξεργαστεί από χρήστες του συστήματος με σκοπό την υλοποίηση πειραμάτων και τη λύση προβλημάτων όπως στα web site github και stack overflow, με αριθμό θεμάτων που πραγματεύονται την περιοχή της εξόρυξης γνώσης, 8.305 και 102.989, αντίστοιχα. Επιπλέον, υπάρχει το φόρουμ R-bloggers με 583 εγγεγραμμένους χρήστες και τέλος η κοινότητα του reddit.gr στο Twitter: TextDataMiningReddit με 10.876 εγγεγραμμένους χρήστες.

Από την άλλη πλευρά, η κοινότητα της Python στο Twitter απαριθμεί 82.100 χρήστες, επίσης η Python έχει ομάδες (Python user groups) [40] ανά τον κόσμο σε 191 πόλεις, 37 χώρες με πάνω από 127.100 εγγεγραμμένα μέλη. Ενώ, το Ίδρυμα της Python ενθαρρύνει τους χρήστες να γίνουν μέλη της mailing list στο http://mail.python.org/mailman/listinfo/diversity. Οι χρήστες της γλώσσας προγραμματισμού Python συναντιούνται περιοδικά σε διάφορες πόλεις όπως καθορίζεται από το meetup.com κάτω από την ομάδα Python User Groups 1, με σκοπό όλων των επιπέδων να μπορούν να μοιράζονται τον τρόπο με τον οποίο χρησιμοποιούν τη γλώσσα και να μαθαίνουν νέες υλοποιήσεις. Το comp.lang.python είναι ένα ανοιχτό Usenet newsgroup για γενικές συζητήσεις και ερωτήσεις για τη Python και στη συνέχεια το φόρουμ http://www.python-forum.org απαριθμεί 5.588 εγγεγραμμένα μέλη. Παράλληλα, τα θέματα που πραγματεύονται την υλοποίηση πειραμάτων σε Python στο github και stackoverflow είναι 1.247 και 459.000, αντίστοιχα. Γενικά όμως, η Python είναι μια γλώσσα η οποία χρησιμοποιείται σε πολλά πεδία εφαρμογής, οπότε κατέχει πολύ μεγάλη κοινότητα χρηστών, όμως ένα μικρό ποσοστό αναφέρεται στο πεδίο εφαρμογής της εξόρυξης κειμένου και συγκεκριμένα του πακέτου scikit-learn.

Ακολούθως, η κοινότητα του Rapid miner απολαμβάνει μια ανοδική πορεία με περίπου 250.000 μέλη ανά τον κόσμο σύμφωνα με τα στατιστικά στοιχεία του Rapid-I. Πιο συγκεκριμένα , το RapidMiner Academia το οποίο ξεκίνησε να λειτουργεί στα τέλη του 2014, απαριθμεί ήδη 45.000 ενεργούς χρήστες, ενώ το rapidminer στο Twitter έχει 12.400 ακολούθους. Επίσης, υπάρχει το ηλεκτρονικό φόρουμ http://forum.rapid-

i.com/, όπου μπορούν να αλληλεπιδράσουν μεταξύ τους οι χρήστες του συστήματος. Σε αντιδιαστολή το Weka μέσω το Πανεπιστημίου του Pentaho έχει usergroups http://community.pentaho.com/user-groups/ και ηλεκτρονικό φόρουμ http://forums.pentaho.com/, όπου μπορούν οι χρήστες να αναρτήσουν ερωτήσεις, παρατηρήσεις και σφάλματα ώστε να λάβουν βοήθεια και απαντήσεις από άλλους χρήστες. Το ηλεκτρονικό φόρουμ του Weka έχει τα περισσότερα εγγεγραμμένα μέλη από τα άλλα τρία εργαλεία, που φτάνουν τον αριθμό των 58.289 μελών, ενώ τα θέματα που άπτονται στο Weka μέσω του stackoverflow αγγίζουν τα 4.657. Επιπλέον, υπάρχουν πολλές αναφορές σε youtube και websites όπου άπτονται υλοποιήσεις πειραμάτων και στα δύο λογισμικά.

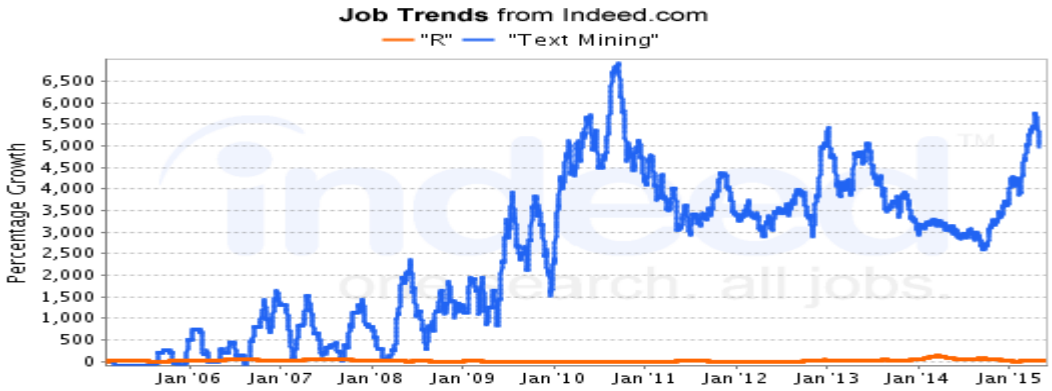
Επιπροσθέτως, συνδεδεμένη με την κοινότητα των χρηστών είναι η αγορά εργασίας που αναφέρεται στους αναλυτές δεδομένων των οποίων τα τεχνικά χαρακτηριστικά είναι η γνώση τουλάχιστον ενός εκ των τεσσάρων συστημάτων. Αυτό συμβαίνει διότι η αύξηση των δεδομένων και της πληροφορίας τα τελευταία χρόνια καθιστά επιτακτική ανάγκη των εταιριών να αντλούν γνώση από τα δεδομένα συνδυάζοντας την επιστήμη των Η/Υ, τη στατιστική ανάλυση και την πρόβλεψη των ενδεχόμενων ρίσκων και αποτελεσμάτων. Αυτό έχει ως αποτέλεσμα όλο και περισσότερες εταιρίες να προσλαμβάνουν αναλυτές δεδομένων με ραγδαίο ρυθμό, οπότε οι εν λόγω αγγελίες να έχουν αυξηθεί κατά πολύ από το 2011 και έπειτα, σύμφωνα με το κορυφαίο web site εύρεσης εργασίας Indeed.com [33].

Στα παρακάτω γραφήματα φαίνεται η τάση των θέσεων εργασίας που έχουν δημοσιευτεί κατά την περίοδο 2010-2015, και έχουν άμεση συνάφεια με την επιστήμη των δεδομένων, της εξόρυξης κειμένου και τη γνώση των τεσσάρων συστημάτων ξεχωριστά. Η πηγή είναι το Indeed.com όπου ανατρέχει εκατομμύρια δημοσιεύσεις αγγελιών από χιλιάδες ιστοσελίδες ευρέσεως εργασίας και απεικονίζει σε γραφήματα τα στοιχεία αναζήτησης που εισάγει ο χρήστης.

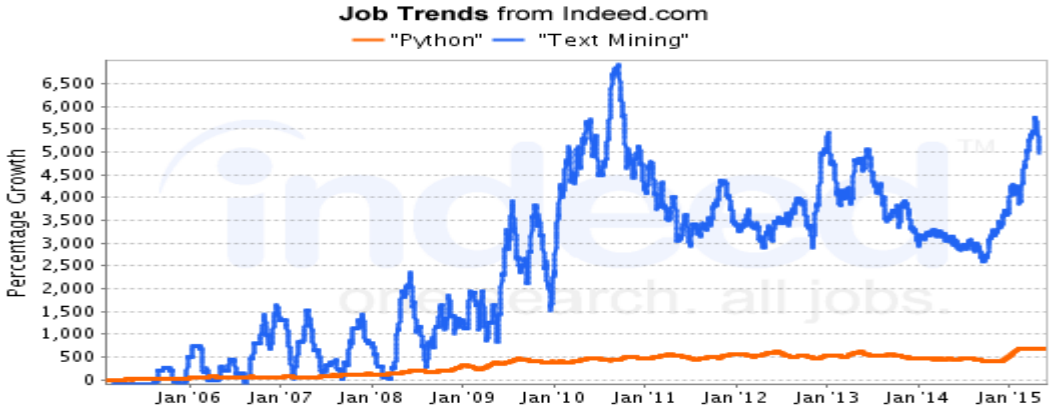
Γράφημα 12. Job trends –Data science. Πηγή: [33].

Στο παραπάνω γράφημα παρατηρείται η ποσοστιαία αύξηση των αγγελιών που ζητούν επιστήμονες με πεδίο την ανάλυση των δεδομένων. Η εν λόγω αύξηση από σχεδόν μηδενική που ήταν για τα έτη 2010-2012, αυξάνεται για τα έτη 2013 -2014 από 10% σε 40% ενώ κατά το τελευταίο εξάμηνο του 2014 η αύξηση ακουμπά το 85%.

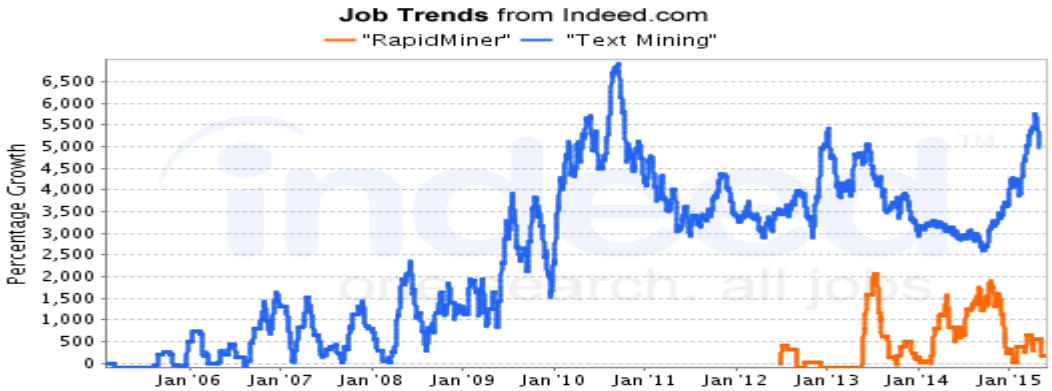
Στα παρακάτω γραφήματα, επιλέχθηκε η συνάφεια των θέσεων αναφορικά με την εξόρυξη κειμένου σε συνδυασμό με τη ζήτηση των τεσσάρων εργαλείων που εξετάζονται, όπως αναφέρονται στις αγγελίες, κατά τη διάρκεια των ετών 2006 – 2015. Γενικά, η εξόρυξη κειμένου κατέχει μια ανοδική πορεία κατά τη διάρκεια των ετών με μεγαλύτερη έξαρση να σημειώνεται τον Ιανουάριο του 2011 που ενδεχομένως να οφείλεται στη ραγδαία πλέον ανάπτυξη των κοινωνικών μέσων δικτύωσης όπως το twitter και της ανάγκης εξόρυξης της πολικότητας της κοινής γνώμης. Έπειτα ξεκινά μια καθοδική πορεία η οποία παρατηρείται να ανακάμπτει κατά τη διάρκεια του 2013 και τον Ιανουάριο του 2015, που ενδεχομένως οι διακυμάνσεις οφείλονται στο κύκλο ζωής της εργασίας στην ίδια εταιρία που κατά μέσο όρο είναι δύο με τρία έτη για κάθε εργαζόμενο ανά εταιρία. Η μεγάλες διακυμάνσεις της ζήτησης της εξόρυξης κειμένου που σημειώνονται ανά εξάμηνο ενδεχομένως οφείλονται στο γεγονός ότι γενικά η ζήτηση στελεχών από τις εταιρίες λαμβάνει χώρα πάντα στις αρχές του έτους ενώ το καλοκαίρι κατά βάση οι αγγελίες είναι ελάχιστες.



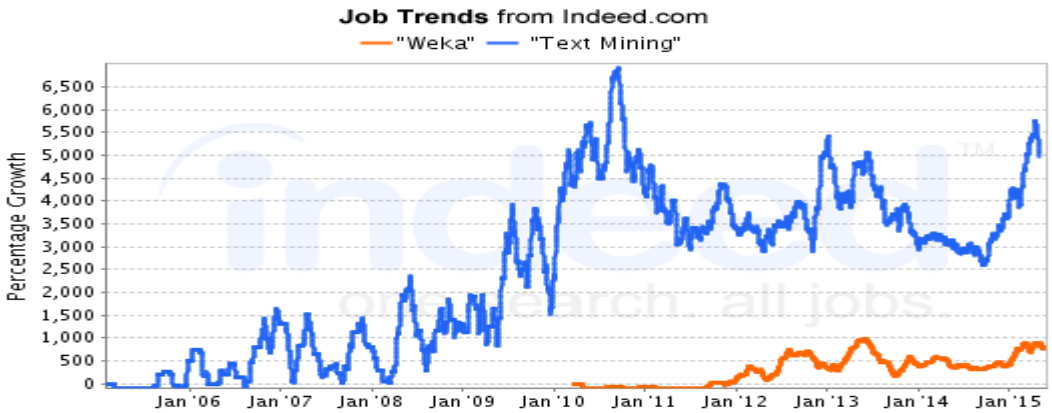
Γράφημα 13. Job trends R – Text Mining. Πηγή[33].



Γράφημα 14. Job trends Python – Text Mining. Πηγή: [33].



Γράφημα 15. Job trends RapidMiner– Text Mining. Πηγή: [33].



Γράφημα 16. Job trends Weka– Text Mining. Πηγή: [33].

Αναφορικά με το ποσοστό των αγγελιών που αναφέρουν τη γνώση της γλώσσας R στα τεχνικά χαρακτηριστικά των προφίλ των υποψηφίων με βασικό πεδίο εφαρμογής την εξόρυξη κειμένου, παρατηρείται ότι δεν υπάρχει καμία συνάφεια, που σημαίνει ότι ενώ η R είναι μια από τις γλώσσες που απολαμβάνει μεγάλη ζήτηση, χρησιμοποιείται για άλλα πεδία εφαρμογής πέραν της εξόρυξης κειμένου. Από την άλλη πλευρά, η γνώση της γλώσσας Python κατά τη διαδικασία εξόρυξης κειμένου έχει λίγο μεγαλύτερη ζήτηση από την R αλλά και πάλι φαίνεται ότι η Python χρησιμοποιείται για άλλα πεδία εφαρμογής. Βέβαια και οι δύο γλώσσες συναντώνται σε αγγελίες από το 2006 και έπειτα.

Ακολούθως, οι αγγελίες που ζητούν τη γνώση του Weka με βασικό πεδίο εφαρμογής την εξόρυξη κειμένου, ξεκινάνε από το 2010 έχοντας μια ανοδική πορεία ανά τα έτη με υψηλότερο σημείο κατά το 2013 και τον Ιανουάριο του 2015. Ενώ το Rapid Miner έχει τη μεγαλύτερη διακύμανση αλλά και ζήτηση σε σύγκριση με τα άλλα τρία εργαλεία, το οποίο ξεκινά να αναφέρεται στις αγγελίες από το 2012 και έπειτα. Το υψηλότερο σημείο βάση του γραφήματος παρουσιάζεται κατά τα μέσα του 2013 και τα τέλη του 2014 ενώ στις αρχές του 2014 και του 2015 παρουσιάζονται τα χαμηλότερα σημεία.

Τέλος σύμφωνα με την έρευνα του marketing dice για το 2014 [05], αναφορικά με τους μισθούς των επιστημόνων των Η/Υ, οι προγραμματιστές της R τείνουν να λαμβάνουν τους υψηλότερους μισθούς και ακολουθούν οι προγραμματιστές της Java και της Python, με ετήσιο εισόδημα 115.121 δολάρια, 102.889 δολάρια και 101.312 δολάρια αντίστοιχα. Έτσι λοιπόν, φαίνεται ότι η επένδυση στη εκμάθηση της γλώσσας R παρέχει αντίκρισμα.

3.2 Παρουσίαση του συστήματος Rapid Miner

Το Rapid Miner είναι μια πλατφόρμα λογισμικού που αναπτύχθηκε από την ομώνυμη εταιρεία, η οποία παρέχει ένα ολοκληρωμένο περιβάλλον για την μηχανική μάθηση, την εξόρυξη δεδομένων και κειμένων, την προβλεπτική ανάλυση και την επιχειρησιακή ανάλυση [39]. Χρησιμοποιείται σε επιχειρήσεις και βιομηχανικές εφαρμογές, καθώς και για την έρευνα, την εκπαίδευση, την κατάρτιση, την ταχεία

προτυποποίηση και την ανάπτυξη εφαρμογών, και υποστηρίζει όλα τα στάδια της διαδικασίας εξόρυξης δεδομένων, συμπεριλαμβανομένων των αποτελεσμάτων απεικόνισης, της επικύρωσης και της βελτιστοποίησης.

Το Rapid Miner ξεκίνησε το 2001 από τους Ralf Klinkenberg, Ingo Mierswa και Simon Fischer στο Τμήμα Τεχνητής Νοημοσύνης του Πανεπιστήμιου του Dortmund. Το 2006 οι Ralf Klinkenberg και Ingo Mierswa ίδρυσαν την εταιρία Rapi-I, που σήμερα είναι ο κύριος συνεισφέρων στη περεταίρω ανάπτυξη του Rapid Miner μαζί με άλλους πενήντα προγραμματιστές παγκοσμίως.

Το Rapid Miner παρέχει το 99% της προηγμένης αναλυτικής λύσης μέσω προτύπων βασισμένων στην επιτάχυνση της παράδοσης αποτελεσμάτων και τη μείωση των σφαλμάτων σχεδόν εξαλείφοντας την ανάγκη να γραφτεί κώδικας. Το Rapid Miner παρέχει μια πλατφόρμα για τους προγραμματιστές ώστε να δημιουργήσουν αλγόριθμους ανάλυσης των δεδομένων, και να τους δημοσιεύουν σε μια ευρύτερη κοινότητα. Επίσης, διανέμεται υπό την άδεια ανοικτού κώδικα AGPL και έχει φιλοξενηθεί από το SourceForge.

Τέλος, το Rapid Miner είναι μια πλήρης πλατφόρμα επιχειρησιακής ανάλυσης, με ιδιαίτερη έμφαση στην εξόρυξη δεδομένων, κειμένων και στην προβλεπτική ανάλυση. Χρησιμοποιεί μια μεγάλη ποικιλία περιγραφικών και προβλεπτικών τεχνικών για να δώσει στους χρήστες τη διορατικότητα ώστε να πάρουν κερδοφόρες αποφάσεις.

3.2.1 Το περιβάλλον του συστήματος Rapid Miner

Το Rapid Miner είναι ένα περιβάλλον που ευνοεί την μηχανική μάθηση και την εξόρυξη δεδομένων. Πιο συγκεκριμένα, μια γενική ιδέα ενός modular τελεστή που επιτρέπει το σχεδιασμό εμφωλιασμένων αλυσίδων τελεστών, για ένα μεγάλο αριθμό προβλημάτων μηχανικής μάθησης. Το Rapid Miner εισάγει νέες ιδέες για τον χειρισμό των δεδομένων και την μοντελοποίηση των διαδικασιών για τους τελικούς χρήστες. Επιπρόσθετα, οι σαφείς διεπιφάνειες και η εν μέρει γλώσσα συγγραφής σεναρίων βασισμένων στη xml, κάνει την πλατφόρμα Rapid Miner ένα ενοποιημένο περιβάλλον ανάπτυξης για τη εξόρυξη δεδομένων και τη μηχανική μάθηση. Πιο συγκεκριμένα, η πλατφόρμα του Rapid Miner είναι [39]:

- Γραμμένη σε Java,

- Περιλαμβάνει μια εσωτερική xml αναπαράσταση ώστε να εξασφαλίζεται η τυποποιημένη μορφή ανταλλαγής εξόρυξης δεδομένων σε διάφορα πειράματα,

- Εξασφαλίζεται η αποτελεσματική διαχείριση των δεδομένων αφού υπάρχει δυνατότητα προβολής αυτών σε πολλά επίπεδα,

- GUI γραμμή εντολών mode (λειτουργία batch) και Java API για τη χρήση του από άλλα προγράμματα,

- Περιέχει ποικίλα επιπρόσθετα plug-ins,

- Μια μεγάλη σειρά αναπαράστασης των δεδομένων με λεπτομερή διάσταση.

Το Rapid Miner μπορεί να διαβάσει ένα μεγάλο αριθμό αρχείων εισόδου, να διαβάσει και να γράψει μοντέλα, σύνολα παραμέτρων και σύνολα χαρακτηριστικών. Γενικότερα, μπορεί να διαβάσει όλα τα αρχεία που παράγει.

Το Rapid Miner είναι ένα περιβάλλον εργασίας στο οποίο δημιουργούνται και υλοποιούνται οι διαδικασίες επεξεργασίας δεδομένων, αφού πρώτα υπάρχει η κατάλληλη σύνδεση με τη Βάση και το Αποθετήριο Δεδομένων "repository" και κυρίως αφού έχουμε επιλέξει και κατάλληλα συνδέσει τους απαραίτητους τελεστές.

3.2.2 Βασική λειτουργία του συστήματος Rapid Miner

Το RapidMiner παρέχει διαδικασίες μάθησης εξόρυξης δεδομένων όπως οι εξής [39]:

1. Φόρτωση δεδομένων και μετατροπής αυτών.

2. Προ-επεξεργασία των δεδομένων.

3. Απεικόνιση των δεδομένων.

4. Προβλεπτική και στατιστική ανάλυση.

5. Αξιολόγηση και ανάπτυξη των δεδομένων.

Το Rapid Miner είναι γραμμένο στη γλώσσα προγραμματισμού Java, ενώ παρέχει ένα γραφικό περιβάλλον ώστε να σχεδιαστούν και να εκτελεστούν οι αναλυτικές ροές εργασίας. Κάθε τελεστής εκτελεί μία εργασία στο πλαίσιο της διαδικασίας του και η έξοδος του κάθε τελεστή αποτελεί την είσοδο του επόμενου, ενώ επιμέρους λειτουργίες μπορεί να κληθούν από τη γραμμή εντολών. Η λειτουργικότητα του Rapid Miner μπορεί να επεκταθεί με επιπλέον plug-ins.

i. **Τελεστές του RapidMiner**

Το RapidMiner παρέχει πάνω από 400 τελεστές συμπεριλαμβανομένων των παρακάτω [04, 53]:

- Είσοδος/Έξοδος: Περιλαμβάνονται ευέλικτοι τελεστές για την είσοδο και την έξοδο των δεδομένων, επίσης παρέχεται τεχνική υποστήριξη αρκετών τύπων αρχείων, συμπεριλαμβανομένων των arff, csv, bibtex, dbase, αλλά και απευθείας ανάγνωση δεδομένων από βάσεις δεδομένων.

- Τελεστές προ-επεξεργασίας δεδομένων: Παρέχονται τελεστές με λειτουργίες όπως: αναπλήρωση χαμένων και άπειρων τιμών, αφαίρεση άχρηστων ιδιοτήτων, δειγματοληψία, μείωση πολυδιαστατικότητας και άλλα.

- Αλγόριθμοι μηχανικής μάθησης: Παρέχεται ένας μεγάλος αριθμός αλγορίθμων μηχανικής μάθησης για έργα παλινδρόμησης και κατηγοριοποίησης. Επίσης, περιλαμβάνονται αρκετοί αλγόριθμοι για εξόρυξη κανόνων συσχέτισης και συσταδοποίησης.

- Τελεστές του WEKA: Παρέχονται όλες οι λειτουργίες του WEKA.

- Τελεστές χαρακτηριστικών: Περιλαμβάνονται αλγόριθμοι επιλογής, γενετικοί αλγόριθμοι, τελεστές για εξαγωγή χαρακτηριστικών από χρονικές σειρές, στάθμιση και συνάφεια χαρακτηριστικών, και παραγωγή νέων.

- Μετά-τελεστές: Περιλαμβάνονται αλγόριθμοι βελτιστοποίησης για σχεδιασμό διαδικασιών.Οπτικοποίηση: Παρέχονται τελεστές καταγραφής και

παρουσίασης αποτελεσμάτων. On line δημιουργία 2D και 3D γραφημάτων των δεδομένων, των μοντέλων μηχανικής και άλλων διαδικαστικών αποτελεσμάτων.

- Αξιολόγηση απόδοσης: Περιλαμβάνονται τεχνικές αξιολόγησης, με κριτήρια απόδοσης κατηγοριοποίησης και παλινδρόμησης.

Ένας τελεστής δέχεται ως είσοδο ένα σύνολο αντικειμένων και παράγει ως έξοδο κάποια αντικείμενα. Τα αντικείμενα μπορούν αν είναι πηγαία αρχεία δεδομένων ή ενδιάμεσα αποτελέσματα, ενώ οι τελεστές μπορούν να συνδυάζονται δημιουργώντας πιο σύνθετους τελεστές, μοντέλα, κριτήρια απόδοσης και άλλα. Όλα τα παραπάνω οπτικοποιούνται στην προοπτική σχεδίασης του εργαλείου.

ii. **Διαδικασίες**

Οι διαδικασίες στο Rapid Miner αποτελούνται από ένα σύνολο εμφωλιασμένων τελεστών. Ένας τελεστής δέχεται ένα σύνολο αντικειμένων εισόδου και παράγει κάποια αντικείμενα εξόδου. Αυτά τα αντικείμενα μπορούν να είναι αρχεία δεδομένων, μοντέλα, κριτήρια απόδοσης και άλλα. Κάποιοι τελεστές μπορούν να έχουν και εσωτερικούς τελεστές, για παράδειγμα, χωρίζεται ένα παράδειγμα σε δεδομένα εκπαίδευσης και δοκιμών και εφαρμόζονται οι εσωτερικοί τελεστές, οι οποίο είναι ένας τελεστής μηχανικής μάθησης και ένας τελεστής εφαρμογής. Κάθε φορά χρησιμοποιούνται ασυνεχή δεδομένα δοκιμών.

Πριν εκτελεστεί μια διαδικασία, θα πρέπει πρώτα να επικυρωθεί. Η επικύρωση διαδικασιών είναι πολύ σημαντική διαδικασία για τη δημιουργία σωστών ορισμών διαδικασιών και μπορεί να βοηθήσει στην κατανόηση του Rapid Miner. Οπότε, είναι καλό να κάνουμε χρήση της εγκυροποίησης των διαδικασιών όσο πιο συχνά γίνεται, τουλάχιστον μια φορά πριν το άνοιγμα μιας διαδικασίας. Μαζί με τα σημεία παύσης από τα μενού των τελεστών, ο σχεδιασμός νέων ή περίπλοκων διαδικασιών γίνεται πιο εύκολος.

Η εκτέλεση μιας διαδικασίας είναι αρκετά εύκολη, απλά επιλέγοντας Εκτέλεση Run από το μενού Process ή το αντίστοιχο κουμπί αναπαραγωγής. Μπορεί ο χρήστης να ακολουθήσει την πρόοδο της διαδικασίας του παρατηρώντας την έξοδο, η οποία

εμφανίζεται στο Message Viewer. Σημειώστε ότι σε λειτουργία GUI, η έξοδος δεν χρειάζεται να γραφτεί σε ένα αρχείο καταγραφής "log file". Εάν δεν καθοριστεί ένα αρχείο καταγραφής "log file", μπορούν να αποθηκεύονται πάντα τα περιεχόμενα των μηνυμάτων του θεατή σε ένα αρχείο επιλέγοντας το αντίστοιχο στοιχείο του μενού, στο μενού του Message Viewer.

Όταν η διαδικασία φτάσει στο τέλος, τα αποτελέσματα εμφανίζονται αυτόματα. Αυτό μπορεί να γίνει με στατιστική απόδοση, δέντρο απόφασης ή οτιδήποτε άλλο. Το Rapid Miner επιλέγει αυτόματα τη λειτουργία αποτελεσμάτων Results Mode. Αν κάποιος θέλει να παρατηρήσει μια διαδικασία πιο προσεκτικά, μπορεί να τοποθετήσει σημεία παύσης πριν και μετά τον τελεστή, μέσω του μενού των τελεστών. Σε αυτή την περίπτωση, κάθε φορά που ο έλεγχος φτάνει σε σημείο παύσης, παρουσιάζονται ενδιάμεσα αποτελέσματα, κατά όμοιο τρόπο με το μήνυμα στο τέλος της διαδικασίας. Μπορεί κανείς να μελετήσει το διάγραμμα χρήσης της μνήμης και τη μπάρα προόδου.

3.2.3 Τύποι τιμών του συστήματος Rapid Miner

Υπάρχουν διαφορετικοί τύποι τιμών για τα διάφορα χαρακτηριστικά γνωρίσματα. Αναφερόμαστε σε τιμή τύπου κειμένου στην περίπτωση ελευθέρου κειμένου, σε τιμή αριθμητικού τύπου στην περίπτωση αριθμών και σε τιμές ονομαστικού τύπου στην περίπτωση που είναι πιθανές πολύ λίγες τιμές.

Παρακάτω αναφέρονται συνολικά όλοι οι τύποι τιμών που υποστηρίζονται από το Rapid Miner [04, 53]:

- Κατηγοριακός (Nominal): Μη-αριθμητικές τιμές που χρησιμοποιούνται συνήθως για πεπερασμένες ποσότητες διαφορετικών χαρακτηριστικών.

- Αριθμητικός (Numeric): Τιμές αριθμητικού τύπου.

- Ακέραιος (Integer): Θετικοί και αρνητικοί ακέραιοι αριθμοί.

- Πραγματικός (Real): Θετικοί και αρνητικοί πραγματικοί αριθμοί.

- Κείμενο (Text): Τυχαίο κείμενο χωρίς δομή.

- Διωνυμικός (Binominal): Ειδική περίπτωση του ονομαστικού τύπου, όπου επιτρέπονται μόνο δύο τιμές.

- Πολυωνυμικός (Polynominal): Ειδική περίπτωση του ονομαστικού τύπου, όπου επιτρέπονται περισσότερες από δύο τιμές.

- Ημερομηνία/Χρόνος (Date_Time): Ημερομηνία και χρόνος.

- Ημερομηνία (Date): Ημερομηνία.

- Χρόνος (Time): Χρόνος.

3.2.4 Το αποθετήριο αποτελεσμάτων του συστήματος Rapid Miner

Το αποθετήριο των δεδομένων του Rapid Miner, αναλαμβάνει τη διαχείριση όλων των δεδομένων και των διαδικασιών. Επίσης, μπορεί να βρίσκεται είτε σε τοπικό είτε σε κοινό σύστημα αρχείων είτε ακόμη και στον εξωτερικό διακομιστή ανάλυσης του Rapid Miner. Παρόλο που τα δεδομένα μπορούν να εισαχθούν στις διαδικασίες και εκτός του αποθετηρίου, η χρήση του αποθετηρίου προσφέρει κάποια πολύ σημαντικά πλεονεκτήματα:

- Όλα τα δεδομένα εισόδου/εξόδου αλλά και τα ενδιάμεσα αποτελέσματα υπομνημονίζονται με μετά – πληροφορίες.

- Ο χρήστης μπορεί να έχει μια επισκόπηση των αποθηκευμένων δεδομένων, των χαρακτηριστικών και υποσημειώσεων τους οποιαδήποτε στιγμή, χωρίς να πρέπει να ανοίξει ξεχωριστά το αρχείο.

- Τα δεδομένα, τα αποτελέσματα και οι αναφορές αποθηκεύονται σε τοποθεσίες που υποδεικνύονται ως σχετικές μεταξύ τους, παρέχοντας εύκολη πρόσβαση στο χρήστη.

3.2.5 Γραφική Απεικόνιση στο σύστημα Rapid Miner

Κάθε αποτέλεσμα προβάλλεται μέσα από τη δική του κάρτα αρχείου. Για παράδειγμα, για το σύνολο δεδομένων υπάρχουν τρεις επιλογές εμφάνισης, η εμφάνιση των μεταδεδομένων και στατιστικών, η εμφάνιση των ίδιων των δεδομένων και η εμφάνιση διαφορετικών οπτικοποιήσεων των δεδομένων.

a. Κείμενο: Η πιο βασική μορφή οπτικοποίησης είναι αυτή σε μορφή κειμένου. Κάποια μοντέλα, καθώς και πολλά άλλα αποτελέσματα μπορούν να προβληθούν σε μορφή κειμένου. Αυτό γίνεται με την αποκαλούμενη Data View (καρτέλα του RapidMiner στην εμφάνιση αποτελεσμάτων).

b. Πίνακας: Το Rapid Miner έχει ως πρωταρχικό στόχο την ανάλυση δεδομένων σε μορφή πινάκων. Παρόλα αυτά, οι πίνακες δε χρησιμοποιούνται μόνο για την εμφάνιση συνόλων δεδομένων αλλά και για την εμφάνιση μετα-δεδομένων, παραγόντων που επηρεάζουν τη στάθμιση μητρών όπως οι συσχετίσεις μεταξύ γνωρισμάτων και πολλών άλλων. Αυτές οι μορφές εμφανίσεων έχουν συνήθως τον όρο Table στο όνομα τους.

c. Διαγράμματα: Τα διαγράμματα είναι ένα από τα δυνατότερα χαρακτηριστικά του Rapid Miner, τα οποία βρίσκονται στην καρτέλα PlotView. Υπάρχουν δύο λειτουργίες 3D διαγραμμάτων ενσωματωμένες στο Rapid Miner:

    1. Παράγει 3D έγχρωμα διαγράμματα, με δυνατότητα περιστροφής με χρήση του ποντικιού.

    2. Παράγει 2D έγχρωμα διαγράμματα. Οι πρώτες δύο διαστάσεις κατασκευάζουν ένα 2D στρώμα και η τρίτη διάσταση απεικονίζεται σε διαφορετικά χρώματα ή μεγέθη.

    3. Υπάρχουν επίσης και συστατικά για σχεδιασμό διαγραμμάτων διασποράς και ιστογραμμάτων.

d. Γραφήματα: Τα γραφήματα είναι μια ακόμη μορφή εμφάνισης που συναντάται αρκετά συχνά στο RapidMiner. Με τον όρο γραφήματα εννοούμε κυρίως όλες τις οπτικοποιήσεις που απεικονίζουν κόμβους και τις μεταξύ τους

σχέσεις. Μπορεί να είναι κόμβοι εντός μιας ιεραρχικής συσταδοποίησης ή κόμβοι ενός δέντρου απόφασης.

3.3 Παρουσίαση του συστήματος Weka

Το Weka (Waikato Environment for Knowledge Analysis) είναι μια δημοφιλής σουίτα λογισμικού μηχανικής μάθησης γραμμένη σε Java, που αναπτύχθηκε στο Πανεπιστήμιο του Waikato της Νέας Ζηλανδίας. Το Weka είναι ελεύθερο λογισμικό υπό την άδεια GNU General Public License [32].

Πιο συγκεκριμένα παρέχει τη δυνατότητα για:

- Προ-επεξεργασία δεδομένων (τα εργαλεία για την προ-επεξεργασία στο περιβάλλον του WEKA ονομάζονται filtrers).

- Δημιουργία μοντέλων από τα δεδομένα με κάποια διαδικασία εκπαίδευσης.

- Χρησιμοποίηση στατιστικών μεγεθών για την αξιολόγηση των διαφόρων αλγορίθμων μάθησης.

- Απεικόνιση τόσο των αρχικών δεδομένων όσο και των αποτελεσμάτων μετά τη διαδικασία εκπαίδευσης.

- Το λογισμικό είναι γραμμένο εξ ολοκλήρου σε Java για να διευκολύνει τη διαθεσιμότητα των εργαλείων εξόρυξης δεδομένων, ανεξαρτήτως του χρησιμοποιούμενου συστήματος.

Η βιβλιοθήκη αλγορίθμων Weka αναπτύχθηκε το 1999 με σκοπό να καλύψει τις πειραματικές ανάγκες του εργαστηρίου Μηχανικής Μάθησης του Πανεπιστημίου του Waikato στη Νέα Ζηλανδία. Σήμερα, είναι ένα δημοφιλές εργαλείο μηχανικής μάθησης και χρησιμοποιείται ευρέως στην ερεύνα, στην εκπαίδευση αλλά και σε εφαρμογές. Τα χαρακτηριστικά που το καθιέρωσαν ως εργαλείο είναι ο μεγάλος αριθμός των αλγορίθμων που υλοποιεί και η ευχρηστία του γραφικού του περιβάλλοντος (GUI).

3.3.1 Το περιβάλλον του συστήματος Weka

Ο επιλογέας περιβάλλοντος του Weka παρέχει ένα σημείο έναρξης για το άνοιγμα των κύριων διεπιφανειών και υποστηριζόμενων εργαλείων του.

Εικόνα 5. Επιλογέας περιβάλλοντος WEKA.

Το Weka έχει πολλαπλές διεπιφάνειες, που περιλαμβάνουν τα παρακάτω [32]:

- Exporer: Ο Explorer είναι η κύρια διεπιφάνεια χρήστη του Weka. Η διεπιφάνεια του Explorer παρέχει αρκετές καρτέλες, παρέχοντας έτσι πρόσβαση στα βασικά συστατικά του Weka, τα οποία αναλύονται παρακάτω εκτενώς.

- Experimenter: Ένα περιβάλλον για τη διεξαγωγή πειραμάτων και στατιστικών δοκιμών σε τεχνικές εκμάθησης. Το experimenter παρέχει τη δυνατότητα στο χρήστη να δημιουργήσει, εκτελέσει, τροποποιήσει και να αναλύσει πειράματα με μια πιο απλή προσέγγιση από τη μεμονωμένη επεξεργασία σχημάτων, και να εφαρμόζει διάφορα μαθησιακά σχήματα σε πολλά διαφορετικά σύνολα δεδομένων και συχνά με διαφορετικές παραμέτρους. Το περιβάλλον πειραμάτων μπορεί να εκτελεστεί από τη γραμμή εντολών του WEKA, ενώ οι εντολές μπορούν να τυπωθούν απευθείας από τη γραμμή εντολών. Ενώ το KnowledgeFlow ξεπερνά τους περιορισμένους σχετικά με το μέγεθος του αρχείου που μπορεί να επεξεργαστεί ο Experimenter ξεπερνά τους χρονικούς περιορισμούς. Επιπλέον, περιέχει υποδομές για προχωρημένους χρήστες, ώστε να διαμοιράσουν το

υπολογιστικό φορτίο που απαιτείται από μεγάλου εύρους πειράματα, σε διάφορους υπολογιστές.

- KnowledgeFlow: Αυτό το περιβάλλον υποστηρίζει τις ίδιες λειτουργίες με αυτό του Explorer, αλλά με χρήση διεπιφάνειας drag-and-drop. Ένα πλεονέκτημα είναι ότι υποστηρίζει και αυξητική μάθηση. Η διεπιφάνεια KnowledgeFlow είναι εμπνευσμένη από τη ροή δεδομένων. Ο χρήστης μπορεί να επιλέξει συστατικά του WEKA από μια εργαλειοθήκη, να τα τοποθετήσει σε έναν καμβά διάταξης και να τα συνδέσει μεταξύ τους ώστε να σχηματίσει μια ροή γνώσης για την επεξεργασία και ανάλυση δεδομένων. Επίσης, μπορούν να εξεταστεί όλη η διαδικασία λεπτομερώς και όχι μόνο το αποτέλεσμα που προκύπτει από αυτή.

  Το περιβάλλον KnowledgeFlow διαχειρίζεται δεδομένα είτε επαυξητικά είτε σε δέσμες, ενώ ο Explorer διαχειρίζεται μόνο δεδομένα δέσμης. Μια τέτοια διάταξη μπορεί επομένως να επεξεργαστεί αρχεία οποιουδήποτε μεγέθους, ακόμα και μεγαλύτερου της κύριας μνήμης του συστήματος, καθώς δεν χρειάζεται να τα αποθηκεύσει εσωτερικά για να ξεκινήσει τη διαδικασία.

- SimpleCLI: Παροχή μιας απλής διεπιφάνειας γραμμής εντολών που επιτρέπει την άμεση εκτέλεση των εντολών του WEKA για λειτουργικά συστήματα που δεν έχουν μια τέτοια διεπιφάνεια. Η απλή γραμμή εντολών παρέχει πλήρη πρόσβαση σε όλες τις κλάσεις του WEKA (ταξινομητές, φίλτρα, συσταδοποιητές και άλλα.).Πρόκειται για το πιο απλό και χωρίς γραφικά βοηθήματα περιβάλλον εργασίας και απευθύνεται σε χρήστες που γνωρίζουν εις βάθος το WEKA και τις εντολές του.

3.3.2 Βασική λειτουργία του συστήματος Weka

Η πλατφόρμα Weka εμπεριέχει μια συλλογή από εργαλεία οπτικοποίησης και αλγορίθμων για την ανάλυση δεδομένων και την προγνωστική μοντελοποίηση, μαζί με γραφικά περιβάλλοντα χρήστη για εύκολη πρόσβαση σε αυτές τις λειτουργίες [23, 32].

Το Weka υποστηρίζει διάφορες πρότυπες εργασίες εξόρυξης δεδομένων, και πιο συγκεκριμένα, την προεπεξεργασία δεδομένων, την συσταδοποίηση, τη Κατηγοριοποίηση, την οπισθοδρόμηση, την οπτικοποίηση και την επιλογή χαρακτηριστικών. Όλες οι τεχνικές του Weka στηρίζονται στην υπόθεση ότι τα δεδομένα είναι διαθέσιμα ως ένα ενιαίο επίπεδο αρχείο ή σχέση, όπου κάθε σημείο δεδομένων περιγράφεται από ένα σταθερό αριθμό από χαρακτηριστικά (συνήθως, αριθμητικό ή ονομαστικό χαρακτηριστικό και άλλα.). Το Weka παρέχει πρόσβαση σε βάσεις δεδομένων SQL χρησιμοποιώντας Java Database Connectivity και μπορεί να επεξεργαστεί το αποτέλεσμα που επιστρέφεται από ένα ερώτημα βάσης δεδομένων. Δεν είναι σε θέση να διαχειριστεί την εξόρυξη δεδομένων πολλαπλών σχέσεων, αλλά υπάρχει ξεχωριστό λογισμικό για τη μετατροπή μιας συλλογής συνδεδεμένων πινάκων της βάσης δεδομένων σε ένα ενιαίο πίνακα που είναι κατάλληλο για επεξεργασία χρησιμοποιώντας το Weka. Ένας άλλος σημαντικός τομέας που αυτή τη στιγμή δεν καλύπτεται από τους αλγορίθμους που περιλαμβάνονται στο Weka είναι τα μοντέλα ακολουθίας.

Πιο αναλυτικά, μια σειρά από τεχνικές μηχανικής μάθησης παρουσιάζονται στο χρήστη με τέτοιο τρόπο ώστε να κρύψει τις ιδιαιτερότητες των μορφών εισόδου και εξόδου, καθώς και να επιτρέψει μια διερευνητική προσέγγιση στην εφαρμογή της τεχνολογίας. Κάθε εφαρμογή αλγορίθμου μηχανικής μάθησης απαιτεί τα στοιχεία να βρίσκονται στη δική του μορφή, και έχει το δικό του τρόπο προσδιορισμού παραμέτρων. Το σύστημα Weka έχει σχεδιαστεί για να φέρει μια σειρά από τεχνικές μηχανικής μάθησης ή καθεστώτα ενισχύσεων δυνάμει μιας κοινής διεπαφής.

Το Weka Explorer είναι ένα εύκολο στη χρήση γραφικό περιβάλλον χρήστη που αξιοποιεί τη δυνατότητα του λογισμικού Weka. Η διεπαφή Explorer διαθέτει τα παρακάτω πακέτα-πλαίσια που παρέχουν την πρόσβαση στα κύρια συστατικά της επιφάνειας εργασίας του :

Pre-process Panel: Το πακέτο «προ-επεξεργασία» είναι το σημείο εκκίνησης για την εξερεύνηση της γνώσης. Από αυτόν το πακέτο μπορούν να φορτωθούν σύνολα δεδομένων, να αναζητηθούν τα χαρακτηριστικά γνωρίσματα και να εφαρμοστούν οποιοσδήποτε συνδυασμός φίλτρων χωρίς επίβλεψη στα δεδομένα. Το πακέτο προ-επεξεργασία διαθέτει εγκαταστάσεις για την εισαγωγή δεδομένων από μια βάση

δεδομένων, για παράδειγμα ένα αρχείο CSV, και λοιπά., καθώς και για την προ-επεξεργασία αυτών των δεδομένων χρησιμοποιώντας τους λεγόμενους αλγόριθμους φιλτραρίσματος. Τα φίλτρα αυτά μπορούν να χρησιμοποιηθούν για τη μετατροπή των δεδομένων (για παράδειγμα, στρέφοντας αριθμητικά χαρακτηριστικά σε διακριτά) και να καταστήσει δυνατή να διαγράφουν περιπτώσεις και χαρακτηριστικά σύμφωνα με συγκεκριμένα κριτήρια που θα οριστούν από τον χρήστη [23].

- Classify Panel: Το πακέτο Κατηγοριοποίησης επιτρέπει να ρυθμιστούν και να εκτελεστούν οποιοιδήποτε από τις ταξινομητές του Weka για την τρέχουσα δέσμη. Το πακέτο Κατηγοριοποίησης επιτρέπει στο χρήστη να εφαρμόσει αλγορίθμους Κατηγοριοποίησης και παλινδρόμησης στο προκύπτον σύνολο δεδομένων, για να εκτιμηθεί η ακρίβεια του προκύπτοντος μοντέλου πρόβλεψης, και να απεικονίσει εσφαλμένες προβλέψεις, ROC καμπύλες, και λοιπά., ή το μοντέλο το ίδιο (αν το μοντέλο μπορεί να αποτελέσει αντικείμενο οπτικοποίησης, όπως ένα δέντρο απόφασης).

- Cluster Panel: Το πακέτο συσταδοποίησης δίνει πρόσβαση στις τεχνικές συσταδοποίησης στο Weka, για παράδειγμα το απλό αλγόριθμο k -means. Υπάρχει επίσης μια υλοποίηση του αλγορίθμου μεγιστοποίησης προσδοκίας για την εκμάθηση ενός μίγματος κανονικών κατανομών. Από το πακέτο συσταδοποίησης, μπορούν να ρυθμιστούν και να εκτελεστούν οποιοιδήποτε από τους αλγορίθμους συσταδοποίησης για την τρέχουσα δέσμη. Οι αλγόριθμοι μπορούν να απεικονιστούν σε ένα εργαλείο οπτικοποίησης δεδομένων.

- Associate Panel: Παρέχει πρόσβαση στους εκπαιδευόμενους κανόνες συσχέτισης που προσπαθούν να εντοπίσουν όλες τις σημαντικές αλληλεξαρτήσεις μεταξύ των χαρακτηριστικών των δεδομένων.

- Select Attributes Panel: Παρέχει αλγόριθμους για τον εντοπισμό των πιο έξυπνων χαρακτηριστικών σε ένα σύνολο δεδομένων. Επίσης, επιτρέπει να ρυθμιστεί και να εφαρμοστεί οποιοσδήποτε συνδυασμός χαρακτηριστικών αξιολογητή από το χρήστη, αλλά και η μέθοδος αναζήτησης για να επιλέξει τα πιο συναφή χαρακτηριστικά στο σύνολο δεδομένων. Εάν ένα σύστημα επιλογής χαρακτηριστικών μετατρέπει τα δεδομένα, τότε τα

μετασχηματισμένα δεδομένα μπορούν να απεικονιστούν σε ένα εργαλείο οπτικοποίησης δεδομένων.

- Visualization Panel: Παρέχει μια πίνακας γραφικών παραστάσεων, όπου μπορούν να επιλεγούν και να διευρυνθούν επιμέρους διαγράμματα διασποράς, τα οποία αναλύονται περαιτέρω χρησιμοποιώντας διάφορους τελεστές επιλογής. Σε πολλές πρακτικές εφαρμογές, η οπτικοποίηση των δεδομένων παρέχει σημαντικές γνώσεις. Μπορεί ακόμη και να καταστήσει δυνατή την αποφυγή περαιτέρω ανάλυσης των δεδομένων με τη χρήση μηχανικής μάθησης και αλγορίθμων εξόρυξης δεδομένων. Επίσης, παρέχει μια πίνακας διαγραμμάτων διασποράς με χρωματική κωδικοποίηση, μαζί με την επιλογή επιμέρους γραφημάτων σε αυτή τη πίνακας και επιλέγοντας επιμέρους τμήματα των δεδομένων προς απεικόνιση, το οποίο επιτρέπει στα σύνολα των δεδομένων και των προβλέψεων των ταξινομητών να απεικονιστούν σε δύο διαστάσεις. Οι μορφές που υποστηρίζονται είναι BMP, EPS, JPEG και PNG. Είναι επίσης δυνατή η εξαγωγή διαφόρων διαγραμμάτων, όπως PNG αρχεία μη αλληλεπιδραστικά από μια διαδικασία ροής της γνώσης.

3.3.3 Αντικείμενα Δεδομένων του συστήματος Weka

Τα σύνολα δεδομένων με τα οποία τροφοδοτείται το Weka και στη συνέχεια επεξεργάζεται είναι σε συγκεκριμένη μορφή. Έχουν την απλή μορφή του κειμένου (plain text) αλλά με μία ορισμένη δομή. Τις περισσότερες φορές τα αρχεία αυτά έχουν κατάληξη .arff (Attribute-RelationFile Format) και πρόκειται για ένα αρχείο κειμένου ASCII, το περιέχει μια σειρά από παραδείγματα (instances) τα οποία περιγράφονται από χαρακτηριστικά (attributes) [23, 32].

Τα αρχεία .arff αποτελούνται από δύο τμήματα. Το πρώτο μέρος είναι η Επικεφαλίδα (Header) και ακολουθεί το τμήμα των δεδομένων (Data). Το τμήμα της επικεφαλίδας ενός ARFF αρχείου περιλαμβάνει πληροφορίες όπως το όνομα του συνόλου δεδομένων (relation), και μία λίστα των ιδιοτήτων (attributes) των δειγμάτων με τους τύπους τους.

Πιο συγκεκριμένα, οι γραμμές οι οποίες ξεκινάνε με % είναι σχόλια και δε λαμβάνονται υπόψη κατά τη φόρτωση του αρχείου. Μετά τα εισαγωγικά

σχόλια ακολουθεί το όνομα που περιγράφει το αρχείο στη γραμμή που ξεκινάει το @ relation.

- **Τύποι Χαρακτηριστικών**

  Ακολουθεί η δήλωση όλων των χαρακτηριστικών που περιγράφουν το σύνολο των δεδομένων. Η δήλωση γίνεται χρησιμοποιώντας την παρακάτω σύμβαση:

  @ attribute < attribute –name> <datatype>

  Το όρισμα <attribute – name>, είναι το όνομα του χαρακτηριστικού, ενώ το όρισμα <datatype> καθορίζει τον τύπο του χαρακτηριστικού. Το Weka υποστηρίζει τέσσερις τύπους χαρακτηριστικών:

  1. Αριθμητικά Χαρακτηριστικά (numeric) : Ενδέχεται να είναι είτε πραγματικοί είτε ακέραιοι αριθμοί.
  2. Ονομαστικά Χαρακτηριστικά (nominal) : Ορίζονται χρησιμοποιώντας αγκύλες εντός των οποίων ορίζονται όλες οι δυνατές τιμές.

     @ attribute { <nominal-name1>, ….., <nominal-name2>}

  3. Αλφαριθμητικά Χαρακτηριστικά (string): Επιτρέπουν τη δημιουργία αυθαίρετων αλφαριθμητικών δομών.

     @ attribute string

  4. Ημερομηνίες (date): Ο καθορισμός τιμών που παίρνουν ως τιμή ημερομηνίες έχουν την παρακάτω μορφή.

     @ attribute <name> date

- **Δεδομένα**

  Έπειτα από τη δήλωση των χαρακτηριστικών ακολουθεί η δήλωση των δεδομένων, ως εξής:

  @ data

Το τμήμα των δεδομένων του αρχείου στην περίπτωση της μάθησης με επίβλεψη έχει την ακόλουθη μορφή:

<Τιμή_1>,...,<Τιμή_N>,<Κλάση_αντικειμένου>,

ενώ στην περίπτωση της μη επιβλεπόμενης μάθησης, που η κλάση κάθε παρατήρησης είναι άγνωστη, έχει την εξής μορφή:

<Τιμή_1>,...,<Τιμή_N>.

Γενικά τα δεδομένα μπορεί να βρίσκονται σε διάφορες μορφές και τύπους αρχείων. Το WEKA περιέχει ενσωματωμένους μετατροπείς για τους πιο κοινούς τύπους αρχείων για να τα τυποποιήσει ως .arff. Πιο συγκεκριμένα, η τυποποίηση .arff αποτελεί τη φυσική μέθοδο αποθήκευσης δεδομένων του WEKA, υποστηρίζοντας τόσο αριθμητικά όσο και ονομαστικά χαρακτηριστικά.

3.4 Παρουσίαση του συστήματος R

Η R είναι μια γλώσσα προγραμματισμού που χρησιμεύει στην επεξηγηματική ανάλυση δεδομένων καθώς και στην εφαρμογή διάφορων στατιστικών μοντέλων. Το λογισμικό R δημιουργήθηκε το 1993 από τους Ross Ihaka και Robert Gentleman στο Πανεπιστήμιο του Auckland στη Νέα Ζηλανδία. Λέγεται R για τον απλό λόγο ότι οι δύο δημιουργοί του έχουν ονόματα που ξεκινούν με το γράμμα "R." Ορισμένοι πιστεύουν ότι το όνομα και μονόγραμμα R αντιπροσωπεύει ένα είδος φόρος τιμής στη γλώσσα S, δεδομένου ότι η γλώσσα R είναι ένας απόγονος ανοιχτού κώδικα S και ένα μεγάλο μέρος του κώδικα της S έχει προστεθεί αυτούσιο στην R. Η S γλώσσα προγραμματισμού αναπτύχθηκε από την ομάδα Bell Labs, για την οποία κέρδισε το βραβείο ACM System Software το 1998 [20].

Οι Ihaka και Gentleman δημιούργησαν μια γλώσσα προγραμματισμού που κατέστησε ευκολότερο για αυτούς να διδάξουν τα μαθήματα τους στην εισαγωγική ανάλυση των δεδομένων. Το R διανέμεται ελεύθερα υπό τα πλαίσια και τους όρους της οργάνωσης GNU General Public License του Ιδρύματος Ελεύθερου Λογισμικού.

Η ανάπτυξη και η διανομή του πραγματοποιείται από διάφορους στατιστικολόγους και επιστήμονες ηλεκτρονικών υπολογιστών, ως μέλη της βασικής ομάδας ανάπτυξης του R (R Development Core Team), αλλά και μέσω της εθελοντικής συνεισφοράς πολλών ανθρώπων ανά τον κόσμο, οι οποίο είναι υπεύθυνοι για την ανάπτυξη της. Επίσης, παρέχεται καθοδήγηση και συμβουλευτική υποστήριξη στους χρήστες του λογισμικού R μέσα από μια ενεργή λίστα ηλεκτρονικού ταχυδρομείου. Η υψηλή δημοτικότητά του R, μεταφράζεται σε μια πολυάριθμη κοινότητα χρηστών η οποία δημιουργεί νέα προγράμματα R (που ονομάζονται "πακέτα") με έναν εκπληκτικό ρυθμό.

Ακριβώς επειδή το R είναι μια γλώσσα προγραμματισμού, και όχι ένα προκατασκευασμένο κομμάτι του λογισμικού νέων αναλυτικών τεχνικών που είναι γραμμένα σε R, όταν οι χρήστες του ανακαλύψουν κάτι καινούριο και συναρπαστικό, έχουν δύο επιλογές:

1. Μπορούν να μοιραστούν τις νέες τεχνικές με άλλους χρήστες του R.

2. Μπορούν να αναπαράγουν και να επαναχρησιμοποιούν τις νέες τεχνικές που έχουν ανακαλύψει.

Η ικανότητα των χρηστών να μοιράζονται ότι νέο έχουν ανακαλύψει αναφορικά με τον κώδικα R, πραγματοποιείται μέσω του φόρουμ το οποίο φιλοξενείται από το CRAN (Comprehensive R Archive Network) και εξασφαλίζει μια κατάσταση συνεχούς εξέλιξης της γλώσσας και των πακέτων.

Η ιστοσελίδα http://www.r-project.org/ περιέχει περαιτέρω πληροφορίες καθώς και συνδέσμους, για τα σχετικά προγράμματα που αφορούν την αποθήκευση και την εκτέλεση του προγράμματος σε διάφορα λειτουργικά συστήματα. Το R μπορεί να τρέξει σε περιβάλλον Linux, Mac OS και Windows.

Τέλος, το R είναι ένα όχημα ανάπτυξης νέων μεθόδων για τη διαδραστική ανάλυση δεδομένων, το οποίο έχει αναπτυχτεί ραγδαία και έχει επεκταθεί από μια μεγάλη συλλογή από πακέτα.

3.4.1 Το περιβάλλον του συστήματος R

Το περιβάλλον R είναι μια ολοκληρωμένη σουίτα λογισμικού για το χειρισμό των δεδομένων, τον υπολογισμό των στατιστικών συναρτήσεων και την γραφική απεικόνιση. Χαρακτηρίζεται από [20]:

- ένα αποτελεσματικό χειρισμό των δεδομένων αλλά και από ιδιότητες αποθήκευσης δεδομένων,

- μια ολοκληρωμένη συλλογή εργαλείων στατιστικής ανάλυσης,

- μια σουίτα τελεστών για τους υπολογισμούς σε συστοιχίες, ιδίως μήτρες,

- μια μεγάλη, συναφής, ολοκληρωμένη συλλογή ενδιάμεσων εργαλείων για την ανάλυση δεδομένων,

- ιδιότητες γραφικών απεικονίσεων για την ανάλυση δεδομένων και την απεικόνιση είτε απευθείας στον υπολογιστή είτε σε έντυπη μορφή,

- μια καλά αναπτυγμένη, απλή και αποτελεσματική γλώσσα προγραμματισμού (που ονομάζεται "S") η οποία περιλαμβάνει υποθέσεις, βρόχους, αναδρομικές συναρτήσεις που ορίζονται από το χρήστη και εγκαταστάσεις εισόδου και εξόδου,

- ένα εξαιρετικό ενσωματωμένο σύστημα βοήθειας μέσω του CRAN web site,

- και δυνατότητα προσθήκης πακέτων/βιβλιοθηκών, που λόγω του μεγάλου αριθμού τους, καλύπτουν οποιοδήποτε πρόβλημα ανακύψει.

Γενικά, το R είναι ένα πλήρως οργανωμένο και συναφές σύστημα, παρά μια σταδιακή προσαύξηση των πολύ συγκεκριμένων και άκαμπτων εργαλείων του, όπως συμβαίνει συχνά με άλλα λογισμικά ανάλυσης δεδομένων.

3.4.2 Βασική Λειτουργία του συστήματος R

Η R εφαρμόζει μια διάλεκτο της γλώσσας S, η οποία είναι μια διερμηνέας γλώσσα προγραμματισμού, που σημαίνει ότι οι εντολές διαβάζονται και εκτελούνται αμέσως. Αντίθετα, η C και Fortran είναι μεταγλωττίστριες γλώσσες προγραμματισμού, όπου ολοκληρωμένα προγράμματα μεταφράζονται με τη βοήθεια ενός μεταγλωττιστή στην κατάλληλη γλώσσα μηχανής. Το μεγάλο πλεονέκτημα των διερμηνέων γλωσσών προγραμματισμού είναι ότι επιτρέπουν σταδιακή ανάπτυξη. Από την άλλη μεριά όμως, ο μεταγλωττισμένος κώδικας τρέχει πιο γρήγορα και χρειάζεται λιγότερη μνήμη.

Όταν το R βρίσκεται σε λειτουργία, οι μεταβλητές, τα δεδομένα, τα αποτελέσματα και λοιπά, αποθηκεύονται στην ενεργό μνήμη του υπολογιστή υπό τη μορφή αντικειμένων, ο χρήστης μπορεί να αλληλεπιδράσει με τα αντικείμενα αυτά μέσω των συναρτήσεων και των διαχειριστών. Όλες οι ενέργειες αλληλεπιδρούν με τα αντικείμενα που έχουν αποθηκευτεί στην ενεργή μνήμη του υπολογιστή, δεν υπάρχουν προσωρινά αρχεία που χρησιμοποιούνται.

Έτσι λοιπόν, ο χρήστης εκτελεί κάποιες συναρτήσεις μέσω κάποιων εντολών και τα αποτελέσματα εμφανίζονται άμεσα στην οθόνη, και αποθηκεύονται σε ένα αντικείμενο, ή στο δίσκο (κυρίως τα γραφικά). Δεδομένου ότι τα αποτελέσματα είναι αντικείμενα μπορούν να θεωρηθούν ως δεδομένα και να αναλυθούν ως τέτοια. Αρχεία δεδομένων μπορούν να προσπελαστούν από τον τοπικό δίσκο ή από έναν απομακρυσμένο server μέσω του διαδικτύου. Οι συναρτήσεις που είναι διαθέσιμες στο χρήστη αποθηκεύονται σε μια βιβλιοθήκη και εντοπίζονται στο δίσκο σε έναν κατάλογο που ονομάζεται R_home/βιβλιοθήκη (είναι ο κατάλογος όπου είναι εγκατεστημένο το R). Αυτός ο κατάλογος περιέχει τα πακέτα των συναρτήσεων όπου είναι δομημένα σε καταλόγους.

Προκειμένου να εκτελεστεί μια λειτουργία πρέπει πάντα να γράφεται με παρενθέσεις, ακόμη και αν δεν υπάρχει τίποτα μέσα τους. Εάν επιλέξουμε το όνομα της συνάρτησης, χωρίς παρενθέσεις θα εμφανίσει το περιεχόμενο της συνάρτησης.

Αξίζει να σημειωθεί ότι η R είναι ευαίσθητη στα κεφαλαία γράμματα, δηλαδή το x και το X είναι διαφορετικά αντικείμενα. Μια συνάρτηση καλείται γράφοντας το όνομα της ακολουθούμενη από μια λίστα ορισμάτων. Οι μαθηματικές πράξεις είναι συναρτήσεις με δύο ορίσματα τα οποία έχουν ειδικό κάλεσμα. Ένα από τα σύμβολα που

χρησιμοποιείται πιο συχνά είναι το σύμβολο εκχώρησης <-, το οποίο καταχωρεί στις μεταβλητές συγκεκριμένες τιμές ή αποτελέσματα πράξεων. Ακόμη ένα συνηθισμένο σύμβολο είναι το σύμβολο του δείκτη, το οποίο χρησιμοποιείται για να εξάγει υποσύνολα από ένα αντικείμενο. Η τοποθέτηση δεικτών είναι πολύ σημαντική στην αποτελεσματική χρήση της R γιατί δίνει έμφαση στην επεξεργασία αντικειμένων δεδομένων σαν ολοκληρωμένες οντότητες, παρά σαν μια συλλογή από ξεχωριστές παρατηρήσεις.

Τέλος, τονίζεται ότι κάθε έκφραση της R ερμηνεύεται από τον αξιολογητή και επιστρέφει ένα αντικείμενο δεδομένων. Τα αντικείμενα δεδομένων έχουν τις παρακάτω μορφές: λογική, αριθμητική, μιγαδική, κειμένου. Οι μορφές είναι γραμμένες από αυτήν που παρέχει την λιγότερη πληροφορία έως εκείνη που παρέχει την περισσότερη. Όταν είναι ανάγκη να συνδυαστούν διαφορετικές μορφές, τότε η R χρησιμοποιεί εκείνη με την περισσότερη πληροφορία.

3.4.3 Αντικείμενα Δεδομένων του συστήματος R

Το R λειτουργεί με αντικείμενα που χαρακτηρίζονται από τα ονόματά τους και το περιεχόμενό τους, αλλά και από τα χαρακτηριστικά που προσδιορίζουν το είδος των δεδομένων που αντιπροσωπεύουν. Γενικότερα ισχύει ότι η δράση μιας συνάρτησης σε ένα αντικείμενο εξαρτάται από τις ιδιότητες του τελευταίου. Όλα τα αντικείμενα έχουν δύο εγγενή χαρακτηριστικά (ιδιότητες): τη μορφή και το μήκος. Τα αντικείμενα δεδομένων είναι οι διάφορες μορφές στις οποίες μπορούν να αποθηκευτούν δεδομένα στη R. Οι κύριες μορφές αντικειμένων που υπάρχουν είναι οι ακόλουθες [20,38]:

- Διάνυσμα (vector): Είναι ένα διατεταγμένο σύνολο τιμών σε σειρά. Η εσωτερική διάταξη του διανύσματος υποδεικνύει ότι υπάρχει ένας κατάλληλος τρόπος με τον οποίο μπορούν να εξαχθούν μερικά ή όλα από τα στοιχεία του.

- Πίνακας (matrix): Χρησιμοποιούνται για να τακτοποιήσουν τιμές κατά γραμμές και στήλες σε έναν ορθογώνιο πίνακα. Στην ανάλυση δεδομένων, οι διάφορες μεταβλητές συνήθως παρουσιάζονται σε διαφορετικές στήλες και οι διάφορες περιπτώσεις ή τιμές σε διαφορετικές γραμμές. Οι πίνακες διαφέρουν

από τα διανύσματα, γιατί έχουν διαστάσεις και σε αυτούς μπορεί να εφαρμοστεί η συνάρτηση διάστασης dim.

- Συστοιχία (array): Επεκτείνεται η έννοια της διάστασης σε παραπάνω από δύο. Κατά συνέπεια, μεγαλώνει και η διάσταση της συνάρτησης dim. Δεν υπάρχει κανένας περιορισμός στον αριθμό των διαστάσεων ενός πίνακα μεγαλύτερης διάστασης. Η πρώτη διάσταση (γραμμές) συμπληρώνεται πρώτη, η δεύτερη διάσταση συμπληρώνεται δεύτερη. Η τρίτη διάσταση όμως συμπληρώνεται με τη δημιουργία ενός πίνακα για κάθε επίπεδο της τρίτης διάστασης. Σους πίνακες μεγαλύτερης διάστασης, εφαρμόζονται οι ίδιες εντολές για την αναγνώριση του μεγέθους, των διαστάσεων και τη μορφή των τιμών τους όπως και στην περίπτωση των πινάκων, αλλά και με τον ίδιο τρόπο δίνονται ονόματα στις διαστάσεις τους.

- Λίστα (list): Αποτελείται από διάφορες συνιστώσες, η κάθε μια από τις οποίες περιέχει διαφορετική μορφή δεδομένων.

- Παράγοντας (factor): Για σκοπούς ανάλυσης δεδομένων, μερικές από τις μεταβλητές μπορεί να μην είναι ποσοτικές αλλά ποιοτικές ή κατηγορικές. Οι κατηγορικές μεταβλητές παρουσιάζονται με το δεδομένων παράγοντας. Για να κατασκευαστεί ένας παράγοντας εφαρμόζεται η συνάρτηση factor.

- Πλαίσιο δεδομένων (data frames): Επιτρέπει τον συνδυασμό δεδομένων διαφορετικών μορφών μέσα σε ένα αντικείμενο για να χρησιμοποιηθεί για ανάλυση και μοντελοποίηση. Η ιδέα του πλαισίου δεδομένων είναι η Κατηγοριοποίηση των τιμών κατά μεταβλητή (στήλη) ανεξάρτητα της μορφής τους. Έπειτα, όλες οι παρατηρήσεις ενός συγκεκριμένου συνόλου μεταβλητών ταξινομούνται σε πλαίσιο δεδομένων.

3.4.4 Γραφική Απεικόνιση στο σύστημα R

Το R είναι ιδιαίτερα χρήσιμο για τη δημιουργία διαγραμμάτων και γραφικών, γρήγορα και εύκολα. Η ικανότητα να δημιουργεί οπτικά οικόπεδα των σύνθετων δεδομένων είναι κάτι περισσότερο από ένα εύχρηστο τέχνασμα, είναι ένα εξαιρετικά σημαντικό

βήμα για την ανάλυση των δεδομένων, διότι δίνει τη δυνατότητα κυριολεκτικά να «βλέπει» τα πρότυπα και τις ανωμαλίες που κρύβονται μέσα στα δεδομένα [38].

Το R καθιστά δυνατή για τους ανθρώπους που δεν είναι επαγγελματίες αναλυτές να δημιουργούν υψηλής ποιότητας διαγράμματα και γραφικά, όπως χάρτες, επιφάνειες 3-D, οικόπεδα εικόνα, διαγράμματα διασποράς, ιστογράμματα, οικόπεδα μπαρ και διαγράμματα πίτας.

Οι γραφικές απεικονίσεις και λειτουργίες είναι ένα σημαντικό και εξαιρετικά ευέλικτο στοιχείο του περιβάλλοντος R. Μπορεί κάποιος να χρησιμοποιήσει τις γραφικές λειτουργίες για να εμφανίσει μια μεγάλη ποικιλία στατιστικών γραφημάτων, και επίσης να οικοδομήσει εντελώς νέους τύπους γραφήματος. Το R προσφέρει μια αξιόλογη ποικιλία γραφικών απεικονίσεων. Για να πάρει κανείς μια ιδέα για τις επιλογές που παρέχονται, μπορεί κανείς να πληκτρολογήσει demo (graphics) ή demo (persp). Στο R κάθε συνάρτηση γραφικής απεικόνισης έχει ένα μεγάλο αριθμό επιλογών καθιστώντας την εξαγωγή των γραφικών πολύ ευέλικτη.

Το αποτέλεσμα μιας γραφικής συνάρτησης δεν μπορεί να εκχωρηθεί σε ένα αντικείμενο αλλά αποστέλλεται σε μία γραφική διάταξη. Μια γραφική διάταξη είναι ένα γραφικό παράθυρο ή ένα αρχείο. Υπάρχουν τρία είδη εντολών σχεδίασης:

1. οι υψηλού επιπέδου σχεδίασης λειτουργίες, οι οποίες δημιουργούν ένα νέο γράφημα, και

2. οι χαμηλού επιπέδου σχεδίασης λειτουργίες, οι οποίες προσθέτουν στοιχεία σε ένα υπάρχων γράφημα, όπως σημεία γραμμές, ετικέτες και λοιπά.

3. Διαδραστικές λειτουργίες γραφικών, οι οποίες επιτρέπουν να προστεθούν ή να αποσπάσουν πληροφορίες, από ένα υπάρχων γράφημα, χρησιμοποιώντας μια συσκευή κατάδειξης, όπως ένα ποντίκι.

Γενικά, τα διαγράμματα παράγονται με σεβασμό προς τις γραφικές παραμέτρους που ορίζονται από προεπιλογή και μπορούν να τροποποιηθούν με τη συνάρτηση par. Όταν εκτελείται μια γραφική λειτουργία, εάν δεν είναι ανοιχτή μια γραφική διάταξη, το R ανοίγει ένα παράθυρο και εμφανίζει το γράφημα. Μια γραφική διάταξη μπορεί να

ανοίξει με μια κατάλληλη λειτουργία. Η λίστα των διαθέσιμων γραφικών διατάξεων εξαρτάται από το λειτουργικό σύστημα.

Από προεπιλογή, το R εξάγει γραφικές παραστάσεις με ένα «έξυπνο» τρόπο: τα διαστήματα μεταξύ των σημάτων στους άξονες, η τοποθέτηση ετικετών, και λοιπά., υπολογίζονται έτσι ώστε το προκύπτον γράφημα να είναι όσο το δυνατόν κατανοητό. Ο χρήστης μπορεί, παρ 'όλα αυτά, να αλλάξει τον τρόπο που παρουσιάζεται ένα γράφημα, για παράδειγμα, να συμμορφώνεται με ένα προκαθορισμένο συντακτικό ύφος, ή να του δώσει μια προσωπική πινελιά. Ο απλούστερος τρόπος για να αλλάξει η παρουσίαση ενός γραφήματος είναι να προστεθούν επιλογές που θα τροποποιήσουν τις προεπιλεγμένες.

3.4.5 Στατιστική Ανάλυση στο σύστημα R

Επειδή το R δημιουργήθηκε από τους στατιστικολόγους για τους στατιστικολόγους, είναι ήδη φορτωμένο με πολλά από τα σημαντικά χαρακτηριστικά που απαιτούνται για να ολοκληρωθούν οι καθημερινές εργασίες της στατιστικής ανάλυσης. Με άλλα λόγια, το R είναι σε αρμονία με τον τρόπο, όπου οι στατιστικολόγοι σκέφτονται και εργάζονται.

Το R περιέχει λειτουργίες για ένα ευρύ φάσμα βασικών στατιστικών αναλύσεων: κλασικές δοκιμές, γραμμικά μοντέλα (συμπεριλαμβανομένων των ελαχίστων τετραγώνων, γενικευμένα γραμμικά μοντέλα, και ανάλυση της διακύμανσης), διασπορά, ιεραρχική συσταδοποίηση, ανάλυση χρονοσειρών, μη γραμμικών ελαχίστων τετραγώνων, και πολυπαραγοντική ανάλυση. Άλλες στατιστικές μέθοδοι είναι διαθέσιμες σε ένα μεγάλο αριθμό πακέτων. Μερικά από αυτά είναι κατανεμημένα με μια βασική εγκατάσταση του R και είναι χαρακτηρισμένα ως συστημένα, και πολλά άλλα πακέτα συμβάλλουν και πρέπει να εγκατασταθούν από το χρήστη.

Οι τύποι είναι ένα βασικό στοιχείο στις στατιστικές αναλύσεις με το R : ο συμβολισμός που χρησιμοποιείται είναι ο ίδιος για (σχεδόν) όλες τις λειτουργίες. Ένας τύπος είναι συνήθως της μορφής y ~ model όπου y είναι η ανάλυση απόκρισης και το model είναι ένα σύνολο όρων για το οποίο ορισμένοι παράμετροι πρόκειται να εκτιμηθούν .

Οι λειτουργίες του R ενεργούν σε σχέση με τις ιδιότητες των αντικειμένων. Οι R στατιστικές συναρτήσεις επιστρέφουν ένα αντικείμενο της κλάσης με το ίδιο όνομα (για παράδειγμα AOV επιστρέφει ένα αντικείμενο της κλάσης "AOV», lm επιστρέφει ένα της κατηγορίας "lm"). Οι λειτουργίες που χρησιμοποιούνται στη συνέχεια για να εξαχθούν τα αποτελέσματα θα ενεργήσουν με σεβασμό στην κλάση του αντικειμένου. Οι λειτουργίες αυτές ονομάζονται γενικές.

Για παράδειγμα, η συνάρτηση η οποία χρησιμοποιείται περισσότερο, για την άντληση αποτελεσμάτων από την ανάλυση είναι η “summary”, η οποία εμφανίζει στην οθόνη τα αναλυτικά αποτελέσματα. Είτε το αντικείμενο, το οποίο δίνεται ως ερώτημα, είναι της κατηγορίας "lm" (γραμμικό μοντέλο) είτε "AOV" (analysis of variance), είναι προφανές ότι οι πληροφορίες που θα εμφανιστούν στην οθόνη, δεν θα είναι οι ίδιες. Το πλεονέκτημα των γενικών συναρτήσεων είναι ότι η σύνταξη είναι η ίδια σε όλες τις περιπτώσεις.

Το αντικείμενο που περιέχει τα αποτελέσματα μιας ανάλυσης είναι γενικά μια λίστα, και ο τρόπος που εμφανίζεται στην οθόνη προσδιορίζεται από την κλάση. Η δράση μιας συνάρτησης εξαρτάται από το είδος του αντικειμένου που δίνεται ως όρισμα. Είναι ένα γενικό χαρακτηριστικό του R .

Πολλά τα πακέτα χαρακτηρίζονται ως «συνιστώμενα» δεδομένου ότι καλύπτουν στατιστικές μεθόδους που χρησιμοποιούνται συχνά στην ανάλυση των δεδομένων. Τα συνιστώμενα πακέτα διανέμονται συχνά με μια βασική εγκατάσταση του R.

3.5 Παρουσίαση του συστήματος Python

Η γλώσσα προγραμματισμού Python επινοήθηκε στα τέλη της δεκαετίας του 1980 και η εφαρμογή της ξεκίνησε το Δεκέμβριο του 1989 από τον Guido van Rossum στο CWI στην Ολλανδία, ως διάδοχος της γλώσσας ABC. Το όνομα της γλώσσας προέρχεται από την ομάδα άγγλων κωμικών Monty Python. Αρχικά, η Python ήταν μια γλώσσα σεναρίων που χρησιμοποιούνταν στο λειτουργικό σύστημα Amoeba, ικανή και για κλήσεις συστημάτων [40].

Το λογισμικό Python αναπτύσσεται ως ανοιχτό λογισμικό και η διαχείρισή του γίνεται από τον μη κερδοσκοπικό οργανισμό Python Software Foundation. Ο κώδικας διανέμεται με την άδεια Python Software Foundation License, η οποία είναι συμβατή με την GPL. Ο κύριος στόχος της είναι η αναγνωσιμότητα του κώδικα της και η ευκολία χρήσης της. Διακρίνεται λόγω του ότι έχει πολλές βιβλιοθήκες που διευκολύνουν ιδιαίτερα, αρκετές συνηθισμένες εργασίες, αλλά και για την ταχύτητα εκμάθησης της.

Μεταξύ των διερμηνέων γλωσσών, η Python διακρίνεται από τη μεγάλη και τη δραστήρια επιστημονική κοινότητα της. Η υιοθέτηση της Python για την επιστημονική κοινότητα της πληροφορικής και στις δύο εφαρμογές της, της βιομηχανίας και της ακαδημαϊκής έρευνας έχει αυξηθεί σημαντικά από τις αρχές της δεκαετίας του 2000.

Τα τελευταία χρόνια, η βελτιωμένη υποστήριξη των βιβλιοθηκών της Python (κυρίως pandas) ανέπτυξε μια ισχυρή εναλλακτική λύση για εργασίες διαχείρισης δεδομένων. Σε συνδυασμό με τη δύναμη της Python στον προγραμματισμό γενικού σκοπού, είναι μια εξαιρετική επιλογή ως μια ενιαία γλώσσα για την κατασκευή εφαρμογών επεξεργασίας δεδομένων.

3.5.1 Το περιβάλλον του συστήματος Python

Η Python είναι μια εύκολη στην εκμάθηση, ισχυρή γλώσσα προγραμματισμού. έχει αποδοτικές δομές δεδομένων υψηλού επιπέδου και μια απλή αλλά αποτελεσματική προσέγγιση στον αντικειμενοστραφή προγραμματισμό. Η κομψή σύνταξη της Python και οι δυναμικοί τύποι της, μαζί με τη λειτουργία της ως διερμηνευόμενης (αντί μεταγλωττιζόμενης) γλώσσας, την καθιστούν την ιδανική γλώσσα για δημιουργία σεναρίων εντολών και για ταχεία ανάπτυξη εφαρμογών σε πολλούς τομείς και στις περισσότερες πλατφόρμες. Η σχεδιαστική φιλοσοφία της τονίζει την αναγνωσιμότητα του κώδικα, και η σύνταξη του επιτρέπει στους προγραμματιστές να εκφράσουν έννοιες σε λιγότερες γραμμές κώδικα από ό, τι θα ήταν δυνατό σε γλώσσες όπως C. Η Python παρέχει δομές που επιτρέπουν σαφή προγράμματα τόσο σε μικρή όσο και μεγάλη κλίμακα [40].

Επίσης, η Python χρησιμοποιεί δυναμικό σχεδιασμό γραψίματος, καταμέτρησης αναφορών και κύκλου διαχείρισης σκουπιδιών “cycle-detecting garbage collector” για τη σωστότερη διαχείριση της μνήμης. Επιπλέον, ένα σημαντικό στοιχείο της Python είναι η δυναμική δέσμευση, με την οποία δεσμεύει τις μεθόδους και τα ονόματα μεταβλητών.

Αντί να απαιτείται όλη η επιθυμητή λειτουργικότητα να κατασκευαστεί στον πυρήνα της γλώσσας, η Python έχει σχεδιαστεί για να είναι εξαιρετικά επεκτάσιμη. Η Python μπορεί επίσης να ενσωματώνεται σε υφιστάμενες εφαρμογές που χρειάζονται μια προγραμματιζόμενη διασύνδεση. Αυτός ο σχεδιασμός ενός μικρού πυρήνα γλώσσας με μια μεγάλη πρότυπη βιβλιοθήκη η οποία είναι εύκολα επεκτάσιμη προορίζεται από τον Van Rossum λόγω των απογοητεύσεων του με τη γλώσσα ABC.

Οι περισσότερες εφαρμογές Python μπορούν να λειτουργήσουν ως μια γραμμή διερμηνέα εντολών, για την οποία ο χρήστης εισάγει διαδοχικά τις δηλώσεις και λαμβάνει αμέσως τα αποτελέσματα. Εν ολίγοις, Python λειτουργεί ως ένα κέλυφος.

Ενώ προσφέρει την επιλογή μεθοδολογίας κωδικοποίησης, η φιλοσοφία Python απορρίπτει την πληθωρική σύνταξη, όπως στην Perl, είναι υπέρ μιας πιο αραιής, λιγότερης σωριασμένης γραμματικής. Η βασική φιλοσοφία της γλώσσας Python συνοψίζεται στο έγγραφο " PEP 20 (Το Zen της Python).

Η Python 2.0 κυκλοφόρησε στις 16 Οκτωβρίου του 2000. Στις 3 Δεκεμβρίου 2008 κυκλοφόρησε η έκδοση 3.0 (γνωστή και ως py3k ή python 3000). Πολλά από τα καινούργια χαρακτηριστικά αυτής της έκδοσης έχουν μεταφερθεί στις εκδόσεις 2.6 και 2.7 που είναι προς τα πίσω συμβατές. Η Python 3.0 είναι ιστορικά η πρώτη γλώσσα προγραμματισμού που σπάει την προς τα πίσω συμβατότητα με προηγούμενες εκδόσεις ώστε να διορθωθούν κάποια λάθη που υπήρχαν σε προγενέστερες εκδόσεις και να καταστεί ακόμα πιο σαφής ο απλός τρόπος με τον οποίο μπορούν να γίνουν κάποια πράγματα. Σε αυτή τη μεταπτυχιακή διατριβή θα χρησιμοποιήσουμε την Python 3.4.

Η Python είναι πραγματικά μια συναρπαστική και ισχυρότατη γλώσσα. Έχει το σωστό συνδυασμό απόδοσης και χαρακτηριστικών που κάνουν τη δημιουργία προγραμμάτων σε Python διασκεδαστική και εύκολη.

- **Το περιβάλλον της Python 3.4**

Στον φάκελο Έναρξη > Προγράμματα > Python (που αποτελεί το βασικό μενού εικονιδίων της Python) θα βρούμε τις εξής επιλογές [40]:

1. IDLE (Python GUI)

2. Module Docs

3. Python (command line)

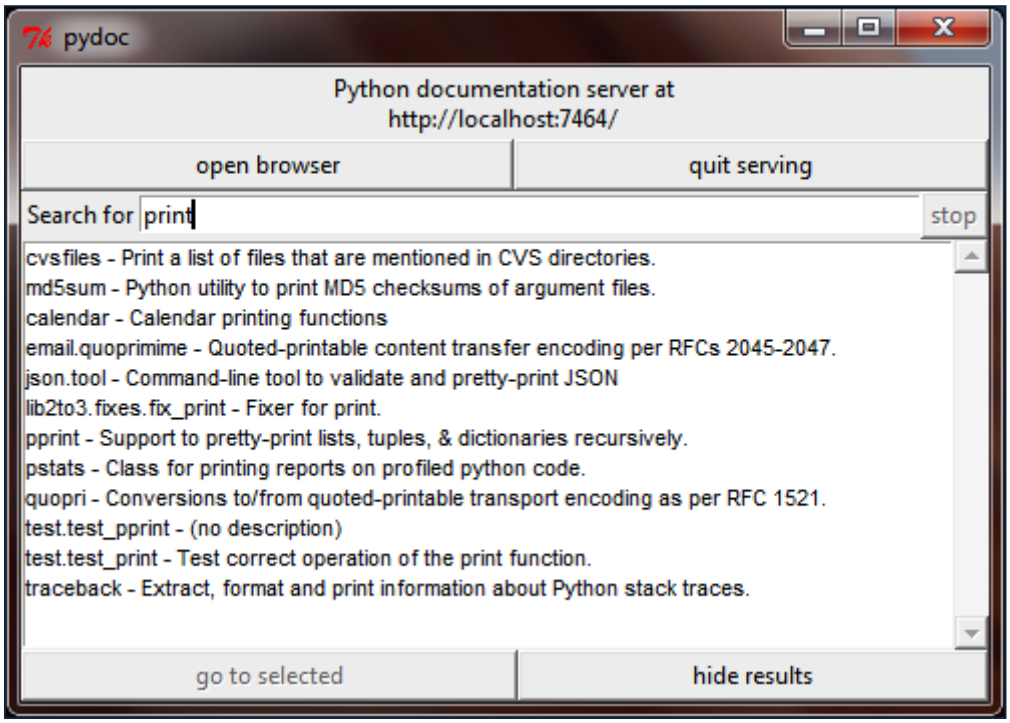
4. Python Manuals

5. Uninstall Python

Οι δύο τελευταίες επιλογές είναι προφανές ότι αντιστοιχούν σε προβολή του εγχειριδίου χρήσης της γλώσσας και απεγκατάσταση της Python από τον υπολογιστή. Ας δούμε λίγο αναλυτικότερα τις υπόλοιπες τρεις.

1. IDLE (Python GUI): Αποτελεί μια διεπαφή χρήστη που, εκτός από το γεγονός ότι ανοίγει τον διερμηνευτή της Python, προσφέρει κάποιες ευκολίες στον προγραμματιστή όπως να ανοίγει και να επεξεργάζεται εφαρμογές Python, να κάνει debugging και πολλές από τις ευκολίες που παρέχει ένας μέσος επεξεργαστής κειμένου (εύρεση / αντικατάσταση, λειτουργίες αντιγραφής / αποκοπής / επικόλλησης, και λοιπά). Φυσικά, για την συγγραφή των Python εφαρμογών μπορεί να χρησιμοποιηθεί εναλλακτικά οποιοσδήποτε άλλος επεξεργαστής κειμένου.

2. Module Docs: Πρόκειται για ένα εργαλείο που επιτρέπει στον προγραμματιστή να περιηγηθεί στα modules που περιέχονται εγγενώς στην Python καθώς και να εμφανίσει πληροφορίες για τις λειτουργίες τους. Τέλος δίνει τη δυνατότητα στον χρήστη να δει τα πάντα συγκεντρωμένα μέσω του αγαπημένου του web browser.



Εικόνα 6. Σελίδα εγχειριδίων χρήσης και λίστα τελεστών του Python.

3. Python (command line):Ουσιαστικά ανοίγει τον διερμηνέα της Python σε ένα περιβάλλον DOS. Ωστόσο δεν προσφέρει καμία επιπλέον δυνατότητα πέρα από το να τρέξει το εκάστοτε πρόγραμμα. Χρησιμοποιείται συνήθως για να τρέξουν προγράμματα τα οποία έχουν ήδη δοκιμαστεί για σφάλματα.

3.5.2 Βασική Λειτουργία του συστήματος Python

Το σύστημα Python χρησιμοποιεί μεταγλωττιστή για την δημιουργία του εκτελέσιμου κώδικα και σχετίζεται με τις γλώσσες προγραμματισμού Tcl, Perl, Scheme, Java και Ruby, καθώς και με την ABC η οποία υπήρξε η αρχική πηγή έμπνευσης για τη δημιουργία της.

Ένα από τα πιο απλά προγράμματα στην γλώσσα Python είναι η εμφάνιση ενός γραπτού αποτελέσματος (π.χ. Γεια σου, κόσμε!):

```
>>>print("Γεια σου, κόσμε!")
Γεια σου, κόσμε!
```

Ένα ιδιαίτερο χαρακτηριστικό της γλώσσας είναι η χρήση κενών διαστημάτων για τον διαχωρισμό των συντακτικών δομών που προγράμματος, σε αντίθεση με την πρακτική σε άλλες γλώσσες όπου για τον ίδιο σκοπό χρησιμοποιούνται ειδικά σύμβολα (π.χ. αγκύλες). Αυτό, σε συνδυασμό με το ότι χρησιμοποιεί πλήρεις αγγλικές λέξεις στη θέση συμβόλων, καθιστούν τον κώδικα της Python ευανάγνωστο από όσους έχουν βασική γνώση της αγγλικής γλώσσας.

Η Python προορίζεται να είναι μια εξαιρετικά ευανάγνωστη γλώσσα. Είναι σχεδιασμένη για να έχει μια λιτή οπτική διάταξη, συχνά χρησιμοποιώντας αγγλικές λέξεις-κλειδιά , όπου άλλες γλώσσες χρησιμοποιούν σημεία στίξης. Επιπλέον, η Python έχει έναν μικρότερο αριθμό συντακτικών εξαιρέσεων και ειδικών περιπτώσεων από τη C ή τη Pascal. Η Python χρησιμοποιεί κενές εσοχές, αντί για άγκιστρα ή τις λέξεις-κλειδιά για να οριοθετήσει τα μπλοκ. Μία αύξηση στην εσοχή έρχεται μετά από ορισμένες δηλώσεις. Ενώ μια μείωση στην εσοχή σηματοδοτεί το τέλος του τρέχοντος μπλοκ.

Οι εκφράσεις στην Python είναι παρόμοιες με γλώσσες όπως η C και η Java [40].

- Η πρόσθεση, η αφαίρεση, ο πολλαπλασιασμός εκφράζονται με τον ίδιο τρόπο, αλλά η συμπεριφορά της διαίρεσης διαφέρει. Η Python προσθέτει, επίσης, τον φορέα (**) για την ύψωση σε δύναμη .

- Στην Python, == συγκρίνει με την αξία , σε αντίθεση με την Java , όπου συγκρίνει με την αναφορά. Οι συγκρίσεις μπορούν να συνδυαστούν αλυσιδωτά, για παράδειγμα, ένα < = b < = c.

- Η Python χρησιμοποιεί τις λέξεις and, or, not για boolean συναρτήσεις και όχι τη συμβολική && , | | , ! που χρησιμοποιείται σε Java και C.

- Οι εκφράσεις υπό συνθήκη στην Python γράφεται ως x If c else y.

- Η Python κάνει διάκριση μεταξύ των λιστών και των πλειάδων. Οι λίστες γράφονται ως [1, 2, 3], είναι ευμετάβλητες και δεν μπορούν να χρησιμοποιηθούν ως κλειδιά στα λεξικά. Οι πλειάδες γράφονται ως (1, 2, 3), είναι αμετάβλητες και έτσι μπορούν να χρησιμοποιηθούν ως κλειδιά των λεξικών. Οι παρενθέσεις γύρω από το πλειάδα είναι προαιρετικές σε ορισμένα

πλαίσια. Οι πλειάδες εμφανίζονται στην αριστερή πλευρά του συμβόλου ίσο, ως εκ τούτου μια δήλωση, όπως x , y = y , x μπορεί να χρησιμοποιηθεί για να ανταλλάξει δύο μεταβλητές.

Στην Python, η διάκριση μεταξύ εκφράσεων και δηλώσεων είναι σταθερή και τηρείται, σε αντίθεση με γλώσσες όπως η Common Lisp , Scheme , ή Ruby . Το γεγονός αυτό οδηγεί σε κάποια επικάλυψη των λειτουργιών .

Οι μέθοδοι για τα αντικείμενα είναι λειτουργίες που συνδέονται με την κλάση του αντικειμένου. Η Python επιτρέπει στους προγραμματιστές να καθορίσουν τους δικούς τους τύπους τιμών, χρησιμοποιώντας τις κλάσεις. Οι πιο βασικοί τύποι τιμών είναι:

- Συναρτήσεις: Οι συναρτήσεις είναι επαναχρησιμοποιήσιμα μέρη προγραμμάτων. Επιτρέπουν να εκχωρείται ένα όνομα σε ένα σύνολο εντολών και να τρέχει το σύνολο εντολών χρησιμοποιώντας το όνομά του, οπουδήποτε στο πρόγραμμα και όσες φορές επιθυμεί ο χρήστης. Αυτό είναι γνωστό σαν κλήση “calling” της συνάρτησης. Οι συναρτήσεις ορίζονται χρησιμοποιώντας τη λέξη κλειδί <tt>def</tt>, μετά την οποία ακολουθεί ένα όνομα που ταυτοποιεί την εκάστοτε συνάρτηση και κατόπιν ακολουθεί ένα ζευγάρι παρενθέσεων που μπορούν να περικλείουν μερικά ονόματα μεταβλητών, και η γραμμή τελειώνει με διπλή τελεία

- Παράμετροι: Μια συνάρτηση μπορεί να δεχθεί παραμέτρους, οι οποίες είναι τιμές που δίνονται στη συνάρτηση, έτσι ώστε αυτή να μπορεί να πραγματοποιεί κάτι αξιοποιώντας αυτές τις τιμές. Αυτές οι παράμετροι μοιάζουν με τις μεταβλητές, διαφέροντας ως προς το ότι οι τιμές αυτών των μεταβλητών ορίζονται όταν καλούμε τη συνάρτηση και τους έχουν ήδη εκχωρηθεί τιμές όταν τρέχει η συνάρτηση.

  Οι παράμετροι καθορίζονται μέσα στο ζευγάρι των παρενθέσεων στον ορισμό της συνάρτησης και διαχωρίζονται με κόμμα. Όταν δηλώνονται μεταβλητές μέσα σε ένα ορισμό συνάρτησης, αυτές δεν έχουν καμία σχέση με άλλες μεταβλητές που έχουν την ίδια ονομασία και χρησιμοποιούνται έξω από αυτή τη συνάρτηση, δηλαδή τα ονόματα των μεταβλητών χρησιμοποιούνται μόνο τοπικά στη συνάρτηση. Αυτό ονομάζεται εμβέλεια των μεταβλητών. Όλες οι

μεταβλητές έχουν την εμβέλεια του τμήματος όπου έχουν δηλωθεί, αρχίζοντας από το σημείο στο οποίο ορίζεται το όνομα.

Εάν υπάρχουν συναρτήσεις με πολλές παραμέτρους και πρέπει να καθοριστούν με ακρίβεια μόνο μερικές από αυτές ,τότε μπορούν να δοθούν τιμές για τέτοιες παραμέτρους, δηλαδή χρησιμοποιείται η ονομασία αντί της θέσης τους για να καθοριστούν τα ορίσματα στη συνάρτηση. Έτσι υπάρχουν δύο πλεονεκτήματα, πρώτον η χρήση της συνάρτησης γίνεται ευκολότερη επειδή δεν χρειάζεται να ανησυχεί ο χρήστης για τη διάταξη των ορισμάτων. Δεύτερον, δίνονται τιμές μόνο σε εκείνες τις παραμέτρους που θέλει ο χρήστης, προνοώντας ότι οι άλλες παράμετροι έχουν τις προεπιλεγμένες τιμές ορίσματος.

Οι βασικές εντολές της Python περιλαμβάνουν :

- Εντολή if, η οποία εκτελεί υπό όρους ένα μπλοκ του κώδικα , μαζί με τις δηλώσεις else και elif (συρρίκνωση των else-if).

- Η εντολή for, η οποία επαναλαμβάνεται σε ένα αντικείμενο, συλλαμβάνοντας κάθε στοιχείο σε μια τοπική μεταβλητή για χρήση από το συνημμένο μπλοκ.

- Η εντολή while, η οποία εκτελεί ένα μπλοκ κώδικα για όσο ισχύει η συνθήκη.

- Η εντολή try, η οποία επιτρέπει εξαιρέσεις που ανέκυψαν στο συνημμένο μπλοκ κώδικα, να χειρίζονται εξαιρώντας τους όρους, και εξασφαλίζοντας επίσης ότι καθαρίζοντας τον κώδικα σε ένα τελικό μπλοκ πάντα θα τρέχει ο κώδικας ανεξάρτητα από τις εξόδους του μπλοκ.

- Η εντολή class, η οποία εκτελεί ένα μπλοκ του κώδικα και αποδίδει το τοπικό namespace της κλάσης, για χρήση σε αντικειμενοστραφή προγραμματισμό.

- Η εντολή def, η οποία ορίζει μια συνάρτηση ή μια μέθοδο.

- Η εντολή with, η οποία περικλείει ένα μπλοκ κώδικα μέσα σε ένα πλαίσιο διαχείρισης.

- Η εντολή pass, η οποία χρησιμεύει ως NOP. Συντακτικά χρησιμεύει προκειμένου να δημιουργηθεί ένα άδειο μπλοκ κώδικα.

- Η εντολή assert, που χρησιμοποιείται κατά τη διάρκεια του debugging για να ελέγξει τους όρους που θα ισχύουν.Η εντολή import, η οποία χρησιμεύει για την εισαγωγή βιβλιοθηκών.

3.5.3 Αντικείμενα Δεδομένων του συστήματος Python

Τα αντικείμενα δεδομένων μπορούν να κρατήσουν κάποια στοιχεία μαζί. Με άλλα λόγια, χρησιμοποιούνται για να αποθηκευτεί μια συλλογή ίδιων δεδομένων. Υπάρχουν τέσσερις ενσωματωμένες δομές δεδομένων στην Python: οι λίστες, οι πλειάδες, λεξικά και τα σύνολα [40].

- Λίστα : Μια λίστα είναι μια δομή δεδομένων που συγκρατεί μια διατεταγμένη συλλογή στοιχείων. Η λίστα των στοιχείων πρέπει να κλείνεται σε αγκύλες, έτσι ώστε να καταλαβαίνει η Python ότι καθορίζεται μια λίστα. Αφού δημιουργηθεί μια λίστα, μπορούν να προστεθούν, να μετακινηθούν ή να αναζητηθούν στοιχεία σ' αυτή τη λίστα. Από τη στιγμή που μπορούμε να προσθέσουμε και να μετακινήσουμε στοιχεία, λέμε ότι η λίστα είναι ένας μεταβλητός τύπος δεδομένων. Επίσης, η λίστα είναι ένα παράδειγμα χρήσης αντικειμένων και κλάσεων. Όταν χρησιμοποιούμε τη μεταβλητή i και εκχωρούμε σ' αυτήν μια τιμή, ας πούμε τον ακέραιο αριθμό 5, αυτό μπορούμε να το σκεφτούμε σαν τη δημιουργία ενός αντικειμένου (δηλ. υπόστασης (instance)) i της κλάσης (δηλ. τύπου (type)) int.

- Πλειάδα : Οι πλειάδες χρησιμοποιούνται για να συγκρατήσουν μαζί πολλαπλά αντικείμενα. είναι παρόμοιες με τις λίστες, αλλά χωρίς την εκτεταμένη λειτουργικότητα που η κλάση της λίστας δίνει. Ένα κύριο χαρακτηριστικό των πλειάδων είναι ότι είναι αμετάβλητες όπως οι συμβολοσειρές. Οι πλειάδες ορίζονται καθορίζοντας στοιχεία που διαχωρίζονται με κόμματα, μέσα σε ένα προαιρετικό ζευγάρι παρενθέσεων. Επιπλέον, χρησιμοποιούνται συνήθως, στις περιπτώσεις όπου μια εντολή ή μια συνάρτηση οριζόμενη από το χρήστη, μπορεί με ασφάλεια να θεωρήσει ότι η συλλογή των τιμών δηλαδή η πλειάδα των τιμών που χρησιμοποιούνται δε θα αλλάξει. Αν και οι παρενθέσεις είναι

προαιρετικές, προτιμάται να υπάρχουν για να γίνεται φανερό ότι αυτή είναι μια πλειάδα, ειδικά επειδή αποφεύγεται η ασάφεια. Μια άδεια πλειάδα δομείται από ένα άδειο ζευγάρι παρενθέσεων όπως το myempty.

- Λεξικό : Ένα λεξικό είναι σαν ένας τηλεφωνικός κατάλογος όπου μπορεί να βρεθεί η διεύθυνση ή άλλα στοιχεία επικοινωνίας για ένα άτομο, γνωρίζοντας μόνο το όνομά του/της, δηλαδή συσχετίζονται κλειδιά (ονομασία) με τιμές (λεπτομέρειες). Να σημειωθεί ότι το κλειδί πρέπει να είναι μοναδικό, με τον ίδιο τρόπο που δε θα βρεθούν τα σωστά στοιχεία επικοινωνίας κάποιου αν δύο άτομα με το ίδιο όνομα. Επίσης, μπορούν να χρησιμοποιηθούν μόνο αμετάβλητα αντικείμενα (όπως συμβολοσειρές) για τα κλειδιά του λεξικού, αλλά μπορούν να χρησιμοποιηθούν είτε αμετάβλητα είτε μεταβλητά αντικείμενα για τις τιμές του λεξικού. Ζευγάρια κλειδιών και τιμών καθορίζονται στο λεξικό χρησιμοποιώντας το συμβολισμό d = {key1: value1, key2: value2 }. Διαφαίνεται ότι τα ζευγάρια κλειδί-τιμή διαχωρίζονται με διπλή τελεία, και τα ζευγάρια γενικά διαχωρίζονται μεταξύ τους με κόμματα και όλα αυτά περικλείονται σε ένα ζευγάρι άγκιστρων. Υπενθυμίζεται ότι τα ζευγάρια κλειδί-τιμή σε ένα λεξικό δεν ταξινομούνται με κανένα τρόπο. Τα λεξικά που χρησιμοποιούνται είναι υποστάσεις/αντικείμενα της κλάσης dict.

- Ακολουθίες : Οι λίστες, οι πλειάδες και οι συμβολοσειρές είναι παραδείγματα ακολουθιών. Τα κυρίαρχα χαρακτηριστικά είναι ότι έχουν δοκιμές ένταξης (membership tests, δηλ. τις εκφράσεις in και not in) και λειτουργίες ευρετηρίασης “indexing operations”. Η λειτουργία ευρετηρίασης επιτρέπει στο χρήστη να λάβει απ’ ευθείας ένα συγκεκριμένο στοιχείο στην ακολουθία. Οι τρεις τύποι ακολουθιών που αναφέρθηκαν παραπάνω, λίστες, πλειάδες και συμβολοσειρές έχουν επίσης μια λειτουργία τεμαχισμού που επιτρέπει στο χρήστη να ανακτάται ένα μέρος της ακολουθίας. Οποτεδήποτε καθορίζεται ένα νούμερο σε μια ακολουθία μέσα σε αγκύλες η Python θα φέρνει το στοιχείο που αντιστοιχεί σε αυτή τη θέση, στην ακολουθία. Αυτό αναφέρεται επίσης σαν συνδρομητική λειτουργία. Υπενθυμίζεται ότι η Python αρχίζει να μετράει τα νούμερα από το μηδέν.

- Σύνολα : Τα σύνολα είναι μη ταξινομημένες συλλογές απλών αντικειμένων. Αυτά χρησιμοποιούνται όταν η ύπαρξη ενός αντικειμένου σε μια συλλογή είναι πιο σπουδαία από την εντολή ή πόσες φορές αυτή συμβαίνει.

- Παραπομπές : Όταν δημιουργείται ένα αντικείμενο και εκχωρείται σε μια μεταβλητή, η μεταβλητή απλά παραπέμπει στο αντικείμενο και δεν αντιπροσωπεύει αυτό καθ' αυτό το αντικείμενο. Η ονομασία της μεταβλητής δείχνει σε εκείνο το σημείο της μνήμης του υπολογιστή, όπου αποθηκεύεται το αντικείμενο. Αυτό ονομάζεται συσχέτιση της ονομασίας με το αντικείμενο.

3.5.4 Γραφική Απεικόνιση στο σύστημα Python

Η Python διαθέτει ένα μεγάλο αριθμό πακέτων που περιλαμβάνουν χαρακτηριστικά για την κατασκευή γραφημάτων. Ένα από τα πιο σημαντικά καθήκοντα στην ανάλυση των δεδομένων είναι τα σχέδια και οι στατικές ή διαδραστικές απεικονίσεις. Μπορεί να είναι ένα μέρος της διερευνητικής διαδικασίας όπως για παράδειγμα, ο εντοπισμός των ακραίων τιμών, μετασχηματισμοί δεδομένων, ή οικοδόμηση μιας διαδραστικής απεικόνισης για το web χρησιμοποιώντας ένα σύνολο εργαλείων. Η Python έχει πολλά εργαλεία οπτικοποίησης. Αυτά περιλαμβάνονται στα πακέτα: matplotlib , Chaco, PyX, Bokeh.

Κεφάλαιο 4

Λειτουργικότητα – Μελέτη Περίπτωσης - Ευχρηστία

Σε συνέχεια της ολοκλήρωσης της περιγραφής και τη μελέτης των επιλεγμένων συστημάτων, ακολουθεί η σύγκριση τους με βάση τα ποιοτικά χαρακτηριστικά τους, την ευχρηστία και τη λειτουργικότητα τους. Οι συγκρίσεις που διενεργήθηκαν αποκαλύπτουν ορισμένες διαφορές και ομοιότητες μεταξύ των διαφόρων εργαλείων, είτε κατά τη χρήση τους ή τις μεθόδους μηχανικής μάθησης. Παράλληλα, αναλύεται και εξετάζεται βήμα προς βήμα η υλοποίηση μιας μελέτης περίπτωσης κατηγοριοποίησης κειμένου με τη χρήση του αλγόριθμου Naïve Bayes, με σκοπό την αναλυτική περιγραφή των διεργασιών που δύναται να ακολουθήσουν οι χρήστες χρησιμοποιώντας τα τέσσερα συστήματα.

4.1 Λειτουργικότητα των συστημάτων

Γενικά για την υποστήριξη εξόρυξης κειμένου και τη διερευνητική ανάλυση, μια σύγχρονη σουίτα εξόρυξης δεδομένων θα πρέπει να παρέχει μια διεπαφή εύκολη στη χρήση για όλους τους χρήστες είτε αρχάριους ή προχωρημένους, που υποστηρίζεται με μοντέλα απεικονίσεων όπως ιστογράμματα, scatterrplots, απεικονίσεις δέντρων απόφασης κ.α., προσφέρει τυποποιημένες τεχνικές προ-επεξεργασίας δεδομένων και τις πιο αντιπροσωπευτικές τεχνικές ανάλυσης δεδομένων με επίβλεψη ή χωρίς επίβλεψη και τέλος να φιλοξενεί μια εργαλειοθήκη για την αξιολόγηση του μοντέλου (ακρίβεια της Κατηγοριοποίησης, η ευαισθησία, Brier βαθμολογία, και άλλα), που περιλαμβάνει επίσης γραφική ανάλυση των αποτελεσμάτων. Εκτός του ότι τα εργαλεία πρέπει να είναι απλά, πρέπει να είναι και ευέλικτα, επιτρέποντας στους χρήστες να καθορίσουν τη δικά τους μοντέλα για την ανάλυση δεδομένων. Γενικά, τα ανοιχτού κώδικα συστήματα εξόρυξης δεδομένων είναι σχεδόν εξ ορισμού επεκτάσιμα.

Στη συνέχεια εξετάζεται η λειτουργικότητα των συστημάτων μέσω ενός πίνακα που αντικατοπτρίζει την ποικιλία των δυνατοτήτων, των τεχνικών και των μεθοδολογιών για να εφαρμοστεί η εξόρυξη κειμένου. Η λειτουργικότητα του λογισμικού θα βοηθήσει να εκτιμηθεί κατά πόσο τα εξεταζόμενα εργαλεία θα προσαρμοστούν σε διαφορετικές περιοχές εξόρυξης κειμένου. Σε περίπτωση που τα εργαλεία υλοποιούν έναν αλγόριθμο χρησιμοποιείται το σύμβολο (√) και σε περίπτωση που χρειάζεται να εγκατασταθεί επιπλέον πακέτο τότε αναφέρεται ως (Add on) και το όνομα του πακέτου, ενώ όταν δεν υπάρχει καθόλου υλοποίηση χρησιμοποιείται το σύμβολο (-). Θα πρέπει να σημειωθεί εδώ ότι τα δεδομένα στον Πίνακα θα πρέπει να θεωρηθούν προσωρινά, επειδή τα περισσότερα από τα εργαλεία βρίσκονται σε διαρκή κατάσταση αναβάθμισης. Παρ 'όλα αυτά, είναι σημαντικό και χρήσιμο να συνοψιστούν οι δυνατότητές τους, έτσι ώστε οι ενδιαφερόμενοι χρήστες να μπορούν να επιλέξουν το κατάλληλο περιβάλλον για το χειρισμό του προβλήματός τους. Οι αλγόριθμοι και οι τεχνικές που φαίνονται στον Πίνακα επιλέχθηκαν με βάση την σημασία και την παρουσία τους στα περισσότερα από τα εργαλεία.

Λειτουργίες		Rapid Miner	Weka Explorer	R	scikit learn (Python)
Pre-processing	Tokenization	√	√	√	√
	Stemming	√	√	√	√
	Filtering Stopwords	√	√	√	√
	Transform Cases	√	√	√	√
	Punctuation	√	√	√	√
	TF - IDF	√	√	√	√
Bayesian Networks	Naïve Bayes	√	√	**√ (Add on e1071, Rweka)**	√
	Bernouli Naïve Bayes	-	-	-	√
	Multinomial Naïve Bayes	**√ (Add on Weka)**	√	**√ (add on e1071, RWeka)**	√
	AODE	**√ (Add on Weka)**	√	**√ (Add on AnDE)**	-
Decision Tree Learning	ID3	√	√	-	-
	C4.5	√	√	**√ (Add on RWeka)**	-
	CART	**√ (Add on Weka)**	√	**√ (Add on RWeka)**	√
Rules Induction	1Rule	√	√	**√ (Add on RWeka)**	-
	PART	**√ (Add on Weka)**	√	**√ (Add on RWeka)**	-
	RIPPER	√	√	**√ (Add on RWeka)**	-
Lazy Modeling	K-NN	√	√	**√ (Add on class, Rweka)**	√
	LWL	**√ (Add on Weka)**	√	**√ (Add on stats)**	-
Neural Net Training	Neural Net	√	-	**√ (Add on nnet, RSNNS)**	√
	Perceptron	√	√	**√ (Add on monmlp)**	√
Support Vector Modeling	SVM	√	-	**√ (Add on e1071, Rweka)**	√
	Linear SVM	√	-	**√ (Add on kernlab, e1071)**	√
	SMO	√	√	**√ (Add on kernlab)**	-
	LibSVM	√	**√ (Add on LibSVM)**	**√ (Add on LibSVM, e1071)**	√
Hierarchical Clustering	Agglomerative	√	√	**√ (Add on cluster)**	√
	Divisive	√	√	**√ (Add on cluster)**	√

	Birch	-	-	**√ (Add on birch)**	√
Centroid Partition based Clustering	K-means	√	√	√	√
	X-means	√	√	**√ (Add on RWeka)**	-
	K-medoids	√	√	**Add on (cluster)**	-
Distribution based clustering	Expectation Maximization	√	√	**√ (Add on mclust)**	√
Density based Clustering	DBSCAN	√	√	**√ (Add on fpc)**	√
	OPTICS	**√ (Add on Weka)**	√	-	-
Association Rules	A-priori	**√ (Add on Weka)**	√	**√ (Add on Rweka, arules)**	√
	FP-Growth	√	√	-	-
	GSP	√	√	-	-
	ECLAT	-	-	**√ (Add on arules)**	-
	Tertius	**√ (Add on Weka)**	√	**√ (Add on RWeka)**	-
Evaluation Methods and Metrics	Holdout	√	√	√	√
	Cross validation	√	√	√	√
	Class to clusters	-	√	**√ (Add on clusteval)**	-
	Classification accuracy: mean, MSE	√	√	√	√
	Clustering: Internal metrics	-	-	√	√
	TP, FP, FN, TN configuration : matrix, precision, recall	√	√	√	√
	ROC curve, Lifts charts,Cost-Benefit chart	√	√	√	**√ (Roc curve)**
Data Visualisation	Histogramms	√	√	√	√
	Scatter plots	√	√	√	√
	Tree Based models	√	√	√	√
Σύνολο Λειτουργιών		**43**	**41**	**43**	**32**

Πίνακας 2. Λειτουργικότητα των Συστημάτων. Πηγή: [28, 32, 38, 39].

Σύμφωνα με τον παραπάνω Πίνακα 2, εξάγεται όλα τα συστήματα υλοποιούν τις βασικές μεθόδους προ-επεξεργασίας και αξιολόγησης, όπως επίσης τις μετρικές κατηγοριοποίησης αλλά και τι βασικές απεικονίσεις του πεδίου εξόρυξης κειμένου. Ενώ,

αναφορικά με τους αλγορίθμους κατηγοριοποίησης το Rapidminer και το R με την προσθήκη πακέτων υλοποιούν όλους τους επιλεγόμενους αλγορίθμους. Σε αντιδιαστολή με το Weka και scikit learn όπου το πρώτο για υλοποίηση Νευρωνικών Δικτύων υλοποιεί μόνο τον αλγόριθμο Perceptron και δεν έχει υλοποίηση του linear SVM, ενώ το δεύτερο δεν υλοποιεί αλγορίθμους κανόνων εκμάθησης.

Αναφορικά με τους αλγορίθμους συσταδοποίησης το Rapidminer και το Weka δεν υλοποιούν από τους ιεραρχικούς αλγορίθμους το Birch, σε αντίθεση με το R και το scikit learn που δεν υλοποιούν τον OPTICS από την ομάδα των βασιζόμενων στην πυκνότητα αλγορίθμων και επιπλέον το scikit learn δεν υλοποιεί τους X-means και k-medoids (Centroid Partition). Ενώ, παρατηρώντας τη λειτουργία της αξιολόγησης της συσταδοποίησης class to clusters η οποία ορίζει τις προκαθορισμένες κλάσεις στις εξαχθείσες συστάδες, βρίσκεται διαθέσιμη στον πυρήνα του Weka Explorer και στο R με την προσθήκη του πακέτου clusteval. Οι εσωτερικές μετρικές της συσταδοποίησης υπολογίζονται στο R και το scikit learn, ενώ στα άλλα δυο δεν υπάρχουν διαθέσιμες προσθήκες. Στη συνέχεια, παρατηρώντας τις λειτουργίες των κανόνων συσχέτισης παρατηρείται ότι το scikit learn υλοποιεί μόνο τον A-priori αλγόριθμο.

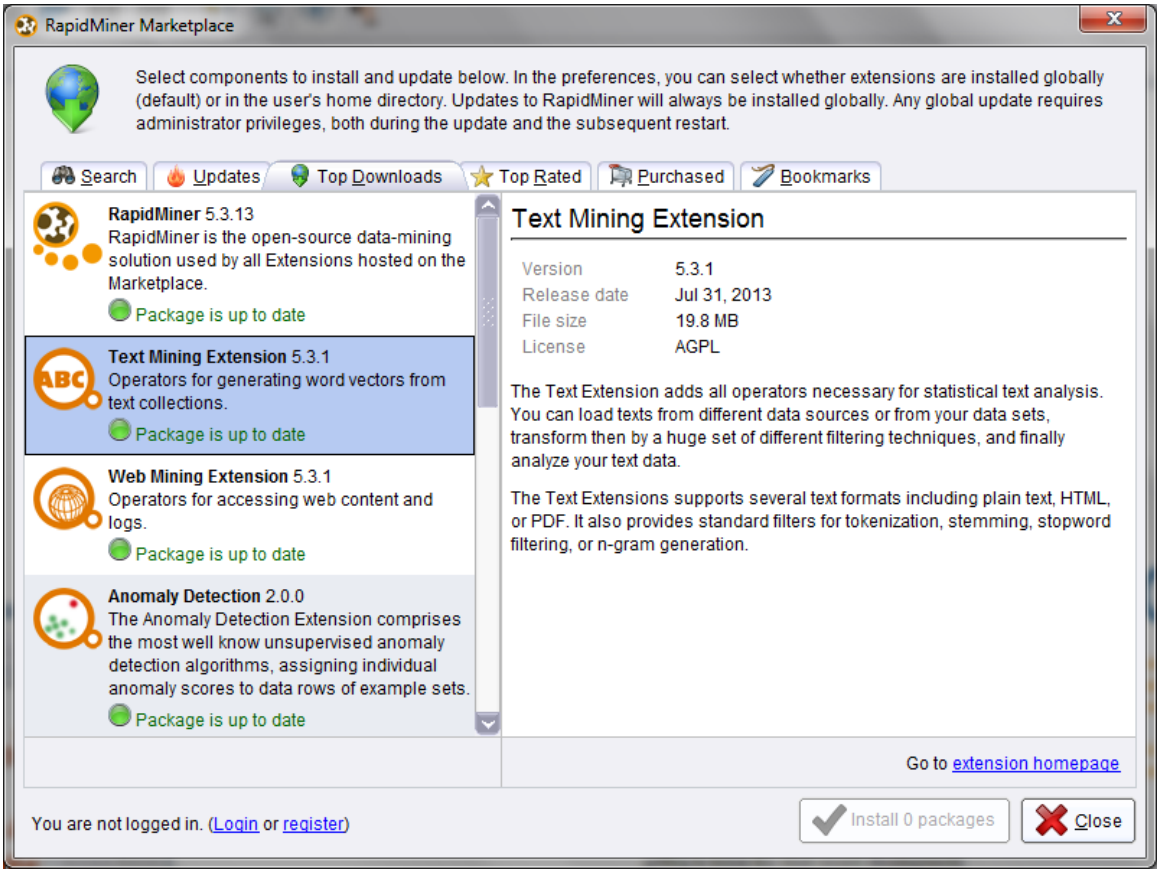
Τέλος, μετρώντας το σύνολο των λειτουργιών που υλοποιούνται και από τα τέσσερα συστήματα το Rapidminer και το R υλοποιούν τις περισσότερες λειτουργίες, ενώ πολύ κοντά ακολουθεί δεύτερο το Weka, ενώ το scikit learn έχει τη χαμηλότερη λειτουργικότητα καλύπτοντας όμως τους χαρακτηριστικούς εκπροσώπους των αλγορίθμων μηχανικής μάθησης, εκτός των κατηγοριοποιητών κανόνων εκμάθησης.

4.2 Μελέτη περίπτωσης κατηγοριοποίησης κειμένου

Ακολούθως, θα υλοποιηθεί ένα παράδειγμα αναλυτικής επεξεργασίας του συνόλου των δεδομένων των κριτικών των ταινιών από το IMDB, που σημάνθηκαν με βάση την ανατροφοδότηση των συγγραφέων, σε αρνητικές και θετικές κριτικές, με εφαρμογή του αλγορίθμου κατηγοριοποίησης Naive Bayes. Επισημαίνονται 1.000 έγγραφα για κάθε κατηγορία (θετικές και αρνητικές κριτικές) και τα δεδομένα παρουσιάζονται σε μορφή απλού κειμένου.

4.2.1 Κατηγοριοποίηση κειμένου με τη βοήθεια του RapidMiner

Για να ξεκινήσει η εξόρυξη κειμένου με τη βοήθεια του συστήματος Rapid Miner, αρχικά πρέπει να εγκατασταθεί το πακέτο "text mining". Γενικά, για να πραγματοποιηθεί η εγκατάσταση των διαθέσιμων πακέτων του Rapid Miner, επιλέγεται από το κεντρικό μενού το "Help" και έπειτα "Updates and Extensions (Marketplace)", και με τη βοήθεια της αναζήτησης επιλέγουμε το πακέτο που μας ενδιαφέρει, ενώ δίπλα αναφέρονται τα χαρακτηριστικά του πακέτου. Στην προκειμένη περίπτωση θα εγκατασταθεί το πακέτο "text mining", το οποίο θα προσθέσει όλους τους απαραίτητους τελεστές για να πραγματοποιηθεί η ανάλυση του κειμένου. Μπορούν να εισαχθούν κείμενα από διαφορετικές πηγές δεδομένων, τα οποία θα αναλυθούν από ένα τεράστιο σύνολο διαφορετικών τεχνικών φιλτραρίσματος. Το "text mining" πακέτο υποστηρίζει διαφορετικές μορφές αρχείων, όπως απλό κείμενο, HTML ή PDF. Επιπλέον, παρέχει βασικά φίλτρα και αλγορίθμους προ-επεξεργασίας.



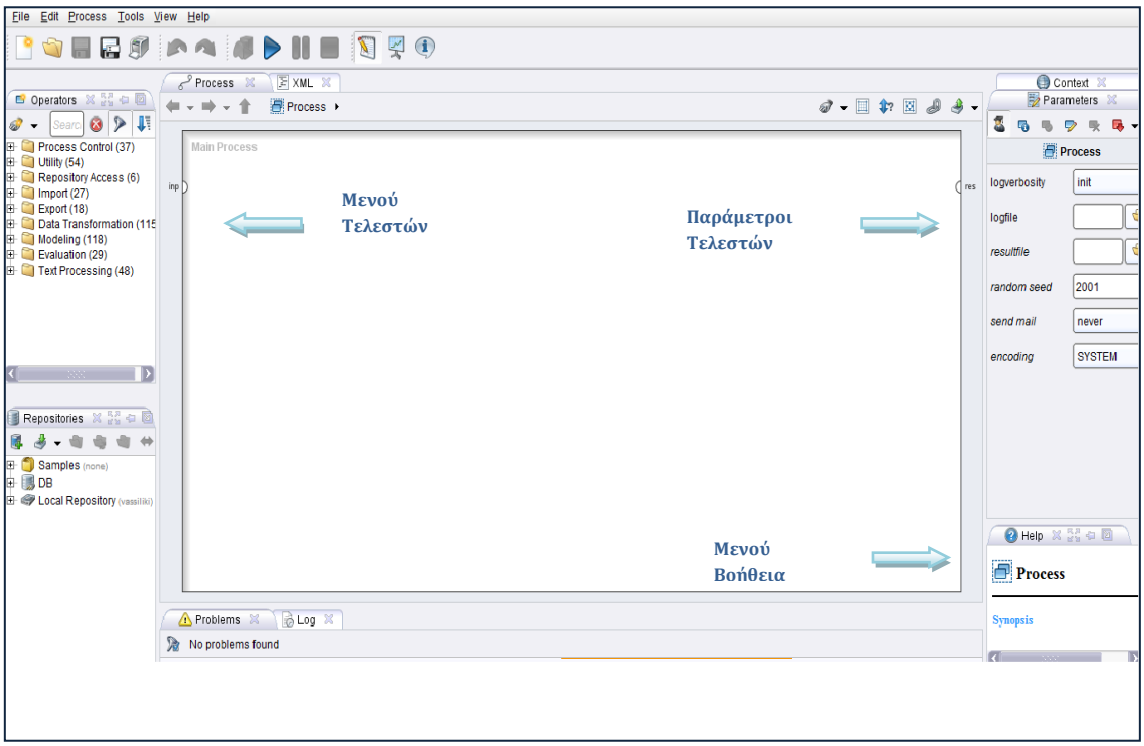
Εικόνα 7. Επιλογή και εγκατάσταση των βιβλιοθηκών μέσω της σελίδας RapidMiner Marketplace.

Για να πραγματοποιηθεί η υλοποίηση του σεναρίου εφαρμογής Κατηγοριοποίησης κειμένου πρέπει να επιλεγούν, να συνδεθούν με την ορθή σειρά, και να εισαχθούν τα

αντίστοιχα δεδομένα στο σύνολο των τελεστών που κρίνονται απαραίτητοι. Σημειώνεται εδώ ότι συνολικά χρειάστηκε η δημιουργία δύο ξεχωριστών διεργασιών. Στην πρώτη εφαρμόζεται η προ-επεξεργασία των δεδομένων και στη δεύτερη δημιουργείται, εκπαιδεύεται και αξιολογείται το μοντέλο αυτόματης Κατηγοριοποίησης.

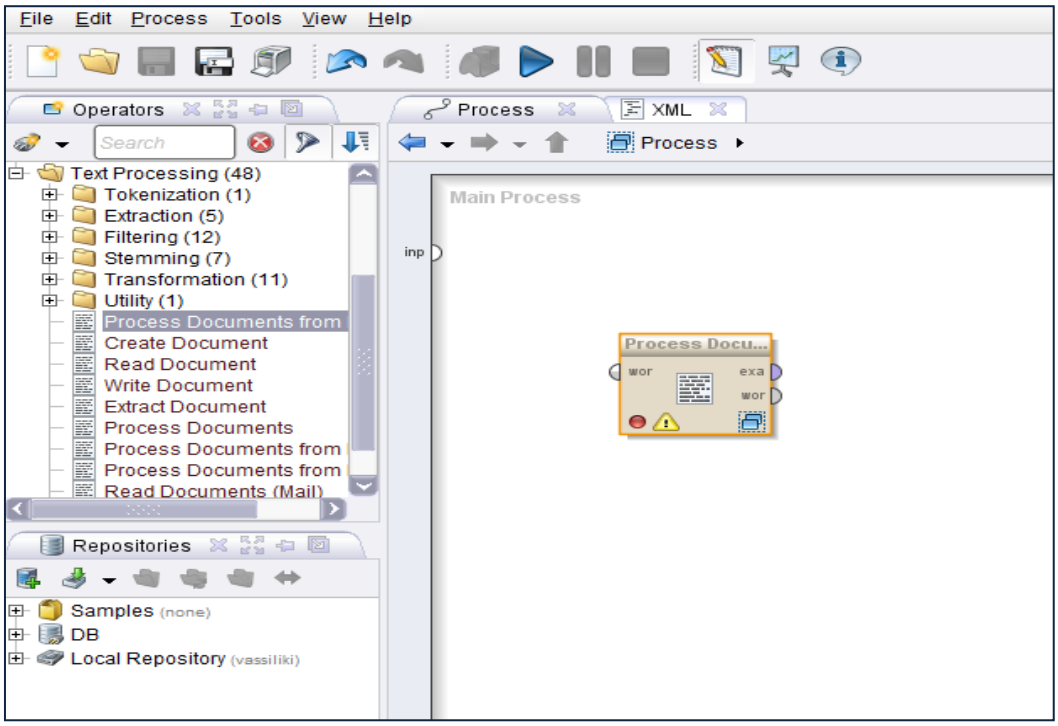
- **1η Διεργασία: Δημιουργία διανυσματικού μοντέλου λέξεων από το σύνολο των δεδομένων και προ-επεξεργασία**

Η πρώτη διεργασία δημιουργεί το διανυσματικό μοντέλο των λέξεων από το σύνολο των δεδομένων, τα οποία αποθηκεύονται στους καταλόγους. Όπως φαίνεται στην παρακάτω Εικόνα 8, η κύρια προβολή του Rapid Miner αποτελείται από τρεις κατακόρυφες επιφάνειες. Ο αριστερός πίνακας περιλαμβάνει ένα κατάλογο των τελεστών και ένα πλαίσιο αναζήτησης για εύκολο φιλτράρισμα. Στον κεντρικό πίνακα δημιουργούνται οι διαδικασίες όπου μπορεί κανείς να καταχωρήσει τους τελεστές. Οι εκάστοτε παράμετροι του χρήστη μπορούν να καθοριστούν στο δεξί πάνελ.



Εικόνα 8. Κύρια Οθόνη του συστήματος Rapid Miner.

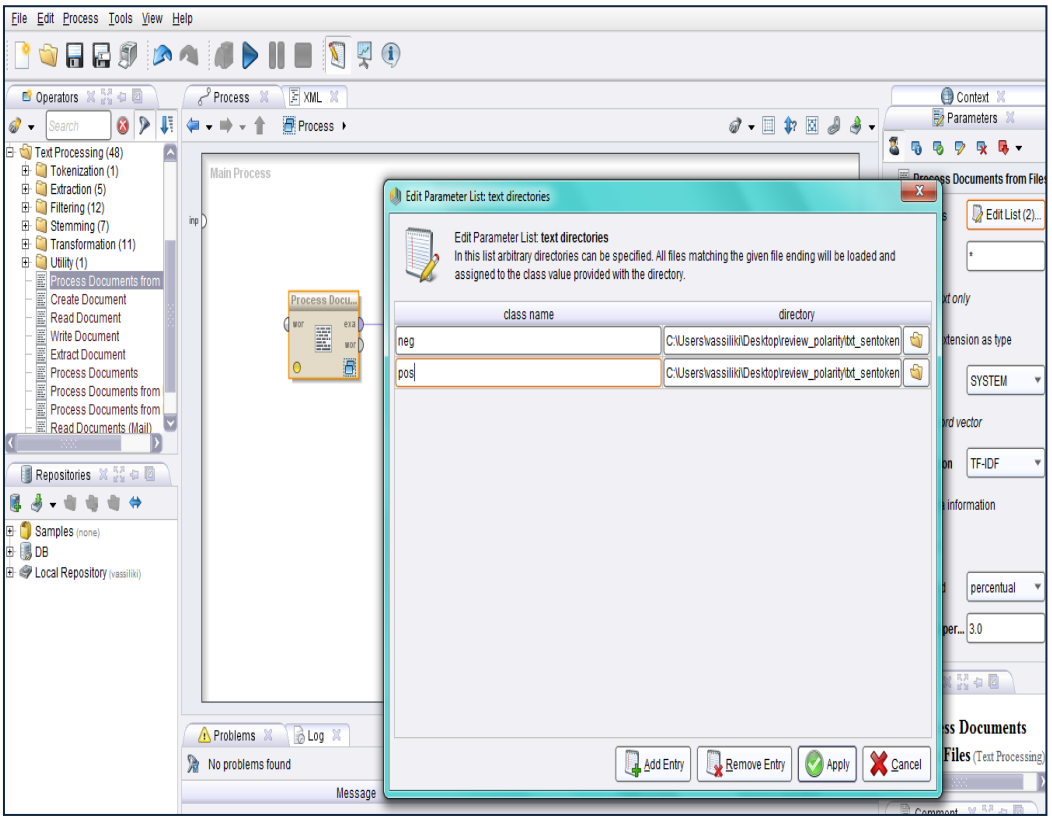
Η διεργασία ξεκινάει φορτώνοντας στον τελεστή “Process Documents from Files” τα σύνολα των δεδομένων. Γενικά, αυτή η διεργασία είναι υπεύθυνη για να διαβάσει τα δεδομένα από τα αρχεία txt και να κάνει όλη την προ-επεξεργασία.



Εικόνα 9. Εισαγωγή Τελεστή "Process Documents from files" στην κύρια διεργασία.

Επιλέγοντας τη διαδικασία, στο δεξί πάνελ των παραμέτρων μπορούν να οριστούν τα εξής:

i. Η τοποθεσία που βρίσκονται τα αρχεία με τα σύνολα δεδομένων προς επεξεργασία. Έτσι λοιπόν στο πεδίο "Text Directories" αποθηκεύονται οι δύο κλάσεις του συνόλου των δεδομένων. Το σύνολο των δεδομένων, που στη δική μας περίπτωση είναι οι θετικές και οι αρνητικές αξιολογήσεις των θεατών, αποτελούν τη βάση για την εκπαίδευση του μοντέλου, γι' αυτό το λόγο οι αξιολογήσεις χωρίζονται σε δύο αρχεία προσθέτοντας την κατάλληλη ετικέτα: positive και negative, προκειμένου να χρησιμοποιηθούν στη συνέχεια κατά την εφαρμογή του κατηγοριοποιητή Naive Bayes.



Εικόνα 10. Οθόνη αποθήκευσης των διευθύνσεων του συνόλου των δεδομένων.

ii. Ο τρόπος που γίνεται η δημιουργία του διανυσματικού μοντέλου. Ο ορισμός των όρων εξαρτάται από την εφαρμογή. Συνήθως, οι όροι είναι μεμονωμένες λέξεις, λέξεις-κλειδιά ή μεγαλύτερες φράσεις. Αν οι λέξεις επιλέγονται να είναι οι όροι, τότε η διάσταση του διανύσματος είναι ο αριθμός των λέξεων στο λεξιλόγιο (ο αριθμός των διακριτών λέξεων που απαντώνται στο σώμα). Υπάρχουν τέσσερις επιλογές:

1. Binary Term Occurrences: Όπου οι όροι λαμβάνουν τη τιμή ένα (1) αν υπάρχουν στο έγγραφο, ή τη τιμή μηδέν (0) αν δεν υπάρχουν.

2. Term Occurrences: Όπου αναφέρονται η αξία είναι ο αριθμός των όρων που απαντώνται στο σύνολο δεδομένων.

3. Term Frequency: Όπου η αξία του βασίζεται στον κανονικό αριθμό των εμφανίσεων του όρου στο έγγραφο.

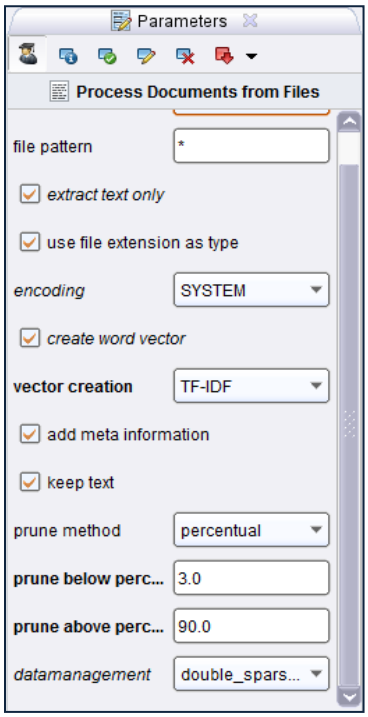
4. TF – IDF: Είναι ένα αριθμητικό στατιστικό στοιχείο που έχει ως στόχο να αντανακλά τη σημαντικότητα μιας λέξης σε ένα έγγραφο σε μια συλλογή δεδομένων. Η αξία tf-idf αυξάνει αναλογικά με τον αριθμό που

μια λέξη εμφανίζεται στο έγγραφο, αλλά αντισταθμίζεται από τη συχνότητα της λέξης στο σώμα, το οποίο βοηθά στον έλεγχο ορισμένων λέξεων που είναι πιο κοινές από ό, τι κάποιες άλλες.

iii. Pruning: Πραγματοποιείται το καθάρισμα των όρων, με την κατάργηση των λιγότερο και των περισσότερο συχνών όρων στο σύνολο των εγγράφων. Από προεπιλογή στο RapidMiner δεν γίνεται pruning. Το καθάρισμα έχει αρνητικό αντίκτυπο λόγω του ότι αυξάνει την πολυπλοκότητα υπολογισμού της διαδικασίας.

iv. Λοιπές άλλες ρυθμίσεις που έχουν να κάνουν κυρίως με τον τρόπο που εισάγονται και κρατούνται τα δεδομένα.

Στο συγκεκριμένο σενάριο θα πραγματοποιήσουμε τη μέθοδο στάθμισης "TF-IDF," ώστε να μειωθεί το βάρος των όρων που απαντώνται πολύ συχνά και αντίστοιχα να αυξήσει το βάρος των όρων που απαντώνται σπάνια. Επίσης, θα χρησιμοποιήσουμε την μέθοδο "pruning"και θα επιλεγούν όλες οι λέξεις με βάρη που είναι κάτω από το 3,00% και πάνω από 90,00%, με την αφαίρεση των λιγότερο και των περισσότερο συχνών όρων στο σύνολο των εγγράφων. Τέλος, για τη διαχείριση των δεδομένων επιλέγεται το "double_sparse_array".



Εικόνα 11. Ορισμός Παραμέτρων Τελεστών.

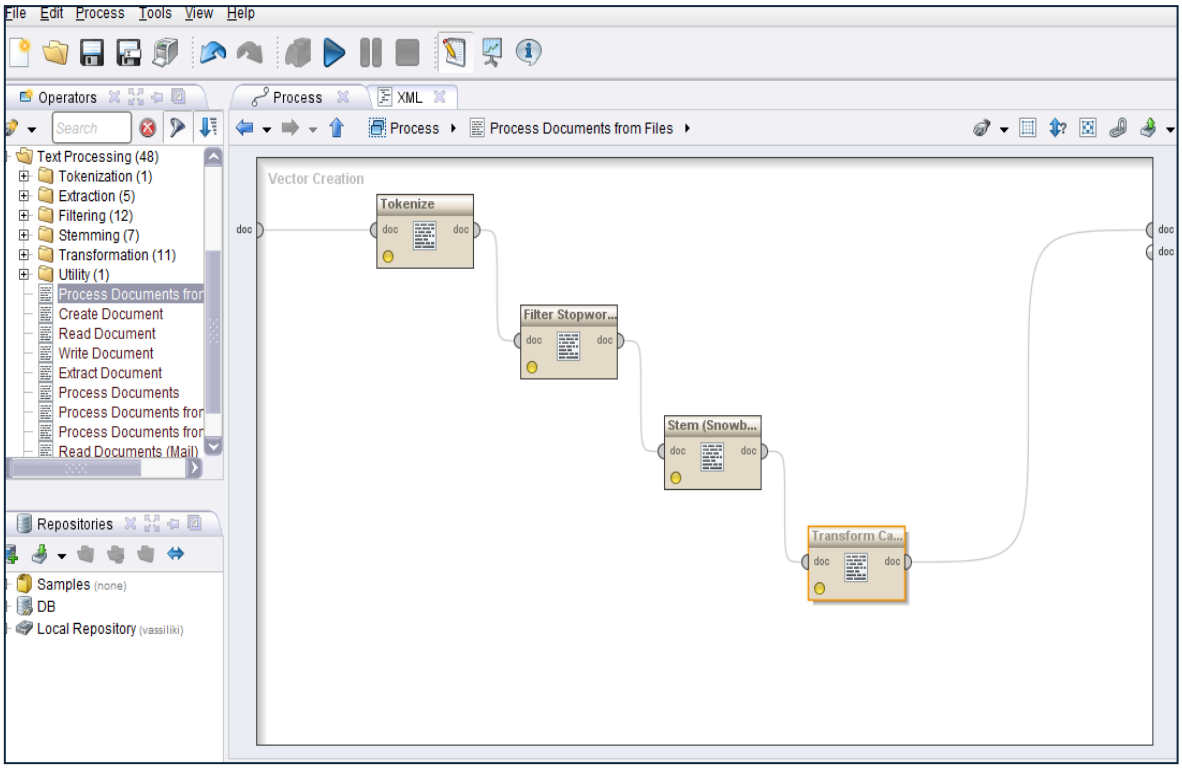
Για να βελτιστοποιηθεί η διεργασία της προ-επεξεργασίας των δεδομένων, οι λέξεις υπόκεινται σε κάποια επεξεργασία, όπως αναλύεται παρακάτω:

- Tokenize: Ο εν λόγω τελεστής χωρίζει το κείμενο ενός εγγράφου σε μια σειρά από ενδείξεις. Μπορούν να χρησιμοποιηθούν όλοι οι χαρακτήρες που δεν είναι γράμματα, για παράδειγμα τα σημεία στίξης, που είναι και η προεπιλεγμένη ρύθμιση. Αυτό θα οδηγήσει σε ενδείξεις που αποτελούνται από μία μόνο λέξη, και είναι η πιο κατάλληλη επιλογή, την οποία θα χρησιμοποιήσουμε στο μελέτη περίπτωσης.

- English Stop words: Ο εν λόγω τελεστής αφαιρεί πολύ κοινές λέξεις οι οποίες δεν έχουν νοηματική αξία (άρθρα, αντωνυμίες, και άλλα), από την ενσωματωμένη “stopword”λίστα.

- Transform cases lower case: Ο εν λόγω τελεστής μετατρέπει όλους τους χαρακτήρες σε ένα έγγραφο είτε σε πεζά είτε σε κεφαλαία, στη συγκεκριμένη περίπτωση επιλέχθηκε να μετατραπούν όλοι οι χαρακτήρες σε πεζά.

- Snowball Stemmer: Ο εν λόγω τελεστής ανακαλύπτει τις ρίζες των λέξεων. Το στέλεχος της λέξης δεν χρειάζεται να είναι ταυτόσημο με τη μορφολογική ρίζα της λέξης.



Εικόνα 12. Επιλογή Τεχνικών Προ-επεξεργασίας Δεδομένων.

- **2η Διεργασία: Εφαρμογή και Αξιολόγηση Μοντέλου**

Η δεύτερη διεργασία αναφέρεται στον τελεστή ο οποίος ανήκει στην ομάδα "Evaluation" και ονομάζεται "Split-Validation". Ο συγκεκριμένος τελεστής εκτιμά την απόδοση του μοντέλου που θα εκπαιδευτεί και θα εφαρμοστεί στη συνέχεια, στα μη κατηγοριοποιημένα δεδομένα.

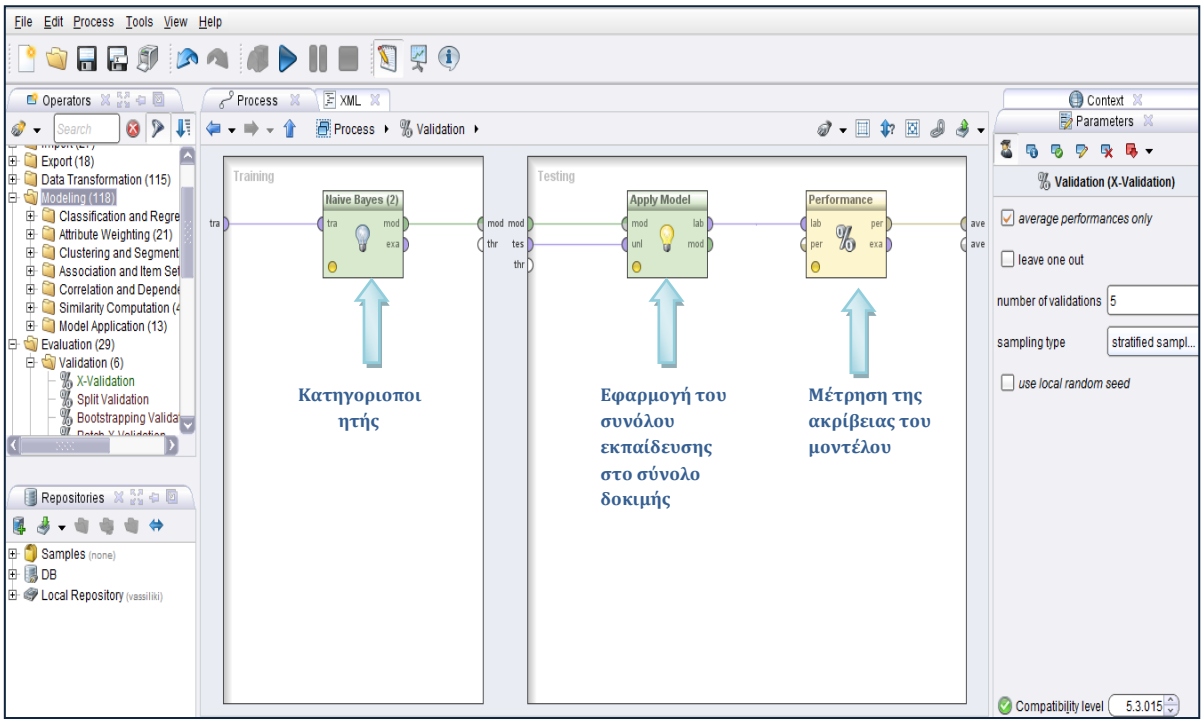
Ο τελεστής "Split-Validation" περιέχει εσωτερικά μία υποδιεργασία, που αποτελείται από δύο φάσεις, τη φάση της εκπαίδευσης του μοντέλου και τη φάση της εφαρμογής αυτού στο σύνολο των δεδομένων. Η υποδιεργασία επιστρέφει το ζητούμενο μοντέλο το οποίο εκπαιδεύεται μέσω των δεδομένων που εισήχθησαν αρχικά στη διεργασία, εν προκειμένω τα αρχεία που περιέχουν τις θετικές και αρνητικές αξιολογήσεις, όπως αυτές έχουν συγκεντρωθεί. Το σύνολο των δεδομένων εισόδου επιλέχθηκε να διαχωριστεί σε πέντε υποσύνολα του ίδιου μεγέθους. Από τα πέντε υποσύνολα μόνο ένα υποσύνολο διατηρείται ως το σύνολο δεδομένων δοκιμής και τα υπόλοιπα τέσσερα

χρησιμοποιούνται στο σύνολο δεδομένων εκπαίδευσης. Αυτή η διαδικασία κατόπιν επαναλαμβάνεται πέντε φορές και κάθε ένα από τα υποσύνολα χρησιμοποιείται ακριβώς μια φορά, όπως τα δεδομένα δοκιμών. Τα αποτελέσματα προέρχονται από τις πέντε επαναλήψεις της παραπάνω διαδικασίας, και στη συνέχεια, μπορεί να υπολογιστεί η μέση τιμή αυτών, προκειμένου να παραχθεί μια ενιαία εκτίμηση. Επιπλέον, στην εν λόγω μελέτη περίπτωσης επιλέχθηκε η μέθοδος "stratisfied sampling". Η διαστρωμάτωση είναι η διαδικασία της διαίρεσης των μελών του συνόλου των δεδομένων σε ομογενείς υποομάδες πριν από τη δειγματοληψία.

Για να πραγματοποιηθεί η εκπαίδευση του συνόλου των δεδομένων του σεναρίου εφαρμογής θα χρησιμοποιηθεί ο αλγόριθμος Naive Bayes, ο οποίος αποτελεί ένα μοντέλο που βασίζεται στην αποτίμηση πιθανότητας ώστε να παράγει την ορθότερη πρόβλεψη, και επιπλέον συνιστά μία απλή και ιδιαιτέρως δημοφιλή μέθοδο μηχανικής μάθησης που δύναται να εφαρμοστεί και στο πεδίο κατηγοριοποίησης φυσικής γλώσσας.

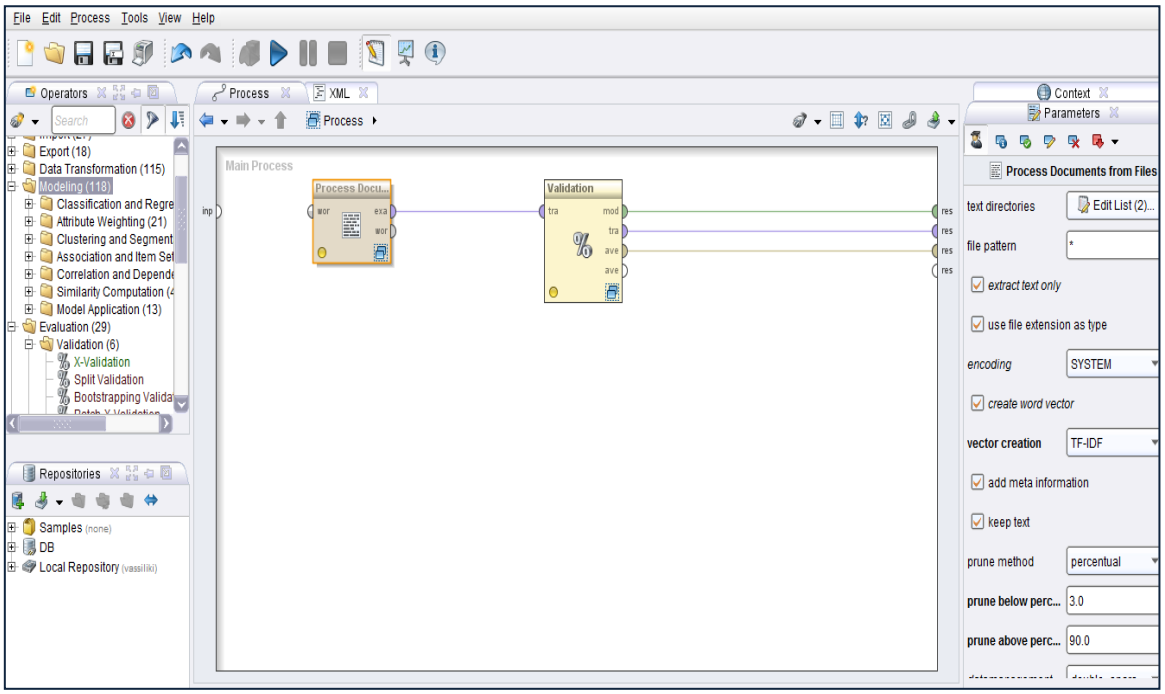
Στη συνέχεια, χρησιμοποιείται ο τελεστής "Apply Model" από την ομάδα "Modeling - Model Application", και ουσιαστικά εφαρμόζεται το μοντέλο που δημιουργήθηκε μέσω του αλγορίθμου Naive Bayes στο σύνολο των δεδομένων της μελέτης περίπτωσης. Ο αλγόριθμος Naive Bayes είναι αυτός που περιέχει τις απαραίτητες πληροφορίες και τα δεδομένα με τα οποία πραγματοποιήθηκε η εκπαίδευση, και θα χρησιμοποιηθεί αργότερα για την πρόβλεψη Κατηγοριοποίησης, της νέας πληροφορίας που θα εισαχθεί, σε θετική ή αρνητική.

Ο τελευταίος τελεστής αυτής της υποδιεργασίας είναι ο "Performance" που βρίσκεται αντίστοιχα στην ομάδα "Evaluation - Performance Measurement" και υπολογίζει μία πρώτη εκτίμηση της ακρίβειας της πρόβλεψης του μοντέλου.



Εικόνα 13. Κατηγοριοποίηση του συνόλου των δεδομένων και μέτρηση της ακριβείας του μοντέλου.

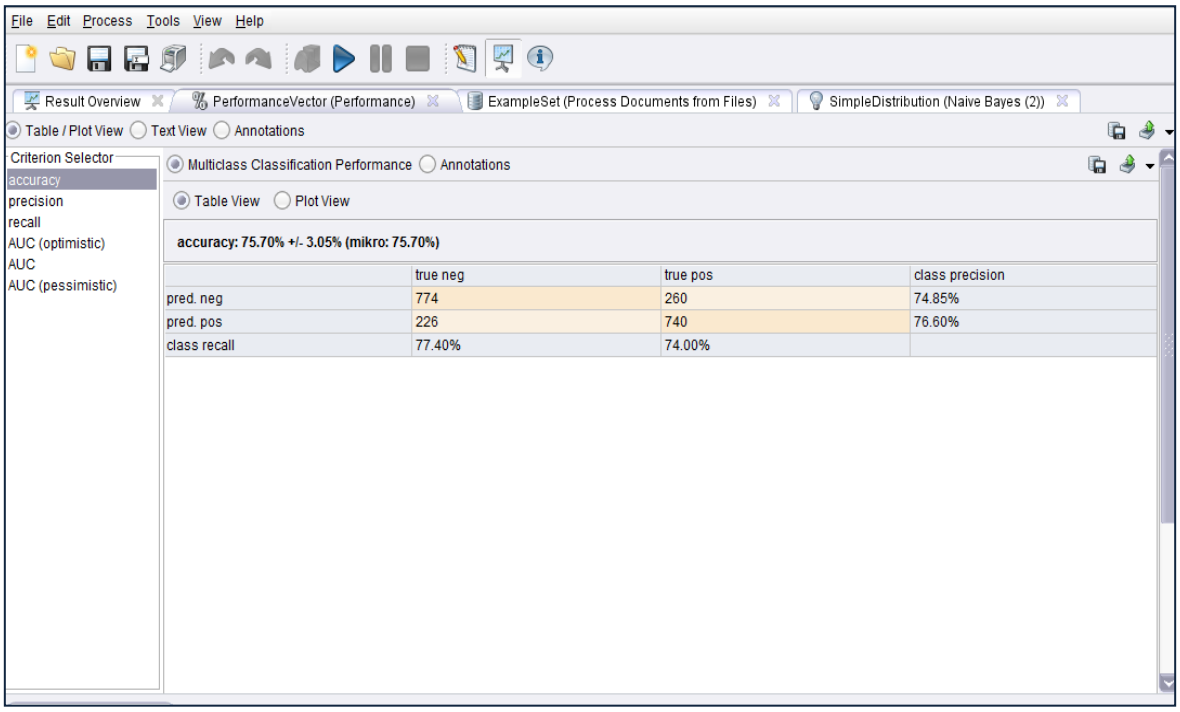
Η διεργασία λοιπόν, έφτασε στο τέλος με τη δημιουργία, εκπαίδευση και τελικά την αποθήκευση του κατηγοριοποιητή. Η ολοκληρωμένη διαδικασία φαίνεται στην Εικόνα 15 που ακολουθεί.



Εικόνα 14. Ολοκληρωμένη διαδικασία Κατηγοριοποίησης του συνόλου των δεδομένων με τον αλγόριθμο Naive Bayes.

Τέλος, έχει ολοκληρωθεί η κατασκευή των δύο διεργασιών μέσω του Rapid Miner ώστε να προβλεφθεί το συναίσθημα θετικό ή αρνητικό που εκφράζεται στα σχόλια που θα ταξινομηθούν αυτόματα.

Στη συνέχεια επιλέγεται το κουμπί αναπαραγωγής για να τρέξει η διαδικασία και να ανοίξει το παράθυρο με τα αποτελέσματα, όπως φαίνεται στην Εικόνα 16.



Εικόνα 15. Πίνακας αποτελεσμάτων, "Confusion Matrix".

Η διαδικασία χρειάστηκε 43 δευτερόλεπτα για να εμφανίσει τα αποτελέσματα. Σύμφωνα με τη πίνακας των αποτελεσμάτων, το ποσοστό ακριβείας της Κατηγοριοποίησης του σεναρίου εφαρμογής, ανέρχεται στο 75,70%, το οποίο είναι ένα πολύ ικανοποιητικό ποσοστό. Τέλος, δίνεται η δυνατότητα να αποθηκευθούν οι διεργασίες αυτές ώστε να μπορούν να χρησιμοποιηθούν μελλοντικά οποιαδήποτε στιγμή.

4.2.2 Κατηγοριοποίηση κειμένου με τη βοήθεια του Weka

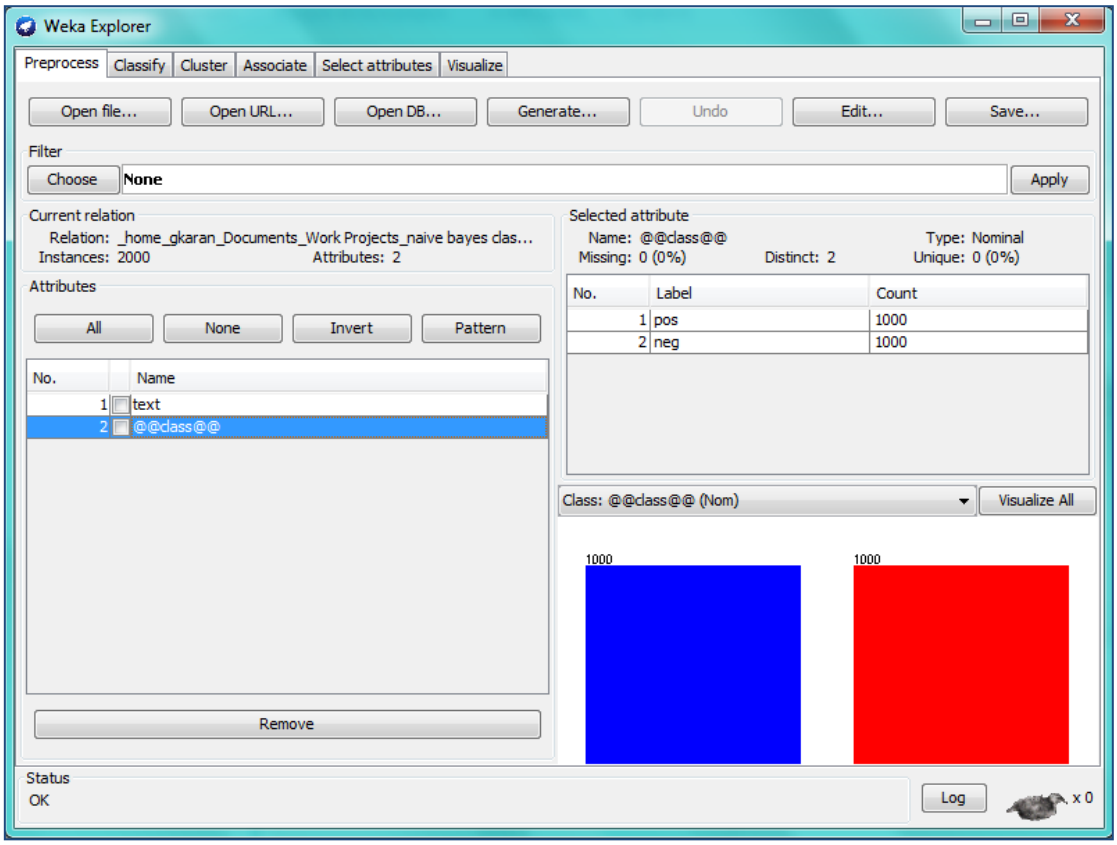
Για να πραγματοποιηθεί η υλοποίηση του σεναρίου εφαρμογής Κατηγοριοποίησης κειμένου πρέπει να επιλεγούν, να συνδεθούν με την ορθή σειρά, και να εισαχθούν τα αντίστοιχα δεδομένα στο σύνολο των τελεστών που κρίνονται απαραίτητοι. Σημειώνεται εδώ ότι συνολικά χρειάστηκε η δημιουργία τριών ξεχωριστών διεργασιών. Στην πρώτη δημιουργείται το αρχείο arff για το σύνολο των δεδομένων, στη δεύτερη εισάγονται οι τελεστές που θα χρησιμεύσουν στην προ-επεξεργασία των δεδομένων και τη βελτίωση του συνόλου των δεδομένων και στη τρίτη διεργασία εκπαιδεύεται το μοντέλο των δεδομένων και πραγματοποιείται Κατηγοριοποίηση του με τον αλγόριθμο Naive Bayes.

- **1η Διεργασία: Δημιουργία διανυσματικού μοντέλου λέξεων από το σύνολο των δεδομένων και Διαδικασία Προ-επεξεργασίας**

Όπως αναφέρθηκε παραπάνω, τα σύνολα δεδομένων με τα οποία τροφοδοτείται το Weka βρίσκονται σε συγκεκριμένη μορφή και πρέπει να έχουν κατάληξη .arff. Οπότε, προκειμένου να τροποποιηθούν τα 2.000 αρχεία txt του συνόλου των δεδομένων, χρησιμοποιείται ο τελεστής TextDirectoryLoader.

Ο τελεστής TextDirectoryLoader εισάγει τα έγγραφα στο Weka και χρησιμοποιεί τους επισημασμένους υποφακέλους (neg και pos) προκειμένου να διατηρήσει τις θετικές και αρνητικές κατηγορίες. Επίσης, εξοικονομεί χρόνο προς το χρήστη διότι προσθέτει στα δεδομένα μια πιο συμβατή σύνταξη ώστε να χρησιμοποιηθούν από το Weka. Η τροφοδότηση του συστήματος με αρχεία arff είναι η καλύτερη επιλογή διότι τα εν λόγω αρχεία είναι λιγότερο απαιτητικά σε μνήμη, ταχύτερα και με καλύτερη ανάλυση επειδή περιλαμβάνουν μεταδεδομένα σχετικά με τις κεφαλίδες των στηλών.

Έτσι λοιπόν, επιλέγοντας τη διεπαφή Explorer και έπειτα “Open File”, διαλέγουμε το αρχείο IMDB.arff που δημιουργήθηκε προ ολίγου, φορτώνοντας το σύνολο των δεδομένων στο σύστημα.



Εικόνα 16. Εισαγωγή συνόλου δεδομένων στο Weka.

Κατά τη δεύτερη διεργασία χρησιμοποιείται το φίλτρο StringToVector, όπου θα μετατρέψει το τυχαίο κείμενο σε σύνολο γνωρισμάτων, προκειμένου να ξεκινήσει η επιλογή των χαρακτηριστικών. Πιο συγκεκριμένα, μετατρέπει τα έγγραφα σε σειρές, τις λέξεις σε στήλες και το κείμενο σε αριθμούς. Κάθε λειτουργία του τελεστή έχει σχεδιαστεί για την εκμάθηση του κατηγοριοποιητή προκειμένου να διακρίνονται οι θετικές από τις αρνητικές αξιολογήσεις.

Ορίζοντας τα επιμέρους χαρακτηριστικά του εν λόγω φίλτρου, αφαιρούνται τα αλφαριθμητικά δεδομένα που έχουν οριστεί και παραμένουν οι λέξεις ως αριθμητικές τιμές. Επιπλέον, τα κατηγορικά χαρακτηριστικά ορίζουν ετικέτες σε κάθε έγγραφο, ως αρνητικές ή θετικές. Παρακάτω αναλύονται τα επιμέρους χαρακτηριστικά του φίλτρου StringToVector, προκειμένου να υλοποιηθεί η προ-επεξεργασία του συνόλου των δεδομένων.

- “WordsToKeep” : Αναφέρεται στην επιλογή αριθμού λέξεων που θέλει ο χρήστης να κρατήσει ανά κλάση, θεωρώντας ότι οι περισσότερες λέξεις δεν είναι πολύτιμες για την εκπαίδευση του μοντέλου. Στην προκειμένη περίπτωση, επιλέγεται η τιμή χίλια, η οποία είναι και η αρχική τιμή που ορίζει

το σύστημα. Έτσι λοιπόν, επιλέγονται χίλιες λέξεις ανά κλάση, που σημαίνει χίλιες λέξεις από τις θετικές και χίλιες από τις αρνητικές κριτικές.

- "OutputWordCounts" : Αναφέρεται στην υπόδειξη της συχνότητας μια λέξης στο έγγραφο.

- "DoNotOperateOnPerClassBasis" : Αναφέρεται στο μέγιστο αριθμό των λέξεων και στην ελάχιστη συχνότητα των όρων όπου δεν επιβάλλονται ανά κλάση, αλλά με βάση το έγγραφο σε όλες τις κλάσεις.

- " IDFTransform" και "TFTransform" : Αναφέρεται στο πόσο συχνά ένας όρος βρίσκεται σε ένα έγγραφο, και όχι αν ένας όρος απλά βρίσκεται στο έγγραφο. Η αξία των λέξεων είναι η TF-IDF βαθμολογία τους. Στην προκειμένη περίπτωση έχει επιλεγεί True και στους δυο τελεστές.

- "NormalizeDocLength" : Αναφέρεται σε μετρήσεις όπως πόσο συχνά βρίσκεται μια λέξη σε ένα έγγραφο, όπου μπορεί να επανεμετρηθεί με βάση το μήκος του εγγράφου. Αν μετρηθεί η συχνότητα του όρου μιας λέξης, αφού ομαλοποιηθεί από το μήκος του εγγράφου οι τιμές θα είναι καλύτερες, εφόσον το έγγραφο θα είναι πιο σύντομο.

- " Stemmer" : Αναφέρεται στην προσπάθεια να χρησιμοποιηθούν οι λέξεις καλύτερα, με την ανάλυσή τους σε μια μικρότερη μορφή με βάση τη ρίζα των λέξεων. Επιλέγεται η τιμή Snowball Stemmer.

- "Stopwords" : Αναφέρεται στην επιλογή ενός λεξιλογίου όπου με την εφαρμογή του θα αφαιρούνται οι λέξεις που δεν έχουν καμία αξία, όπως αντωνυμίες, άρθρα και λοιπά. Παρέχεται η δυνατότητα προσαρμογής του λεξιλογίου με βάση τις ανάγκες του χρήστη, προσθέτοντας ή αφαιρώντας όρους στο εκάστοτε λεξιλόγιο. Επιλέγεται η τιμή True.

- " Tokenizer" : Διαχωρίζει τα δεδομένα σε ενδείξεις. Στην προκειμένη περίπτωση επιλέγεται ο Alphabetic Tokenizer, ο οποίος κρατάει μόνο τις ενδείξεις που εμπεριέχονται στο αλφάβητο.

- " MinTermFreq" : Ορίζει την ελάχιστη συχνότητα των όρων. Στην προκειμένη περίπτωση επιλέχθηκε το τρία, που σημαίνει ότι οι λέξεις θα πρέπει να εμφανίζονται τρεις φορές και άνω για να θεωρούνται ως χαρακτηριστικά του συνόλου των δεδομένων.

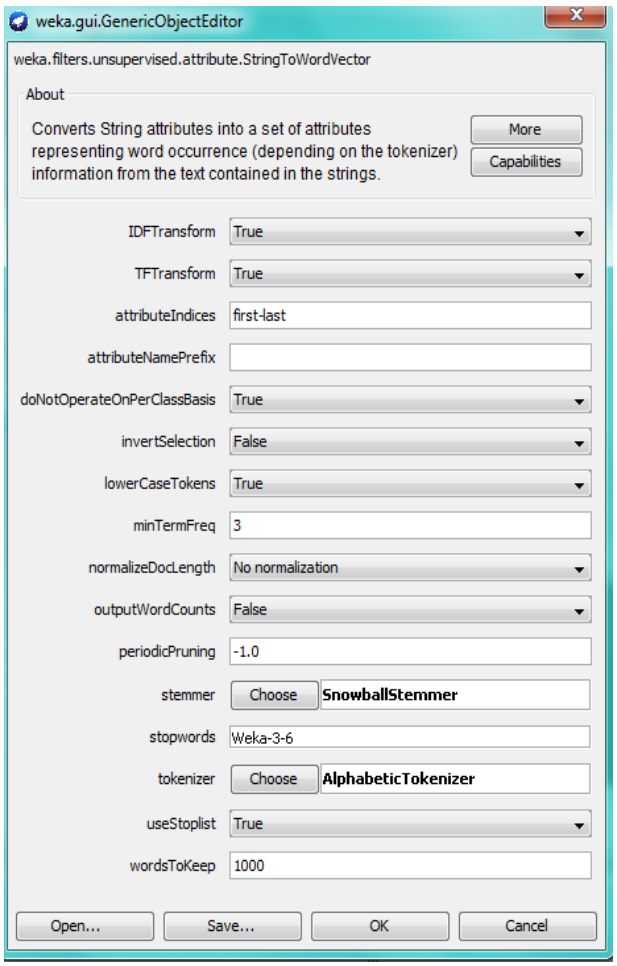
- " PeriodicPruning" : Αφαιρεί λέξεις χαμηλής συχνότητας. Συνήθως, χρησιμοποιείται όταν επεξεργάζονται μεγάλα σύνολα δεδομένων και η μνήμη του υπολογιστή μπορεί να μην είναι σε θέση να καθαρίσει το μεγάλο όγκο των δεδομένων μονομιάς.

- "AttributeNamePrefix" : Αλλάζει το όνομα των χαρακτηριστικών για λόγους οργάνωσης.

- "LowerCaseTokens" : Μετατρέπει τα κεφαλαία γράμματα που ενδέχεται να απαντώνται στο σύνολο των δεδομένων, σε πεζά.

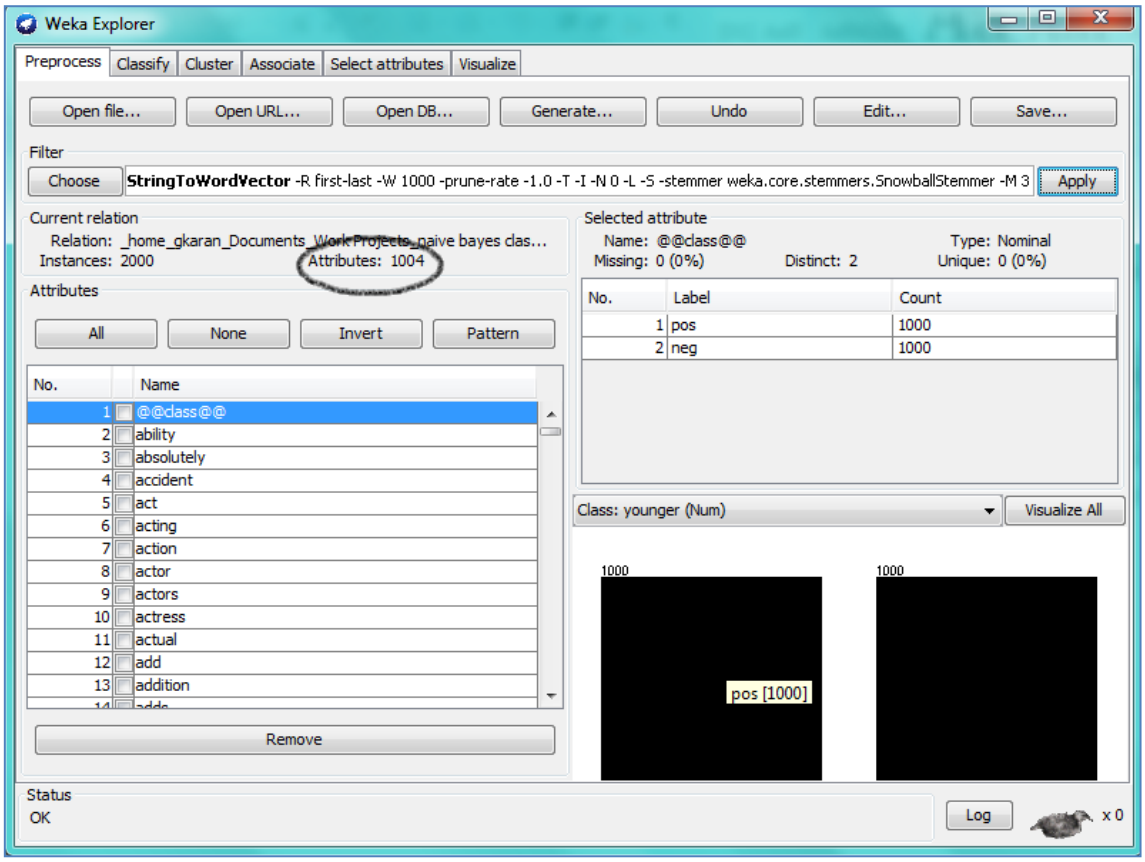
- "AttributeIndices" : Προσδιορίζει μια σειρά από χαρακτηριστικά γνωρίσματα για να ενεργήσει.

Οι επιλογές των τελεστών κατά τη διαδικασία της προ-επεξεργασίας του συνόλου των δεδομένων έχουν οριστεί όπως φαίνεται στην παρακάτω Εικόνα.



Εικόνα 17. Επιλογές τελεστών προ-επεξεργασίας των δεδομένων.

Στην παρακάτω Εικόνα, φαίνεται το σύνολο των χαρακτηριστικών που κρατούνται μετά την ολοκλήρωση τη διαδικασίας της προ-επεξεργασίας των δεδομένων.

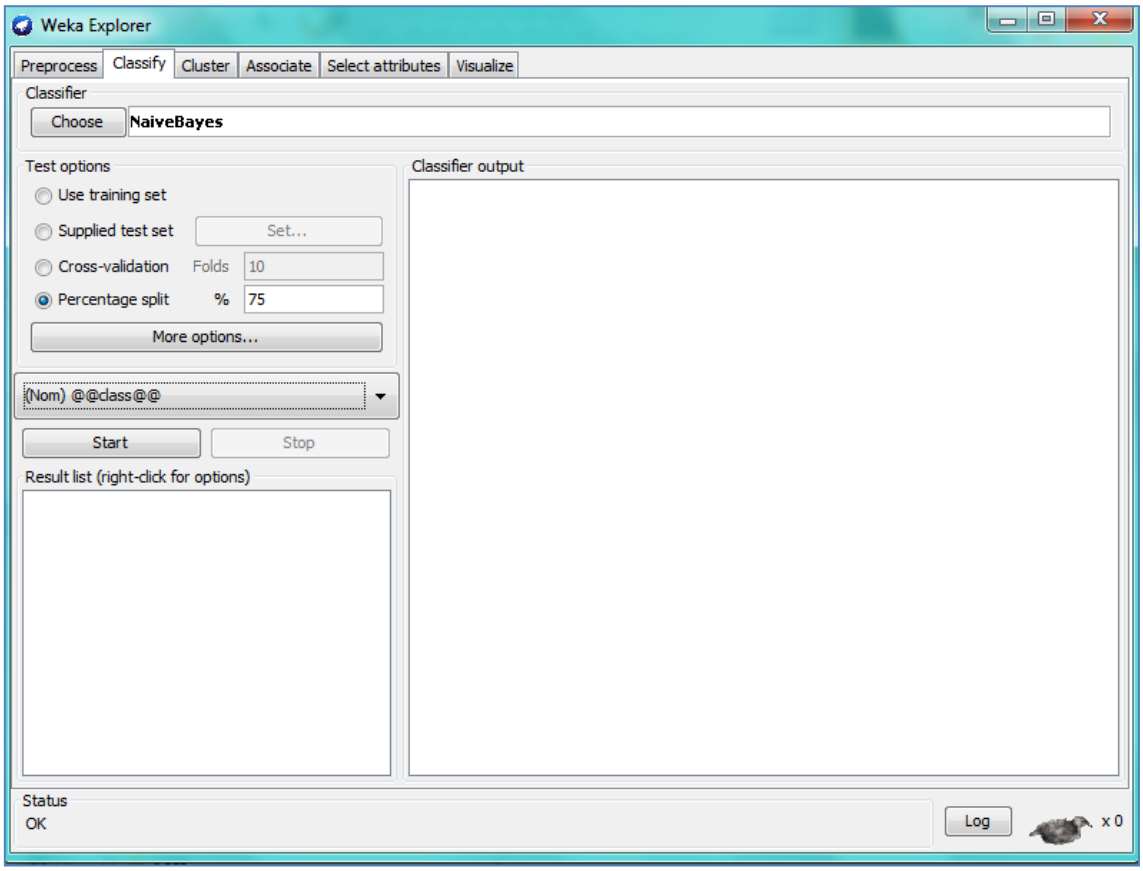


Εικόνα 18. Σύνολο χαρακτηριστικών μετά την εφαρμογή της προ-επεξεργασίας.

- **2η Διεργασία : Υλοποίηση Κατηγοριοποίησης Naïve Bayes στο σύνολο των δεδομένων εκπαίδευσης**

Στη συνέχεια, προκειμένου να ολοκληρωθεί η εφαρμογή του κατηγοριοποιητή Naive Bayes στο σύνολο των προ-επεξεργασμένων δεδομένων, επιλέγεται η καρτέλα "Classify", και ακολούθως από το μενού επιλογής των ταξινομητών, ο κατηγοριοποιητής Naive Bayes.

Επιπλέον, προκειμένου να δημιουργηθεί το προβλεπτικό μοντέλο και έπειτα με το σύνολο δεδομένων δοκιμής να αξιολογηθεί η αντοχή και η χρησιμότητα της πρόβλεψης και να προσομοιώσει την πρόβλεψη των αξιολογήσεων στο μέλλον, δηλώνεται το ποσοστό διαχωρισμού του συνόλου των προ-επεξεργασμένων δεδομένων στο σύνολο των δεδομένων εκπαίδευσης σε 75%, ενώ το εναπομείναντα ποσοστό περιλαμβάνει το σύνολο των δεδομένων δοκιμής.



Εικόνα 19. Επιλογή κατηγοριοποιητή Naive Bayes από την καρτέλα Classify.

Εφόσον, ολοκληρωθεί η διαδικασία Κατηγοριοποίησης των δεδομένων με τον αλγόριθμο Naive Bayes, απεικονίζονται τα αποτελέσματα όπως φαίνονται στις Εικόνες 64 και 65. Αρχικά, ο χρόνος που χρειάστηκε για να δημιουργηθεί το προβλεπτικό μοντέλο ήταν 2,16 δευτερόλεπτα, κατά την αξιολόγηση του συνόλου δεδομένων δομικής 500 ενδείξεων, το 76% των δεδομένων ταξινομήθηκε σωστά και το 24% λάθος.

```
Time taken to build model: 2.16 seconds

=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances         380               76      %
Incorrectly Classified Instances       120               24      %
Kappa statistic                          0.5197
Mean absolute error                      0.2444
Root mean squared error                  0.4631
Relative absolute error                 48.8727 %
Root relative squared error             92.6113 %
Total Number of Instances              500
```

Εικόνα 20. Περίληψη αξιολόγησης μοντέλου κατηγοριοποιητή Naive Bayes.

Ακολούθως αναλύονται τα αποτελέσματα αξιολόγησης δημιουργώντας τη πίνακας αποτελεσμάτων "Confusion Matrix", απεικονίζοντας τους σταθμισμένους μέσους όρους των τιμών "Precision", "Recall" και " F- Measure", με τα αντίστοιχα ποσοστά 76,50 %, 76,00 % και 75,90 %. Τα ποσοστά ακρίβειας του μοντέλου που παρουσιάστηκε είναι αρκετά ικανοποιητικά.

```
=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.829     0.309      0.73      0.829     0.776      0.844    pos
                 0.691     0.171      0.8       0.691     0.741      0.844    neg
Weighted Avg.    0.76      0.241      0.765     0.76      0.759      0.844

=== Confusion Matrix ===

   a   b   <-- classified as
 208  43 |   a = pos
  77 172 |   b = neg
```

Εικόνα 21. Ανάλυση σταθμισμένων μέσων όρων τιμών "Precision", "Recall" και " F- Measure", και απεικόνιση της πίνακαςς αποτελεσμάτων.

4.2.3 Κατηγοριοποίηση κειμένου με τη βοήθεια του R

Η εξόρυξη κειμένου είναι ένα ερευνητικό πεδίο που χρησιμοποιεί τεχνικές από την επιστήμη των υπολογιστών, της γλωσσολογίας, και τη στατιστική. Ωστόσο, μέχρι πρόσφατα, το R υστερούσε σαφούς πλαισίου για τους σκοπούς της εξόρυξης κειμένου.

Αυτό άλλαξε με τη δημιουργία του πακέτου tm (Feinerer ,2008), το οποίο παρέχει την υποδομή για την επεξεργασία της εξόρυξης κειμένου στο R. Πιο συγκεκριμένα, επιτρέπει στο R και στους χρήστες του να εργάζονται αποτελεσματικά με τα κείμενα και τα αντίστοιχα μετα-δεδομένα, και να μετατρέπουν τα κείμενα σε δομημένες αναπαραστάσεις, όπου μπορούν να εφαρμοστούν υπάρχουσες μέθοδοι της R.

- **1η Διεργασία: Δημιουργία διανυσματικού μοντέλου λέξεων από το σύνολο των δεδομένων και Διαδικασία Προ-επεξεργασίας**

Προκειμένου να ξεκινήσει η διαδικασία της εξόρυξης κειμένου με τη βοήθεια του συστήματος R, αρχικά πρέπει να προετοιμαστεί το περιβάλλον εργασίας για να δεχτεί το σύνολο των δεδομένων, εγκαθιστώντας τις βιβλιοθήκες tm, SnowballC και e1071. Η βιβλιοθήκη SnowballC είναι μια διεπαφή για τη βιβλιοθήκη των stemmers στη C, προκειμένου να χρησιμοποιηθεί ο αλγόριθμος του Porter και να βοηθήσει στη προ-επεξεργασία των δεδομένων ανακαλύπτοντας τις ρίζες των λέξεων, ενώ η βιβλιοθήκη e1071, εμπεριέχει συναρτήσεις για την ανάλυση των κλάσεων, όπως ο κατηγοριοποιητής Naïve Bayes, ο οποίος θα χρησιμοποιηθεί στη συγκεκριμένη περίπτωση. Ενώ η ποιο σημαντική βιβλιοθήκη είναι η tm. Η βιβλιοθήκη tm προσφέρει λειτουργικότητα στη διαχείριση των εγγράφων κειμένου και διευκολύνει τη χρήση των ετερογενών μορφών κειμένου στο R. Η βιβλιοθήκη έχει ενσωματωθεί με μια υποστηρικτική βάση δεδομένων, ώστε να ελαχιστοποιηθούν οι απαιτήσεις μνήμης. Μια προηγμένη διαχείριση μετά-δεδομένων εφαρμόζεται για τις συλλογές των εγγράφων κειμένου, ώστε να μην είναι απαραίτητη η χρήση των μεγάλων και εμπλουτισμένων με μετα-δεδομένα, συνόλων εγγράφων. Οι δομές δεδομένων και οι αλγόριθμοι μπορούν να επεκταθούν για να χωρέσουν τις προσαρμοσμένες απαιτήσεις του χρήστη, δεδομένου ότι το πακέτο είναι σχεδιασμένο με έναν αρθρωτό τρόπο, ώστε να επιτρέπει την εύκολη ενσωμάτωση των νέων μορφών αρχείων.

Γενικά, το πακέτο tm επεξεργάζεται έγγραφα κειμένου με αποτελεσματικό τρόπο. Οι εργασίες του πακέτου συνοψίζονται στις κατώθι: α) δημιουργεί ένα «σώμα» εγγράφων, μια συλλογή εγγράφων κειμένου, β) παρέχει διάφορες εργασίες προεπεξεργασίας, γ) δημιουργεί μια πίνακας όρου – εγγράφου, δ) Επιθεωρεί τη πίνακας όρου-εγγράφου (για παράδειγμα, μετατροπή των δεδομένων σε ένα πλαίσιο δεδομένων που απαιτείται από τους ταξινομητές), ε) εκπαιδεύει ένα κατηγοριοποιητή και στ) εφαρμόζει τον

εκπαιδευμένο κατηγοριοποιητή για νέα έγγραφα κειμένου ώστε να αποκτήσουν τις προβλέψεις κλάσης και την αξιολόγηση των επιδόσεων.

```
# Import the required libraries
library("tm")
library("Snowballc")
library("e1071")
```

Εικόνα 22. Εισαγωγή απαιτούμενων βιβλιοθηκών R.

Για να πραγματοποιηθεί η υλοποίηση της εν λόγω μελέτης περίπτωσης κατηγοριοποίησης κειμένου πρέπει να επιλεγούν και να εισαχθούν τα αντίστοιχα δεδομένα στο σύνολο των τελεστών και συναρτήσεων που κρίνονται απαραίτητοι.

Στη συνέχεια, δημιουργείται η συνάρτηση η οποία έχει στόχο την προ-επεξεργασία του συνόλου των δεδομένων. Τα έγγραφα αντιπροσωπεύονται συνήθως με τη χρήση λέξεων, όρων ή εννοιών. Λαμβάνοντας υπόψη όλες τις πιθανές λέξεις ως πιθανούς δείκτες μιας κατηγορίας μπορεί να δημιουργήσει προβλήματα στην κατάρτιση ενός συγκεκριμένου κατηγοριοποιητή. Η tmMap λειτουργία (διαθέσιμη σε πακέτο "tm") μπορεί να χρησιμοποιηθεί να πραγματοποιήσει διάφορα στάδια προ-επεξεργασίας. Η λειτουργία αυτή εφαρμόζεται σε ολόκληρο το σώμα, δεν υπάρχει λόγος να προγραμματιστεί χρησιμοποιώντας ένα βρόχο. Πιο συγκεκριμένα, η συνάρτηση που ορίστηκε πραγματοποιεί τις παρακάτω λειτουργίες:

- Αφαιρεί τις λέξεις που δεν έχουν καμία αξία, όπως αντωνυμίες, άρθρα και άλλα.,
- Αφαιρεί τα πλεονάζοντα κενά,
- Αφαιρεί τα νούμερα,
- Αφαιρεί τα σημεία στίξης,
- Μετατρέπει όλα τα κεφαλαία γράμματα σε πεζά,
- Εκτελεί ετικετοποίηση (stemming).

```
# Function to perform preprocessing in the data
preProcess <- function(corp) {
  x <- corp
  x <- tm_map(x, content_transformer(tolower))
  x <- tm_map(x, removeWords, stopwords("english"))
  x <- tm_map(x, content_transformer(function(t) gsub("[^a-zA-Z]+", " ", t)))
  x <- tm_map(x, removeNumbers)
  x <- tm_map(x, stemDocument)
  x <- tm_map(x, stripWhitespace)
  return(x)
}
```

Εικόνα 23. Εισαγωγή συνάρτησης προ-επεξεργασίας του συνόλου των δεδομένων.

Ακολούθως, δημιουργείται και εισάγεται το σώμα του συνόλου των δεδομένων μέσω των διευθύνσεων που είναι αποθηκευμένα.

```
createCorpus <- function(dir) {
  corp <- Corpus(DirSource(dir), readerControl=list(language="english", reader=readPlain))
  return(corp)
}

# Read command lines arguments
args <- commandArgs(trailingOnly = TRUE)
dataset.dir <- file.path(args[1])
print(dataset.dir)


# Create the data directory paths
dataset.categories  <- list.dirs(path = dataset.dir, recursive = F, full.names = F)
dataset.dirs        <- file.path(dataset.dir, dataset.categories)
names(dataset.dirs) <- dataset.categories

# Create corpus from data directories
print("Reading data from filesystem...")
dataset.corpus <- sapply(dataset.dirs, createCorpus)
```

Εικόνα 24. Εισαγωγή του συνόλου των δεδομένων και δημιουργία του σώματος των δεδομένων.

Επίσης, δημιουργείται το σύνολο των δεδομένων εκπαίδευσης και δοκιμής, διαχωρίζοντας το μήκος του συνόλου των δεδομένων σε 75% δεδομένα εκπαίδευσης και 25% δεδομένα δοκιμής.

```
# Create split
split.percentage  <- 0.75
split.sizes       <- sapply(dataset.corpus, length)
split.train.sizes <- floor(split.sizes * split.percentage)
split.test.sizes  <- split.sizes - split.train.sizes

train.corpuses <- mapply(function(corp, size) corp[1:size]$content,
                         dataset.corpus, split.train.sizes)

test.corpuses  <- mapply(function(corp, start, size) corp[start+1:size]$content,
                         dataset.corpus, split.train.sizes, split.test.sizes)

# Merge splitted datasets
train.corpus <- c(unlist(train.corpuses, recursive = F))
test.corpus  <- c(unlist(test.corpuses,  recursive = F))

stopifnot(length(train.corpus) == sum(split.train.sizes))
stopifnot(length(test.corpus)  == sum(split.test.sizes))
```

Εικόνα 25. Δημιουργία συνόλου δεδομένων εκπαίδευσης και δοκιμής.

Εφόσον, δημιουργηθεί το σύνολο των δεδομένων εκπαίδευσης και δοκιμής, καλείται η συνάρτηση της προε-επεξεργασίας, για να εφαρμοστεί στο σύνολο των δεδομένων.

```
# Perform the preprocessing
print("Pre-processing corpuses...")
train.corpus <- preProcess(VCorpus(VectorSource(train.corpus)))
test.corpus  <- preProcess(VCorpus(VectorSource(test.corpus)))
```

Εικόνα 26. Εφαρμογή της συνάρτησης προ-επεξεργασίας στο σύνολο των δεδομένων.

Εφόσον πραγματοποιηθεί και η διαδικασία της προ-επεξεργασίας των δεδομένων, δημιουργείται το “Document Term Matrix”. Οι ταξινομητές που εκμεταλλεύονται την προτασιακή αναπαράσταση “prepositional representation”, όπως KNN, NaiveBayes, SVM και λοιπά, απαιτούν τα δεδομένα να εκπροσωπούνται με τη μορφή ενός πίνακα, όπου κάθε σειρά περιέχει μία περίπτωση, στην προκειμένη περίπτωση ένα έγγραφο, και κάθε στήλη αντιπροσωπεύει ένα συγκεκριμένο χαρακτηριστικό γνώρισμα, στην προκειμένη περίπτωση, μια λέξη. Οι σημαντικότεροι παράμετροι της συνάρτησης “Document Term Matrix” που χρησιμοποιούνται στη συγκεκριμένη περίπτωση είναι οι εξής:

- Η στάθμιση TF-IDF όπου αναφέρεται στο πόσο συχνά ένας όρος βρίσκεται σε ένα έγγραφο, και όχι αν ένας όρος απλά βρίσκεται στο έγγραφο,

- Το ελάχιστο μήκος λέξης που στην προκειμένη περίπτωση ορίζεται στη τιμή δυο,

- και τέλος η κάθε λέξη να εμφανίζεται τουλάχιστον μια φορά στο εκάστοτε έγγραφο.

```
# Create the Document Term Matrices
print("Creating document term matrices...")
train.corpus.dtm <- DocumentTermMatrix(train.corpus, control=list(weighting=weightTfIdf,
                                                                  wordLengths=c(4, 15),
                                                                  bounds = list(global = c(5,Inf))))
test.corpus.dtm  <- DocumentTermMatrix(test.corpus, control=list(weighting=weightTfIdf,
                                                                 wordLengths=c(4, 15),
                                                                 bounds = list(global = c(5,Inf))))
```

Εικόνα 27. Δημιουργία " Document Term Matrix".

Επιπλέον, οι ταξινομητές στην R όπως KNN, Naive Bayes, SVM, απαιτούν τα δεδομένα να εκπροσωπούνται με τη μορφή ενός πλαισίου δεδομένων. Έτσι, θα πρέπει να μετατραπεί η παραπάνω πίνακας δεδομένων "Document Term Matrix" σε ένα πλαίσιο δεδομένων "Data Frame", όπως φαίνεται παρακάτω. Παράλληλα, θα πρέπει να προσαρτηστούν οι πληροφορίες των κλάσεων, και αυτή η διαδικασία περιλαμβάνει δύο στάδια:

1. Δημιουργία ενός διανύσματος το οποίο θα εμπεριέχει τις πληροφορίες των κλάσεων της κάθε κριτικής στο σύνολο δεδομένων εκπαίδευσης και δοκιμής,

2. Προσάρτηση της κλάσης ως τελευταία στήλη του πλαισίου δεδομένων.

```
# Create training and testing corpuses
print("Creating training and testing corpuses...")

print("Creating data matrices...")
train.corpus.df <- as.matrix(train.corpus.dtm)
test.corpus.df  <- as.matrix(test.corpus.dtm)

# Generate vector with class values
print("Creating class information...")
train.corpus.class <- rep(dataset.categories, split.train.sizes)
test.corpus.class  <- rep(dataset.categories, split.test.sizes)
```

Εικόνα 28. Δημιουργία πλαισίου δεδομένων και κλάσεων.

- **2η Διεργασία : Υλοποίηση Κατηγοριοποίησης Naïve Bayes στο σύνολο των δεδομένων εκπαίδευσης**

Στη συνέχεια και εφόσον τα δεδομένα εκπαίδευσης είναι πλέον σε μορφή που μπορούν να επεξεργαστούν από τους ταξινομητές, καλείται ο κατηγοριοποιητής Naive Bayes.

```
# Train classifier
print("Training classifier...")
start.time <- Sys.time()
classifier <- naiveBayes(train.corpus.df, as.factor(train.corpus.class))
```

Εικόνα 29. Εκπαίδευση κατηγοριοποιητή Naive Bayes, χρησιμοποιώντας τα δεδομένα εκπαίδευσης.

Αφού ολοκληρωθεί η διαδικασία εκπαίδευσης του κατηγοριοποιητή Naive Bayes, το επόμενο στάδιο είναι η αξιολόγηση του μοντέλου με τα δεδομένα δοκιμής.

```
# Evaluate Classifier
print("Evaluating... Please be patient. This will take a while...")
start.time <- Sys.time()
corpus.predictions <- predict(classifier, test.corpus.df)
end.time <- Sys.time()
time.taken <- end.time - start.time
print(time.taken)

# Print results
table(corpus.predictions, test.corpus.class)
```

Εικόνα 30. Αξιολόγηση του μοντέλου εκπαίδευσης.

Ολοκληρώνοντας την προσπάθεια, τρέχουμε το πρόγραμμα, το οποίο διαρκεί πολύ ώρα μέχρι να παράξει τα αποτελέσματα της αξιολόγησης. Στο τέλος, εμφανίζεται η πίνακας αποτελεσμάτων "Confusion Matrix", όπως φαίνεται στην Εικόνα 31. Πιο συγκεκριμένα, καλείται η συνάρτηση "corpus.predictions" η οποία αποτελείται από τα αποτελέσματα του κατηγοριοποιητή και από το πλαίσιο δεδομένων του συνόλου των δεδομένων δοκιμής, όπου στη συνέχεια δημιουργείται η πίνακας αποτελεσμάτων όπου οι γραμμές εκπροσωπούν τις θετικές και αρνητικές κριτικές του μοντέλου εκπαίδευσης και οι στήλες τις αντίστοιχες κριτικές στα πραγματικά δεδομένα.

```
[1] "Training classifier..."
   user  system elapsed
  6.684   0.037   6.771
[1] "Evaluating... Please be patient. This will take a while..."
   user  system elapsed
169.332   0.153 170.735
[1] "Confusion Matrix:"
                    test.corpus.class
corpus.predictions neg pos
               neg 187  76
               pos  63 174
[1] "Recall per class:"
  neg   pos
0.748 0.696
[1] "Precision per class:"
     neg       pos
0.7110266 0.7341772
[1] "Average Recall Score    : 0.722"
[1] "Average Precision Score : 0.723"
[1] "Accuracy Score          : 0.722"
```

Εικόνα 31. Πίνακας αποτελεσμάτων του κατηγοριοποιητή Naive Bayes.

Τέλος, με βάση τον πίνακα αποτελεσμάτων που εξάχθηκε υπολογίζονται οι Δείκτες Ακριβείας, Ανάκλησης και Ορθότητας του μοντέλου που υλοποιήθηκε και ανέρχονται σε 72,20 %, 72,20% και 72,30%, αντίστοιχα.

4.2.4 Κατηγοριοποίηση Κειμένου με τη βοήθεια του Python

Κάθε νέα έκδοση της Python αναπτύσσει όλο και περισσότερο τη δύναμη της γλώσσας στην εξόρυξη κειμένου. Έχουν αναπτυχθεί μερικές χρήσιμες μέθοδοι επεξεργασίας κειμένου, με τη βελτίωση των παλιών ή την προσθήκη νέων μεθόδων. Πραγματοποιήθηκαν βελτιώσεις και προσθήκες στην πρότυπη βιβλιοθήκη για να καλύψει τις πιο συνηθισμένες εργασίες στην εξόρυξη κειμένου.

- **1η Διεργασία: Δημιουργία διανυσματικού μοντέλου λέξεων από το σύνολο των δεδομένων και Διαδικασία Προ-επεξεργασίας**

Για την ανάπτυξη της εν λόγω μελέτης περίπτωσης, επιλέχθηκε να εγκατασταθεί το πακέτο scikit learn το οποίο είναι μια κορυφαία πλατφόρμα για τη δημιουργία προγραμμάτων σε Python, προκειμένου να εργαστεί κάποιος με δεδομένα φυσικής γλώσσας στο πεδίο της εξόρυξης κειμένου. Παρέχει πάνω από πενήντα σύνολα δεδομένων και λεξιλογικούς πόρους, όπως το WordNet, μαζί με μια σειρά από βιβλιοθήκες προ-επεξεργασίας κειμένου, όπως επίσης και ένα ενεργό φόρουμ

συζήτησης αλλά και πολύ καλή τεκμηρίωση. Προκειμένου να εγκατασταθεί το εν λόγω πακέτο απαιτείται η εγκατάσταση της Python έκδοση >= 2.6 ή >= 3.3, επίσης το πακέτο NumPy (>= 1.6.1) και SciPy (>= 0.9). Αρχικά, πρέπει να εγκατασταθεί το NumPy και SciPy από την επίσημη σελίδα εγκατάστασης τους. Τα wheel πακέτα (.whl αρχεία) για scikit από το PyPI μπορούν να εγκατασταθούν με το pip utility. Ανοίγοντας μια διεπαφή χρήστη της Python πληκτρολογείται η ακόλουθη εντολή ώστε να εγκατασταθεί ή να αναβαθμιστεί το scikit learn σύμφωνα με την τελευταία σταθερή έκδοση :

- pip install -U scikit-learn

Στη γλώσσα προγραμματισμού Python, αντί να δημιουργηθεί ένας πίνακας όρων, για την οικοδόμηση του κατηγοριοποιητή Naive Bayes, θα οικοδομηθεί μια ένωση χαρακτηριστικών των δεδομένων εκπαίδευσης, ώστε να τροφοδοτηθεί μια λίστα από αυτά κατά τη λειτουργία του NaiveBayesClassifier. Η λίστα είναι ο πιο ευέλικτος τύπος δεδομένων που διατίθεται σε Python διότι τα στοιχεία σε μια λίστα, δεν χρειάζεται να έχουν όλα τον ίδιο τύπο.

Αρχικά, εισάγονται τα απαιτούμενα πακέτα και βιβλιοθήκες που θα προετοιμάσουν το περιβάλλον ώστε να πραγματοποιηθεί η μελέτη περίπτωσης.

```
import os
import random
import sys
import textwrap
import time
from collections import namedtuple
from os import path
from timer import Timer

from sklearn.feature_extraction.text import (CountVectorizer, TfidfVectorizer)
from sklearn import (ensemble, naive_bayes, neighbors, neural_network, svm, tree)
from sklearn import metrics
from sknn.mlp import Classifier, Layer

def enum(**enums):
    return type('Enum', (), enums)
```

Εικόνα 32. Εισαγωγή των απαιτούμενων βιβλιοθηκών.

Γενικά, μια συνάρτηση στη Python είναι ένα μπλοκ οργανωμένου, επαναχρησιμοποιήσιμου κώδικα που ορίζεται για να καθορίσει μια λειτουργία κατά την εκτέλεση ενός ενιαίου κώδικα. Μια συνάρτηση δίνει μόνο ένα όνομα, προσδιορίζει τις παραμέτρους που πρέπει να συμπεριληφθούν στη λειτουργία και τις δομές των μπλοκ

του κώδικα. Έτσι λοιπόν, στη συνέχεια ορίζεται ένα πλήθος συναρτήσεων για να καθοριστούν συγκεκριμένες λειτουργίες προκειμένου να ολοκληρωθεί το μοντέλο κατηγοριοποίησης κειμένου με τη βοήθεια του Naive Bayes.

Αρχικά, ορίζεται μια κενή κλάση η οποία θα περιέχει τα αποτελέσματα της εκτέλεσης. Κάθε στιγμιότυπο θα αποκτήσει πεδία κατά τη δημιουργία του και όχι νωρίτερα, εκμεταλλευόμενοι έτσι τη δυναμική φύση της python. Έπειτα, ορίζεται η κλάση configuration η οποία περιέχει τις βασικές ρυθμίσεις εκτέλεσης, όπως τον κατάλογο ο οποίος περιέχει τις αρνητικές και θετικές κριτικές (reviews_dir), και το ποσοστό αυτών που θα χρησιμοποιηθεί για το στάδιο της εκπαίδευσης κατά τη κατηγοριοποίηση (training_percentage), το οποίο ορίζεται σε 75%.

```
class Results():
    pass


class Configuration(object):
    def __init__(self, reviews_dir = '.', training_percentage = .75):
        self.reviews_dir = reviews_dir
        self.pos_reviews_dir = path.join(reviews_dir, Category.POSITIVE)
        self.neg_reviews_dir = path.join(reviews_dir, Category.NEGATIVE)
        self.training_percentage = training_percentage
```

Εικόνα 33. Ορισμός διαδρομών του σύνολου των δεδομένων κι συνόλου εκπαίδευσης μέσω της κλάσης configuration.

Οι βασικές ρουτίνες της κατηγοριοποίησης περιέχονται σε μία κλάση `Classification`, η οποία αρχικοποιείται:

1. Με κάποιες ρυθμίσεις (βλέπε κλάση Configuration) `config` (αν δεν δοθούν συγκεκριμένες ρυθμίσεις κατασκευάζεται ένα νέο στιγμιότυπο ρυθμίσεων έχοντας τις default επιλογές),

2. με κάποιον vectorizer ο οποίος θα αναλάβει να σπάσει τα κείμενα των κριτικών σε λέξεις, να μετασχηματίσει τα κείμενα (αφαιρώντας τα stopwords κλπ) και μετά να αναπαραστήσει την όλη πληροφορία σε έναν δισδιάστατο πίνακα λέξεων-κριτικών όπου μία θέση θα είναι είτε 1 είτε 0 ανάλογα με το αν περιέχεται στην εκάστοτε κριτική ή όχι (αλλιώς μπορεί να έχει το πλήθος των εμφανίσεών της στο κείμενο). Επίσης, ο πίνακας μπορεί να υποστεί και έναν μετασχηματισμό tf-idf, ανάλογα με τον `vectorizer`.

```
class Classification(object):
    def __init__(self, vectorizer, config=None):
        # Default configuration, if none was given
        self.config = config or Configuration()
        self.vectorizer = vectorizer
```

Εικόνα 34. Αρχικοποίηση των βασικών ρουτινών της κλάσης classification.

Στη συνέχεια, ορίζονται κάποιες ιδιότητες (μέθοδοι python οι οποίες μπορούν να προσπελαύνονται ως απλά πεδία) της κλάσης για εύκολη προσπέλαση. Έπειτα, η μέθοδος read_reviews διαβάζει τις κριτικές από το δίσκο αναδρομικά και επιστρέφει τα κείμενα (μαζί με το αντίστοιχο μονοπάτι) ένα ένα τη φορά, δημιουργώντας έναν generator με χρήση της `yield`.

```
    @property
    def training_percentage(self):
        return self.config.training_percentage

    def _read_reviews(self, directory):
        for root, dirs, files in os.walk(directory):
            for review in files:
                with open(path.join(root, review)) as f:
                    yield review, f.read()

    def read_positive_reviews(self):
        return self._read_reviews(self.config.pos_reviews_dir)

    def read_negative_reviews(self):
        return self._read_reviews(self.config.neg_reviews_dir)
```

Εικόνα 35. Ορίζεται η προσπέλαση του συνόλου των δεδομένων και επιστροφή του κειμένου μαζί με το αντίστοιχο μονοπάτι.

Η παραλλαγή read_reviews δέχεται ως όρισμα όνομα κατηγορίας αντί για κατάλογο. Στη συνέχεια, η μέθοδος vectorize δέχεται ένα σύνολο δεδομένων ως παράμετρο (σε μορφή ακολουθίας ζευγών κείμενο-κατηγορία). Στην πρώτη γραμμή σπάει τα ζεύγη σε δύο λίστες, μία για τα κείμενα, και μία για τις κατηγορίες, διατηρώντας την ίδια σειρά που είχαν. Έπειτα, χρησιμοποιεί τον vectorizer που περιγράφηκε παραπάνω για να κατασκευάσει το διάνυσμα που αντιστοιχεί στο σύνολο δεδομένων. Αυτό το αποθηκεύει σε ένα πεδίο ενός στιγμιοτύπου του συνόλου δεδομένων, περιλαμβάνοντας επίσης τη λίστα των κατηγοριών.

```
def read_reviews(self, category):
    if category == Category.POSITIVE:
        directory = self.config.pos_reviews_dir
    elif category == Category.NEGATIVE:
        directory = self.config.neg_reviews_dir
    else:
        raise ValueError('Illegal category %s' % category)
    return self._read_reviews(directory)

def vectorize(self, dataset, fit=False):
    data, labels = map(list, zip(*dataset))
    V = self.vectorizer

    # Vectorize data
    vector = V.fit_transform(data) if fit else V.transform(data)

    # Sanity checks
    assert vector.shape[0] == len(labels)
    return Dataset(vector,labels)
```

Εικόνα 36. Ορίζεται η διανυσματοποίηση του συνόλου των δεδομένων.

Η μέθοδος process_reviews κατανέμει τις αρνητικές και θετικές κριτικές σε δύο σύνολα, (training set και testing set) αναθέτοντας περιοδικά τις κριτικές στο κατάλληλο σύνολο το οποίο επιλέγεται και αποθηκεύεται σε κάθε βήμα στην μεταβλητή `dataset`. Για παράδειγμα, για training percentage = 0.75, η περίοδος (period) είναι 4. Επομένως, αποθηκεύει 1 ανά 4 στοιχεία στο test set. Έπειτα, διανυσματοποιεί τα σύνολα και αποθηκεύει τα διανύσματα σε κάποια πεδία της κλάσης.

```
def process_reviews(self, config=None):
    '''Define training and testing sets.'''

    # Apply different configuration
    if config is not None:
        self.config = config

    # Read the data
    train_set, test_set = [], []
    period = 1 / (1 - self.training_percentage)

    # Loop over categories
    for category in (Category.POSITIVE, Category.NEGATIVE):
        counter = 1
        # Loop over category reviews
        for fname, content in self.read_reviews(category):
            # Choose appropriate dataset
            dataset = test_set if counter >= period else train_set
            counter = 1 + (counter % period)
            # Append to dataset
            dataset.append((content, category))

    # print '# Training Set Items  : %4d' % len(train_set)
    # print '# Testing Set Items   : %4d' % len(test_set)

    # Store processed datasets
    self.train_set = self.vectorize(train_set, fit=True)
    self.test_set = self.vectorize(test_set)

    return self
```

Εικόνα 37. Δημιουργία των συνόλων δεδομένων εκπαίδευσης και δοκιμής.

Η call είναι η βασική μέθοδος όπου καλεί ότι έχει οριστεί προηγουμένως. Αρχικά, υπολογίζεται το πλήθος των στοιχείων κάθε κατηγορίας και στη συνέχεια

διαχωρίζονται τα πεδία στα αντίστοιχα διανύσματα-κατηγορίες. Παράλληλα, ορίζεται η φάση της εκπαίδευσης με το σύνολο εκπαίδευσης. Ορίζονται δυο προσπάθειες προσπέλασης της φάσης της εκπαίδευσης, σε περίπτωση που η πρώτη αποτύχει επειδή ο αλγόριθμος απαιτεί συμπαγείς μορφές πινάκων, μετασχηματίζονται οι πίνακες και επαναλαμβάνεται η διαδικασία.

```
def __call__(self, classifier):
    nPos = len([l for l in self.test_set.labels if l == Category.POSITIVE])
    nNeg = len(self.test_set.labels) - nPos

    train_vector, train_labels = self.train_set
    test_vector, test_labels = self.test_set

    print

    # print train_vector.shape
    # print test_vector.shape
    # print len(train_labels)

    # Run classifier (at most two times)
    for i in range(2):
        try:
            t0 = time.time()
            with Timer('Training classifier'):
                classifier.fit(train_vector, train_labels)
            break
        except TypeError:
            # Try dense vectors
            train_vector = self.dense_train_set.vector
            test_vector = self.dense_test_set.vector

    t1 = time.time()
    with Timer('Using classifier to perform predictions'):
        prediction = classifier.predict(test_vector)
    t2 = time.time()

    assert prediction.size == len(test_labels)
```

Εικόνα 38. Ορισμός της συνάρτησης call, στα πλαίσια της οποίας τρέχει η φάση της εκπαίδευσης του συνόλου των δεδομένων.

Επίσης, κατασκευάζεται ο πίνακας αποτελεσμάτων, υπολογίζοντας τα αποτελέσματα που πρόκειται να απεικονιστούν (συνολικό πλήθος σωστών προβλέψεων, συνολικό πλήθος κριτικών, ακρίβεια ανά κατηγορία). Στα πλαίσια της αξιολόγησης του μοντέλου υπολογίζονται και απεικονίζονται στην οθόνη τα ποσοστά ακρίβειας του συνόλου των δεδομένων δοκιμής των θετικών και αρνητικών αξιολογήσεων. Επίσης, υπολογίζεται η πίνακας αποτελεσμάτων *"Confusion Matrix"* των τιμών *true_positive, true_negative, false_positive* και *false_negative* και τέλος εκτυπώνονται τα δέκα χαρακτηριστικά, που συγκεντρώνουν τη μεγαλύτερη πληροφορία.

```
# Build confusion matrix
confusion_matrix = metrics.confusion_matrix(test_labels, prediction)

# Gather results
results = Results()
results.train_time, results.pred_time = t1-t0, t2-t1
results.negative_found, results.negative_missed = confusion_matrix[0]
results.positive_missed, results.positive_found = confusion_matrix[1]
results.positive_accuracy = results.positive_found / float(nPos)
results.negative_accuracy = results.negative_found / float(nNeg)
results.overall_accuracy = 0.5 * (results.positive_accuracy + results.negative_accuracy)

# print(metrics.classification_report(test_labels, prediction))
# print self.vectorizer.get_feature_names()

# Calculate stop words
stopwords = list(self.vectorizer.get_stop_words())
stopword_shortlist = ', '.join(stopwords[:3])

if len(stopwords) > 3:
    stopword_shortlist += ", ..."
```

Εικόνα 39. Δημιουργία πίνακα αποτελεσμάτων και υπολογισμός αποτελεσμάτων.

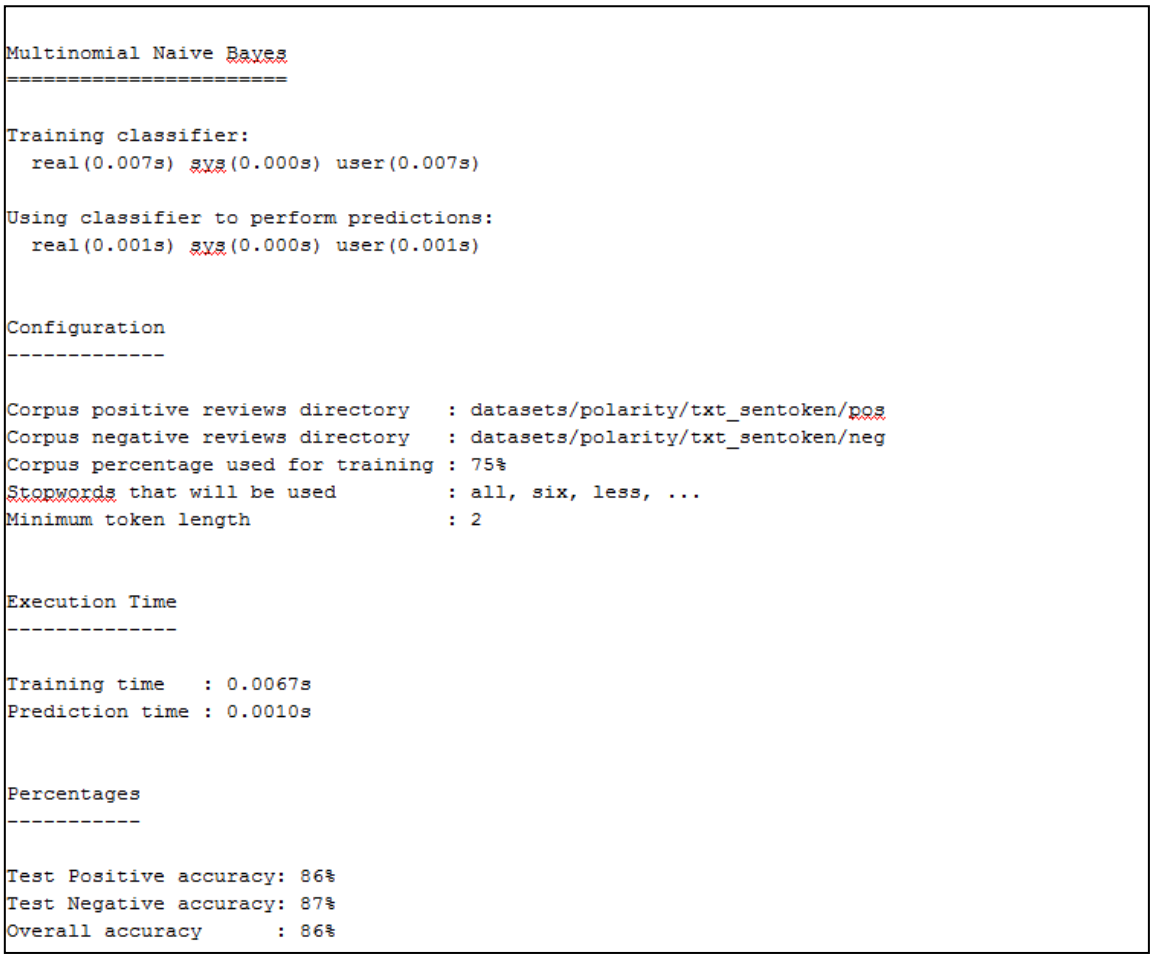
- **2η Διεργασία : Υλοποίηση Κατηγοριοποίησης Naïve Bayes στο σύνολο των δεδομένων εκπαίδευσης**

Εφόσον έχει ολοκληρωθεί η διαδικασία της προετοιμασίας των δεδομένων, θα προχωρήσουμε στην εκπαίδευση του κατηγοριοποιητή *Naive Bayes*, και στη συνέχεια στην αξιολόγηση του μοντέλου που οικοδομήθηκε. Η συνάρτηση main θα ξεκινήσει ουσιαστικά την εκτέλεση της κατηγοριοποίησης. Στην αρχή, κατασκευάζεται ο vectorizer (tf-idf) που θα χρησιμοποιηθεί, επίσης κατασκευάζεται ένα στιγμιότυπο ρυθμίσεων και ένα κατηγοριοποίησης. Εκτελείται με χρονομέτρηση η προ-επεξεργασία του συνόλου των δεδομένων και η κατηγοριοποίηση Naïve Bayes.

```
def main(reviews_dir, option = None):
    # vectorizer = TfidfVectorizer(min_df=.3,max_df=.9,stop_words='english')
    vectorizer = TfidfVectorizer(min_df=5,
                                 max_df = 0.9,
                                 # max_features = 2000,
                                 sublinear_tf=True,
                                 smooth_idf=True,
                                 use_idf=True,
                                 stop_words='english')

    # Create and print configuration
    config = Configuration(reviews_dir)
    classification = Classification(vectorizer, config)
    with Timer('Processing reviews'):
        classification.process_reviews()

    schemes = [('Multinomial Naive Bayes', naive_bayes.MultinomialNB()),
               ]

    scheme = None
```

Εικόνα 40. Εκπαίδευση του κατηγοριοποιητή Naive Bayes και αξιολόγηση του μοντέλου Κατηγοριοποίησης.

Τέλος, ορίζεται η συνάρτηση *printConfiguration,* η οποία είναι υπεύθυνη για την απεικόνιση των αποτελεσμάτων και των επιμέρους οριζόμενων στοιχείων της μελέτης περίπτωσης, όπως φαίνεται στην Εικόνα 40.



Εικόνα 41. Απεικόνιση αποτελεσμάτων της κατηγοριοποίησης Naïve Bayes με το σύνολο δεδομένων IMDb reviews.

4.3 Ευχρηστία των συστημάτων

Έπειτα από την ολοκλήρωση των υλοποιήσεων της κατηγοριοποίησης κειμένου με τον αλγόριθμο Naïve Bayes για κάθε σύστημα ξεχωριστά, καταρτίστηκε ο παρακάτω πίνακας, ο οποίος περιέχει κάποια κριτήρια ευχρηστίας που έχουν επιλεγεί από τον συγγραφέα και βαθμολογίες. Γενικά, η ευχρηστία περιγράφεται ως την ικανότητα των συστημάτων να εκπληρώνουν τις προσδοκίες του χρήστη. Η βαθμολογία είναι μια υποκειμενική αξία και βασίζεται στην προοπτική του συγγραφέα και την εμπειρία σε συνδυασμό με αντικειμενικά δεδομένα και στοιχεία, όπου αυτά είναι διαθέσιμα. Η παρακάτω αξιολόγηση βασίστηκε σε ένα σύστημα κλίμακας από το 1 έως το 5, όπου οι υψηλότερες βαθμολογίες σημαίνουν υψηλή αξιολόγηση με κριτήρια την ευκολία και την απλότητα, ενώ οι χαμηλότερες βαθμολογίες παρουσιάζουν αρνητικά αποτελέσματα ως προς τα κριτήρια της δυσκολίας και της πολυπλοκότητας.

Σημειώνεται ότι ο συγγραφέας δεν είχε πρότερη εμπειρία με τα εξεταζόμενα συστήματα, οπότε λογίζεται ως αρχάριος.

Κριτήρια Ευχρηστίας	R	RapidMiner	Weka Explorer	Python (scikit-learn)
Ευκολία Εγκατάστασης	5	5	5	5
Εγχειρίδια χρήσης / Τεκμηρίωση	5	4	5	5
Online εκπαίδευση	5	4	5	3
Αναφορά σφαλμάτων	2	5	3	2
Επεκτασιμότητα	5	5	5	5
Διαλειτουργικότητα με άλλα συστήματα	5	5	4	4
Εύρος Οπτικοποίησης Αποτελεσμάτων	5	5	5	5
Ευκολία Μάθησης	2	3	4	2
Ευχρηστία	3	3	4	2

Πίνακας 3. Βαθμολόγηση κριτηρίων ευχρηστίας.

Όπως αναφέρθηκε στα γενικά χαρακτηριστικά των συστημάτων, όλα τα συστήματα μπορούν να εγκατασταθούν σε όλα τα λειτουργικά συστήματα, που σημαίνει ότι ο βαθμός ευκολίας στην εγκατάσταση και των τεσσάρων συστημάτων είναι υψηλός. Επίσης, όλα τα συστήματα έχουν δωρεάν και διαθέσιμα προς όλους τους χρήστες, εγχειρίδια χρήσης όπως επίσης και πολλά παραδείγματα υλοποιήσεων αλλά και online εκπαίδευση. Το scikit learn έχει διαθέσιμο ένα μικρό εύρος των λειτουργιών του σε online εκπαίδευση και το RapidMiner δεν έχει τόσο μεγάλο εύρος τεκμηρίωσης σε σύγκριση με το R και το Weka.

Στη συνέχεια εξετάστηκε η ικανότητα και το εύρος της αναφοράς των σφαλμάτων, η οποία αυξάνει το βαθμό μάθησης των συστημάτων για έναν αρχάριο χρήστη. Το Rapidminer διαθέτει μια απεικόνιση των σφαλμάτων όπου καταγράφεται η λάθος χρήση κάποιου τελεστή ή η απώλεια δεδομένων κ.λ.π. άμεσα και προτείνονται διάφορες λύσεις του σφάλματος. Ενώ το Weka σε περίπτωση λάθους χρήσης διαθέτει pop up μηνύματα όπου αναφέρεται και καταγράφεται το λάθος σε log file, χωρίς όμως να δίνεται η λύση στο πρόβλημα. Τέλος, το R και scikit learn λόγω της γραμμής εντολών

διεπαφής χρήστη, καταγράφονται και αναφέρονται τα σφάλματα αναφορικά με τη σύνταξη του κώδικα χωρίς να δίνεται λύση.

Γενικά, δίνεται ως δυνατότητα σε όλα τα ανοιχτού κώδικα λογισμικά να επεκτείνονται και να βελτιώνονται από τους χρήστες, οπότε όλα τα εξεταζόμενα συστήματα απολαμβάνουν υψηλό βαθμό επεκτασιμότητας. Αναφορικά με το κριτήριο της διαλειτουργικότητας, όλα τα συστήματα κατέχουν υψηλό βαθμό διαλειτουργικότητας, διότι παρέχουν τη δυνατότητα μέσω επιπροσθέτων πακέτων, να ενσωματώνουν λειτουργίες άλλων συστημάτων όπως Hadoop, SAS, systat, R, Weka και Python. Το R και το Rapidminer συνδέονται με περισσότερα συστήματα σε σύγκριση με το scikit learn και το Weka. Στη συνέχεια, εξετάζεται το εύρος της οπτικοποίησης των αποτελεσμάτων συγκεκριμένα για το πεδίο εφαρμογής της εξόρυξης κειμένου. Σύμφωνα με τον πίνακα της λειτουργικότητας των συστημάτων, όλα τα συστήματα παρέχουν τις βασικές απεικονίσεις όπως scatterplots, histograms, Roc curves, accuracy – precision and recall tables και tree based απεικονίσεις. Γενικά, το R παρέχει πολυάριθμες απεικονίσεις και σε άλλα πεδία εφαρμογής, αλλά αυτή τη στιγμή εξετάζεται μόνο η εξόρυξη κειμένου.

Τέλος, εξετάζεται η ευκολία μάθησης των συστημάτων όπου είναι άρρηκτα συνδεδεμένη με τη διεπαφή χρήστη. Γενικά, το R και η Python απευθύνονται σε γνώστες των αντίστοιχων γλωσσών προγραμματισμού δηλαδή τουλάχιστον ενδιάμεσου επιπέδου χρήστες. Αυτό συμβαίνει διότι η διεπαφή χρήστη βασίζεται σε γραμμή εντολών και ακόμη και η επιλογή των πακέτων ειδικά για το R είναι μια δύσκολη υπόθεση λόγω του μεγάλου αριθμού των πακέτων που υπάρχουν. Ενώ, το Weka και το Rapidminer απευθύνεται και σε αρχάριους χρήστες λόγω της γραφικής διεπαφής χρήστη. Γενικά, δεν είναι απαραίτητη η γνώση της Java για να τρέξει κάποιος πειράματα Κατηγοριοποίησης και συσταδοποίησης στο Rapidminer και το Weka, βέβαια η χρήση του Weka είναι ακόμη πιο απλή από το Rapidminer, διότι απλά με δυο επιλογές μπορεί να τρέξει όλα τα πειράματα κατευθείαν ενώ στο Rapidminer χρειάζεται να γνωρίζει ποιες διαδικασίες θα εισάγει για κάθε πείραμα και εμπεριέχει ένα βαθμό δυσκολίας μέχρι να καταλάβει ο χρήστης ποιοι τελεστές μπορούν να συνδυαστούν σε συνάρτηση με το πείραμα που διεξάγεται. Σύμφωνα με τα παραπάνω, συνάγεται ότι ο βαθμός ευχρηστίας του κάθε συστήματος είναι ανάλογος με το γνωστικό υπόβαθρο του κάθε χρήστη όπως επίσης με το πείραμα που θέλει να υλοποιήσει αλλά και των επιμέρους κριτηρίων που αναλύθηκαν παραπάνω.

Κεφάλαιο 5

Πειραματική Αξιολόγηση

Επί του παρόντος, περισσότερα από πενήντα εργαλεία και λογισμικά εξόρυξης δεδομένων και ανακάλυψης γνώσης, είναι διαθέσιμα για διαφορετικές χρήσεις, όπως απαριθμούνται στο KDNuggets web site (http://www.kdnuggets.com).

Η ταχεία εισαγωγή νέων και αναβαθμισμένων εργαλείων είναι μια συναρπαστική εξέλιξη, που όμως δημιουργεί δυσκολίες και σύγχυση στους πιθανούς χρήστες που προσπαθούν να αξιολογήσουν τις ικανότητες των εργαλείων και να επιλέξουν το κατάλληλο για τις ανάγκες τους. Παρακάτω συνοψίζεται μια πρόσφατη εκτεταμένη αξιολόγηση των κορυφαίων εργαλείων εξόρυξης κειμένου ανοιχτού κώδικα, χρησιμοποιώντας αλγορίθμους μηχανικής μάθησης για προβλήματα κατηγοριοποίησης και συσταδοποίησης. Τα εργαλεία που επιλέχθηκαν είναι τα τέσσερα κορυφαία εργαλεία ανοιχτού κώδικα για την εξόρυξη δεδομένων: R, Weka, Rapid Miner και Python, με βάση τη ψηφοφορία του 2014 που διεξήχθη ανάμεσα σε χρήστες μέσω του KDNuggets web site, αλλά και τη τάση των ψηφοφοριών από το 2011 έως 2015.

Πιο αναλυτικά, περιγράφονται πολλαπλές κατηγορίες αξιολόγησης συμπεριλαμβανομένης της καταλληλότητας του εργαλείου στους προβλεπόμενους χρήστες και το περιβάλλον του υπολογιστή, τις δυνατότητες αυτοματοποίησης, τα είδη και η ποιότητα των αλγορίθμων που εφαρμόζονται, και την ευκολία χρήσης. Τέλος, κάθε εργαλείο αξιολογήθηκε εκτενώς σε πραγματικά σύνολα δεδομένων προκειμένου να διακριθούν με βάση την ακρίβειά τους και τις δυνάμεις τους.

5.1 Υλοποίηση Γενικού Πλαισίου

Η παρακάτω ενότητα παρέχει μια σύντομη εισαγωγή στις έννοιες και τους αλγορίθμους μηχανικής μάθησης που σχετίζονται με το αντικείμενο της μεταπτυχιακής διατριβής. Η Μηχανική Μάθηση αποτελεί έναν από τους παλαιότερους τομείς έρευνας της Τεχνητής Νοημοσύνης. Στόχος της είναι η δημιουργία συστημάτων τα οποία θα μπορούν να βελτιώνουν της απόδοσή τους στην εργασία που επιτελούν εκμεταλλευόμενα αυτόματα προηγούμενη εμπειρία από την εκτέλεση της εργασίας.

Ανάλογα με το είδος της γνώσης που παρέχεται για εκπαίδευση, διαιρούμε το πεδίο της μηχανικής μάθησης σε δύο μεγάλες κατηγορίες: στη μάθηση με επίβλεψη “supervised learning”, η οποία έρχεται σε αντιδιαστολή με τη μάθηση χωρίς επίβλεψη “unsupervised learning”.

Ολοκληρώνοντας τη σύντομη αυτή αναφορά στις θεμελιώδεις έννοιες της μηχανικής μάθησης, θα πραγματοποιηθεί μια πειραματική αξιολόγηση όπου θα εξεταστούν οι πιο αντιπροσωπευτικοί αλγόριθμοι μηχανικής μάθησης σε προβλήματα κατηγοριοποίησης και συσταδοποίησης, οι οποίοι θα αποτελέσουν τη βάση της αξιολόγησης των συστημάτων εξόρυξης κειμένου R, Weka, Rapid Miner και Python. Τα εργαλεία εξόρυξης κειμένου ελέγχθηκαν σύμφωνα με την ίδια αρχή, όπως οι υλοποιήσεις των αλγορίθμων μηχανικής μάθησης και οι μέθοδοι προ-επεξεργασίας, προκειμένου να διασφαλιστεί ότι τα αποτελέσματα μπορούν να συγκριθούν και είναι σχετικά ανθεκτικά με την πάροδο του χρόνου. Οι δοκιμές πραγματοποιήθηκαν από ένα μόνο Η/Υ και για κάθε σύνολο δεδομένων, ελέγχθηκαν οι λειτουργίες του κάθε λογισμικού ώστε να συμπληρωθούν τα διάφορα σημεία των πινάκων σύγκρισης που αναλύονται παρακάτω.

5.1.1 Σχεδίαση Συστήματος Μηχανικής Μάθησης

Το πρώτο στάδιο της σχεδίασης ενός συστήματος Μηχανικής Μάθησης συνίσταται στον προσδιορισμό της γνώσης που θα χρησιμοποιηθεί κατά την εκπαίδευσή του. Θα πρέπει να σημειωθεί πως οι όποιες επιλογές ακολουθηθούν, θα έχουν άμεση επίδραση στην απόδοση του συστήματος.

Το πρωταρχικό πολύ βασικό θέμα είναι η αναπαράσταση του κειμένου, κατά την οποία θα πραγματοποιηθεί η ανεύρεση προτύπων "patterns" ανάμεσα στα σύνολα δεδομένων που περιλαμβάνονται στα κείμενα, τα οποία αποτελούν γραπτή μορφή της φυσικής γλώσσας. Έπειτα, θα εισαχθούν μέθοδοι επεξεργασίας φυσικής γλώσσας στις διαδικασίες της εξόρυξης κειμένου, ώστε να μπορούν να μελετηθούν τα νοήματα των δεδομένων που περιέχονται στα κείμενα με σκοπό να εξαχθεί χρήσιμη γνώση. Κατά τη διαδικασία της προ-επεξεργασίας των συνόλων των δεδομένων, στόχος είναι η εξαγωγή των γνωρισμάτων "features" προκειμένου να δημιουργηθούν στη συνέχεια οι διανυσματικές αναπαραστάσεις των αρχείων κειμένου.

Στην πειραματική αξιολόγηση που ακολουθεί, θα χρησιμοποιηθεί το μοντέλο διανυσματικού χώρου για την αναπαράσταση των κειμένων, προκειμένου να κατασκευαστεί ο πίνακας όρων-εγγράφων. Προκειμένου να μειωθεί η πολυπλοκότητα χρόνου και χώρου, κρίνεται απαραίτητη η μείωση της διάστασης των διανυσμάτων των όρων στο αρχικό μοντέλο διανυσματικού χώρου. Παράλληλα, για να επιτευχθεί στη συνέχεια μια ικανοποιητική συσταδοποίηση και κατηγοριοποίηση των εγγράφων, θα χρησιμοποιηθούν οι πιο γνωστές τεχνικές προ-επεξεργασίας. Οι κυριότερες μέθοδοι προ-επεξεργασίας υποστηρίζονται και από τα τέσσερα υπό εξέταση λογισμικά και χρησιμοποιούνται σε όλα τα πειράματα, είναι οι εξής:

- Δεδομένου ότι υπάρχουν λέξεις όμοιας ρίζας αλλά με διαφορετική μορφή (λόγω διαφορετικής κλίσης, καταλήξεων κλπ), οι οποίες παρουσιάζουν προφανώς ομοιότητα, επιλέγεται η τεχνική "Snowball Stemmer" ώστε να συγχωνευθούν οι όροι βασιζόμενοι στην εξαγωγή της ρίζας κάθε λέξης,

- στη συνέχεια απομακρύνονται οι κοινές, μη σημαντικές λέξεις "stop words" οι οποίες δεν προσδίδουν καμία θεματική ιδιότητα στα σύνολα των δεδομένων,

- έπειτα, πραγματοποιείται η μετατροπή του τύπου των γραμμάτων σε πεζά "transform cases",

- και τέλος, επιλέγεται η ετικετοποίηση των μερών του λόγου των ακατέργαστων μέχρι αυτή τη στιγμή κειμένων προκειμένου να βρεθούν οι λέξεις που απαρτίζουν κάθε κείμενο και να διαχωριστούν από μη λεκτικά σύμβολα όπως είναι τα σημεία στίξης, "Tokenization".

Επίσης, αποδίδεται βάρος σε κάθε όρο "term weighting", με βάση τη συχνότητα εμφάνισής του σε κάθε έγγραφο ξεχωριστά αλλά και στη συλλογή των εγγράφων συνολικά, χρησιμοποιώντας τη συνάρτηση tfidf "term frequency inverse document frequency" (Sehgal, 2004) η οποία εγγυάται ότι:

- όσο πιο συχνά εμφανίζεται ένας όρος σε κάποιο κείμενο, τόσο πιο αντιπροσωπευτικός είναι για το περιεχόμενο του κειμένου και,

- όσο πιο συχνά εμφανίζεται ένας όρος συνολικά στο σώμα εκπαίδευσης, τόσο μικρότερη είναι η διαχωριστική του ικανότητα,

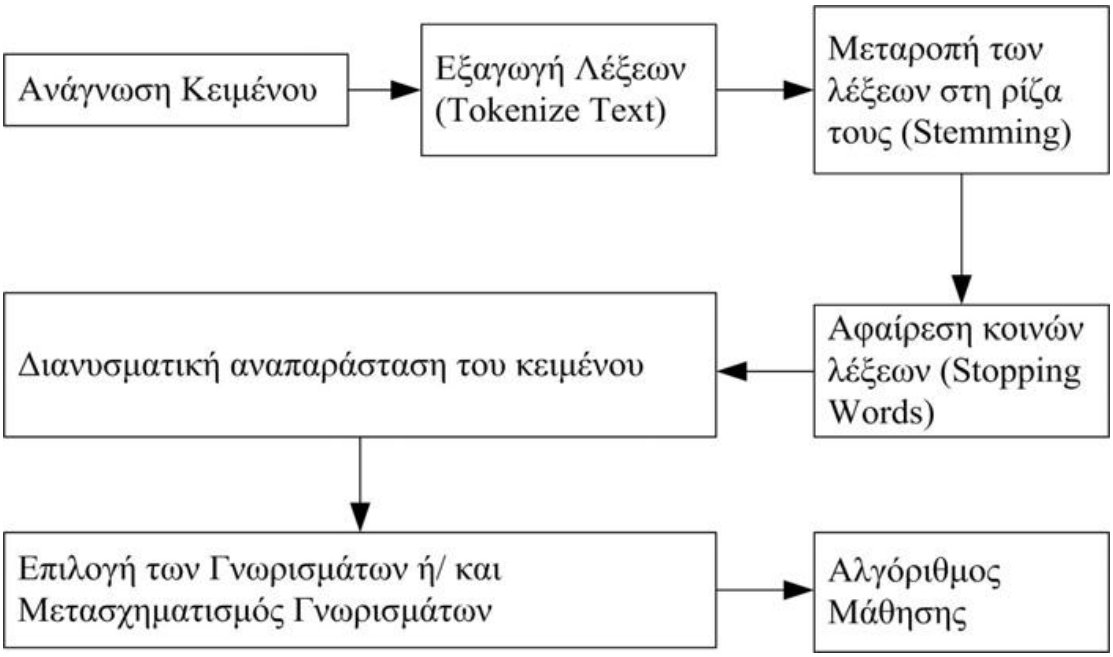
Ενώ στη συνέχεια, αφαιρούνται οι όροι οι οποίοι λόγω της πολύ σπάνιας και της πολύ συχνής εμφάνισης τους συνολικά στα έγγραφα δεν έχουν ιδιαίτερο ενδιαφέρον για την εύρεση ομοιότητας μεταξύ των κειμένων "pruning", με ποσοστό ελάχιστης και μέγιστης συχνότητας εμφάνισης κάτω του 30% και άνω του 70%, διότι έχει παρατηρηθεί πως οι 10 συχνότερες λέξεις της αγγλικής γλώσσας αποτελούν το 20-30% των λεκτικών μονάδων σε ένα κείμενο.

Το επόμενο βήμα είναι η διανυσματική αναπαράσταση του κειμένου "vector representation". Έτσι, κάθε αρχείο κειμένου από τα σύνολα δεδομένων που εξετάζονται παρακάτω μετατρέπεται σε ένα διάνυσμα όρων "term vector" στο οποίο κάθε όρος αποτελεί ένα μοναδικό ανεξάρτητο γνώρισμα. Κάθε στοιχείο σε αυτό το διάνυσμα έχει και μια τιμή η οποία αντιστοιχεί στην εμφάνιση του όρου μέσα στο κείμενο.

Οι προαναφερθείσες διαδικασίες επιλέχθηκαν έχοντας ως στόχο τη μείωση της διαστασιμότητας του χώρου και την ευκολότερη ανάδειξη των σημαντικών όρων, που

μπορούν βέβαια κάποιες φορές να οδηγήσουν σε μείωση της αποτελεσματικότητας των αλγορίθμων μηχανικής μάθησης.

Ο Sebastiani [18] έδωσε μια αρκετά καλή γραφική αναπαράσταση της διαδικασίας των επιμέρους δραστηριοτήτων της εξόρυξης κειμένου, χρησιμοποιώντας τους αλγορίθμους μηχανικής μάθησης, όπως φαίνεται στην Εικόνα 42.



Εικόνα 42. Διαδικασία Δραστηριοτήτων Συστήματος Μηχανικής Μάθησης.

5.1.2 Αλγόριθμοι Κατηγοριοποίησης

Στο πρώτο μέρος των πειραμάτων που διεξήχθησαν στην παρούσα μεταπτυχιακή διατριβή, το αντικείμενο είναι η διαδικασία της κατηγοριοποίησης, για σύνολα δεδομένων των οποίων τα γνωρίσματα των κλάσεων αποτελούνται από δύο αλλά και από πολλές τιμές. Επίσης, υλοποιήθηκαν οι πιο αντιπροσωπευτικοί αλγόριθμοι κατηγοριοποίησης οι οποίοι αναλύονται παρακάτω: Δέντρα Απόφασης (Decision Tree), Αυτόματη Εκμάθηση Κανόνων (Rule Induction), Μέθοδος Κοντινότερου Γείτονα (K-nn), Αφελής Κατηγοριοποιητής Bayes (Naïve Bayes) και Μηχανές Διανυσματικής Υποστήριξης (SVM).

Η κατηγοριοποίηση είναι μία μέθοδος εξόρυξης δεδομένων κατά την οποία ένα στοιχείο ανατίθεται σε ένα προκαθορισμένο σύνολο κατηγοριών “target category”. Γενικά, κατά

τη διαδικασία της κατηγοριοποίησης το σύνολο δεδομένων χωρίζεται σε ένα σύνολο εκπαίδευσης “training set”, και ένα σύνολο ελέγχου “test set” το οποίο χρησιμοποιείται για επικύρωση “validation” της απόδοσης του αλγορίθμου κατηγοριοποίησης. Στο σύνολο εκπαίδευσης, το συνολικό κείμενο μετατρέπεται σε ξεχωριστές λέξεις. Όλες οι λέξεις που περιλαμβάνονται στα κείμενα αναπαριστούνται ως ένας «σάκος λέξεων» “bag-of-words”.

Στη συνέχεια, τα δεδομένα εκπαίδευσης αναλύονται από τους αλγορίθμους κατηγοριοποίησης που έχουν επιλεγεί, προκειμένου να σχηματιστεί το μοντέλο το οποίο ονομάζεται κατηγοριοποιητής “classifier” [22].

Για την αξιολόγηση του μοντέλου, χρησιμοποιείται ένα σύνολο δοκιμαστικών δεδομένων “test data” το οποίο είναι διαφορετικό από το σύνολο εκπαίδευσης που χρησιμοποιείται για τη δημιουργία του μοντέλου. Το μοντέλο κατηγοριοποιεί τα δεδομένα. Στη συνέχεια συγκρίνεται η τιμή της πρόβλεψης της κατηγορίας που σχηματίστηκε από τα δοκιμαστικά δεδομένα με την υπάρχουσα τιμή των δεδομένων εκπαίδευσης. Για την αξιολόγηση του μοντέλου χρησιμοποιούνται διάφορες μετρικές οι οποίες αναλύονται παρακάτω και κρίνουν εάν το μοντέλο είναι αποδεκτό για την συγκεκριμένη χρήση. Η απόδοση των κατηγοριοποιητών οφείλεται αποκλειστικά στα χαρακτηριστικά των δεδομένων.

i. Δέντρα απόφασης

Τα δέντρα απόφασης είναι μια τεχνική κατηγοριοποίησης που χρησιμοποιείται ευρέως. Μπορούν να θεωρηθούν ως ένα ισχυρό και δημοφιλές εργαλείο για διαδικασίες κατηγοριοποίησης και πρόβλεψης. Είναι ένας κατηγοριοποιητής που έχει παρόμοια μορφή με αυτή ενός δέντρου και έχει τα ακόλουθα δομικά συστατικά [22,41]:

- Κόμβος ρίζα (Root node): ο πιο αριστερός κόμβος σε ένα δέντρο απόφασης.
- Κόμβος απόφασης (Decision node): καθορίζει μία δοκιμή σε ένα μεμονωμένο χαρακτηριστικό.
- Κόμβος φύλλο (Leaf node): υποδεικνύει τη τιμή του χαρακτηριστικού προορισμού.

- Ακμές (Edge): διαχωρισμός ενός χαρακτηριστικού γνωρίσματος.

- Κόμβος τέλους (End-point): ο πιο δεξιός κόμβος που αναπαριστά το τελικό αποτέλεσμα.

Κατασκευάζεται με χρήση της προσέγγισης του "διαίρει και βασίλευε". Κάθε διαδρομή στο δέντρο απόφασης καθορίζει και ένα κανόνα απόφασης. Συνήθως, ακολουθεί μια άπληστη προσέγγιση, από την κορυφή στον πυθμένα. Αναδρομικά, από τον κόμβο ρίζα ως τον τελικό κόμβο για τον καθορισμό του τελικού αποτελέσματος και επομένως την αντιμετώπιση αβεβαιοτήτων. Κατ' ουσίαν, ένα δένδρο απόφασης αναπαριστά μια διάζευξη συζευγμένων περιορισμών επί ενός συνόλου δεδομένων απόφασης.

Κάποια από τα πλεονεκτήματα του δέντρου αποφάσεων είναι ότι παρέχει αντικειμενική ανάλυση για την λήψη αποφάσεων, επιτρέπει ευελιξία και είναι αποτελεσματικό στη λήψη αποφάσεων. Ένα μεγάλο μειονέκτημα είναι ότι όλη η διαδικασία βασίζεται στην ακρίβεια των δεδομένων εισόδου και απαιτεί ποιοτικά δεδομένα για τον καθορισμό της ακρίβειας του αποτελέσματος.

ii. Bayesian Δίκτυα

Το δίκτυο Bayesian είναι μια γραφική αναπαράσταση της κατανομής πιθανοτήτων. Το Bayesian δίκτυο αποτελείται από δύο συστατικά. Το πρώτο είναι ένας κατευθυνόμενος άκυκλος γράφος του οποίου οι κόμβοι ονομάζονται τυχαίες μεταβλητές και οι ακμές μεταξύ των κόμβων ή τυχαίων μεταβλητών αναπαριστούν τις πιθανοτικές εξαρτήσεις μεταξύ των αντίστοιχων τυχαίων μεταβλητών. Το δεύτερο συστατικό είναι ένα σύνολο παραμέτρων που περιγράφουν την υποθετική πιθανότητα κάθε μεταβλητής που κληρονομεί από τους γονείς της. Οι υποθετικές εξαρτήσεις στο γράφο υπολογίζονται από στατιστικές και υπολογιστικές μεθόδους. Άρα, τα δίκτυα αυτά συνδυάζουν τις ιδιότητες της επιστήμης των υπολογιστών και της στατιστικής [22,41].

Η προσέγγιση της απλής κατηγοριοποίησης κατά Bayes έχει πλεονεκτήματα. Πρώτον, είναι εύκολο να χρησιμοποιηθεί. Δεύτερον, αντίθετα με άλλες προσεγγίσεις κατηγοριοποίησης, απαιτείται μόνο ένα πέρασμα των δεδομένων εκπαίδευσης. Η προσέγγιση του Bayes μπορεί εύκολα να χειριστεί ελλιπή δεδομένα, απλά παραλείποντας εκείνη την πιθανότητα όταν υπολογίζει τις πιθανοφάνειες του μέλους

για κάθε κατηγορία. Σε εκείνες τις περιπτώσεις που υπάρχουν απλές συσχετίσεις, η τεχνική συνήθως δίνει καλά αποτελέσματα.

Από την άλλη πλευρά, παρόλο που η απλοϊκή προσέγγιση του Bayes είναι αρκετά απλή στη χρήση της, δεν δίνει πάντα ικανοποιητικά αποτελέσματα. Πρώτον, τα γνωρίσματα δεν είναι ανεξάρτητα. Θα μπορούσαμε να χρησιμοποιήσουμε ένα υποσύνολο των γνωρισμάτων αγνοώντας εκείνα που εξαρτώνται από άλλα. Η τεχνική αυτή δεν μπορεί να χειριστεί συνεχή δεδομένα. Η διαίρεση των συνεχών τιμών σε διαστήματα θα μπορούσε να χρησιμοποιηθεί για να λύσει αυτό το πρόβλημα, αλλά η διαίρεση του πεδίου σε διαστήματα δεν είναι μια απλή λειτουργία, και ο τρόπος με τον οποίο θα γίνει αυτό μπορεί φυσικά να επηρεάσει τα αποτελέσματα.

iii. Αυτόματη Εκμάθηση Κανόνων

Μια συγγενική μεθοδολογία επαγωγικής κατασκευής ταξινομητών με αυτή των δένδρων απόφασης αποτελεί η αυτόματη εκμάθηση κανόνων, χαρακτηριστική για την ικανότητά της να παράγει ιδιαίτερα εύληπτα μοντέλα με τη μορφή κανόνων συμπερασμού (if-then rules). Πιο συγκεκριμένα, αν ένας κανόνας κατηγοριοποίησης (classification rule) r = a,c, αποτελείται από το if ή αλλιώς πρότερο τμήμα (antecedent), α, και από το then ή το επακόλουθο τμήμα (consequent), c , η υπόθεση περιέχει ένα κατηγόρημα το οποίο μπορεί να αξιολογηθεί σαν αληθές ή ψευδές ως προς τα δεδομένα της εκπαίδευσης [41].

Όταν δημιουργούνται κανόνες, μόνο μια κατηγορία πρέπει να εξετάζεται κάθε φορά. Οι αλγόριθμοι κάλυψης όπως λέγονται προσπαθούν να δημιουργήσουν κανόνες έτσι ώστε να καλύψουν μια συγκεκριμένη κατηγορία. Αυτοί δημιουργούν τον καλύτερο δυνατό κανόνα με τη βελτιστοποίηση της επιθυμητής πιθανότητας κατηγοριοποίησης. Συνήθως επιλέγεται το «καλύτερο» ζευγάρι γνωρίσματος- τιμής, αντίθετα από την επιλογή του καλύτερου γνωρίσματος που λαμβάνει χώρα σε αλγορίθμους βασισμένους σε δένδρα. Η βασική ιδέα είναι η επιλογή του καλύτερου γνωρίσματος για την εκτέλεση της κατηγοριοποίησης με βάση τα δεδομένα εκπαίδευσης.

iv. Κατηγοριοποιητές k-Κοντινότερου γείτονα

Ο μηχανισμός k-κοντινότερου γείτονα είναι ένας μηχανισμός που χρησιμοποιείται για την ταυτοποίηση άγνωστων σημείων δεδομένων σύμφωνα με τον κοντινότερο γείτονα του οποίου η τιμή είναι γνωστή. Ο k-κοντινότερος γείτονας δουλεύει με την υπόθεση ότι τα δεδομένα περιέχονται σε ένα χώρο χαρακτηριστικών [22]. Ως εκ τούτου, όλα τα σημεία περιέχονται σε αυτόν. Για την εύρεση της απόστασης μεταξύ των σημείων χρησιμοποιείται η Ευκλείδεια ή Hamming απόσταση σύμφωνα με τον χρησιμοποιούμενο τύπο δεδομένων των κλάσεων δεδομένων. Ο μηχανισμός του k-κοντινότερου γείτονα είναι εύκολος να υλοποιηθεί, πράγμα που κάνει τη διαδικασία υλοποίησης και αποσφαλμάτωσης γρηγορότερη. Μπορεί επίσης να βοηθήσει στην εύκολη ανάλυση των γειτονικών σημείων.

Ως εκ τούτου, το κύριο πλεονέκτημα αυτής της μεθόδου είναι ότι η εκπαίδευση δεδομένων μπορεί να γίνει με πιο γρήγορο τρόπο, απλό και εύκολο στη μάθηση. Μεγάλα σύνολα δεδομένων εκπαίδευσης μπορούν να προσδιοριστούν, πράγμα που τον κάνει ένα εύρωστο μηχανισμό. Αρκετές τεχνικές μείωσης θορύβου μπορούν να χρησιμοποιηθούν για τη βελτίωση του μηχανισμού Κατηγοριοποίησης. Κάποια από τα μειονεκτήματα του είναι η εξαρτώμενη μνήμη, υπολογιστική περιπλοκότητα και η εξάρτηση του από τη k-τιμή. Επίσης, απαιτεί μεγάλο υπολογιστικό χρόνο και άρα είναι μια αργή τεχνική αφού όλες οι διαδικασίες πραγματοποιούνται κατά τη διάρκεια του χρόνου εκτέλεσης.

v. Μηχανές Υποστήριξης Διανυσμάτων

Στόχος του αλγορίθμου αυτού είναι η επιλογή ενός μικρού αριθμού στιγμιοτύπων εκπαίδευσης από κάθε κλάση, των διανυσμάτων υποστήριξης (support vectors), που συνορεύουν στο χώρο του προβλήματος με στιγμιότυπα άλλων κλάσεων. Τα επιλεγμένα στιγμιότυπα χρησιμοποιούνται για την κατασκευή μιας γραμμικής συνάρτησης διάκρισης (discriminant function), ικανής να τα διαχωρίσει όσο το δυνατόν περισσότερο [22,41].

Τα συστήματα Κατηγοριοποίησης που βασίζονται στον αλγόριθμο αυτό αποτελούν σήμερα μια από τις δημοφιλέστερες προσεγγίσεις στο χώρο της κατηγοριοποίησης κειμένου, λόγω της ευρωστίας, της αποτελεσματικότητας και της ταχύτητας που

επιδεικνύουν, αλλά και της ικανότητάς τους να παράγουν μη γραμμικές επιφάνειες απόφασης, καθιστώντας έτσι υπολογιστικά εφικτή την επίλυση ενός μεγάλου αριθμού πρακτικών προβλημάτων μάθησης που δεν μπορούν να αντιμετωπιστούν από γραμμικά μοντέλα.

Καθοριστική σημασία για την ικανότητα γενίκευσης του αλγορίθμου φέρει η επιλογή της παραμέτρου c, καθώς όσο μεγαλύτερη είναι η τιμή της, τόσο πιο αυστηρό είναι το επαγόμενο μοντέλο στον προσδιορισμό ενός υπερεπιπέδου ικανού να διαχωρίσει σωστά την πλειοψηφία των διανυσμάτων εκπαίδευσης, ακόμα και αυτών εντός του περιθωρίου.

Όπως απέδειξαν οι Boser, Guyon και Vapnik [03], ο υπό εξέταση αλγόριθμος είναι εφαρμόσιμος και στην περίπτωση που η συνάρτηση διάκρισης δεν είναι γραμμική ως προς τα δεδομένα εκπαίδευσης. Αυτό που απαιτείται είναι ο μετασχηματισμός του χώρου του προβλήματος σε έναν άλλο χώρο, μεγαλύτερης ή και άπειρης διάστασης μέσω μιας απεικόνισης $\Phi : S \rightarrow H$. Εφαρμόζοντας το τέχνασμα αυτό, επιτυγχάνουμε την κατασκευή μιας μηχανής διανυσμάτων υποστήριξης σ' ένα χώρο απείρων διαστάσεων, ανάγοντας τη μη γραμμική επιφάνεια διάκρισης του αρχικού χώρου S σε γραμμική, χωρίς να εισάγουμε επιπλέον υπολογιστικό φόρτο στο σύστημα. Ένα ακόμα πλεονέκτημα των SVMs είναι η ικανότητά τους να χειρίζονται πολύ μεγάλους χώρους χαρακτηριστικών, καθιστώντας το στάδιο της επιλογής χαρακτηριστικών, που συνήθως προηγείται αυτού της εκπαίδευσης, περιττό. Επίσης, αξιοσημείωτη είναι και η ανεκτικότητα που παρουσιάζουν όσον αφορά στο πλήθος των στιγμιοτύπων εκπαίδευσης, ιδιαίτερα όταν αυτό διαφέρει μεταξύ των δύο κλάσεων, καθώς τα SVMs δεν επιδιώκουν να ελαχιστοποιήσουν το σφάλμα των δεδομένων εκπαίδευσης, αλλά να τα διαχωρίσουν αποτελεσματικά σε ένα χώρο μεγάλης διάστασης. Όσον αφορά στους χρόνους εκπαίδευσης και ελέγχου του αλγορίθμου, αυτοί αποδεικνύονται κάπως αυξημένοι, ιδιαίτερα όταν η διάσταση του χώρου είναι μεγάλη, ή όταν η συνάρτηση διάκρισης δεν είναι γραμμική.

5.1.3 Αλγόριθμοι Συσταδοποίησης

Στο δεύτερο μέρος της πειραματικής αξιολόγησης που διεξήχθη, το αντικείμενο είναι η διαδικασία της συσταδοποίησης. Επιλέχθηκαν οι παρακάτω αλγόριθμοι

συσταδοποίησης προς υλοποίηση, a. Ιεραρχική συσταδοποίηση και πιο συγκεκριμένα η σωρευτική μέθοδος (agglomerative), b. k-means, c. DBSCAN.

Γενικά, η συσταδοποίηση είναι η διαδικασία διαίρεσης ενός συνόλου δεδομένων σε αμοιβαία αποκλειόμενες ομάδες, τέτοιες ώστε τα μέλη κάθε ομάδας να είναι όσο κοντά γίνεται το ένα με το άλλο, ενώ οι διαφορετικές ομάδες να είναι όσο το δυνατόν πιο μακριά η μία από την άλλη, όπου η απόσταση μετράται σε σχέση με όλες τις διαθέσιμες μεταβλητές. Με την αναπαράσταση των δεδομένων με λιγότερες συστάδες σίγουρα χάνονται ορισμένες μικρολεπτομέρειες, αλλά επιτυγχάνεται απλοποίηση (μία συστάδα είναι μια ταξινομημένη λίστα αντικειμένων, τα αντικείμενα της οποίας έχουν κάποια κοινά χαρακτηριστικά. Τα αντικείμενα ανήκουν σε ένα διάστημα [a,b])[22, 41].

Από τη πλευρά της μηχανικής εκμάθησης, οι συστάδες αντιστοιχούν σε κρυμμένα μοτίβα, η αναζήτηση συστάδας ανήκει στις τεχνικές μη-καθοδηγούμενης εκμάθησης και το σύστημα που προκύπτει αναπαριστά ένα σχήμα δεδομένων.

i. Ιεραρχική συσταδοποίηση (Hierarchical Clustering)

Οι ιεραρχικοί αλγόριθμοι δημιουργούν μία ιεραρχική αποσύνθεση των αντικειμένων. Είναι είτε συσωρευτικοί (agglomerative) (από κάτω προς τα πάνω) είτε διαιρετικοί (divisive) (από πάνω προς τα κάτω)[22,41]:

1. Οι συσωρευτικοί αλγόριθμοι ξεκινούν με το κάθε αντικείμενο σαν ξεχωριστή συστάδα, και διαδοχικά συγχωνεύουν ομάδες σύμφωνα με ένα μέτρο απόστασης. Η συσταδοποίηση μπορεί να σταματήσει όταν όλα τα αντικείμενα είναι σε μία ομάδα ή σε οποιοδήποτε άλλο σημείο θέλει ο χρήστης. Αυτοί οι μέθοδοι συνήθως ακολουθούν μία άπληστη από κάτω προς τα πάνω συγχώνευση.

2. Οι διαιρετικοί αλγόριθμοι ακολουθούν την αντίθετη στρατηγική. Ξεκινούν από μία ομάδα με όλα τα αντικείμενα και διαδοχικά χωρίζουν τις ομάδες σε μικρότερες, μέχρι κάθε αντικείμενο να εμπίπτει σε μία συστάδα, ή όπως επιθυμείται. Οι διαιρετικές προσεγγίσεις χωρίζουν τα αντικείμενα των δεδομένων σε πολλές ομάδες σε κάθε βήμα, και ακολουθούν το ίδιο μοτίβο μέχρι όλα τα αντικείμενα να εμπίπτουν σε διαφορετικές συστάδες. Αυτό είναι

παρόμοιο με τη τεχνική που χρησιμοποιούν οι αλγόριθμοι του διαίρει και βασίλευε.

Η ομοιότητα μεταξύ ομάδων είναι βασικό σημείο του αλγορίθμου πάνω στο οποίο έχουν προταθεί διάφορα μέτρα όπως τα ακόλουθα:

- MIN: Λαμβάνει τη μικρότερη απόσταση ή ισοδύναμα τη μεγαλύτερη ομοιότητα.

- MAX: Λαμβάνει την ομοιότητα ομάδων δύο εγγράφων που βρίσκονται πιο μακριά.

- GROUP: Λαμβάνει το μέσο όρο των αποστάσεων όλων των εγγράφων.

- WARD: Η ομοιότητα μεταξύ ομάδων λαμβάνεται ως η αύξηση του κόστους από τη συνένωση των δύο ομάδων.

Γενικά, η πολυπλοκότητα των αλγορίθμων σε χώρο και σε χρόνο θέτει περιορισμούς για το μέγεθος των δεδομένων που μπορούν να χρησιμοποιηθούν, πόσο μάλλον για κειμενικά δεδομένα που οι διαστάσεις είναι εκ των προτέρων μεγάλες. Επίσης, δε μπορεί να καθοριστεί ένας στόχος εκ των προτέρων ο οποίος θα βελτιστοποιηθεί. Παρότι οι αλγόριθμοι αποφασίζουν τοπικά για το ποιες ομάδες είναι καλύτερο να συγχωνευθούν, η απόφαση αυτή είναι τελική και δε μπορεί να αναστραφεί σε επόμενο βήμα. Αυτό εμποδίζει ένα τοπικό κριτήριο βελτιστοποίησης να γίνει καθολικό.

Παρόλα αυτά το μεγάλο πλεονέκτημα των μεθόδων αυτών της τμηματικής συσταδοποίησης είναι η παραγωγή ενός δενδρογράμματος, το οποίο παρουσιάζει τόσο τις σχέσεις ομάδων – υποομάδων, όσο και η σειρά με την οποία οι ομάδες ενώθηκαν. Επίσης, οι αλγόριθμοι της κατηγορίας αυτής μπορούν να τροποποιηθούν ώστε να χειρίζονται ομάδες διαφορετικών μεγεθών. Υπάρχουν δύο προσεγγίσεις η μη σταθμισμένη, που αντιμετωπίζει όλες τις ομάδες ισότιμα και η σταθμισμένη, που λαμβάνει υπόψη τον αριθμό των εγγράφων σε κάθε ομάδα.

ii. Συσταδοποίηση βάση πυκνότητας (Density-Based Clustering)

Οι αλγόριθμοι αυτοί ομαδοποιούν αντικείμενα σύμφωνα με συγκεκριμένες αντικειμενικές συναρτήσεις πυκνότητας. Σαν πυκνότητα ορίζεται ο αριθμός

των αντικειμένων σε μια συγκεκριμένη γειτονιά αντικειμένων δεδομένων. Αυτό αντιμετωπίζει το πρόβλημα των απομονωμένων σημείων καθώς εγγυάται ότι ένα τέτοιο σημείο (ή ένα μικρό σύνολο απομονωμένων σημείων) δε θα δημιουργήσει συστάδα. Μία παράμετρος εισόδου, MinPts, δείχνει το ελάχιστο πλήθος σημείων σε κάποια συστάδα. Επιπλέον, για κάθε σημείο συστάδας θα πρέπει να υπάρχει κάποιο άλλο σημείο στη συστάδα, η απόσταση του οποίου από το αρχικό σημείο να είναι μικρότερη απ' το κατώφλι εισόδου, Eps. Η Eps-γειτονιά, ή αλλιώς απλά, γειτονιά, ενός σημείου είναι το σύνολο των σημείων σε απόσταση Eps από το σημείο. Το επιθυμητό πλήθος συστάδων, k, δεν αποτελεί είσοδο αλλά αντιθέτως προσδιορίζεται από τον ίδιο τον αλγόριθμο. Τα σημεία αυτά σχηματίζουν το βασικό τμήμα της συστάδας δεδομένου ότι βρίσκονται όλα κοντά μεταξύ τους. Ένα άμεσα προσεγγίσιμο σημείο με βάση την πυκνότητα θα πρέπει να βρίσκεται κοντά σε ένα απ' αυτά τα σημεία πυρήνες, αλλά δεν χρειάζεται να είναι και το ίδιο πυρήνας. Στην περίπτωση αυτή, καλείται οριακό σημείο (border point). Άρα μία συστάδα ορίζεται να είναι ένα σύνολο από σημεία συνδεδεμένα με βάση την πυκνότητα με τη μέγιστη προσεγγισιμότητα με βάση την πυκνότητα. Ο θόρυβος είναι τα σημεία που δεν ανήκουν σε καμία συστάδα.

Τα πλεονεκτήματα των μεθόδων που είναι βασισμένες στην πυκνότητα είναι ότι μπορούν να ανακαλύψουν συστάδες με αυθαίρετες μορφές, δεν χρειάζεται να προκαθοριστεί ο αριθμός των συστάδων, αναγνωρίζουν τα ακραία σημεία και δεν επηρεάζονται από θόρυβο. Επίσης, ένα μειονέκτημα είναι ότι επηρεάζεται από τις τιμές των παραμέτρων Eps και MinPts, οι οποίες είναι δύσκολο να προσδιοριστούν και η χρήση δείγματος για να περιοριστεί το μέγεθος της εισόδου στην εφαρμογή των αλγορίθμων που βασίζονται στην πυκνότητα δεν είναι εφικτή. Ο λόγος είναι ότι ακόμα και αν το δείγμα είναι μεγάλο, μπορεί να υπάρχουν μεγάλες διακυμάνσεις στην πυκνότητα των σημείων μέσα σε κάθε συστάδα στο τυχαίο δείγμα.

iii. Τμηματική συσταδοποίηση (Partitional Clustering)

Δοσμένου ενός συνόλου δεδομένων n αντικειμένων, ο αλγόριθμος τμηματικής συσταδοποίησης κατασκευάζει k τμήματα των δεδομένων, όπου κάθε συστάδα βελτιστοποιεί ένα κριτήριο συσταδοποίησης, όπως η ελαχιστοποίηση του αθροίσματος των τετραγώνων των αποστάσεων από το μέσο για κάθε συστάδα.

Ένα από τα μειονεκτήματα αυτών των αλγορίθμων είναι η υψηλή τους πολυπλοκότητα, αφού μερικοί από αυτούς εξαντλητικά απαριθμούν όλες τις πιθανές ομαδοποιήσεις και προσπαθούν να βρουν τη βέλτιστη. Ακόμα και για μικρό αριθμό αντικειμένων, ο αριθμός των διαμερίσεων είναι τεράστιος. Για αυτό, οι κοινές λύσεις ξεκινούν με μια αρχική, συνήθως τυχαία διαίρεση και συνεχίζουν με την εξευγένιση της. Μία καλύτερη πρακτική θα ήταν η εκτέλεση του αλγόριθμου τμηματοποίησης για διαφορετικά σύνολα k αρχικών σημείων και διερεύνηση για το αν όλες οι λύσεις οδηγούν στην ίδια τελική διαμέριση.

Οι αλγόριθμοι τμηματικής συσταδοποίησης προσπαθούν να βελτιώσουν τοπικά ένα συγκεκριμένο κριτήριο. Πρώτα, υπολογίζουν τις τιμές ομοιότητας ή απόστασης, διατάσσουν τα αποτελέσματα και επιλέγουν αυτό που βελτιστοποιεί το κριτήριο. Ως εκ τούτου, μπορούν να θεωρηθούν στη πλειοψηφία τους ως άπληστοι αλγόριθμοι.

5.1.4 Μέθοδοι Εκτίμησης της Αποτελεσματικότητας

Ιδιαίτερα σημαντικό στάδιο του κύκλου ζωής ενός συστήματος εξόρυξης κειμένου αποτελεί η αξιολόγηση της αποτελεσματικότητας του, καθώς παρέχει τη δυνατότητα στο σχεδιαστή να προβεί αφενός στις κατάλληλες ρυθμίσεις των παραμέτρων των επιμέρους υποσυστημάτων του και στην αποτίμηση των σχεδιαστικών επιλογών που ακολουθήθηκαν, αφετέρου στη σύγκρισή του με διαφορετικές προσεγγίσεις που ενδεχομένως να έχουν υλοποιηθεί.

Πιο συγκεκριμένα, τα πιο σημαντικά κριτήρια για την αξία ενός συστήματος, τα οποία και θα αναλυθούν παρακάτω, είναι η αποτελεσματικότητά του (effectiveness) στο έργο της κατηγοριοποίησης και της συσταδοποίησης, η οποία αντικατοπτρίζει την ακρίβεια των προβλέψεων αλλά και η αποδοτικότητα (efficiency) η οποία αναφέρεται στην χρονική και χωρική πολυπλοκότητα των επιμέρους αλγορίθμων.

Λόγω της εγγενούς υποκειμενικότητας που χαρακτηρίζει τη διαδικασία της κατηγοριοποίησης και της συσταδοποίησης των κειμένων, η θεωρητική εκτίμηση της αποτελεσματικότητας ενός συστήματος, αποδεικνύοντας την ορθότητα και την πληρότητά του, δεν είναι δυνατή. Γι' αυτό το λόγο χρησιμοποιούνται πειραματικές προσεγγίσεις αξιολόγησης, μέσω των αποφάσεων Κατηγοριοποίησης στις οποίες προβαίνει το σύστημα επί ενός δοκιμαστικού σώματος εγγράφων (test dataset),

διαφορετικού από το σώμα που χρησιμοποιήθηκε κατά την εκπαίδευσή του (training dataset) για κάθε σύνολο δεδομένων. Πολλές φορές κρίνεται απαραίτητη η ύπαρξη ενός ακόμα σώματος εγγράφων, του σώματος επικύρωσης (validation dataset), με τα περιεχόμενά του να μην εμφανίζονται σε κανένα από τα δύο προηγούμενα. Το τελευταίο χρειάζεται όταν η εκπαίδευση του συστήματος περιλαμβάνει τον πειραματικό προσδιορισμό των τιμών κάποιων παραμέτρων του που μεγιστοποιούν το μέτρο αποτελεσματικότητας που έχει επιλεχθεί.

Η διαδικασία της αξιολόγησης ενός αλγορίθμου κατηγοριοποίησης γίνεται πολλές φορές εις βάρος της εκπαίδευσής του, καθώς το σύνολο των δεδομένων που προορίζεται γι' αυτήν περιορίζεται σημαντικά, ιδιαίτερα όταν τα διαθέσιμα στιγμιότυπα του προβλήματος δεν επαρκούν. Μια απλοϊκή προσπάθεια επίλυσης του προβλήματος αυτού στο στάδιο της αξιολόγησης της κατηγοριοποίησης, είναι η μέθοδος holdout, προϋποθέτει την κράτηση ενός ποσοστού των στιγμιοτύπων εκπαίδευσης για την κατασκευή του σώματος ελέγχου. Μειονεκτεί ωστόσο καθώς ένα αρκετά μεγάλο μέρος των δεδομένων δεν συμμετέχει καθόλου στο στάδιο της εκπαίδευσης. Στα πειράματα που θα αναλυθούν παρακάτω τα στιγμιότυπα εκπαίδευσης ορίστηκαν στο 75% του συνόλου των δεδομένων.

Η διαδικασία αξιολόγησης των αποτελεσμάτων ενός αλγορίθμου συσταδοποίησης ονομάζεται αξιολόγηση της εγκυρότητας των συστάδων (cluster validity assessment). Δύο κριτήρια μέτρησης έχουν προταθεί για την αξιολόγηση και την επιλογή ενός βέλτιστου σχήματος συσταδοποίησης [22]:

1. Συνοχή(compactness): Η απόσταση μεταξύ των σημείων κάθε συστάδας πρέπει να είναι όσο το δυνατόν πιο μικρή. Ένα κοινό μέτρο της συνοχής είναι η διακύμανση(variance) που πρέπει να είναι ελάχιστη.

2. Διαχωρισμός(separation): Οι συστάδες πρέπει να είναι μεταξύ τους διαχωρίσιμες.

    i. Μετρικές Αξιολόγησης Αλγορίθμων Κατηγοριοποίησης

Η εφαρμογή ενός αλγορίθμου κατηγοριοποίησης σε ένα σύνολο δεδομένων στοχεύει στην ανακάλυψη των ήδη προκαθορισμένων κλάσεων, για αυτό ονομάζεται και επιβλεπόμενη κατηγοριοποίηση. Στην ενότητα αυτή εξετάζονται τα πλέον σημαντικά

μέτρα αποτελεσματικότητας ενός συστήματος κατηγοριοποίησης, όσον αφορά στην ακρίβεια των προβλέψεών του. Δύο συχνά χρησιμοποιούμενα μέτρα, η Ορθότητα (Precision) και η Ανάκληση (Recall), γνωστά από την περιοχή της Ανάκτησης Πληροφορίας , εκφράζουν το βαθμό ορθότητας (degree of soundness) και πληρότητας (degree of completeness) αντίστοιχα του συστήματος Κατηγοριοποίησης, σε σχέση με το σύνολο των κλάσεων των προς κατηγοριοποίηση εγγράφων.

Εξετάζοντας τον πίνακα αποτελεσμάτων (confusion matrix) απεικονίζονται οι προβλέψεις που γίνονται κατά την κατηγοριοποίηση του μοντέλου. Οι γραμμές του πίνακα απεικονίζουν τις γνωστές κλάσεις των δεδομένων και οι στήλες τις προβλέψεις που έγιναν από το μοντέλο κατηγοριοποίησης. Πιο συγκεκριμένα, τα διαγώνια στοιχεία του πίνακα αποτελεσμάτων απεικονίζουν τον αριθμό των σωστών κατηγοριοποιήσεων που έγιναν για κάθε κλάση και τα υπόλοιπα στοιχεία απεικονίζουν τον αριθμό των λαθών. Οι περιπτώσεις που εμφανίζονται σε ένα πίνακα αποτελεσμάτων είναι οι εξής[22, 30]:

- TP (True Positive) : ο αριθμός των περιπτώσεων που ταξινομούνται σωστά σε αυτή την κατηγορία.
- TN (True Negative) : ο αριθμός των περιπτώσεων που σωστά απορρίφθηκαν από αυτή την κατηγορία.
- FP (False Positive): ο αριθμός των περιπτώσεων που κ απορρίφθηκαν από αυτή την κλάση.
- FN (False Negative) : ο αριθμός των περιπτώσεων που εσφαλμένα ταξινομήθηκαν στην εν λόγω κατηγορία.

Αναλυτικότερα, ως ορθότητα σχετική με μια κλάση, ορίζεται η σχέση [30]:

$$\text{Precision} = \frac{TP}{TP+FP}$$

Στη συνέχεια ως ανάκληση σχετική με μια κλάση ορίζεται η σχέση [30] :

$$\text{Recall} = \frac{TP}{TP+FN}$$

Οι συγκεκριμένες σχέσεις μπορούν να πάρουν τιμές από 0 έως 1.

Γενικότερα, παρατηρείται στην πράξη είναι πως μειώνοντας την ορθότητα, αυξάνεται η ανάκληση και αντίστροφα. Προκύπτει έτσι η ανάγκη για μια

μέτρικη που θα συνδυάζει και θα συνυπολογίζει τις δυο έννοιες. Το πιο απλό συνδυαστικό μέτρο είναι η ακρίβεια (accuracy), η οποία ορίζεται, όπως φαίνεται από την παρακάτω σχέση [30]:

$$\text{Accuracy} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}}$$

Η συγκεκριμένη σχέση μπορεί να λάβει τιμές από 0 έως 1. Το μέτρο της ακρίβειας μπορεί να είναι πολλές φορές παραπλανητικό. Μερικές φορές μπορεί να είναι επιθυμητό να επιλεγεί ένα μοντέλο με χαμηλότερη ακρίβεια, επειδή έχει μεγαλύτερη προβλεπτική δύναμη σε σχέση με το πρόβλημα. Για παράδειγμα, σε ένα πρόβλημα όπου υπάρχει μια μεγάλη ανισορροπία κατηγοριών, ένα μοντέλο μπορεί να προβλέψει την αξία της πλειοψηφίας των κατηγοριών για όλες τις προβλέψεις και να επιτύχει υψηλή ακρίβεια Κατηγοριοποίησης, το πρόβλημα είναι ότι το μοντέλο αυτό δεν είναι χρήσιμο στον εκάστοτε τομέα προβλήματος. Αυτό ονομάζεται το παράδοξο της ακρίβειας. Για αυτό λοιπόν χρησιμοποιείται ένα πρόσθετο μέτρο το F-measure, όπου η σχέση ορίζεται ως εξής [30]:

$$F_1 = 2 * \frac{\text{precision} * \text{recall}}{\text{precision}+\text{recall}}$$

το οποίο μπορεί να ερμηνευθεί ως ο αρμονικός μέσος των μέτρων της ορθότητας και της ανάκλησης, όπου η καλύτερη τιμή της σχέσης λογίζεται το 1 και η χειρότερη το 0.

Στην παρούσα μεταπτυχιακή διατριβή για τους σκοπούς της αξιολόγησης των συστημάτων μέσω της πειραματικής ανάλυσης που διενεργήθηκε, ως μέτρο αποτελεσματικότητας χρησιμοποιήθηκε η ακρίβεια.

ii. Μετρικές αξιολόγησης αλγορίθμων συσταδοποίησης

Η εφαρμογή ενός αλγορίθμου συσταδοποίησης σε ένα σύνολο δεδομένων στοχεύει, στην ανακάλυψη των έμφυτων διαμερισμών του. Ωστόσο, η διαδικασία συσταδοποίησης γίνεται αντιληπτή ως μία ανεπίβλεπτη διαδικασία, καθώς δεν υπάρχουν προκαθορισμένες κλάσεις και παραδείγματα που δείχνουν ποια είναι η

έγκυρη επιθυμητή σχέση των δεδομένων. Είναι εμφανές ότι ένα πρόβλημα που υπάρχει στην συσταδοποίηση είναι το να αποφασίσουμε τον βέλτιστο αριθμό συστάδων που ταιριάζει σε ένα σύνολο δεδομένων. Είναι δύσκολο να ορίσουμε πότε ένα αποτέλεσμα συσταδοποίησης είναι αποδεκτό ή ακριβές, κατά συνέπεια έχουν αναπτυχθεί διάφορες τεχνικές και μετρικές ελέγχου της συσταδοποίησης.

Μια μετρική συσταδοποίησης χαρτογραφεί μια ομαδοποίηση σε έναν πραγματικό αριθμό. Ο αριθμός δείχνει σε ποιο βαθμό οι δομικές ιδιότητες αναπτύσσονται στην συσταδοποίηση. Η εξωτερική και η εσωτερική επικύρωση συσταδοποίησης είναι οι δυο βασικές μέθοδοι αξιολόγησης της συσταδοποίησης. Σε αντίθεση με τις εξωτερικές μετρικές επικύρωσης, που χρησιμοποιούν εξωτερικές πληροφορίες οι οποίες δεν υπάρχουν στα δεδομένα, οι μετρικές εσωτερικής επικύρωσης βασίζονται σε πληροφορίες που βρίσκονται μέσα στα δεδομένα. Οι μετρικές εσωτερικής επικύρωσης αξιολογούν το βαθμό της ακρίβειας του μοντέλου μιας δομής ομαδοποίησης χωρίς να βασίζονται σε εξωτερικές πληροφορίες. Εφόσον οι εξωτερικές μετρικές επικύρωσης γνωρίζουν τον πραγματικό αριθμό των συστάδων εκ των προτέρων, χρησιμοποιούνται κυρίως για την επιλογή ενός βέλτιστου αλγόριθμου για ένα συγκεκριμένο σύνολο δεδομένων. Από την άλλη πλευρά, τα εσωτερικά μέτρα επικύρωσης μπορούν να χρησιμοποιηθούν για να επιλεχθεί ο καλύτερος αλγόριθμος, καθώς και ο βέλτιστος αριθμός συστάδων.

Στην παρούσα μεταπτυχιακή διατριβή, εξετάζεται η εξωτερική μετρική «καθαρότητα» (purity) [26]. Η καθαρότητα (Purity) [Zhao, Karypis, 2001] επικεντρώνεται στη συχνότητα της πιο κοινής κατηγορίας σε κάθε συστάδα. Η καθαρότητα (purity) δείχνει την αναλογία του μεγέθους της επικρατούσας κλάσης μέσα στη συστάδα προς το μέγεθος της ίδιας κλάσης. Κάθε συστάδα μπορεί να περιέχει δείγματα από διαφορετικές κλάσεις. Υψηλές τιμές καθαρότητας σημαίνουν ότι η συστάδα είναι ένα καθαρό υποσύνολο της επικρατούσας κλάσης. Για τον υπολογισμό της καθαρότητας, κάθε συστάδα ανατίθεται στην κλάση που είναι πιο συχνή μέσα στη συστάδα και έπειτα η ακρίβεια αυτής της ανάθεσης μετράται, εξάγοντας τον αριθμό των σωστά κατανεμημένων αρχείων και διαιρώντας με N (σύνολο αρχείων). Η σχέση είναι ως εξής:

$$\text{purity}(\Omega, \mathbb{C}) = \frac{1}{N} \sum_{k} \max_{j} |\omega_k \cap c_j|$$

Όπου Ω = {ω1, ω2, ….., ωj} είναι το σύνολο των συστάδων και C = {c1, c2, …...., cj} είναι το σύνολο των κλάσεων.

Με βάση τον πίνακα αποτελεσμάτων (confusion matrix) ισχύουν τα εξής:

- TP (True Positive): Ο αριθμός των ζευγαριών των ίδιων αρχείων που ανήκουν στην ίδια συστάδα.

- TN (True Negative) : Ο αριθμός των ζευγαριών ανόμοιων αρχείων που ανήκουν σε διαφορετικές συστάδες.

- FP (False Positive): Ο αριθμός των ζευγαριών ανόμοιων αρχείων που ανήκουν στην ίδια συστάδα.

- FN (False Negative): Ο αριθμός των ζευγαριών των ίδιων αρχείων που ανήκουν σε διαφορετικές συστάδες.

Η καθαρότητα συνδυάζεται επίσης με την εξωτερική μετρική Rand Index. Η μετρική Rand Index αντικατοπτρίζει το ποσοστό των σωστών αναθέσεων, όπως περιγράφονται παραπάνω. Είναι σχεδόν ίδια με τη μετρική της ακρίβειας που αναλύθηκε παραπάνω και δίνεται από την παρακάτω σχέση [26]:

$$\mathrm{RI} = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN}}$$

Η επέκταση του δείκτη Rand Index ονομάζεται Adjusted Rand Index και επιχειρεί να ορίσει τα στοιχεία που μπορεί να έχουν συγκεντρωθεί κατά τύχη.

$$\text{Adjusted Rand Index} = \frac{2\,(\mathrm{TP}*\mathrm{TN}-\mathrm{FP}*\mathrm{FN})}{(\mathrm{TP}+\mathrm{FP})(\mathrm{FP}+\mathrm{TN})+(\mathrm{TP}+\mathrm{FN})(\mathrm{FN}+\mathrm{TN})}$$

Μια ακόμη εξωτερική μετρική είναι η V-measure, η οποία συνδυάζει την ομοιογένεια και την πληρότητα. Η ομοιογένεια (homogeneity) δείχνει κατά πόσο κάθε συστάδα περιέχει μόνο τα μέλη μιας κατηγορίας. Ενώ η πληρότητα (completeness) δείχνει κατά πόσο όλα τα μέλη μιας δεδομένης κατηγορίας εκχωρούνται στην ίδια συστάδα. Η παρακάτω σχέση υπολογίζει το V-measure, όπου H η ομοιογένεια και C η πληρότητα [26].

$$V_\beta = \frac{(1+\beta)*H*C}{(\beta*H)+C}$$

Στη συνέχεια θα αναλυθούν κάποιες από τις εσωτερικές μετρικές [24] που υπολογίζονται και εξάγονται αυτόματα από τα συστήματα. Ο συντελεστής σιλουέτας (silhouette coefficient) ενός αντικειμένου μετρά την απόσταση του από τα αντικείμενα της ομάδας του, συγκριτικά όμως με την απόσταση του από τα αντικείμενα όλων των άλλων ομάδων. Επομένως, αποτελεί ένα συνδυαστικό μέτρο για τη συνοχή και την απομόνωση. Μπορούμε να αξιολογήσουμε το αποτέλεσμα της συσταδοποίησης παίρνοντας το μέσο όρο των συντελεστών σιλουέτας για κάθε αντικείμενο του συνόλου δεδομένων. Οι τιμές που βρίσκονται κοντά στο -1, σημαίνουν ότι η συσταδοποίηση είναι λανθασμένη, ενώ τιμές που βρίσκονται κοντά στο +1, η συσταδοποίηση είναι ορθή. Οι τιμές γύρω στο μηδέν δείχνουν ότι οι συστάδες υπερκαλύπτονται.

Η ρίζα της μέσης τετραγωνικής τυπικής απόκλισης (*RMSSTD*) είναι η τετραγωνική ρίζα της διακύμανσης του ομαδοποιημένου δείγματος όλων των attributes. Μετρά την ομοιογένεια του σχηματιζόμενων συστάδων.

Στην παρούσα διατριβή επιλέχθηκε να εξεταστεί μόνο η εξωτερική μετρική καθαρότητα (purity), διότι το confusion matrix δεν εξάγεται από το Rapidminer για να υπολογιστεί η μετρική Rand index, όπως επίσης η ομοιογένεια και η πληρότητα εξάγεται μόνο από το scikit learn. Επιπλέον, οι εσωτερικές μετρικές που υπολογίζονται από τα υπό εξέταση συστήματα δεν είναι ίδιες για όλα τα συστήματα.

5.2 Σύνολα Δεδομένων

Για την υλοποίηση της πειραματικής αξιολόγησης των τεσσάρων επιλεγόμενων συστημάτων, επιλέγονται τρία από τα πιο αντιπροσωπευτικά σύνολα δεδομένων με κριτήρια το τύπο, το πεδίο εφαρμογής και το μέγεθος τους, όπως φαίνεται στον παρακάτω πίνακα, με σκοπό να εξεταστεί η λειτουργία των συστημάτων σε όσο το δυνατό διαφορετικά προβλήματα με διαφορετικές μεταβλητές.

Πιο συγκεκριμένα, για την επίτευξη της υλοποίησης των πεδίων εφαρμογής της κατηγοριοποίησης αλλά και της συσταδοποίησης, επιλέχθηκαν δύο συλλογές μικρού και μεσαίου μεγέθους, για κάθε δυνατή εργασία για κάθε πεδίο εφαρμογής.

Σύνολα Δεδομένων	Τύπος Συνόλου Δεδομένων	Εργασία	Τύπος Γνωρισμάτων	Αριθμός Παραδειγμάτων
IMDb reviews	Κείμενο, δύο μεταβλητών	Δυαδική Κατηγοριοποίηση	Κατηγοριακός, Ονομαστικός	2.000
20 newsgroup	Κείμενο, πολλαπλών μεταβλητών	Κατηγοριοποίηση Πολλαπλών Μεταβλητών, Συσταδοποίηση	Κατηγοριακός, Ονομαστικός	18.828
BBC news	Κείμενο, πολλαπλών μεταβλητών	Συσταδοποίηση	Κατηγοριακός, Ονομαστικός	2.225

Πίνακας 4. Χαρακτηριστικά Συνόλων Δεδομένων.

5.2.1 IMDb reviews

Η συλλογή των IMDB reviews περιλαμβάνει επισημασμένα έγγραφα σε σχέση με τη συνολική πολικότητα του συναισθήματος, θετικού ή αρνητικού, αναφορικά με τις κριτικές των ταινιών. Το συνολικό μέγεθος της συλλογής είναι 3 MB η οποία θα χρησιμοποιηθεί παρακάτω σε πειράματα εξόρυξης συναισθήματος με αλγορίθμους κατηγοριοποίησης.

Αναλυτικότερα, το σύνολο των δεδομένων είναι μοιρασμένο σε ένα σύνολο 1.000 θετικών και 1.000 αρνητικών κριτικών ταινιών [31]. Κάθε κριτική αποτελείται από ένα απλό αρχείο κειμένου (txt) και μια ετικέτα της τάξης που αντιπροσωπεύει τη συνολική άποψη του χρήστη. Οι κλάσεις έχουν μόνο δύο τιμές: θετικές ή αρνητικές. Οι δημιουργοί του συνόλου δεδομένων εξηγούν ότι η διάκριση των κλάσεων σε θετικές και αρνητικές, έχει καθοριστεί με τη χρήση απλούστερων κανόνων, όπως η ψήφος των χρηστών όπως εξάγεται από την αρχική αξιολόγηση.

Το συγκεκριμένο σύνολο δεδομένων πρωτοεμφανίστηκε στην μελέτη των Pang/Lee, 2004, το οποίο είναι γνωστό ως “Sentiment Polarity Dataset version 2.0” (http://www.cs.cornell.edu/People/pabo/movie-review-data) [31].

5.2.2 20 newsgroup

Το σύνολο δεδομένων 20 newsgroup (Lang, 1995) περιέχει 18.828 έγγραφα σε txt μορφή, από τη συλλογή των άρθρων usenet news group, συνολικού μεγέθους 32,30 MB. Τα έγγραφα στην αρχική της μορφή αποτελούνται από επικεφαλίδες, θέμα, υπογραφές και σχόλια από άλλα άρθρα. Κάθε newsgroup ανήκει σε διαφορετική κατηγορία με διαφορετικούς βαθμούς επικάλυψης, ενώ υπολογίζεται ότι το 4% των εγγράφων ανήκουν σε περισσότερες της μια κατηγορίας και ταξινομούνται με βάση το περιεχόμενο του σε 20 υποκατηγορίες. Η κατηγοριοποίηση του συνόλου των δεδομένων αναλύεται στον παρακάτω πίνακα:

comp.graphics comp.os.ms-windows.misc comp.sys.ibm.pc.hardware comp.sys.mac.hardware comp.windows.x (4.881 αρχεία)	rec.autos rec.motorcycles rec.sport.baseball rec.sport.hockey (3.977 αρχεία)	sci.crypt sci.electronics sci.med sci.space (3.868 αρχεία)
misc.forsale (972 αρχεία)	talk.politics.misc talk.politics.guns talk.politics.mideast (2.624 αρχεία)	talk.religion.misc alt.atheism soc.religion.christian (2.353 αρχεία)

Το σύνολο δεδομένων 20 newsgroup http://qwone.com/~jason/20Newsgroups/

θα χρησιμοποιηθεί στην πειραματική αξιολόγηση των τεσσάρων συστημάτων χρησιμοποιώντας αλγορίθμους Κατηγοριοποίησης αλλά και συσταδοποίησης.

5.2.3 BBC news

Τέλος, χρησιμοποιήθηκε το σύνολο δεδομένων BBC news το οποίο αποτελείται από 2.225 έγγραφα σε txt μορφή, που έχουν δημοσιευθεί από το 2004 - 2005 στην ιστοσελίδα του BBC News και τα οποία αντιστοιχούν σε πέντε θεματικούς τομείς [25]. Τα κείμενα που την απαρτίζουν αποτελούνται στο σύνολό τους από άρθρα που αφορούν την καθημερινή ειδησιογραφία, σε επίπεδο κοινωνικό, πολιτικό, τεχνολογικό, αθλητικό και οικονομικό, όπως αναλύεται στον παρακάτω πίνακα:

Πολιτική (417 αρχεία)	Ψυχαγωγία (386 αρχεία)	Επιχειρηματική δραστηριότητα (510 αρχεία)	Αθλητισμός (511 αρχεία)	Τεχνολογία (401 αρχεία)

Η συλλογή περιλαμβάνει πέντε ομάδες συνολικού μεγέθους 4,80 MB, βασιζόμενες στο τομέα που εξετάζουν. Το συγκεκριμένο σύνολο δεδομένων φιλοξενείται στον ιστότοπο http://mlg.ucd.ie/datasets/bbc.html και θα χρησιμοποιηθεί σε πειράματα με αλγορίθμους συσταδοποίησης.

5.3 Πειραματικά Αποτελέσματα

Στην παρούσα μεταπτυχιακή διατριβή παρουσιάζονται αποτελέσματα που έχουν προκύψει από πειραματισμούς και στα τέσσερα συστήματα σε κάθε μέθοδο κατηγοριοποίησης και συσταδοποίησης που αναφέρθηκε παραπάνω με έναν αλγόριθμο που αποτελεί ενδεικτικό αντιπρόσωπο όλων των αλγορίθμων που εφαρμόζουν την εκάστοτε μέθοδο.

5.3.1 Εκτίμηση Αποτελεσματικότητας Κατηγοριοποίησης

Αναφορικά με το πεδίο της κατηγοριοποίησης, οι χαρακτηριστικοί αντιπρόσωποι των παραπάνω κατηγοριοποιητών που επιλέχθηκαν, είναι οι εξής:

- Δέντρα Αποφάσεων: Ο C4.5 (J48 Weka) ο οποίος αναπτύχθηκε από τον Ross Quinlan και είναι μια επέκταση του αλγορίθμου ID3,

- Αλγόριθμοι Αυτόματης Εκμάθησης Κανόνων: Ο Ripper (JRip Weka) ο οποίος προτάθηκε από τον William Cohen ως μία βελτιστοποιημένη έκδοση του IREP,

- Μηχανική Υποστήριξη Διανυσμάτων: Ο αλγόριθμος SVM(SMO Weka) ο οποίος υλοποιήθηκε από το John Platt, ενώ στην Python που δεν υπάρχει υλοποίηση χρησιμοποιήθηκε ο γραμμικός αλγόριθμος SVM του Stefan Ruping.

- Bayesian Δίκτυα: Ο πολυωνομικός Naïve Bayes,

- Lazy μοντεολοποίηση: Ο K-nn ο οποίος προτάθηκε από τους D. Aha και D. Kibler με k ίσο με 5 για όλα τα πειράματα.

Τα σύνολα εκπαίδευσης που χρησιμοποιήθηκαν και αναλύθηκαν παραπάνω, έχουν ήδη υποστεί την προ-επεξεργασία όπως επίσης και την διανυσματική τους αναπαράσταση, χρησιμοποιώντας όσο το δυνατόν τις ίδιες μεθόδους, όπως παρουσιάστηκαν παραπάνω, ώστε τα αποτελέσματα να είναι συγκρίσιμα και όσο πιο αντικειμενικά γίνεται.

Κάθε αλγόριθμος εφαρμόστηκε με τις προκαθορισμένες παραμέτρους. Με σκοπό να υπολογιστεί η ακρίβεια του κατηγοριοποιητή χρησιμοποιήθηκε η μέθοδος hold out. Κατά τη μέθοδο αυτή το σύνολο εκπαίδευσης χωρίζεται είναι το 75% του συνόλου των δεδομένων και ο κατηγοριοποιητής εκπαιδεύεται στο υπόλοιπο υποσυνόλο. Χρησιμοποιήθηκε φιλτράρισμα όπως αναλύθηκε παραπάνω, ώστε να δώσει τη δυνατότητα να μπορέσει να επιλεχθεί πειραματικά ο κατάλληλος αριθμός γνωρισμάτων για την εύρεση της καλύτερης ακρίβειας. Επιπλέον σημειώνεται ότι έχει οριστεί ένα όριο συνολικά μίας ώρας για την εκτέλεση του κάθε μοντέλου και σε περίπτωση που ξεπεραστεί το όριο η διαδικασία τερματίζεται χωρίς την εξαγωγή αποτελεσμάτων. Στους παρακάτω πίνακες καταγράφονται τα αποτελέσματα των βασικότερων μετρικών αξιολόγησης που παράχθηκαν έπειτα από τα πειράματα που υλοποιήθηκαν με τα σύνολα δεδομένων IMDb reviews και 20newsgroup, για κάθε αλγόριθμο ξεχωριστά.

a. Αποτελέσματα πειραμάτων με το σύνολο δεδομένων IMDb reviews

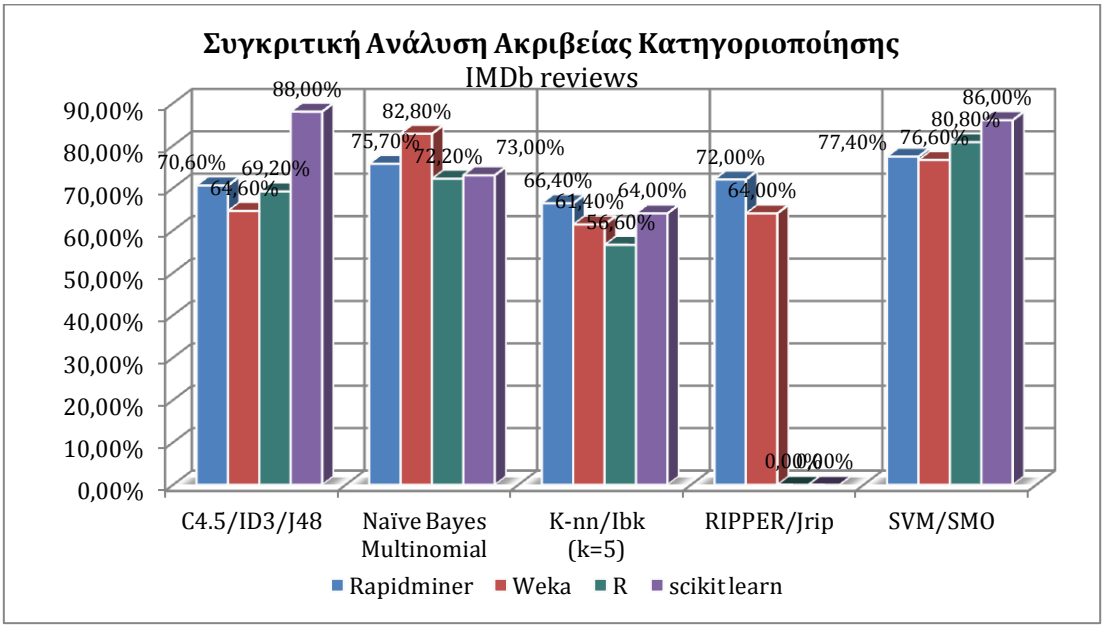
Σημειώνεται ότι δεν υπάρχει διαθέσιμη υλοποίηση του RIPPER στο scikit learn, ενώ δεν εξήχθησαν αποτελέσματα στο R διότι η υλοποίηση του RIPPER δεν είχε καμία συνάφεια με τους υπόλοιπους αλγορίθμους που υλοποιήθηκαν με διαφορετικούς μεθόδους.

		C4.5/ID3/J48	**Naïve Bayes Multinomial**	**K-nn/Ibk (k=5)**	**RIPPER/Jrip**	**SVM/SMO**
Rapid Miner	Ακρίβεια	70,60%	75,70%	66,40%	72,00%	77,40%
	Μέσος όρος Ορθότητα	70,60%	75,73%	65,45%	72,00%	77,77%
	Μέσος όρος Ανάκληση	70,64%	75,70%	66,50%	72,00%	77,47%
	Μέσος όρος F-measure	70,62%	75,71%	65,97%	72,00%	77,62%
Weka	Ακρίβεια	64,60%	82,80%	61,40%	64,00%	76,60%
	Μέσος όρος Ορθότητα	64,60%	82,80%	68,40%	64,30%	76,60%
	Μέσος όρος Ανάκληση	64,60%	82,80%	61,40%	64,00%	76,60%
	Μέσος όρος F-measure	64,60%	82,80%	57,20%	63,80%	76,60%
R	Ακρίβεια	69,20%	72,20%	56,60%	-	80,80%
	Μέσος όρος Ορθότητα	69,30%	72,30%	70,10%	-	80,80%
	Μέσος όρος Ανάκληση	69,20%	72,20%	56,60%	-	80,80%
	Μέσος όρος F-measure	69,25%	72,25%	62,63%	-	80,80%
(scikit learn) Python	Ακρίβεια	88,00%	73,00%	64,00%	-	86,00%
	Μέσος όρος Ορθότητα	89,00%	81,00%	62,00%	-	86,00%

Μέσος όρος Ανάκληση	87,00%	64,00%	67,00%	-	87,00%
Μέσος όρος F-measure	87,99%	71,50%	64,40%	-	86,50%

Πίνακας 5. Συγκριτική ανάλυση επιδόσεων των αλγορίθμων κατηγοριοποίησης ανά σύστημα, για το σύνολο δεδομένων IMDb reviews.

Ειδικότερα, σύμφωνα με το Γράφημα 17, παρατηρείται ότι το scikit learn κατηγοριοποιεί με υψηλότερη ακρίβεια από όλα τα συστήματα τους αλγόριθμους C4.5/J48 και SVM/SMO, το Rapidminer τους αλγόριθμους K-nn και RIPPER και το Weka τον Naïve Bayes multinomial, ενώ το R δεν απαντάται στην πρώτη θέση σε καμία υλοποίηση. Από την άλλη πλευρά, όλα τα συστήματα κατέχουν τα χαμηλότερα επίπεδα ακρίβειας στο μοντέλο K-nn. Το πιο ακριβές μοντέλο για το Rapidminer και το R παρατηρείται για το SVM, ενώ για το Weka για το Naïve Bayes και για το scikit learn για το C4.5.



Γράφημα 17. Συγκριτική ανάλυση ακριβείας των αλγορίθμων κατηγοριοποίησης ανά σύστημα, για το σύνολο δεδομένων IMDb reviews.

Τέλος, εξετάζοντας τη συνολική μέση ακρίβεια του κάθε συστήματος για όλους τους αλγορίθμους για του οποίους έχει ολοκληρωθεί με επιτυχία η υλοποίηση, το scikit learn κατέχει την πρώτη θέση με 77,75% και ακολουθούν το Rapidminer με 72,42 % και με μικρή διαφορά μεταξύ τους το Weka και το R, με αντίστοιχα ποσοστά ακρίβειας 69,88% και 69,70%.

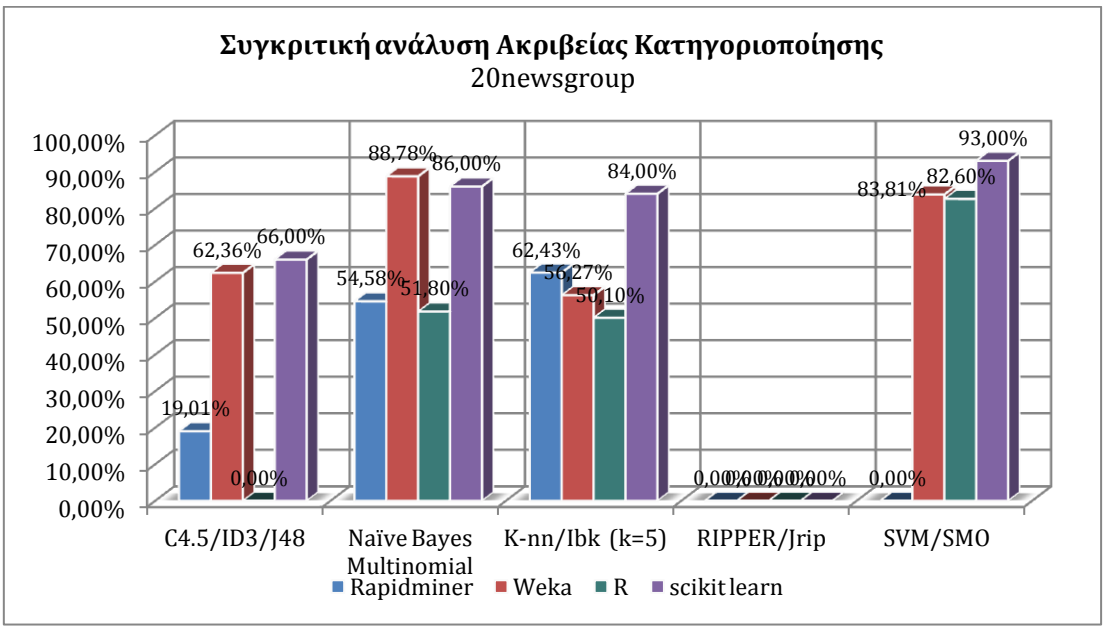
b. Αποτελέσματα πειραμάτων με το σύνολο δεδομένων 20newsgroup

Συνεχίζοντας την ανάλυση των αποτελεσμάτων της μετρικής της ακρίβειας, εξετάζοντας τον Πίνακα 6 έπειτα από τα πειράματα που υλοποιήθηκαν με το σύνολο δεδομένων 20newsgroup, παρατηρείται μια μεγαλύτερη διακύμανση ανώτατου-κατώτατου ποσοστού ακριβείας των αλγορίθμων, σε σύγκριση με το σύνολο δεδομένων IMDb reviews. Σημειώνεται ότι δεν υπάρχει διαθέσιμη υλοποίηση του RIPPER στο scikit learn και στα πακέτα του R που επιλέχθηκε να χρησιμοποιηθούν, ενώ δεν εξήχθησαν αποτελέσματα σε κανένα από τα άλλα δυο συστήματα διότι η διαδικασία ξεπέρασε το όριο της μιας ώρας στο Rapidminer, ενώ στο Weka τερματίστηκαν οι διεργασίες λόγω μη διαθέσιμης μνήμης. Επίσης, δεν σημειώθηκαν αποτελέσματα για το C4.5/J48 στο R και για το SVM στο σύστημα Rapidminer διότι οι διεργασίες ξεπέρασαν το όριο της μιας ώρας.

		C4.5/ID3/J48	**Naïve Bayes Multinomial**	**K-nn/Ibk (k=5)**	**RIPPER/Jrip**	**SVM/SMO**
Rapid Miner	Ακρίβεια	19,01%	54,58%	62,43%	-	-
	Μέσος όρος Ορθότητα	18,48%	54,41%	62,98%	-	-
	Μέσος όρος Ανάκληση	19,30%	54,23%	61,87%	-	-
	Μέσος όρος F-measure	18,88%	54,32%	62,42%	-	-
Weka	Ακρίβεια	62,36%	88,78%	56,27%	-	83,81%
	Μέσος όρος Ορθότητα	62,80%	88,90%	82,00%	-	84,10%
	Μέσος όρος Ανάκληση	62,40%	88,80%	56,30%	-	83,80%
	Μέσος όρος F-measure	62,10%	88,70%	63,10%	-	83,80%
R	Ακρίβεια	-	51,80%	50,10%	-	82,60%
	Μέσος όρος Ορθότητα	-	50,90%	68,00%	-	81,80%
	Μέσος όρος Ανάκληση	-	53,00%	50,10%	-	83,00%
	Μέσος όρος F-measure	-	51,93%	57,69%	-	80,80%
(scikit learn) Python	Ακρίβεια	66,00%	86,00%	84,00%	-	93,00%
	Μέσος όρος Ορθότητα	66,00%	81,15%	81,00%	-	91,50%
	Μέσος όρος Ανάκληση	66,00%	85,00%	83,20%	-	94,00%
	Μέσος όρος F-measure	66,00%	83,03%	82,09%	-	92,73%

Πίνακας 6. Συγκριτική ανάλυση επιδόσεων των αλγορίθμων κατηγοριοποίησης ανά σύστημα για το σύνολο δεδομένων 20newsgroup.

Σύμφωνα με το Γράφημα 18, παρατηρείται ότι το scikit learn κατηγοριοποιεί με υψηλότερη ακρίβεια από όλα τα συστήματα όλους τους αλγόριθμους εκτός του μοντέλου Naïve Bayes που το πιο ακριβές μοντέλο παρατηρείται στο Weka. Σε αντίθεση με τα παραπάνω, το Rapidminer κατέχει το χαμηλότερο ποσοστό ακρίβειας που έχει απαντηθεί, για το μοντέλο C4.5/J48, ενώ το R κατέχει τα χαμηλότερα ποσοστά ακρίβειας για όλα υπόλοιπα μοντέλα (K-nn, Naïve Bayes και SVM). Αξίζει να σημειωθεί ότι το Rapidminer κατηγοριοποιεί τον αλγόριθμο J48 με 19,01% ακρίβεια η οποία είναι πολύ χαμηλή ενώ το scikit learn με 66%.



Γράφημα 18. Συγκριτική ανάλυση ακριβείας των αλγορίθμων κατηγοριοποίησης ανά σύστημα, για το σύνολο δεδομένων 20newsgroup.

Τέλος, εξετάζοντας τη μέση συνολική ακρίβεια του κάθε συστήματος για όλους τους αλγορίθμους το scikit learn κατέχει την πρώτη θέση με υψηλό ποσοστό ακρίβειας 82,25% σε αντίθεση με το 77,75% για το σύνολο δεδομένων IMDb reviews. Ακολουθεί το Weka με 72,81% και διαφορά 2,81 μονάδων σε σύγκριση με το ποσοστό για το IMDb reviews. Τέλος, το R τερματίζει με 61,50% και το Rapidminer με μεγάλη διαφορά από τα υπόλοιπα με ποσοστό 45,34%.

5.3.2 Εκτίμηση Αποτελεσματικότητας Συσταδοποίησης

Στην παρούσα μεταπτυχιακή διατριβή παρουσιάζονται αποτελέσματα που έχουν προκύψει από πειραματισμούς και στα τέσσερα συστήματα σε κάθε μέθοδο συσταδοποίησης που αναφέρθηκε παραπάνω με έναν αλγόριθμο που αποτελεί ενδεικτικό αντιπρόσωπο όλων των αλγορίθμων που εφαρμόζουν την εκάστοτε μέθοδο.

Έτσι λοιπόν, χαρακτηριστικός αντιπρόσωπος της κλάσης των αλγορίθμων κατανομής κέντρου βάρους που χρησιμοποιήθηκε είναι ο K-means, των αλγορίθμων βασισμένων στην πυκνότητα ο DBSCAN και των ιεραρχικών αλγορίθμων ο Agglomerative. Τα σύνολα εκπαίδευσης που χρησιμοποιήθηκαν και αναλύθηκαν παραπάνω, έχουν ήδη υποστεί την προ-επεξεργασία όπως επίσης και την διανυσματική τους αναπαράσταση, χρησιμοποιώντας όσο το δυνατόν τις ίδιες μεθόδους, όπως παρουσιάστηκαν παραπάνω, ώστε τα αποτελέσματα να είναι συγκρίσιμα και όσο πιο αντικειμενικά γίνεται.

Κάθε αλγόριθμος εφαρμόστηκε με τις προκαθορισμένες παραμέτρους και ως μέτρο απόστασης για τον K-means και Agglomerative χρησιμοποιήθηκε η Ευκλείδεια Απόσταση. Ο DBSCAN υπολογίστηκε με τιμές στις παραμέτρους Epsilon και Minpoint 0.9 και 2 αντίστοιχα. Με σκοπό να παραχθεί ο πίνακας αποτελεσμάτων ορίζοντας τις προκαθορισμένες κλάσεις στις εξαχθείσες συστάδες με τα στοιχεία του οποίου υπολογίστηκε η εξωτερική μετρική της καθαρότητας (purity), χρησιμοποιήθηκε η μέθοδος επικύρωσης class to clusters για το Weka, R και scikit learn. Για το Rapidminer χρησιμοποιήθηκε η μέθοδος holdout, διότι δεν υπάρχει διαθέσιμη υλοποίηση του class to clusters, όπου κατά τη μέθοδο αυτή το σύνολο εκπαίδευσης χωρίζεται στο 75% του συνόλου των δεδομένων και ο επιλεγόμενος αλγόριθμος εκπαιδεύεται στο υπόλοιπο υποσυνόλο.

a. Αποτελέσματα πειραμάτων με το σύνολο δεδομένων BBC news

Στον Πίνακα 7 καταγράφεται η εξωτερική μετρική της καθαρότητας της συσταδοποίησης ανά σύστημα, για το σύνολο δεδομένων BBC news.

	RapidMiner	**Weka Explorer**	**R**	**scikit learn**
K-means	87,06%	90,79%	54,40%	96,22%

Agglomerative	73,71%	70,56%	47,10%	92,76%
DBSCAN	36,94%	11,51%	27,20%	0,00%

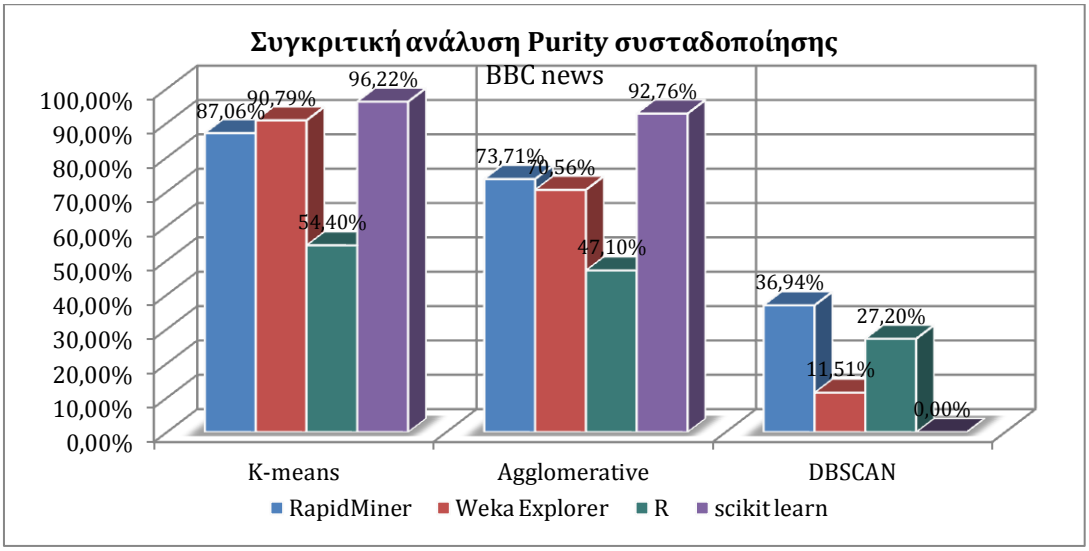
Πίνακας 7. Συγκριτική ανάλυση επιδόσεων των αλγορίθμων συσταδοποίησης ανά σύστημα για το σύνολο δεδομένων BBC news.

Γενικά, ισχύει ότι όσο μεγαλύτερη είναι η τιμή της καθαρότητας τόσο καλύτερο είναι το μοντέλο συσταδοποίησης. Σύμφωνα με το παρακάτω γράφημα, το μοντέλο DBSCAN έχει το χαμηλότερο μέσο όρο καθαρότητας, με το scikit learn να καταγράφει τη τιμή μηδέν που σημαίνει ότι όλα οι περιπτώσεις του συνόλου των δεδομένων χαρακτηρίζονται ως θόρυβος, δηλαδή δεν έχουν κάποιο κοινό σημείο μεταξύ τους. Η υψηλότερη τιμή καθαρότητας για το DBSCAN απαντάται στο Rapidminer και ακολουθούν το R και το Weka. Γενικά όμως, οι τιμές της καθαρότητας είναι πολύ χαμηλές που σημαίνει ότι το μοντέλο DBSCAN δε συσταδοποιεί επιτυχημένα το σύνολο BBC news.

Σε αντίθεση με τα παραπάνω, ο αλγόριθμος K-means κατέχει το υψηλότερο μέσο όρο καθαρότητας με το scikit learn να πετυχαίνει 92,76%, να ακολουθεί το Weka με 90,79%, το Rapidminer με 87,06% και τέλος το R να έχει τη χαμηλότερη τιμή καθαρότητας με 54,40%. Ο k-means λαμβάνει ως δεδομένο τον αριθμό των συστάδων, οπότε είναι λογικό να κατέχει τα υψηλότερα ποσοστά συσταδοποίησης σε σύγκριση με τους άλλους υπό εξέταση αλγορίθμους.

Εξετάζοντας τον αλγόριθμο Agglomerative, παρατηρείται μια υψηλή τιμή καθαρότητας για το scikit learn (92,76%), ενώ το Rapidminer και το Weka έχουν πολύ μικρή διαφορά μεταξύ τους με 73,71% και 70,56%, αντίστοιχα. Το συνεχίζει να κατέχει τη χαμηλότερη τιμή καθαρότητας του μοντέλου με 47,10%, καταγράφοντας μια μεγάλη διακύμανση της τάξεως των 45,66 ποσοστιαίων μονάδων.

Τέλος, εξετάζοντας τους μέσους όρους της καθαρότητας για το κάθε σύστημα ξεχωριστά, η πιο επιτυχημένη συσταδοποίηση καταγράφεται στο Rapidminer με μέσο όρο καθαρότητας 65,90% ακολουθεί πολύ κοντά το scikit learn με 63%, συνεχίζει το Weka με 57,6% και τελευταίο καταλήγει το R με μέσο όρο καθαρότητας 42,90%.



Γράφημα 19. Συγκριτική ανάλυση καθαρότητας των αλγορίθμων συσταδοποίησης ανά σύστημα, για το σύνολο δεδομένων BBC news.

b. Αποτελέσματα πειραμάτων με το σύνολο δεδομένων 20 newsgroup

Στον Πίνακα 8 καταγράφεται η εξωτερική μετρική της καθαρότητας της συσταδοποίησης ανά σύστημα, για το σύνολο δεδομένων 20newsgroup. Για το R δε καταγράφεται κανένα αποτέλεσμα διότι το μοντέλο K-means ξεπέρασε το όριο της μιας ώρας που έχει τεθεί, ενώ τα άλλα δύο μοντέλα τερματίστηκαν λόγω μη διαθέσιμης μνήμης. Γενικά παρατηρείται, ότι για το μοντέλο Agglomerative δεν κατεγράφη αποτέλεσμα σε κανένα σύστημα, διότι για το Rapidminer και το scikit learn ξεπέρασε το όριο της μιας ώρας και το Weka τερμάτισε λόγω μη διαθέσιμης μνήμης.

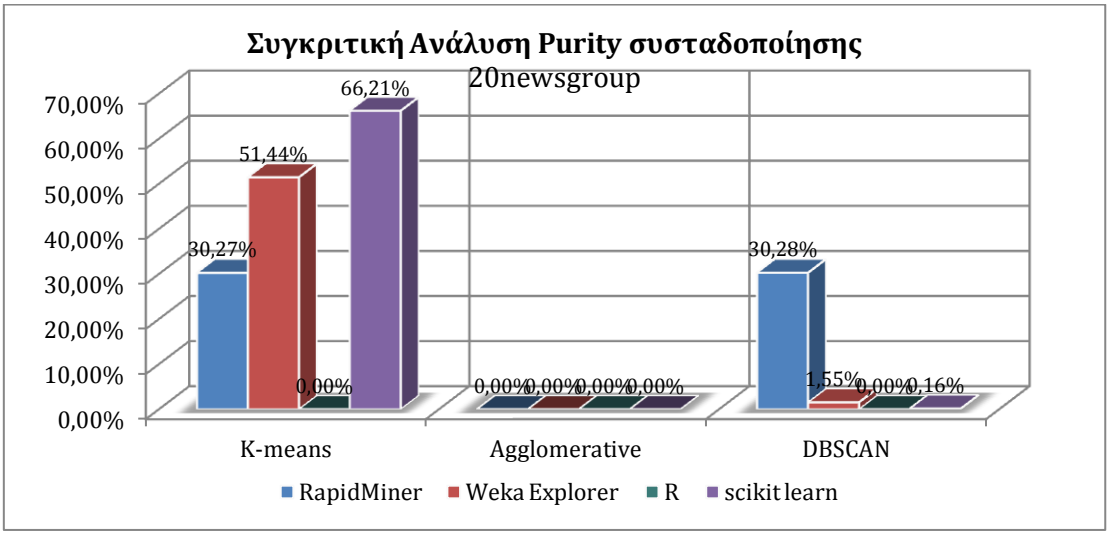
	RapidMiner	**Weka Explorer**	**R**	**scikit learn**
K-means	30,27%	51,44%	-	66,21%
Agglomerative	-	-	-	-
DBSCAN	30,28%	1,55%	-	0,16%

Πίνακας 8. Συγκριτική ανάλυση επιδόσεων των αλγορίθμων συσταδοποίησης ανά σύστημα για το σύνολο δεδομένων 20newsgroup.

Σύμφωνα με το παρακάτω γράφημα, το μοντέλο DBSCAN συνεχίζει να έχει το χαμηλότερο μέσο όρο καθαρότητας όπως και για το σύνολο δεδομένων BBC news, με το scikit learn να καταγράφει τη χαμηλότερη τιμή. Η υψηλότερη τιμή καθαρότητας για το DBSCAN απαντάται στο Rapidminer και ακολουθεί το Weka. Γενικά όμως, οι τιμές

της καθαρότητας είναι πολύ χαμηλές που σημαίνει ότι το μοντέλο DBSCAN δε συσταδοποιεί επιτυχημένα το σύνολο 20newsgroup.

Εξετάζοντας τον αλγόριθμο k-means, το scikit learn συσταδοποιεί επιτυχημένα το σύνολο δεδομένων με ποσοστό 66,21% ενώ το Weka έχει ποσοστό 51,44% και το Rapiminer 30,27%. Τέλος, εξετάζοντας τους μέσους όρους της καθαρότητας για το κάθε σύστημα ξεχωριστά, η πιο επιτυχημένη συσταδοποίηση καταγράφεται στο scikit learn με μέσο όρο καθαρότητας των μοντέλων 33,18% ακολουθεί πολύ κοντά το Rapidminer με 30,27%, συνεχίζει το Weka με 26,49%.



Γράφημα 20. Συγκριτική ανάλυση καθαρότητας των αλγορίθμων συσταδοποίησης ανά σύστημα, για το σύνολο δεδομένων 20newsgroup.

5.3.3 Εκτίμηση αποδοτικότητας

Ένα πολύ σημαντικό κριτήριο πέρα από την ακρίβεια των μοντέλων, κρίνεται ο χρόνος εκπαίδευσης του μοντέλου αλλά και ο συνολικός χρόνος εκτέλεσης. Γενικά, ο κυριότερος στόχος των χρηστών της ανάλυσης δεδομένων και όχι μόνο, είναι η βελτίωση του χρόνου εκτέλεσης των πειραμάτων τους επιδιώκοντας άμεσα αποτελέσματα. Οι τιμές που καταγράφονται στους πίνακες είναι σε δευτερόλεπτα.

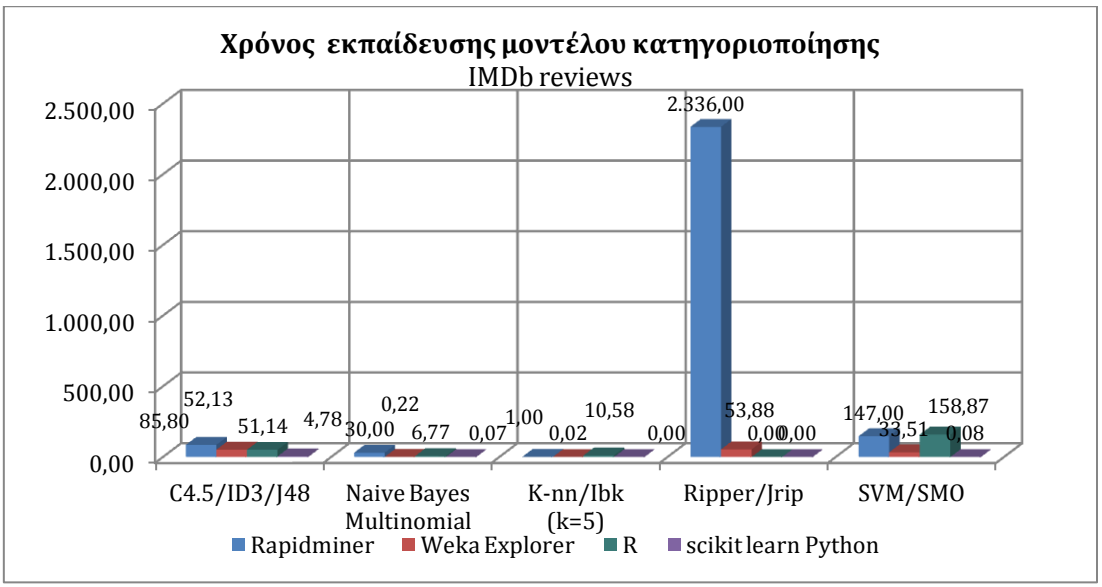
a. Εκτίμηση αποδοτικότητας κατηγοριοποίησης με το IMDb reviews

Στον Πίνακα 9, καταγράφονται αναλυτικά οι χρόνοι εκτέλεσης της εκπαίδευσης και της πρόβλεψης του κάθε αλγόριθμου ξεχωριστά, για το σύνολο δεδομένων IMDb reviews.

		Rapidminer	Weka Explorer	R	scikit learn Python
C4.5/ID3/J48	Χρόνος Εκπαίδευσης του μοντέλου	85,80	52,13	51,14	4,78
	Χρόνος εκτέλεσης Πρόβλεψης του μοντέλου	58,00	9,80	12,58	0,00
	Συνολικός Χρόνος Εκτέλεσης (wall clock time)	144,60	61,93	74,05	4,79
Naive Bayes Multinomial	Χρόνος Εκπαίδευσης του μοντέλου	30,00	0,22	6,77	0,07
	Χρόνος εκτέλεσης Πρόβλεψης του μοντέλου	10,00	3,10	170,74	0,00
	Συνολικός Χρόνος Εκτέλεσης (wall clock time)	82,00	3,32	188,93	0,08
K-nn/Ibk (k=5)	Χρόνος Εκπαίδευσης του μοντέλου	1,00	0,02	10,58	0,00
	Χρόνος εκτέλεσης Πρόβλεψης του μοντέλου	6,00	0,00	121,41	0,15
	Συνολικός Χρόνος Εκτέλεσης (wall clock time)	49,00	0,02	142,58	0,17
Ripper/Jrip	Χρόνος ε Εκπαίδευσης του μοντέλου	2.336,00	53,88	-	-
	Χρόνος εκτέλεσης Πρόβλεψης του μοντέλου	328,00	35,30	-	-
	Συνολικός Χρόνος Εκτέλεσης (wall clock time)	2.706,00	89,18	-	-
SVM/SMO	Χρόνος Εκπαίδευσης του μοντέλου	147,00	33,51	158,87	0,08
	Χρόνος εκτέλεσης Πρόβλεψης του μοντέλου	56,40	13,80	16,84	0,00
	Συνολικός Χρόνος Εκτέλεσης (wall clock time)	245,40	47,31	186,45	0,08

Πίνακας 9. Χρόνοι εκτέλεσης των αλγορίθμων κατηγοριοποίσης ανά σύστημα (sec)για το σύνολο δεδομένων IMDb reviews.

Συγκρίνοντας το χρόνο που χρειάστηκε για να εκπαιδευτούν τα μοντέλα, με βάση το παρακάτω γράφημα, τον υψηλότερο χρόνο εκπαίδευσης τον κατέχει ο RIPPER το Rapid miner να χρειάζεται 2.236 sec. Επίσης, το Rapidminer έχει τους υψηλότερους χρόνους εκπαίδευσης για το C4.5 και Naïve Bayes, ενώ το R για το SVM και K-nn. Οι χρόνοι εκπαίδευσης των αλγορίθμων είναι άρρηκτα συνδεδεμένοι με την πολυπλοκότητα των εκάστοτε μοντέλων. Ο K-nn απαιτεί περισσότερο χρόνο για την πρόβλεψη του μοντέλου αντί για την εκπαίδευση και με το R να φτάνει στην υψηλότερη τιμή του χρόνου πρόβλεψης.



Γράφημα 21. Απεικόνιση χρόνου εκπαίδευσης των αλγορίθμων κατηγοριοποίησης ανά συστημα σε sec, για το σύνολο δεομένων IMDb reviews.

Υπολογίζοντας τους μέσους όρους των συνολικών χρόνων εκπαίδευσης των μοντέλων ανά σύστημα, παρατηρείται ότι το scikit learn παρέχει τη πιο γρήγορη εκπαίδευση σε όλα τα μοντέλα με μέσο χρόνο εκπαίδευσης 1,23s και πιο αργό χρόνο στο μοντέλο C4.5. Το Weka καταλαμβάνει τη δεύτερη θέση με μέσο χρόνο εκπαίδευσης 27,95s με πιο υψηλό χρόνο στο μοντέλο RIPPER και πιο χαμηλό στο K-nn. Ενώ πολύ κοντά ακολουθεί το R 56,84s με πιο υψηλό χρόνο στο μοντέλο SVM και πιο χαμηλό στο μοντέλο Naïve Bayes. Τέλος, το Rapidminer έχει μέσο όρο χρόνου εκπαίδευσης 519,16 s, με πιο υψηλό χρόνο στο μοντέλο RIPPER και πιο χαμηλό στο K-nn.

b. Εκτίμηση αποδοτικότητα ς κατηγοριοποίησης με το 20newsgroup

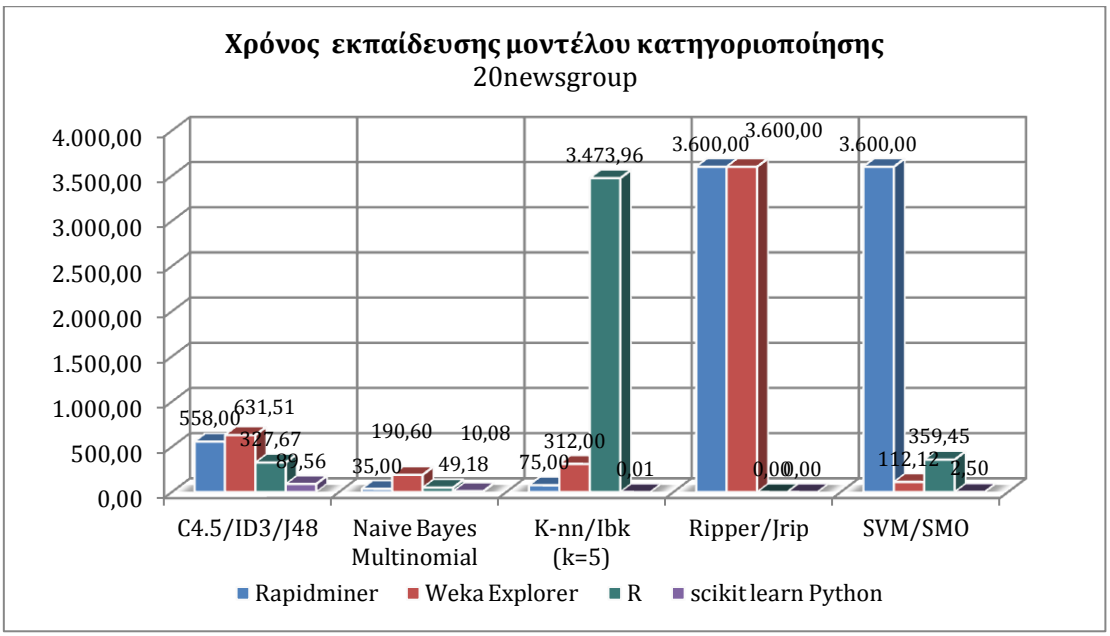
Στον Πίνακα 10 καταγράφονται αναλυτικά οι χρόνοι εκτέλεσης της εκπαίδευσης και της πρόβλεψης του κάθε αλγόριθμου ξεχωριστά, για το σύνολο δεδομένων 20newsgroup.

		Rapidminer	**Weka Explorer**	**R**	**scikit learn Python**
C4.5/ID3/J48	Χρόνος Εκπαίδευσης του μοντέλου	558,00	631,51	327,67	89,56
	Χρόνος εκτέλεσης Πρόβλεψης του μοντέλου	-	65,00	-	0,01
	Συνολικός Χρόνος Εκτέλεσης (wall clock time)	1.166,40	696,51	-	89,56

Naive Bayes Multinomial	Χρόνος Εκπαίδευσης του μοντέλου	193,00	0,23	49,18	10,08
	Χρόνος εκτέλεσης Πρόβλεψης του μοντέλου	134,60	135,00	1.711,48	0,02
	Συνολικός Χρόνος Εκτέλεσης (wall clock time)	327,60	135,23	1.816,45	10,10
K-nn/Ibk (k=5)	Χρόνος Εκπαίδευσης του μοντέλου	75,00	0,02	3.473,96	0,01
	Χρόνος εκτέλεσης Πρόβλεψης του μοντέλου	1.955,40	48,00	98,00	2,83
	Συνολικός Χρόνος Εκτέλεσης (wall clock time)	2.358,00	48,02	3.625,55	2,84
Ripper/Jrip	Χρόνος Εκπαίδευσης του μοντέλου	>3600	>3600	-	-
	Χρόνος εκτέλεσης Πρόβλεψης του μοντέλου	-	-	-	-
	Συνολικός Χρόνος Εκτέλεσης (wall clock time)	>3600	>3600	-	-
SVM/SMO	Χρόνος Εκπαίδευσης του μοντέλου	>3600	112,12	359,45	2,50
	Χρόνος εκτέλεσης Πρόβλεψης του μοντέλου	-	1,50	40,56	0,01
	Συνολικός Χρόνος Εκτέλεσης (wall clock time)	>3600	113,62	457,91	2,51

Πίνακας 10. Χρόνοι εκτέλεσης των αλγορίθμων κατηγοριοποίησης ανά σύστημα (sec)για το σύνολο δεδομένων 20Newsgroup.

Συγκρίνοντας το χρόνο που χρειάστηκε για να εκπαιδευτούν τα μοντέλα, με βάση το παρακάτω γράφημα, τον υψηλότερο χρόνο εκπαίδευσης τον κατέχει ο RIPPER με το Rapidminer και το Weka να έχουν ξεπεράσει το όριο της μίας ώρας. Επίσης, το Rapidminer έχει ξεπεράσει το όριο της μιας ώρας και με το μοντέλο SVM. Το scikit learn συνεχίζει να κατέχει τους πιο γρήγορους χρόνους για όλα τα μοντέλα με μέσο όρο χρόνου εκπαίδευσης 25,55s, πιο γρήγορο χρόνο στον K-nn και πιο αργό στο C4.5. Το Weka είναι δεύτερο σε χρόνους με μέσο όρο 868,78s, με πιο γρήγορο χρόνο στο μοντέλο K-nn και πιο αργό στον RIPPER. Τέλος, ακολουθούν το R και το Rapidminer με μέσο όρος χρόνου εκπαίδευσης 1.052,56 και 1.60520σ, αντίστοιχα. Η πιο γρήγορη εκπαίδευση του R και του Rapidminer γίνεται για το μοντέλο Naïve Bayes και K-nn, αντίστοιχα.



Γράφημα 22. Απεικόνιση χρόνου εκπαίδευσης αλγορίθμων κατηγοριοποίησης ανά σύστημα σε sec, για το σύνολο δεδομένων 20newsgroup.

c. Αποδοτικότητα συσταδοποίησης για το BBC news

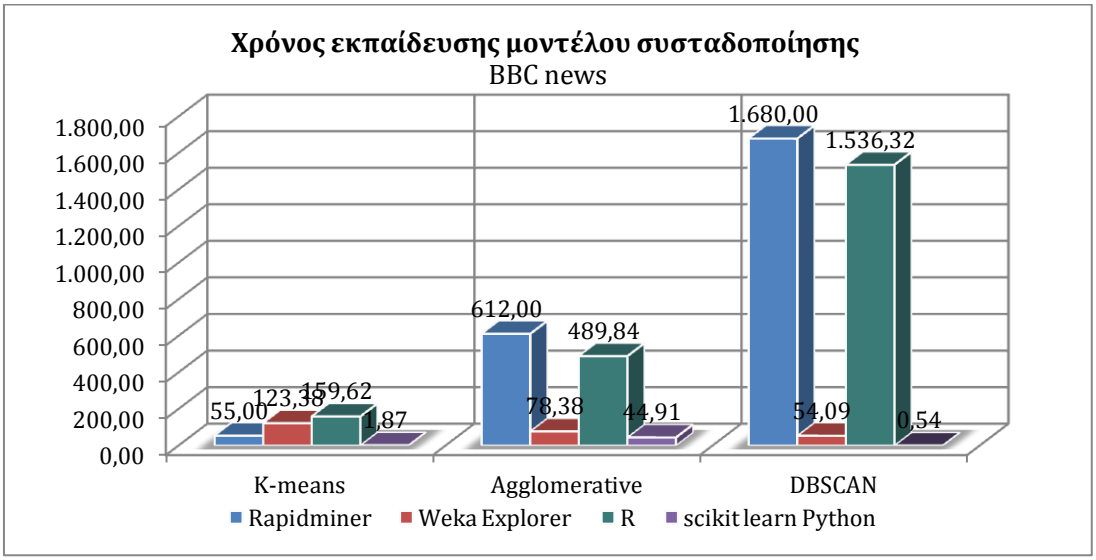
Στον Πίνακα 11 καταγράφονται αναλυτικά οι χρόνοι εκτέλεσης της εκπαίδευσης του μοντέλου και οι συνολικοί χρόνοι εκτέλεσης του κάθε αλγόριθμου συσταδοποίησης ξεχωριστά, για το σύνολο δεδομένων BBC news.

Συγκρίνοντας το χρόνο που χρειάστηκε για να εκπαιδευτούν τα μοντέλα, με βάση το γράφημα 23, τον υψηλότερο χρόνο εκπαίδευσης τον κατέχει ο DBSCAN, λόγω των υψηλών χρόνων που απαντώνται για το Rapidminer και το R. Το Rapidminer και το R έχουν τον υψηλότερο μέσο χρόνο εκπαίδευσης, ενώ ο χαμηλότερος μέσος χρόνος εκπαίδευσης παρατηρείται στο scikit learn.

		Rapidminer	Weka Explorer	R	scikit learn Python
K-means	Χρόνος Εκπαίδευσης του μοντέλου	55,00	98,61	159,62	1,87
	Συνολικός Χρόνος Εκτέλεσης (wall clock time)	76,80	120,13	170,67	6,19
Agglomerative	Χρόνος Εκπαίδευσης του μοντέλου	612,00	64,76	489,84	44,91

	Συνολικός Χρόνος Εκτέλεσης (wall clock time)	1.289,40	120,18	500,30	130,78
DBSCAN	Χρόνος Εκπαίδευσης του μοντέλου	1.680,00	78,19	1.536,32	0,54
	Συνολικός Χρόνος Εκτέλεσης (wall clock time)	2.247,00	120,00	1.750,55	4,44

Πίνακας 11. Χρόνοι εκτέλεσης των αλγορίθμων κατηγοριοποίησης ανά σύστημα (sec)για το σύνολο δεδομένων BBC news.



Γράφημα 23. Απεικόνιση χρόνου εκπαίδευσης αλγορίθμων κατηγοριοποίησης ανά σύστημα σε sec, για το σύνολο δεδομένων 20newsgroup.

d. Αποδοτικότητα συσταδοποίησης 20newsgroup

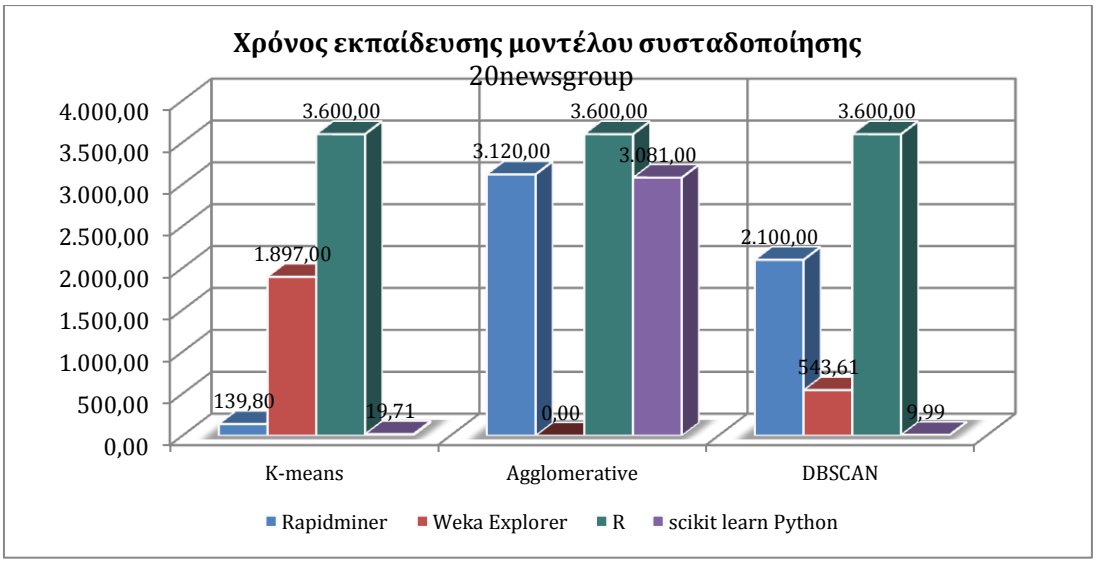
Στον Πίνακα 12 καταγράφονται αναλυτικά οι χρόνοι εκτέλεσης της εκπαίδευσης του μοντέλου και οι συνολικοί χρόνοι εκτέλεσης του κάθε αλγόριθμου συσταδοποίησης ξεχωριστά, για το σύνολο δεδομένων BBC news.

		Rapidminer	**Weka Explorer**	**R**	**scikit learn Python**
K-means	Χρόνος Εκπαίδευσης του μοντέλου	139,80	1.897,00	>3600	19,71
	Συνολικός Χρόνος Εκτέλεσης (wall clock time)	190,20	2.123,00	>3600	69,92
Agglomerative	Χρόνος Εκπαίδευσης του μοντέλου	3.120,00	-	>3600	3.206,00

	Συνολικός Χρόνος Εκτέλεσης (wall clock time)	>3600,00	-	>3600	>3600
DBSCAN	Χρόνος Εκπαίδευσης του μοντέλου	2.100,00	543,61	>3600	9,99
	Συνολικός Χρόνος Εκτέλεσης (wall clock time)	2.302,20	848,00	>3600	47,22

Πίνακας 12. Χρόνοι εκτέλεσης των αλγορίθμων κατηγοριοποίησης ανά σύστημα (sec)για το σύνολο δεδομένων 20newsgroup.

Συγκρίνοντας το χρόνο που χρειάστηκε για να εκπαιδευτούν τα μοντέλα, με βάση το γράφημα 24, το Rapidminer, το scikit learn και το R ξεπέρασαν το όριο της μιας ώρας κατά τη διαδικασία πρόβλεψης και αξιολόγησης του αλγορίθμου Agglomerative, οπότε δεν εξήχθησαν τελικά αποτελέσματα, ενώ το Weka τερματισε τη διεργασία λόγω μη διαθέσιμης μνήμης. Επίσης, το R ξεπέρασε το όριο της μιας ώρας για το μοντέλο DBSCAN. Το scikit learn κατέχει τους πιο γρήγορους χρόνους για τα μοντέλα του K-means και DBSCAN, ενώ το R είναι το πιο αργό. Υπολογίζοντας το μέσο χρόνο εκπαίδευσης ανά σύστημα παρατηρείται ότι η πιο γρήγορη εκπαίδευση παρέχεται από το scikit learn, ακολουθεί το Weka έπειτα το Rapidminer και τέλος το R.

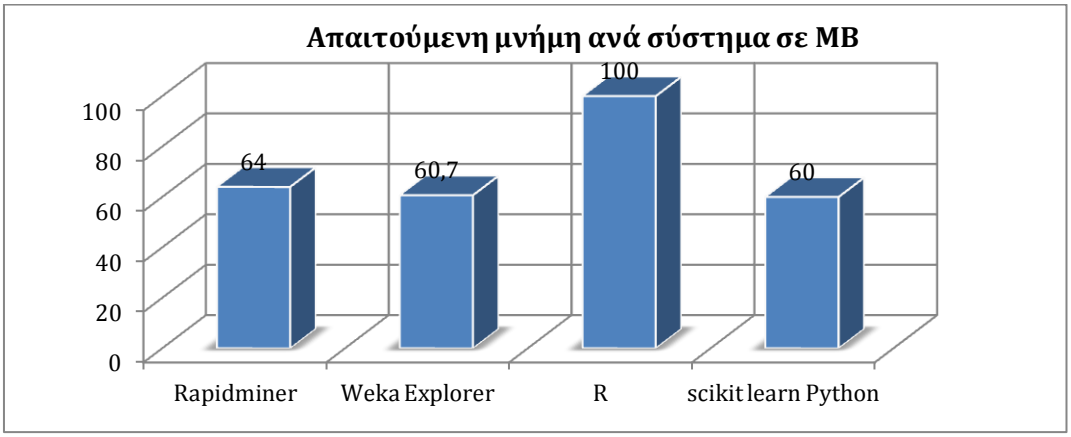


Γράφημα 24. Απεικόνιση χρόνου εκπαίδευσης αλγορίθμων κατηγοριοποίησης ανά σύστημα σε sec, για το σύνολο δεδομένων 20newsgroup.

5.3.4 Σύγκριση απαιτήσεων υπολογιστικών πόρων

Ακολούθως, εξετάζονται οι ελάχιστοι υπολογιστικοί πόροι που απαιτούνται από το κάθε σύστημα ξεχωριστά και η μέγιστη κατανάλωση μνήμης για την υλοποίηση του κάθε αλγόριθμου. Οι απαιτούμενοι και μέγιστοι πόροι μνήμης είναι ένα σημαντικό κριτήριο σύγκρισης των συστημάτων, διότι κρίνεται βασική η δυνατότητα να μπορούν να επεξεργαστούν μεγάλα δεδομένα χωρίς να κρασάρουν ή να τερματίζουν τις διεργασίες λόγω μη διαθέσιμης μνήμης. Όσο μικρότερη μνήμη απαιτείται τόσο πιο λίγες είναι οι απαιτήσεις σε υπολογιστικούς πόρους και τόσο πιο γρήγορα ολοκληρώνεται η επεξεργασία των δεδομένων. Ο έλεγχος των απαιτήσεων των συστημάτων R και scikit learn, σε μνήμη διενεργήθηκε μέσω της εφαρμογής Perfmon των Windows απομονώνοντας κάθε φορά την εκάστοτε διεργασία των δυο συστημάτων. Το Rapidminer και το Weka διαθέτουν ένα εργαλείο ελέγχου του συστήματος όπου καταγράφεται η μέγιστη και η συνολική κατανάλωση μνήμης κατά τη διάρκεια τρεξίματος των διεργασιών. Η καταγραφή των τιμών της μνήμης έγινε σε MB.

Η ελάχιστη απαιτούμενη μνήμη για τα υπό εξέταση συστήματα καταγράφεται στο παρακάτω γράφημα με το R να έχει τη μεγαλύτερη απαίτηση.

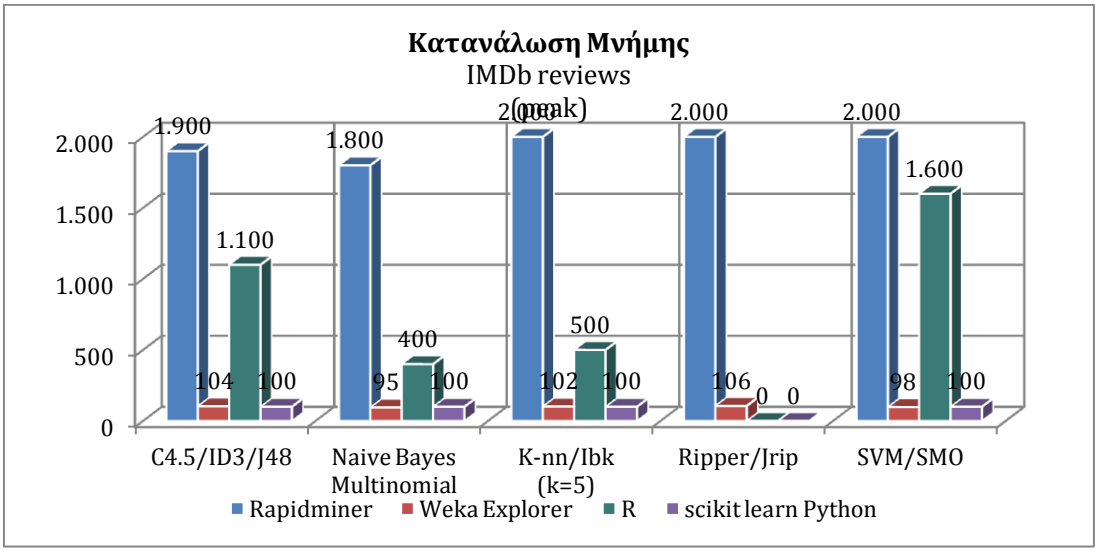


Γράφημα 25. Απαιτούμενη μνήμη ανά σύστημα σε MB.

a. Κατανάλωση μνήμης για τους αλγορίθμους κατηγοριοποίησης

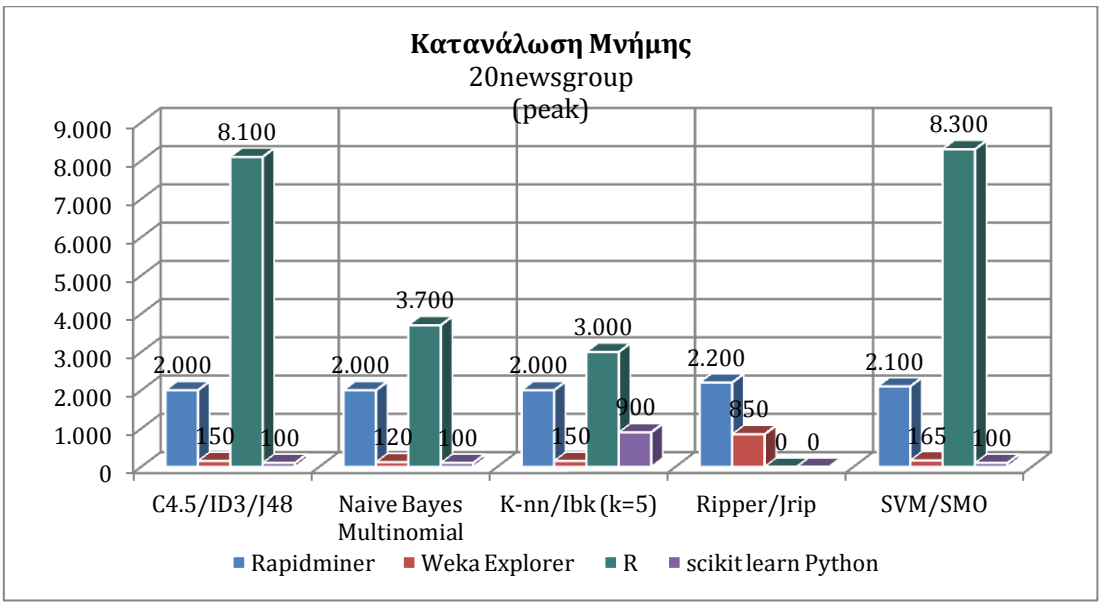
Η μέγιστη κατανάλωση μνήμης ανά σύστημα και αλγόριθμο για το σύνολο δεδομένων του IMDb reviews, απεικονίζεται στο παρακάτω γράφημα, με το Rapidminer και δεύτερο το R να καταναλώνουν τη μεγαλύτερη μνήμη για όλα τα πειράματα, ενώ το Weka και το scikit learn την ελάχιστη. Εξετάζοντας τη μνήμη που καταναλώνεται ανά

αλγόριθμο φαίνεται ότι ο SVM απαιτεί την υψηλότερη μνήμη, ακολουθεί ο C4.5, ο K-nn, ο RIPPER και τέλος ο Naïve Bayes.



Γράφημα 26. Κατανάλωση μνήμης ανά σύστημα και ανά αλγόριθμο κατηγοριοποίησης σε MB, για το σύνολο δεδομένων IMDb reviews.

Στη συνέχεια, εξετάζεται η μέγιστη κατανάλωση μνήμης ανά σύστημα και αλγόριθμο για το σύνολο δεδομένων του 20newsgroup όπως απεικονίζεται στο παρακάτω γράφημα. Υπενθυμίζεται ότι το συγκεκριμένο σύνολο δεδομένων είναι κατά πολύ μεγαλύτερο σε σύγκριση με το IMDb reviews, οπότε καταναλώνεται μεγαλύτερη μνήμη. Το R καταναλώνει τη μεγαλύτερη μνήμη με μεγάλη διαφορά από τα άλλα συστήματα για όλα τα πειράματα, ενώ το Weka την ελάχιστη. Το Rapidminer βρίσκεται στα ίδια επίπεδα σε σύγκριση με το IMDb reviews ενώ το scikit learn αυξάνει την κατανάλωση μνήμης δραματικά για τον αλγόριθμο K-nn.

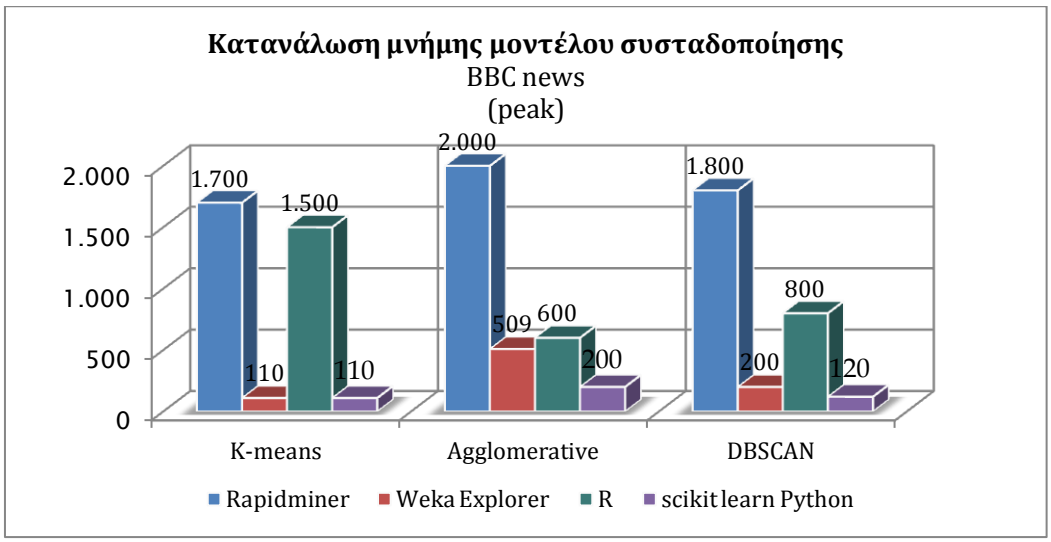


Γράφημα 27. Κατανάλωση μνήμης ανά σύστημα και ανά αλγόριθμο κατηγοριοποίησης σε MB, για το σύνολο δεδομένων 20newsgroup.

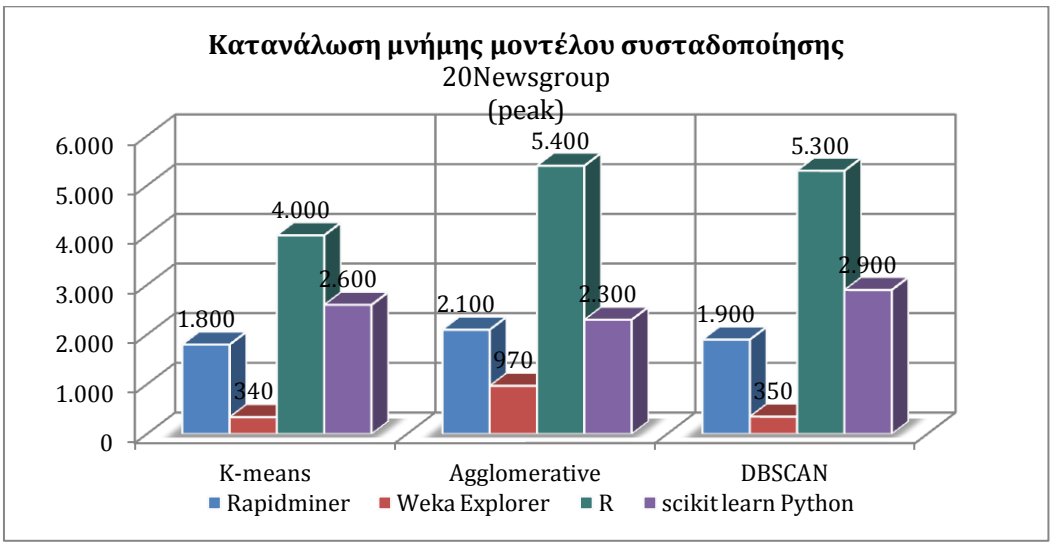
b. Κατανάλωση μνήμης για τους αλγορίθμους συσταδοποίησης

Η μέγιστη κατανάλωση μνήμης ανά σύστημα και αλγόριθμο συσταδοποίησης, για το σύνολο δεδομένων του BBC news και του 20newsgroup, απεικονίζονται στα Γραφήματα 28 και 29, αντίστοιχα. Υπενθυμίζεται ότι το συγκεκριμένο σύνολο δεδομένων είναι κατά πολύ μεγαλύτερο σε σύγκριση με το BBC news, οπότε καταναλώνεται μεγαλύτερη μνήμη.

Για το σύνολο δεδομένων BBC news ,το Rapidminer και δεύτερο το R καταναλώνουν τη μεγαλύτερη μνήμη για όλα τα πειράματα, ενώ το scikit learn και Weka την ελάχιστη. Εξετάζοντας τη μνήμη που καταναλώνεται ανά αλγόριθμο φαίνεται ότι ο K-means απαιτεί την υψηλότερη μνήμη, ακολουθεί ο Agglomerative, και τέλος ο DBSCAN. Ενώ παρατηρώντας τη μέγιστη κατανάλωση μνήμης για το 20newsgroup, οι απαιτήσεις σε υπολογιστικούς πόρους για το R είναι πολύ υψηλές για όλα τα πειράματα σε σύγκριση με τα υπόλοιπα συστήματα, ενώ το Weka έχει τις ελάχιστες απαιτήσεις. Το Rapidminer βρίσκεται περίπου στα ίδια επίπεδα σε σύγκριση με το σύνολο δεδομένων BBC news ενώ το scikit learn αυξάνει την κατανάλωση μνήμης δραματικά για όλα τα πειράματα.



Γράφημα 28. Κατανάλωση μνήμης ανά σύστημα και ανά αλγόριθμο κατηγοριοποίησης σε MB, για το σύνολο δεδομένων BBC news.



Γράφημα 29. Κατανάλωση μνήμης ανά σύστημα και ανά αλγόριθμο κατηγοριοποίησης σε MB, για το σύνολο δεδομένων 20newsgroup.

5.4 Συμπεράσματα

Προκειμένου να ολοκληρωθεί η συγκριτική ανάλυση των πειραματικών αποτελεσμάτων που παρουσιάστηκαν παραπάνω, αναφορικά με τα κριτήρια των μετρικών, του χώρου και του χρόνου των υλοποιήσεων, παρουσιάζονται τέσσερις πίνακες οι οποίοι περιέχουν τις καλύτερες και τις χειρότερες τιμές ανά σύστημα, καταγράφοντας παράλληλα τον αλγόριθμο που αφορούν.

Αρχικά εξετάζονται οι τιμές που εξήχθησαν αναφορικά με την υλοποίηση της κατηγοριοποίησης για το σύνολο δεδομένων IMDb reviews. Συνοψίζοντας τον παρακάτω πίνακα, για ένα σχετικά μικρό σύνολο δεδομένων όπως το IMDb reviews, το scikit learn κατέχει την πιο υψηλή τιμή ακρίβειας για όλα τα μοντέλα, σε αντίθεση με το R που έχει τις χαμηλότερες τιμές ακριβείας. Επίσης, εκτός από την υλοποίηση των επιτυχόντων μοντέλων κατηγοριοποίησης, το scikit learn είναι το πιο γρήγορο σύστημα και καταναλώνει του λιγότερους υπολογιστικούς πόρους. Από την άλλη πλευρά το Rapid Miner είναι το πιο αργό σύστημα όλων και καταναλώνει τους περισσότερους υπολογιστικούς πόρους.

Επιπλέον, εξετάζοντας τα μοντέλα της κατηγοριοποίησης ο αλγόριθμος Naïve Bayes καταναλώνει τους λιγότερους υπολογιστικούς πόρους και ο RIPPER τους περισσότερους, ενώ οι πιο γρήγοροι χρόνοι παρατηρούνται κατά την υλοποίηση του K-nn και οι πιο αργοί κατά την υλοποίηση του RIPPER. Τέλος, παρατηρώντας την ακρίβεια των μοντέλων ο K-nn κατέχει τα χαμηλότερα ποσοστά ακριβείας για όλα τα συστήματα.

		Καλύτερες Τιμές	**Αλγόριθμος**	**Χειρότερες Τιμές**	**Αλγόριθμος**
scikit learn	**Ακρίβεια**	88%	C4.5/J48	64%	K-nn
	Χρόνος εκπαίδευσης	0s	K-nn	4,78s	C4.5/J48
	Μνήμη	100 MB	C4.5/J48, Naïve Bayes, SVM, K-nn	-	-
R	**Ακρίβεια**	80,80%	SVM	56,60%	K-nn
	Χρόνος εκπαίδευσης	6,77s	Naïve Bayes	158,87s	SVM
	Μνήμη	400 MB	Naïve Bayes	1.600 MB	SVM
RapidMiner	**Ακρίβεια**	77,40%	SVM	66,40%	K-nn
	Χρόνος εκπαίδευσης	1s	K-nn	2.336s	RIPPER
	Μνήμη	1.800 MB	Naïve Bayes	2.000 MB	RIPPER
Weka	**Ακρίβεια**	82,80%	Naïve Bayes	61,40%	K-nn
	Χρόνος εκπαίδευσης	0,02s	K-nn	53,88s	RIPPER
	Μνήμη	95MB	Naïve Bayes	106 MB	RIPPER

Πίνακας 13. Συγκεντρωτικά αποτελέσματα κατηγοριοποίησης ανά σύστημα για το σύνολο δεδομένων IMDb reviews.

Συνεχίζοντας την παρατήρηση και αναφορικά με την υλοποίησης της κατηγοριοποίησης για το σύνολο δεδομένων 20newsgroup, σύμφωνα με τον παρακάτω πίνακα, το scikit learn συνεχίζει να κατηγοριοποιεί με επιτυχία τα μοντέλα έναντι των άλλων συστημάτων, όπως επίσης συνεχίζει να είναι το πιο γρήγορο σύστημα. Αναφορικά με τη χαμηλότερη κατανάλωση των υπολογιστικών πόρων, αυτή τη φορά βρίσκεται πρώτο το Weka. Επιπλέον, το Rapidminer συνεχίζει να είναι το πιο αργό σύστημα ενώ το R καταλαμβάνει τους περισσότερους υπολογιστικούς πόρους κατά την εκτέλεση των διεργασιών.

Συγκρίνοντας την ακρίβεια των μοντέλων, το SVM κατέχει τα πιο υψηλά ποσοστά ενώ το μοντέλο K-nn εκπαιδεύεται πιο γρήγορα από τα υπόλοιπα μοντέλα, ενώ μαζί με το Naïve Bayes καταναλώνουν τη λιγότερη μνήμη. Τέλος, ο RIPPER λόγω της πολυπλοκότητας του έχει την πιο αργή εκπαίδευση και καταναλώνει τους περισσότερους υπολογιστικούς πόρους.

		Καλύτερες Τιμές	**Αλγόριθμος**	**Χειρότερες Τιμές**	**Αλγόριθμος**
scikit learn	**Ακρίβεια**	93,00%	SVM	66%	C4.5/J48
	Χρόνος εκπαίδευσης	0,1s	K-nn	89,56s	C4.5/J48
	Μνήμη	100 MB	C4.5/J48, Naïve Bayes, SVM	900 MB	K-nn
R	**Ακρίβεια**	82,60%	SVM	50,10%	K-nn
	Χρόνος εκπαίδευσης	49,18s	Naïve Bayes	3.473s	K-nn
	Μνήμη	3.000 MB	K-nn	8.300 MB	SVM
RapidMiner	**Ακρίβεια**	62,43%	K-nn	19,01%	SVM
	Χρόνος εκπαίδευσης	75s	K-nn	3.600s	RIPPER, SVM
	Μνήμη	2.000 MB	K-nn, Naïve Bayes, C4.5/J48	2.200 MB	RIPPER
Weka Explorer	**Ακρίβεια**	88,78%	Naïve Bayes	62,36%	C4.5/J48
	Χρόνος εκπαίδευσης	0,02s	K-nn	3.600s	RIPPER

	Μνήμη	120 MB	Naïve Bayes	850 MB	RIPPER

Πίνακας 14. Συγκεντρωτικά αποτελέσματα κατηγοριοποίησης ανά σύστημα για το σύνολο δεδομένων 20newsgroup.

Συνεχίζοντας στο πεδίο της συσταδοποίησης για το σύνολο δεδομένων BBC news, παρατηρείται ότι το RapidMiner κατέχει το μέσο όρο των καλύτερων τιμών καθαρότητας των εξεταζόμενων μοντέλων, όμως καταναλώνει τους περισσότερους υπολογιστικούς πόρους και είναι το πιο αργό σύστημα. Το πιο γρήγορο σύστημα είναι το scikit learn, όπως επίσης είναι και αυτό με τη μικρότερη κατανάλωση μνήμης.

Ο αλγόριθμος DBSCAN κατέχει τα χαμηλότερα ποσοστά καθαρότητας άρα σωστής συσταδοποίησης και τους πιο αργούς χρόνους εκπαίδευση. Αντίθετα ο K-means έχει τα καλύτερα ποσοστά καθαρότητας και επίσης καταναλώνει τους λιγότερους υπολογιστικούς πόρους, επιπλέον ο K-means εκπαιδεύεται πιο γρήγορα από τα υπόλοιπα μοντέλα , ενώ ο Agglomerative καταναλώνει το μεγαλύτερο όγκο μνήμης.

		Καλύτερες τιμές	**Αλγόριθμος**	**Χειρότερες τιμές**	**Αλγόριθμος**
scikit learn	**Purity**	96,22%	K-means	0%	DBSCAN
	Χρόνος εκπαίδευσης	0,54s	DBSCAN	44,91s	Agglomerative
	Μνήμη	110 MB	K-means	200 MB	Agglomerative
R	**Purity**	54,40%	K-means	27,20%	DBSCAN
	Χρόνος εκπαίδευσης	159,62s	K-means	1.536s	DBSCAN
	Μνήμη	600 MB	Agglomerative	1.500 MB	K-means
RapidMiner	**Purity**	87,06%	K-means	36,94%	DBSCAN
	Χρόνος εκπαίδευσης	55s	K-means	1.680s	DBSCAN
	Μνήμη	1.700 MB	K-means	2.000 MB	Agglomerative
Weka	**Purity**	90,79%	K-means	11,51%	DBSCAN
	Χρόνος εκπαίδευσης	54,09s	DBSCAN	123,38s	K-means
	Μνήμη	110 MB	K-means	509 MB	Agglomerative

Πίνακας 15. Συγκεντρωτικά αποτελέσματα συσταδοποίησης ανά σύστημα για το σύνολο δεδομένων BBC news.

Τέλος, εξετάζοντας το τελευταίο πίνακα στον οποίο καταγράφονται οι καλύτερες και οι χειρότερες τιμές της υλοποίησης της συσταδοποίησης για το σύνολο δεδομένων 20newsgroup, απεικονίζεται το scikit learn να κατέχει τον καλύτερο μέσο όρο καθαρότητας των μοντέλων, ενώ το Weka είναι το πιο γρήγορο σύστημα όλων και με τη μικρότερη κατανάλωση μνήμης. Από την άλλη πλευρά το R έχει τους υψηλότερους χρόνους εκπαίδευσης και καταναλώνει τη μεγαλύτερη μνήμη.

Ο αλγόριθμος K-means συνεχίζει να εκπαιδεύεται πιο γρήγορα και να δίνει τα καλύτερα ποσοστά καθαρότητας με το σύνολο δεδομένων 20newsgroup. Επιπλέον, καταναλώνει τους λιγότερους πόρους σε αντίθεση με τον Agglomerative ο οποίος παρατηρείται να έχει τους πιο αργούς χρόνους εκπαίδευσης.

		Καλύτερες τιμές	**Αλγόριθμος**	**Χειρότερες τιμές**	**Αλγόριθμος**
scikit learn	**Purity**	66,21%	K-means	0,16%	DBSCAN
	Χρόνος εκπαίδευσης	9,99s	DBSCAN	3.081s	Agglomerative
	Μνήμη	2.300 MB	K-means	2.900 MB	Agglomerative
R	**Purity**	-	-	-	-
	Χρόνος εκπαίδευσης	-	-	-	-
	Μνήμη	4.000 MB	K-means	5.400 MB	Agglomerative
RapidMiner	**Purity**	30,28%	K-means	30,27%	DBSCAN
	Χρόνος εκπαίδευσης	139,8s	K-means	3.120s	Agglomerative
	Μνήμη	1.800 MB	K-means	2.100 MB	Agglomerative
Weka Explorer	**Purity**	51,44%	K-means	1,55%	DBSCAN
	Χρόνος εκπαίδευσης	543,61s	DBSCAN	1.897s	K-means
	Μνήμη	340 MB	K-means	970 MB	Agglomerative

Πίνακας 16. Συγκεντρωτικά αποτελέσματα συσταδοποίησης ανά σύστημα για το σύνολο δεδομένων 20newsgroup.

Πιο συγκεκριμένα, συνδυάζοντας τα αποτελέσματα της υλοποίησης της κατηγοριοποίησης για το μικρό προς μεσαίο σύνολο δεδομένων IMDb reviews, όπως

αναλύθηκαν παραπάνω, προτείνεται για την υλοποίηση των μοντέλων C4.5/J48/ID3 και SVM το scikit learn λόγω των υψηλών ποσοστών ακρίβειας που προσφέρει, των γρήγορων χρόνων εκπαίδευσης αλλά και χαμηλών υπολογιστικών πόρων που καταναλώνει. Στη συνέχεια για πειράματα όπου υλοποιείται ο αλγόριθμος Naïve Bayes, περισσότερο ενδείκνυται το Weka και για το RIPPER το RapidMiner. Επίσης, η εκπαίδευση του αλγόριθμου K-nn απαντάται πιο επιτυχημένη στο Rapidminer, ενώ στο R ως πιο επιτυχημένο μοντέλο αλλά με τη μεγαλύτερη απαίτηση μνήμης και μεγαλύτερο χρόνο εκπαίδευσης, υλοποιείται το SVM.

Επεκτείνοντας τη σύγκριση των αποτελεσμάτων κατηγοριοποίησης για το σύνολο δεδομένων 20newsgroup, το οποίο είναι ένα μεσαίο προς μεγάλο σύνολο, παρατηρείται ότι το scikit learn παρέχει μια κατά πολύ επιτυχημένη κατηγοριοποίηση για το μοντέλο SVM το οποίο είναι το πιο επιτυχημένο μοντέλο που υλοποιεί επίσης το R. Έπειτα, το Weka επιλέγεται για την υλοποίηση του αλγορίθμου Naïve Bayes, όπως ίσχυε και για το IMDb reviews, ενώ το scikit learn προτείνεται για τον αλγόριθμο K-nn.

Τέλος, εξετάζοντας τα αποτελέσματα των πειραμάτων συσταδοποίησης σε συνδυασμό με τα κριτήρια που αναλύθηκαν παραπάνω, το scikit learn και το Weka συσταδοποιούν επιτυχημένα τον αλγόριθμο K-means στην περίπτωση και των δύο συνόλων των δεδομένων που εξετάστηκαν (μικρό προς μεσαίο/ μεσαίο προς μεγάλο), ενώ το Rapidminer εξάγει τα καλύτερα αποτελέσματα για τον αλγόριθμο DBSCAN.

Εξετάζοντας τον αλγόριθμο Agglomerative για το σύνολο δεδομένων BBC news (μικρό προς μεσαίο), παρατηρείται ότι εκπαιδεύεται πολύ επιτυχημένα από το scikit learn, ενώ αντίθετα για το σύνολο δεδομένων 20newsgroup που είναι κατά πολύ μεγαλύτερο του πρώτου, απαιτούνται πολύ υψηλοί χρόνοι εκπαίδευσης και μεγάλοι υπολογιστικοί πόροι. Ο συνδυασμός της πολυπλοκότητας του αλγορίθμου με την πολυπλοκότητα του συνόλου των δεδομένων φαίνεται ότι δεν υποστηρίζεται από κανένα σύστημα επαρκώς. Επιπλέον, το R δεν εκτέλεσε επιτυχημένα κανένα από τους τρεις αλγορίθμους συσταδοποίησης για το εν λόγω σύνολο δεδομένων, λόγω των υψηλών χρόνων εκπαίδευσης που απαιτεί.

Συνοψίζοντας τα αποτελέσματα των πειραμάτων, αναφορικά με τα κριτήρια αποτελεσματικότητας, αποδοτικότητας και κατανάλωσης υπολογιστικών πόρων, εξάγεται ότι το σύστημα scikit learn είναι περισσότερο αποδοτικό και αποτελεσματικό

σε σύγκριση με τα άλλα συστήματα, για όλα τα μοντέλα που εξετάστηκαν, κατέχοντας τους πιο γρήγορους χρόνους εκπαίδευσης, όπως επίσης τα πιο υψηλά ποσοστά ακρίβειας κατηγοριοποίησης και καθαρότητας συσταδοποίησης. Επίσης, εξετάζοντας την κατανάλωση των υπολογιστικών πόρων που απαιτούν τα υπό εξέταση συστήματα το scikit learn και το Weka απαιτούν και καταναλώνουν τη μικρότερη μνήμη.

Κεφάλαιο 6
Επίλογος

Όπως προκύπτει από τις σελίδες της παρούσας μεταπτυχιακής διατριβής, λόγω της μετάβασης στην εποχή της πληροφορίας μέσω του Διαδικτύου τα τελευταία έτη, κρίνεται αναγκαία η ανάπτυξη τεχνικών και εργαλείων επεξεργασίας δεδομένων στο χώρο της μηχανικής μάθησης και συγκεκριμένα στο πεδίο της εξόρυξης κειμένου. Στόχος είναι η αξιοποίηση της ωφέλιμης πληροφορίας από το χρήστη ανάλογα με τις απαιτήσεις του.

Έτσι λοιπόν, στα πλαίσια της μελέτης των διαθεσίμων εργαλείων και τεχνικών του πεδίου εξόρυξης κειμένου, παρουσιάστηκαν, αναλύθηκαν και συγκρίθηκαν τα ποιοτικά χαρακτηριστικά των κορυφαίων διαθέσιμων εργαλείων ελεύθερου λογισμικού/ανοιχτού κώδικα, R, RapidMiner, scikit learn Python και Weka. Με σκοπό την αξιολόγηση των εν λόγω συστημάτων διεξήχθησαν πειράματα καλύπτοντας τα βασικά προβλήματα της κατηγοριοποίησης και συσταδοποίησης, των οποίων τα αποτελέσματα δίνουν στον εκάστοτε χρήστη τη δυνατότητα να καταλήξει στην

επιλογή του καταλληλότερου εργαλείου, αναφορικά με τη διεργασία που θέλει να πραγματοποιήσει.

Γενικά και τα τέσσερα εργαλεία επικεντρώνονται στη χρησιμότητα και διάδραση μέσω του αριθμού και τύπο των λειτουργιών τους, υποστηρίζουν την επεκτασιμότητα μέσω αύξησης του πηγαίου κώδικα ή ακόμη καλύτερα μέσω χρήσης διεπιφανείων, ενώ παρέχουν ευελιξία οπτικού προγραμματισμού και γραφικών διεπιφανειών χρήστη. Επίσης, διαθέτουν τεράστιες εργαλειοθήκες που είναι πολύ καλά τεκμηριωμένες, κοινότητες χρηστών με τη βοήθεια των οποίων γίνονται συνδιασκέψεων ή ομάδων συζητήσεων για υποστήριξη των τελικών χρηστών και ανταλλαγή ιδεών. Ο βαθμός στον οποίο υλοποιούνται όλα τα παραπάνω φυσικά διαφέρει από το ένα λογισμικό στο άλλο.

Το RapidMiner προσφέρει την ενσωμάτωση του περιβάλλοντος του με οπτικά ελκυστική και φιλική προς το χρήστη διεπαφή, έχει ένα μεγάλο σύνολο τελεστών, κάτι που το κάνει κατάλληλο για τη σύγκριση διαφορετικών μεθόδων μηχανικής εκμάθησης και στατιστικής. Τα πάντα στο RapidMiner επικεντρώνονται σε διαδικασίες που περιέχουν υποεπεξεργασίες. Η ροή των δεδομένων κατασκευάστηκαν με drag-and-drop τελεστές οι οποίοι συνδέονται μεταξύ τους μέσω των εισόδων και των εξόδων των αντίστοιχων τελεστών. Υπάρχουν tutorials διαθέσιμα για πολλές ειδικές εργασίες, ώστε το εργαλείο έχει μια σταθερή καμπύλη εκμάθησης. Είναι επίσης πολύ καλό για κατασκευή μοντέλων και εγκυροποίηση δεδομένων και διαδικασιών ενώ παρέχει πολλούς τρόπους απεικόνισης των μοντέλων αυτών και των συνόλων δεδομένων. Η γραφική διεπιφάνεια χρήστη του είναι σχετικά απλή και με τη βοήθεια των εγχειριδίων χρήσης και την αφθονία παραδειγμάτων που υπάρχουν στο διαδίκτυο, μπορεί κάποιος αρχάριος χρήστης να επεξεργαστεί και να αναλύσει δεδομένα. Από την άλλη πλευρά απαιτεί μεγάλο όγκο υπολογιστικών πόρων σε σύγκριση με τα άλλα εργαλεία και οι χρόνοι εκτέλεσης των μοντέλων ειδικά για μεγάλα δεδομένα είναι υψηλοί.

Το Weka Explorer αποτελείται από πολλές διεπιφάνειες χρήστη, ανάλογα με τις ανάγκες του κάθε χρήστη. Περιέχει ένα μεγάλο αριθμό τελεστών και είναι πολύ απλό στη χρήση για ένα αρχάριο χρήστη. Αποτελείται από μία αλληλουχία καρτελών που περιέχουν σχόλια τόσο για τη λειτουργία τους, αλλά και τη λειτουργία των επιμέρους τελεστών τους. Ωστόσο, δε μπορεί να εκτελέσει όσες λειτουργίες εκτελούνται σε άλλα πακέτα (όπως το RapidMiner). Είναι ένα πρόγραμμα ευρέως διαδεδομένο και μπορεί

κανείς να βρει πολύ υλικό για την τεκμηρίωση και την ανάλυση του. Αν και δεν είναι ένα ενιαίο εργαλείο επιλογής σε εξόρυξη κειμένου, το Weka είναι αρκετά ισχυρό και ευέλικτο με την υποστήριξη μιας μεγάλης κοινότητας. Επίσης, υποστηρίζει πολλές διαδικασίες αξιολόγησης των μοντέλων, ενώ είναι περισσότερο προσανατολισμένο προς την κατηγοριοποίηση και τα προβλήματα παλινδρόμησης και λιγότερο προς τις μεθόδους συσταδοποίησης και αυτό φαίνεται από την σύγκριση της ακρίβειας των πειραμάτων που διεξήχθησαν. Δεν απαιτεί υψηλούς υπολογιστικούς πόρους όπως το RapidMiner ή το R και είναι σχετικά γρήγορο τουλάχιστον όσον αφορά την κατηγοριοποίηση.

Το R είναι μια ισχυρή επιλογή για διεργασίες εξόρυξης δεδομένων και κειμένου. Επεκτείνεται με πολυάριθμα πακέτα σε σύγκριση με τα άλλα συστήματα, για όλα τα είδη των υπολογιστικών εργασιών. Το εργαλείο προσφέρει μόνο μια απλή διεπιφάνεια εισόδου γραμμής εντολών με αποτέλεσμα να μην είναι φιλικό προς το αρχάριο χρήστη επειδή όλες οι εντολές πρέπει να εισαχθούν στη γλώσσα R, οπότε ο χρήστης πρέπει να έχει γνώση της γλώσσας. Επίσης, προσφέρει εφαρμογές πολλών αλγορίθμων μηχανικής μάθησης, καθώς και η πλήρης μεθόδους απεικονίσεων των στατιστικών στοιχείων.

Το πακέτο scikit - learn είναι ένα δωρεάν πακέτο σε Python, που επεκτείνει το λειτουργικότητα της NumPy και SciPy με αρκετούς αλγορίθμους εξόρυξης δεδομένων, αλλά σε σύγκριση με τα υπόλοιπα πακέτα έχει τη μικρότερη λειτουργικότητα, διότι έχουν παραληφθεί κανόνες εκμάθησης κατηγοριοποίησης και κανόνες συσχέτισης. Το scikit learn συνεχίζει να βελτιώνεται με την αποδοχή πολύτιμων συνεισφορών από πολλούς συνεργάτες και υποστηρίζεται τόσο από το INRIA και Google Summer of Code. Ένα από τα κύρια δυνατά του σημεία είναι η εμπεριστατωμένη και καλογραμμένη ηλεκτρονική τεκμηρίωση για όλες τις εφαρμόσιμες λειτουργίες του. Είναι επίσης συνολικά το πιο γρήγορο από τα άλλα τρία συστήματα παρά το γεγονός ότι είναι γραμμένο σε μια ερμηνευμένη γλώσσα και απαιτεί ελάχιστους υπολογιστικούς πόρους για τις διεργασίες του. Παρά τα πλεονεκτήματά του η χρήση του scikit learn απαιτεί ένα εξειδικευμένο προγραμματιστή στην Python λόγω της διεπιφάνειας της γραμμής εντολών του.

Συνοψίζοντας, όλα τα πακέτα έχουν τα πλεονεκτήματα και τα μειονεκτήματα τους, με αποτέλεσμα να μην υπάρχει κάποιο που είναι το καλύτερο. Η επιλογή του πακέτου

βασίζεται στην εμπειρία του εκάστοτε χρήστη, στις λειτουργίες που θέλει να αναλύσει, στο χρόνο που έχει στη διάθεση του και στους υπολογιστικούς πόρους που διαθέτει.

Βιβλιογραφία

[01] Sullivan Dan. (2001). Document warehousing and text mining. New York: Wiley.

[02] Aono M., Kobayashi M. (2002). Vector space models for search and cluster mining. In S. o. II. Springer.

[03] Boser B., Guyon I., Vapnik V. (1992). A training algorithm for optimal margin classifiers.

[04] Chisholm Andrew. Exploring Data with RapidMiner. packt publishing, open source community experience.

[05] Dice Tech. (Jan 2015). Dice Tech Salary Survey 2014-2015.

[06] Fayyad M. Usama, Piatetsky-Shapiro Gregory, Smyth Padjraic, Uthurusamy Ramasamy. (1996). Advances in Knowledge Discovery and Data Mining. American Association for Artificial Intelligence Menlo Park.

[07] Hearst Marti. (Jun 1999). Untangling Text Data Mining. Proceedings of ACL '99: the37th Annual Meeting of the Association for computational Linguistics.

[08] Hand D., Mannila H., Smyth P. (2001). Principles of Data Mining. Cambridge: MIT Press.

[09] Hotho A., Nurnberger A. & Paaß G. (2005). A Brief Survey of Text Mining.

[10] Karanikas H., Koundourakis G., Kopanakis I., Mavroudakis T. A Temporal Text Mining Application in Competitive Intelligence.

[11] Karanikas H., Theodoulidis B. Centre for Research in Information Management. (2002). " Knowledge Discovery in Text and Text Mining Software".

[12] Karanikas H., Tjortjis C. and Theodoulidis B. An Approach to Text Mining using Information Extraction. PKDD 2000 Knowledge Management: Theory and Applications.

[13] Kosala Raymond, Blockeel Hendrik. (2000, July). Web Mining Research: A Survey. ACM SIGKDD Newsletter .

[14] Mjolsness Eric, D. D. (2001, September). Machine Learning for Science: State of the Art and Future Prospects. Science , pp. 2051-2055.

[15] Nahm Un Yong, Mooney Raymond. (2003). "Text Mining with information extraction". Multilingualism and Electronic Language Management: Proceedings of the 4th International MIDP Colloquium.

[16] Radovanovic Miloš, Ivanovic Mirjana. (2008). TEXT MINING:APPROACHES AND APPLICATIONS. Novi Sad J. Math , pp. 227-234.

[17] Rajman M., Besançon R.Artificial Intelligence Laboratory, Computer Science Department, Swiss Federal Institute of Technology. (1997). Text Mining: Natural Language techniques and Text Mining applications. Chapman & Hall.

[18] Sebastiani Fabrizio, Consiglio Nazionale delle Ricerche. (2001). Machine Learning in Automated Text Categorization. Italy: ACM Computing Surveys.

[19] Sharp Mark, Rutgers University, School of Communication, Information and Library Studies. (2001). Text Mining.

[20] Venables W. N., Smith D. M. & Core team. An Introduction to R.

[21] Vidhya KA., Aghila G. (2010). Text Mining Process, Techniques and Tools : an Overview. International Journal of Information Technology and Knowledge Management , pp. 613-622.

[22] Vishal Gupta, Gurpreet S. Lehal. (Aug 2009). A Survey of Text Mining Techniques and Applications. Journal of Emerging Technologies in Web Intelligence, Vol 1, No 1 (2009).

[23] Witten H. Ian, Frank Eibe. (2005). Data Mining Practical Machine Learning Tool and Techniques. Morgan Kaufmann.

[24] Yanchi Liu, Zhongmou Li, Hui Xiong, Xuedong Gao, Junjie Wu, School of Economics and Management, University of Science and Technology Beijing, China. Understanding of Internal Clustering Validation Measures. 2010 IEEE International Conference on Data Mining.

[25] URL: http://mlg.ucd.ie/datasets/bbc.html.

[26] URL:http://nlp.stanford.edu/IR-book/html/htmledition/evaluation-of-clustering-1.html.

[27] URL: http://opensource.com.

[28] URL: http://scikit-learn.org/.

[29] URL: http://sourceforge.net/.

[30] URL: http://webdocs.cs.ualberta.ca/~eisner/measures.html.

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

[32] URL: http://www.cs.waikato.ac.nz/.

[33] URL: http://www.indeed.com/jobtrends/.

[34] URL: http://www.kdnuggets.com.

[35] URL: http://www.revolutionanalytics.com/.

[36] URL: http://www.statmethods.net/.

[37] URL: http://www.statsoft.com/Textbook/Text-Mining#applications.

[38] URL: https://cran.r-project.org/.

[39] URL: https://rapidminer.com/, http://docs.rapidminer.com/.

[40] URL: https://www.python.org/.

[41] URL: https://www.wikipedia.org/.

[42] URL: www.indeed.com.

[43] URL:https://www.gnu.org/.

Παράρτημα Α

Οδηγός Εγκατάστασης Συστημάτων

Α.1 Εγκατάσταση του συστήματος RapidMiner και των πακέτων/βιβλιοθηκών του

Η λήψη και στη συνέχεια η εγκατάσταση του συστήματος Rapid Miner, μπορεί να πραγματοποιηθεί από την ιστοσελίδα του Rapid Miner, πρέπει πρώτα όμως ο χρήστης να δημιουργήσει ένα λογαριασμό.

Στη συνέχεια της εγγραφής και της σύνδεσης του χρήστη στην ιστοσελίδα του Rapid Miner/download, μπορεί να επιλέξει την εγκατάσταση της πλατφόρμας η οποία είναι ανοιχτού κώδικα, επιλέγοντας το αντίστοιχο λειτουργικό σύστημα. Παράλληλα, θα πρέπει να εγκατασταθεί η πλατφόρμα Java από τη σελίδα της Oracle.

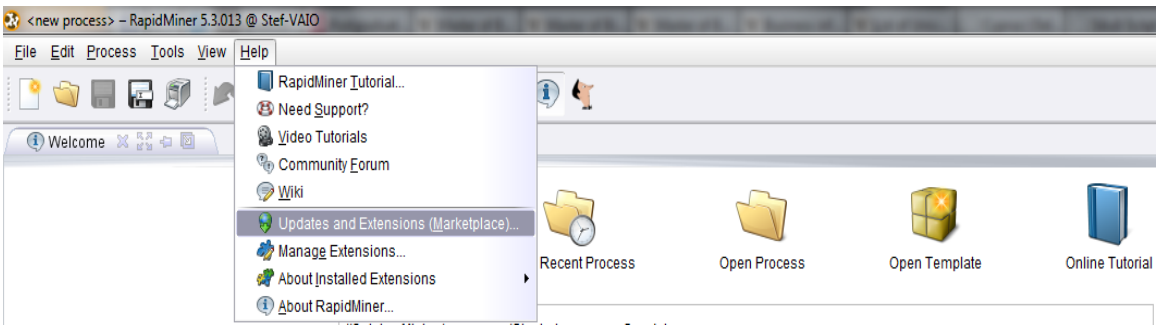
Στη συνέχεια, από τη σελίδα εγκατάστασης της ιστοσελίδας, επιλέγεται το λειτουργικό σύστημα στο οποίο θα εγκατασταθεί το εν λόγω λογισμικό, δηλαδή: Linux, Mac OS ή Windows, κάτω από

την επικεφαλίδα “Community Edition”. Εφόσον επιλεγεί η λήψη του συστήματος, και φτάσει στο τέλος της, επιλέγεται το ληφθέν αρχείο και εμφανίζονται οι οδηγίες που πρέπει να ακολουθήσει ο χρήστης προκειμένου να επιτευχθεί η εγκατάσταση. Μετά την εγκατάσταση θα υπάρχει μια νέα εγγραφή στο μενού έναρξης και ένα νέο εικονίδιο στην επιφάνεια εργασίας.

Με την έναρξη του Rapid Miner, εμφανίζεται μια οθόνη καλωσορίσματος και δίνεται η δυνατότητα επιλογής ανάμεσα στις πέντε παρακάτω λειτουργίες:

6. Έναρξη νέου έργου
7. Άνοιγμα πρόσφατου έργου
8. Άνοιγμα υπάρχοντος έργου
9. Έναρξη του οδηγού δημιουργίας νέου έργου
10. Άνοιγμα του on-line Προγράμματος εκμάθησης του RapidMiner

Γενικά, για να πραγματοποιηθεί η εγκατάσταση των διαθέσιμων πακέτων του Rapid Miner, επιλέγεται από το κεντρικό μενού το “Help” και έπειτα “Updates and Extensions (Marketplace)”, και με τη βοήθεια της αναζήτησης επιλέγουμε το πακέτο που μας ενδιαφέρει, ενώ δίπλα αναφέρονται τα χαρακτηριστικά του πακέτου.



Εικόνα 43. Εγκατάσταση βιβλιοθηκών στο Rapidminer.

Υπάρχουν διαθέσιμα δωρεάν εγχειρίδια χρήσης του συστήματος Rapid Miner, στην ομώνυμη ιστοσελίδα αλλά και στο μενού του προγράμματος όπως απεικονίζεται στην αρχική σελίδα, ενώ παράλληλα η ιστοσελίδα του Rapid Miner περιλαμβάνει ένα πλούσιο υλικό για την εκμάθηση του συστήματος συμπεριλαμβανομένων σεμιναρίων και εισαγωγικών εργασιών.

A.2 Εγκατάσταση του συστήματος Weka και των πακέτων/βιβλιοθηκών του

Η λήψη και στη συνέχεια η εγκατάσταση του συστήματος WEKA μπορεί να πραγματοποιηθεί από το Waikato web site:

http://www.cs.waikato.ac.nz/ml/Weka/downloading.html

Επιπλέον υπάρχει διαθέσιμη μια έκδοση για προγραμματιστές όπου περιέχεται ο πηγαίος κώδικας, και δίνεται η δυνατότητα μετατροπής, προσθήκης χαρακτηριστικών/ αλγορίθμων του WEKA.

Η Εικόνα 54 απεικονίζει την σελίδα εγκατάστασης του συστήματος, επιλέγοντας το λειτουργικό σύστημα στο οποίο θα εγκατασταθεί το εν λόγω λογισμικό, δηλ: Linux, Mac OS ή Windows. Διαθέσιμη επιλογή εγκατάστασης του συστήματος είναι πάντα η τελευταία έκδοση, αλλά αν ο χρήστης το θελήσει μπορεί να ανατρέξει στις παλιές εκδόσεις κάνοντας επιλογή αυτών από το Sourceforge website.

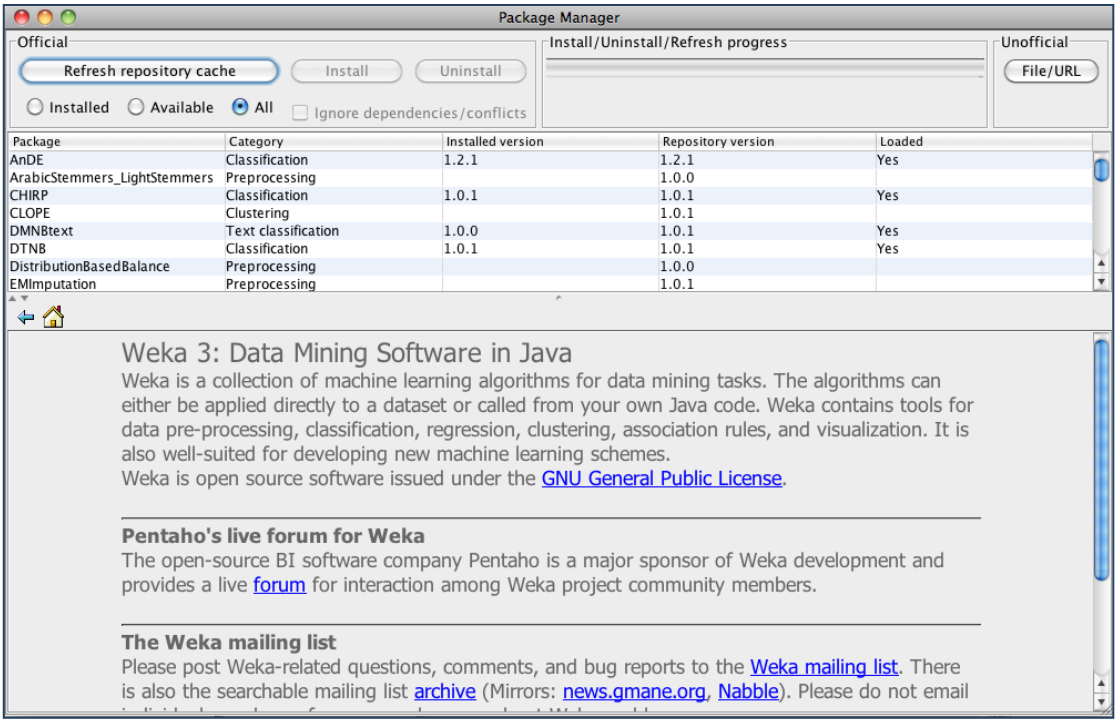
Υπάρχουν διαθέσιμα δωρεάν εγχειρίδια χρήσης του συστήματος Weka στην ομώνυμη ιστοσελίδα και επιλέγοντας το πεδίο documentation, όπου περιλαμβάνει ένα πλούσιο υλικό για την εκμάθηση του Weka συμπεριλαμβανομένων των συχνών ερωτήσεων, σεμιναρίων και εισαγωγικών εργασιών, αλλά και στο φάκελο του προγράμματος κατά την εγκατάσταση του.

Στο Weka ο όρος «πακέτο» αναφέρεται στην έννοια της οργάνωσης των κλάσεων στη Java. Πιο συγκεκριμένα, το Weka έχει την έννοια ενός πακέτου ως μια δέσμης πρόσθετων λειτουργιών, διαφορετικών από εκείνες που παρέχονται στο κύριο αρχείο Weka.jar . Ένα πακέτο αποτελείται από διάφορα .jar αρχεία, εγχειρίδια, μετα-δεδομένα, και ενδεχομένως, τον πηγαίο κώδικα. Έτσι λοιπόν, το κεντρικό σύστημα Weka επιτρέπει στους χρήστες να εγκαταστήσουν ακριβώς ό, τι χρειάζονται ή τους ενδιαφέρει. Επίσης, παρέχεται ένας απλός μηχανισμός για τους χρήστες που συμβάλλουν στην ανάπτυξη του Weka. Υπάρχουν μια σειρά από πακέτα που διατίθενται για το Weka που προσθέτουν συστήματα μάθησης ή επεκτείνουν τη λειτουργικότητα του πυρήνα του συστήματος με κάποιο τρόπο. Πολλές από αυτές προσφέρονται από την ομάδα του Weka και άλλες από την εκτεταμένη κοινότητα του Weka.

Μια εκτεταμένη λίστα πακέτων υποστηριζόμενων από το Weka, περιέχεται στη σελίδα:

http://Weka.sourceforge.net/packageMetaData

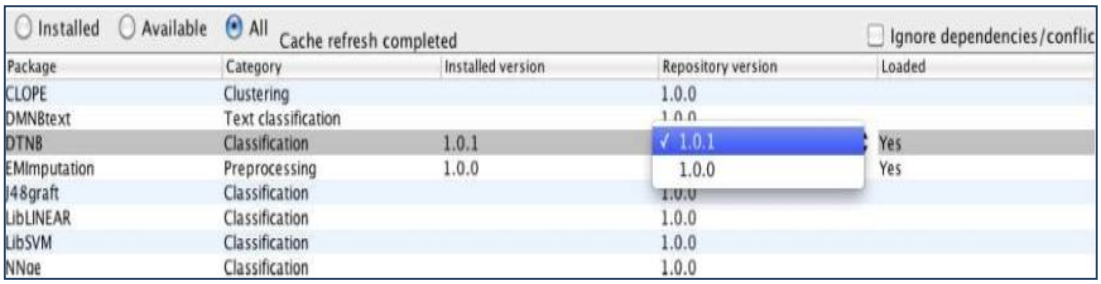
Το Weka περιλαμβάνει μια υπηρεσία για τη διαχείριση των πακέτων και έναν μηχανισμό για να τα φορτώνει δυναμικά κατά το χρόνο εκτέλεσης. Έτσι λοιπόν απαντάται μια γραφική διεπαφή για το σύστημα διαχείρισης πακέτων του Weka, η οποία είναι διαθέσιμη από το μενού Εργαλεία στο GUIChooser. Επιπλέον, είναι διαθέσιμη η δυνατότητα να εγκατασταθούν και να απεγκατασταθούν πολλαπλά πακέτα απλά επιλέγοντας τα.



Εικόνα 44. Υπηρεσία διαχείρισης πακέτων.

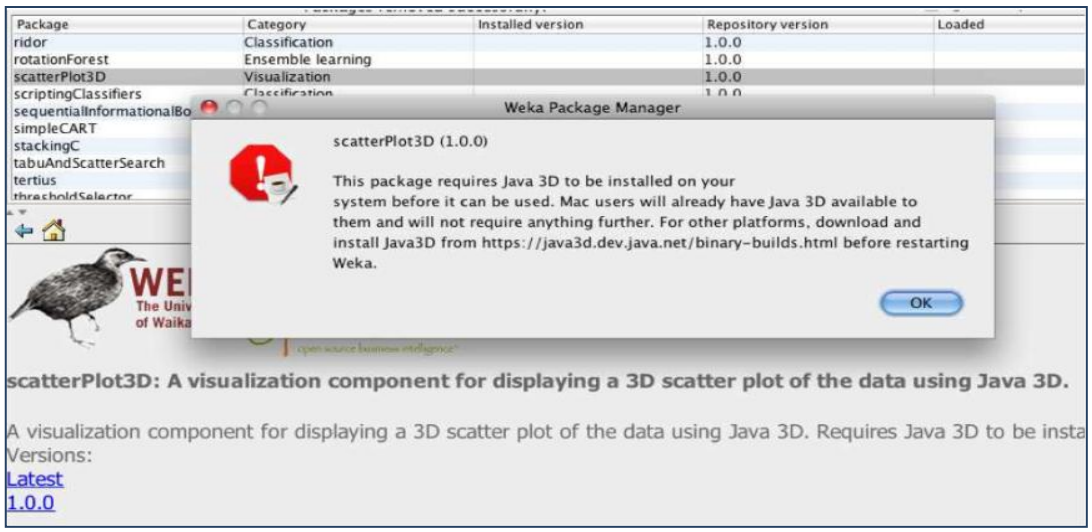
Το παράθυρο του διαχειριστή πακέτων είναι χωρισμένο οριζόντια σε δύο μέρη: α) στην κορυφή είναι η λίστα των πακέτων και β) το κάτω μέρος είναι ένα μίνι πρόγραμμα περιήγησης που μπορεί να χρησιμοποιηθεί για να εμφανίσετε πληροφορίες σχετικά με το επιλεγμένο πακέτο.

Η λίστα πακέτων εμφανίζει το όνομα ενός πακέτου, την κατηγορία του, την τρέχουσα εγκατεστημένη έκδοση (εάν υπάρχει) και την πιο πρόσφατη έκδοση που διατίθεται μέσω του αποθετηρίου. Αν υπάρχουν πολλές εκδόσεις ενός πακέτου οι οποίες είναι διαθέσιμες, μπορούν να προσπελαστούν επιλέγοντας μία καταχώρηση στη στήλη "Repository version":



Εικόνα 45. Υπηρεσία διαχείρισης πακέτων.

Πολλαπλά πακέτα μπορούν να εγκατασταθούν επιλέγοντας ένα πακέτο και χρησιμοποιώντας την επιλογή προσθήκη. Ορισμένα πακέτα μπορεί να έχουν πρόσθετες πληροφορίες σχετικά με το πώς να ολοκληρώσετε την εγκατάσταση ή τις ειδικές οδηγίες που παίρνει εμφανίζεται όταν έχει εγκατασταθεί το πακέτο, όπως φαίνεται στο παρακάτω παράδειγμα :



Εικόνα 46. Διαδικασία επιλογής πακέτου προς εγκατάσταση από την υπηρεσία διαχείρισης πακέτων.

Οι πληροφορίες (μετα-δεδομένα) σχετικά με τα πακέτα αποθηκεύονται σε ένα web server που φιλοξενείται στο Sourceforge. Η εντολή-εγκατάσταση πακέτου επιτρέπει να εγκατασταθεί ένα πακέτο σε μία από τις τρεις θέσεις:

11. καθορίζοντας ένα όνομα ενός πακέτου θα εγκατασταθεί το πακέτο, χρησιμοποιώντας τις πληροφορίες των στοιχείων περιγραφής των μετα δεδομένων του πακέτου που είναι αποθηκευμένα στο διακομιστή.

12. παρέχεται μια διαδρομή σε ένα αρχείο zip που θα επιχειρήσει να εγκαταστήσει το εν λόγω αρχείο ως πακέτο του Weka.

13. παρέχεται ένα URL (αρχίζοντας με http://) για ένα πακέτο σε μορφή αρχείου zip στο διαδίκτυο όπου θα κατεβάσει και θα προσπαθήσει να εγκαταστήσει το αρχείο zip ως πακέτο Weka.

A.3 Εγκατάσταση του συστήματος R και των πακέτων/βιβλιοθηκών του

Η λήψη και στη συνέχεια η εγκατάσταση του συστήματος R, μπορεί να πραγματοποιηθεί από το CRAN web site:

http://cran.r-project.org

Η Εικόνα απεικονίζει την αρχική σελίδα της ιστοσελίδας, επιλέγοντας το λειτουργικό σύστημα στο οποίο θα εγκατασταθεί το εν λόγω λογισμικό, δηλ: Linux, Mac OS ή Windows, κάτω από την επικεφαλίδα "Download and install R".

Υπάρχουν διαθέσιμα δωρεάν εγχειρίδια χρήσης του συστήματος R στην ομώνυμη ιστοσελίδα αλλά και στο μενού βοήθειας του προγράμματος, ενώ παράλληλα το CRAN Web Site περιλαμβάνει ένα πλούσιο υλικό για την εκμάθηση του R συμπεριλαμβανομένων των συχνών ερωτήσεων, σεμιναρίων και εισαγωγικών εργασιών.

Ένα πακέτο στο σύστημα R είναι ένα σχετικό σύνολο δυνατοτήτων, λειτουργιών, σελίδων βοήθειας, και μερικές φορές δεδομένων που είναι ομαδοποιημένα μαζί. Με την αρχική λήψη του λογισμικού R, παράλληλα γίνεται και η εγκατάσταση ενός βασικού συνόλου πακέτων. Βέβαια υπάρχουν πολυάριθμα πακέτα, και ένα μεγάλο μέρος της ευελιξίας του R προέρχεται από το εκτεταμένο σύνολο των πακέτων που αναπτύσσονται από τους ίδιους τους χρήστες.

Μια εκτεταμένη λίστα πακέτων υποστηριζόμενων από το CRAN, περιέχεται στη σελίδα:

http://cran.r-project.org/web/packages/

Ενώ η λίστα των πακέτων που εμπεριέχονται κατά την εγκατάσταση του R, περιέχεται στη σελίδα:

http://www.r-project.org/

Επιπροσθέτως, ανάλογα με την εκάστοτε ανάλυση που θέλει να διεξαγάγει ο χρήστης, στην ιστοσελίδα CRAN βρίσκονται οι βιβλιοθήκες εξειδικευμένων λειτουργιών, καθώς και μια σειρά από εγχειρίδια βοήθειας και παρουσίασης αυτών. Αυτή τη στιγμή, τα διαθέσιμα πακέτα/βιβλιοθήκες του R αγγίζουν τον αριθμό των 5.052 πακέτων.

Available CRAN Packages By Date of Publication

Date	Package	Title
2013-12-26	cpk	Clinical Pharmacokinetics
2013-12-25	HHG	Heller-Heller-Gorfine Tests of Independence
2013-12-25	kmc	Kaplan - Meier estimator with constraints for right censored data – a recursive computational algorithm
2013-12-24	Cairo	R graphics device using cairo graphics library for creating high-quality bitmap (PNG, JPEG, TIFF), vector (PDF, SVG, PostScript) and display (X11 and Win32) output
2013-12-24	climdex.pcic	PCIC implementation of Climdex routines
2013-12-24	confreq	Configural Frequencies Analysis Using Loglinear Modeling
2013-12-24	evt0	Mean of order p, peaks over random threshold Hill and high quantile estimates
2013-12-24	grpreg	Regularization paths for regression models with grouped covariates
2013-12-24	HBSTM	Hierarchical Bayesian Space-Time models for Gaussian space-time data
2013-12-24	PCovR	Principal Covariates Regression
2013-12-24	RcmdrPlugin.sos	Efficiently search the R help pages
2013-12-24	rJava	Low-level R to Java interface
2013-12-24	rtiff	A tiff reader for R
2013-12-24	safeBinaryRegression	Safe Binary Regression
2013-12-24	stilt	Separable Gaussian Process Interpolation (Emulation)
2013-12-24	TSTutorial	Fitting and Predict Time Series Interactive Laboratory

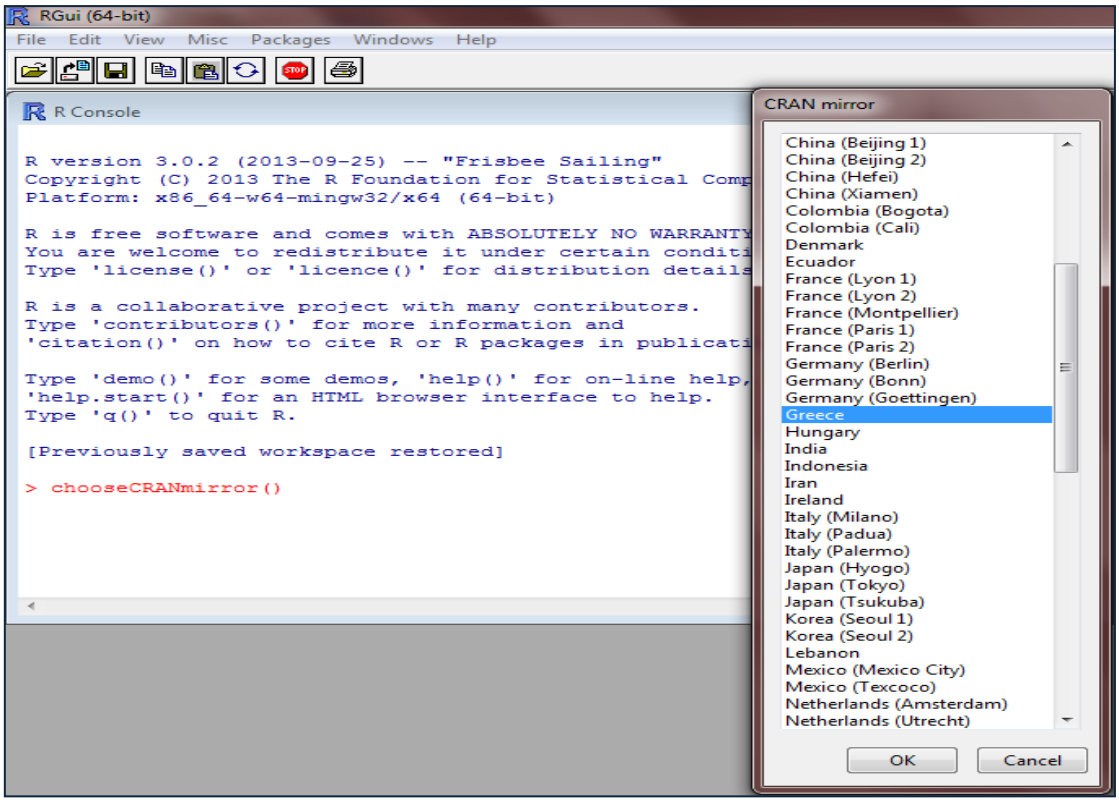
Εικόνα 47. Λίστα διαθέσιμων πακέτων/βιβλιοθηκών στην ιστοσελίδα http://cran.r-project.org.

Έπειτα από την επιλογή του κατάλληλου πακέτου, ο χρήστης προχωράει στην εγκατάσταση του. Υπάρχουν δύο κύριες επιλογές για την εγκατάσταση πακέτων στο R. Κατ 'αρχάς, ένα πακέτο μπορεί να εγκατασταθεί χρησιμοποιώντας την εντολή install.packages:

> Install.packages ("package name")

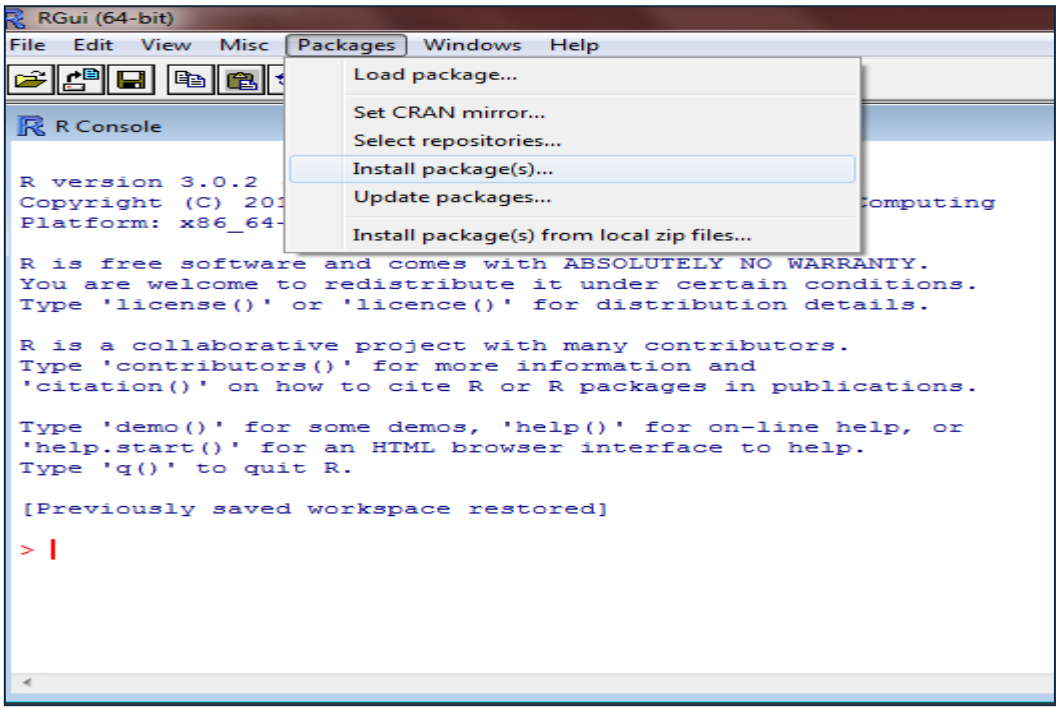
Κατά τη διαδικασία εγκατάστασης ενός πακέτου, ανοίγει ένα παράθυρο διαλόγου, όπου ζητείται να επιλεγεί ένας καθρέφτης. Οι καθρέφτες είναι τοποθεσίες σε όλο τον κόσμο που αποθηκεύουν

τα πακέτα, από τις οποίες μπορείτε κάποιος να τα προμηθευτεί και να τα εγκαταστήσει. Επιλέγοντας τη διαδικασία που απεικονίζεται στην Εικόνα, το R θα εγκαταστήσει αυτόματα τα πακέτα.



Εικόνα 48. Επιλογή CRAN mirror ανάλογα με γεωγραφική θέση του χρήστη.

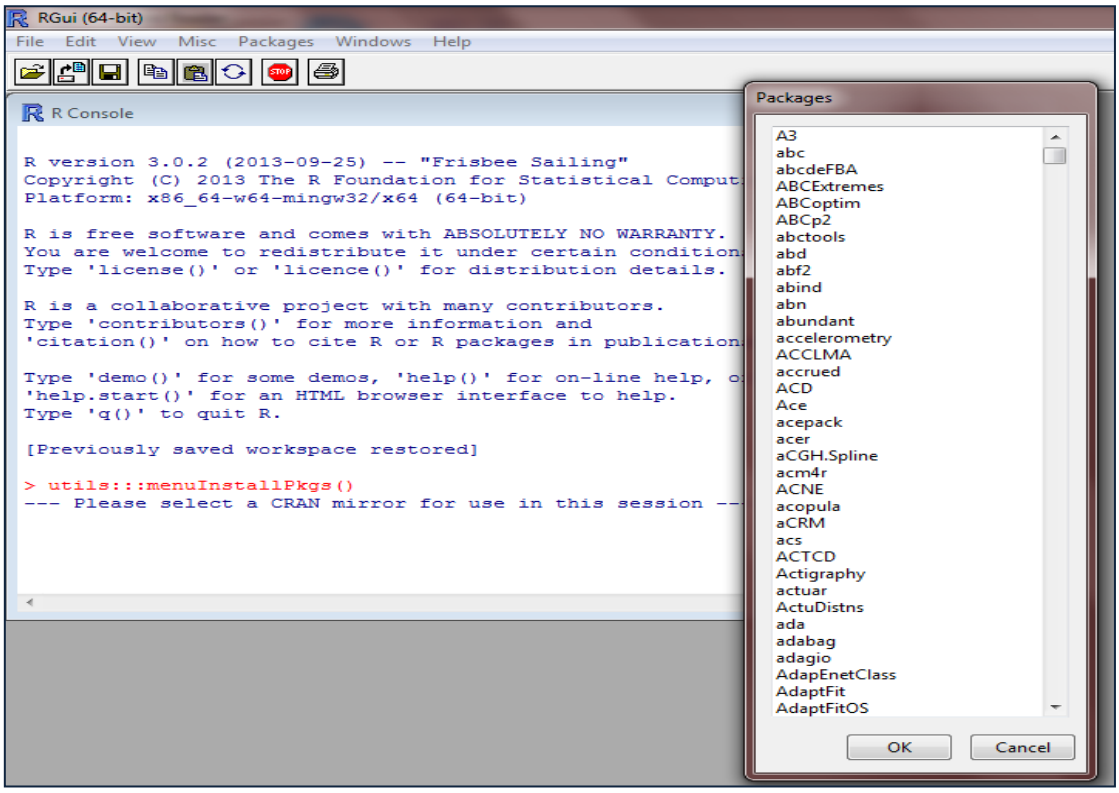
Εναλλακτικά, μπορεί ο χρήστης να επιλέξει τα «πακέτα» από το μενού του R, όπως απεικονίζεται στην παρακάτω Εικόνα, και έπειτα την επιλογή “Install Package(s)”.



Εικόνα 49. Εγκατάσταση πακέτων.

Όπως απεικονίζεται στην Εικόνα, ανοίγει ένα παράθυρο διαλόγου όπου θα πρέπει να επιλεγεί ο CRAN καθρέφτης για αυτή τη συνεδρία. Επιλέγεται πάντα ο καθρέφτης που είναι κοντά στο χρήστη γεωγραφικά ώστε να ελαχιστοποιήσει το χρόνο εγκατάστασης.

Στη συνέχεια, θα ανοίξει ένα δεύτερο παράθυρο συνομιλίας, όπου θα απεικονίζει αλφαβητικά τα διαθέσιμα πακέτα, ώστε να επιλεγούν από τον χρήστη τα επιθυμητά πακέτα προς εγκατάσταση. Με την επιλογή οποιουδήποτε πακέτου γίνεται η αυτόματη εγκατάσταση αυτού.



Εικόνα 50. Επιλογή πακέτου από λίστα πακέτων.

Εάν εγκατασταθεί κάποιο πακέτο, δεν χρειάζεται επανεγκατάσταση. Ωστόσο, θα χρειαστεί να φορτωθεί στη βιβλιοθήκη σε κάθε συνεδρία. Για να φορτωθεί ένα πακέτο χρησιμοποιείται η παρακάτω εντολή:

>library(package name)

Έπειτα από τις βασικές εγκαταστάσεις του συστήματος R, θα συνεχίσουμε με την ανάλυση της βασικής λειτουργίας και των βασικών εντολών του.

A.4 Εγκατάσταση του συστήματος Python και των πακέτων/βιβλιοθηκών του

Η λήψη και στη συνέχεια η εγκατάσταση του συστήματος Python, μπορεί να πραγματοποιηθεί από το Python web site επιλέγοντας το λειτουργικό σύστημα στο οποίο θα πραγματοποιηθεί η εγκατάσταση:

https://www.python.org

Υπάρχουν διαθέσιμα δωρεάν εγχειρίδια χρήσης του συστήματος Python στην ομώνυμη ιστοσελίδα και κατά την εγκατάσταση του συστήματος, συμπεριλαμβάνονται ως «πακέτα» και τα “Module Docs” αλλά και το “Python Manuals”, ενώ παράλληλα το Python.org Web Site περιλαμβάνει ένα πλούσιο υλικό για την εκμάθηση του συστήματος Python συμπεριλαμβανομένων των συχνών ερωτήσεων, σεμιναρίων, εισαγωγικών εργασιών και της του forum της κοινότητας των χρηστών και των προγραμματιστών.

Ένα πακέτο στο σύστημα Python είναι ένα σχετικό σύνολο δυνατοτήτων, λειτουργιών, σελίδων βοήθειας, και μερικές φορές δεδομένων που είναι ομαδοποιημένα μαζί. Με την αρχική λήψη του λογισμικού Python 3.4, παράλληλα γίνεται και η εγκατάσταση ενός βασικού συνόλου πακέτων. Βέβαια υπάρχουν πολυάριθμα πακέτα και ένα μεγάλο μέρος της ευελιξίας του Python προέρχεται από το εκτεταμένο σύνολο των πακέτων που αναπτύσσονται από τους ίδιους τους χρήστες.

Μια εκτεταμένη λίστα πακέτων υποστηριζόμενων από το Python, περιέχεται στη σελίδα:

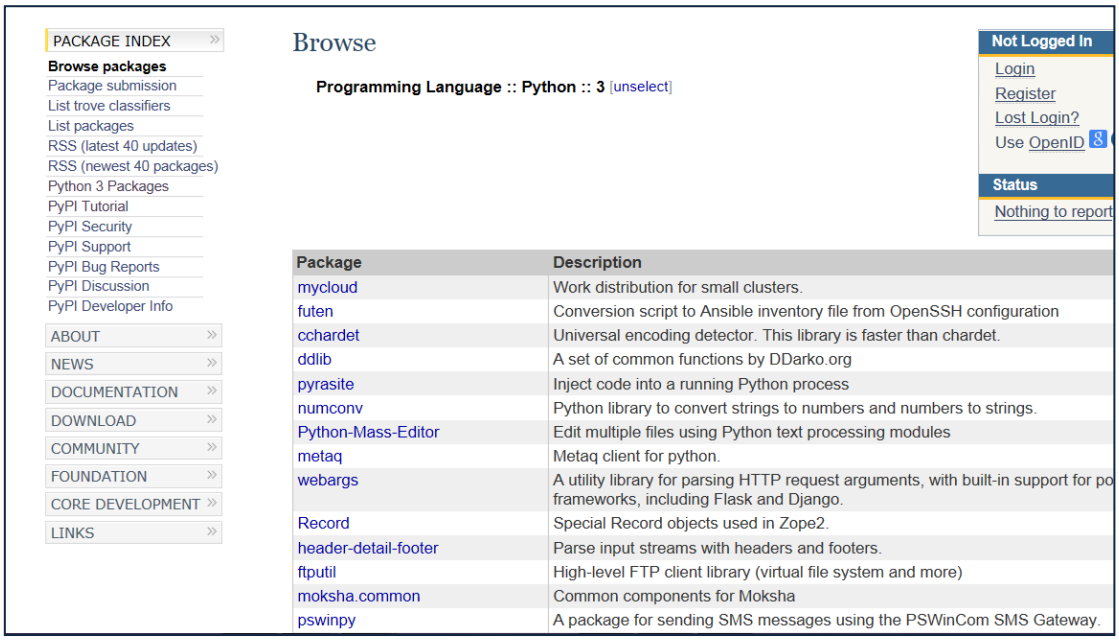
https://pypi.python.org/pypi

Επιπροσθέτως, ανάλογα με την εκάστοτε ανάλυση που θέλει να διεξάγει ο χρήστης, στην παραπάνω ιστοσελίδα βρίσκονται οι βιβλιοθήκες εξειδικευμένων λειτουργιών, καθώς και μια σειρά από εγχειρίδια βοήθειας και παρουσίασης αυτών στο δικό του domain στο pythonhosted.org. Αυτή τη στιγμή, τα διαθέσιμα πακέτα/βιβλιοθήκες του Python αγγίζουν τον αριθμό των 41.444 πακέτων, καλύπτοντας ένα μεγάλο αριθμό λειτουργιών, συμπεριλαμβάνοντας:

14. Γραφική διεπαφή χρήστη, το πλαίσιο των ιστοσελίδων, multimedia εφαρμογές, βάσεις δεδομένων, δικτύωση και επικοινωνία,

15. Δοκιμαστικά πλαίσια, τα εργαλεία τεκμηρίωσης, τη διαχείριση του συστήματος,

16. Επιστημονική πληροφορική, επεξεργασία κειμένου, επεξεργασία εικόνας.



Εικόνα 51. Λίστα διαθέσιμων πακέτων/βιβλιοθηκών στην ιστοσελίδα http://pypi.python.org/pypi.

Εκτός από την πρότυπη βιβλιοθήκη, υπάρχει μια αυξανόμενη συλλογή από αρκετές χιλιάδες συστατικά, από τα επιμέρους προγράμματα και τις ενότητες των πακέτων και ένα ολόκληρο πλαίσιο ανάπτυξης εφαρμογών, που διατίθεται από το ευρετήριο πακέτων της Python .Έπειτα από την επιλογή του κατάλληλου πακέτου, ο χρήστης προχωράει στην εγκατάσταση του.

Κατ 'αρχάς, για να χρησιμοποιηθεί ένα πακέτο από τα περιεχόμενα της Pypi, είτε από την ιστοσελίδα:

http://www.pip-installer.org/en/latest/installing.html

Επιλέγοντας την εγκατάσταση του σημαίνει ότι είτε θα πραγματοποιηθεί η λήψη και αποσυμπίεση του “python setup.py” στο μεταγλωττιστή του Python, ενώ σε Mac OS ή Linux , μπορεί να χρειαστεί να εκτελεστεί ως “getpip.py sudo python”. Έτσι θα εγκατασταθεί ή θα αναβαθμιστεί το setuptools. Στη συνέχεια χρησιμοποιείται η εντολή “easy_install” ώστε να εγκατασταθεί το pip. Ένα πακέτο μπορεί να εγκατασταθεί χρησιμοποιώντας την εντολή:

> import package name

> package name download ()

Έπειτα από τις βασικές εγκαταστάσεις του συστήματος του Python, θα συνεχίσουμε με την ανάλυση της βασικής λειτουργίας και των βασικών εντολών του.