ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΚΑΙ
ΕΠΙΚΟΙΝΩΝΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΕΞΟΡΥΞΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΓΝΩΜΗΣ ΑΠΟ ΗΛΕΚΤΡΟΝΙΚΟΥΣ ΔΗΜΟΣΙΟΥΣ ΔΙΑΛΟΓΟΥΣ

Η Διπλωματική Εργασία
παρουσιάστηκε ενώπιον
του Διδακτικού Προσωπικού του
Πανεπιστημίου Αιγαίου

Σε Μερική Εκπλήρωση
των Απαιτήσεων για το Δίπλωμα του
Μηχανικού Πληροφοριακών και Επικοινωνιακών Συστημάτων

των φοιτητών
ΒΟΥΡΟΥ ΕΛΕΝΗ, Α.Μ.: 321/2005008
ΔΗΜΗΤΡΑΚΗΣ ΓΕΩΡΓΙΟΣ, Α.Μ.: 321/2005015

ΣΕΠΤΕΜΒΡΙΟΣ 2011

Η ΤΡΙΜΕΛΗΣ ΕΠΙΤΡΟΠΗ ΔΙΔΑΣΚΟΝΤΩΝ ΕΓΚΡΙΝΕΙ
ΤΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ ΤΩΝ ΦΟΙΤΗΤΩΝ
ΒΟΥΡΟΥ ΕΛΕΝΗ ΚΑΙ ΔΗΜΗΤΡΑΚΗ ΓΕΩΡΓΙΟ:

Ιωάννης Χαραλαμπίδης, Επιβλέπων, Επίκουρος Καθηγητής
Τμήμα Μηχανικών Πληροφοριακών και
Επικοινωνιακών Συστημάτων

Ευριπίδης Λουκής , Μόνιμος Επίκουρος Καθηγητής
Τμήμα Μηχανικών Πληροφοριακών και
Επικοινωνιακών Συστημάτων

Εμμανουήλ Μαραγκουδάκης, Λέκτορας
Τμήμα Μηχανικών Πληροφοριακών και
Επικοινωνιακών Συστημάτων

ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ
ΣΕΠΤΕΜΒΡΙΟΣ 2011

ΠΕΡΙΛΗΨΗ

Σήμερα, η ολοένα συνεχής ανάπτυξη του Παγκόσμιου Ιστού, ως μέσο που εκθέτει τη δυνατότητα να προσελκύσει και να διατηρήσει την ανάμιξη της κοινωνίας σε συνδυασμό με την ανάγκη για την δημιουργία μιας πολιτικής που στηρίζεται στον πολίτη και την κοινωνία, έχει ανάγκη για νέα εργαλεία τα οποία θα έχουν την ικανότητα να αναλύουν τον ρόλο της κοινωνίας και να προβλέπουν το πιθανό αντίκτυπο στις εκάστοτε πολιτικές. Οι κυβερνήσεις σε παγκόσμια κλίμακα εισήγαγαν την έννοια της ηλεκτρονικής διακυβέρνησης , η οποία θεωρείται σήμερα πολύ σημαντικό εργαλείο για μια ευρεία διοικητική μεταρρύθμιση όπου οι νέες τεχνολογίες διαδραματίζουν ένα νέο ρόλο. Επίσης η ανάγκη να γνωρίζουμε την υποκειμενική άποψη των άλλων σχετικά με κάθε είδους ζητήματος, υπήρξε πάντοτε υψίστης σημασίας για τον άνθρωπο και ως επακόλουθο η ηλεκτρονική διακυβέρνηση στηρίζεται πάνω σε αυτή την ανάγκη. Πολύ καθοριστικό ρόλο έχει και η αλματώδης ανάπτυξη του Παγκόσμιου Ιστού, οπού πλέον οι χρήστες του έχουν την δυνατότητα να διατυπώνουν ελεύθερα κάθε είδους γνώμη και άποψη. Πρόσφορο έδαφος βρίσκει το πρόγραμμα Padgets στόχος του οποίου είναι να σχεδιάσει, να αναπτύξει και να επεκτείνει ένα σύνολο πρωτότυπων εργαλείων, που θα επιτρέψει στους φορείς χάραξης πολιτικής να δημιουργήσουν γραφικά Web εφαρμογές οι οποίες θα επεκταθούν σε περιβάλλον βασικής γνώσης Web 2.0.

Η ευρεία χρήση και η ταχύτατη εξάπλωση του Παγκόσμιου Ιστού είχε σαν αποτέλεσμα τη μετατροπή του σε μια τεράστια αποθήκη πληροφοριών με δισεκατομμύρια σελίδες και περισσότερους από 2,110 εκατομμύρια χρήστες παγκοσμίως. Σήμερα θεωρείται ένα από τα πιο σημαντικά μέσα συλλογής, διαμοίρασης και διάδοσης πληροφοριών και υπηρεσιών. Την ίδια στιγμή ο όγκος των διαθέσιμων πληροφοριών αποτελεί μια τεράστια πρόκληση για τους ερευνητές για την καλύτερη αξιοποίηση αυτών των πληροφοριών. Η εξόρυξη κειμένου είναι ένας νέος ερευνητικός τομέας που προσπαθεί να επιλύσει το πρόβλημα της υπερφόρτωσης πληροφοριών με τη χρησιμοποίηση διάφορων τεχνικών. Ο κύριος στόχος της εξόρυξης κειμένου είναι να βοηθήσει τους χρήστες να εξαγάγουν πληροφορίες από μεγάλους κειμενικούς πόρους. Δύο από τους σημαντικότερους στόχους είναι η κατηγοριοποίηση και η ομαδοποίηση εγγράφων. Μέσα σε αυτά τα πλαίσια έχουν αναπτυχθεί διάφοροι αλγόριθμοι κατηγοριοποίησης αλλά και έτοιμα λογισμικά με σκοπό να επιτύχουν τον παραπάνω στόχο.

Ο εν λόγω κλάδος έχει απασχολήσει τις εταιρίες σε παγκόσμια κλίμακα που επιθυμούν να γνωρίζουν την άποψη των καταναλωτών τους σχετικά με τις υπηρεσίες ή τα προϊόντα που παρέχουν. Προς αυτή την κατεύθυνση κλίνουν οι εκάστοτε κυβερνήσεις, χρησιμοποιώντας την εξόρυξη γνώμης από δημόσιους ηλεκτρονικούς διαλόγους, με στόχο ο πολίτης να είναι πλέον ένα ενεργό μέλος για την βελτιστοποίηση της πολιτική τους.

ABSTRACT

Nowadays, the continuing development of the Web, as a tool that exposes the ability to attract and retain the involvement of society combined with the need to create a policy based on the citizen and society, needs for new tools that will have the ability to analyze the role of society and to predict the impact on current policies. Governments, worldwide have introduced the concept of e-gov, which is now considered as a very important tool for a broad administrative reform where new technologies play a new role. Also the need to know the subjective opinion of others on any issue has always been very important to us and as a result the e-gov field is based on this need. Decisive role has the exponential growth of the Web.

More and more users have the ability to freely express any opinion and viewpoint. Fertile ground finds Padgets program which objective is to design, develop and expand a set of original tools, that will allow policy makers to create Web graphics applications which will be extended to environmental knowledge based on Web 2.0. The widespread use and rapid deployment of the Web has resulted in its conversion into a vast repository of information with billions of pages and more than 2.110 million users worldwide. Today, Web is one of the most important tools of collecting, sharing and dissemination information and services. At the same time the amount of available information is a huge challenge for researchers to make a better use of this information. Text mining is a new research area that tries to solve the problem of overload information by using various techniques. The main purpose of text mining is to help users extract information from large textual resources. Two of the major goals is to classify and group documents.

Within these frameworks have developed various algorithms for classification but also there are plenty of software in order to achieve the above goal. Text mining has employed companies worldwide who want to know the opinion of consumers about services or products they provide. To this end gravitate the respective governments using data mining from public electronic dialogues to make citizen an active member for the optimization of policy.

ΕΥΧΑΡΙΣΤΙΕΣ – ΑΦΙΕΡΩΣΕΙΣ

Η παρούσα διπλωματική εργασία εκπονήθηκε από τους φοιτητές Βούρου Ελένη και Δημητράκη Γεώργιο του Τμήματος Μηχανικών Πληροφοριακών και Επικοινωνιακών Συστημάτων του Πανεπιστημίου Αιγαίου κατά το ακαδημαϊκό έτος 2010-2011 υπό την επίβλεψη του επίκουρου καθηγητή Ιωάννη Χαραλαμπίδη.

Στον κύριο Χαραλαμπίδη οφείλουμε τις θερμές μας ευχαριστίες για την καθοδήγηση και την υποστήριξη του καθώς και για την υπομονή που έδειξε καθ'όλη τη διάρκεια διεκπεραίωσης της παρούσας διπλωματικής.

Ιδιαίτερες ευχαριστίες θέλουμε να δώσουμε στον κύριο Εμμανουήλ Μαραγκουδάκη, Λέκτορα του τμήματος μας για την πολύτιμη βοήθεια που μας έδωσε. Χωρίς τη βοήθεια του η ολοκλήρωση αυτής της μελέτης θα ήταν αδύνατη. Επίσης για το αμείωτο ενδιαφέρον και τη συμπαράσταση του σε όλα τα στάδια της εργασίας μας.

Επιπλέον πολλές ευχαριστίες οφείλουμε να δώσουμε στον συνάδελφο Γεώργιο Λουλουδάκη, για την σημαντική συνεισφορά του στο προγραμματιστικό στάδιο της διπλωματικής μας και για την ηθική υποστήριξη και συνεχής συμπαράσταση που έδειξε όλο αυτό τον καιρό.

Περιεχόμενα

Κεφάλαιο 1
Εισαγωγικά στοιχεία

1.1 *Αντικείμενο / Σκοπός*

Σε μια εποχή που η τεχνολογία αναπτύσσεται εκθετικά και χωρίς τα όρια της εξέλιξης της να διακρίνονται στον ορίζοντα, ολοένα και αυξάνονται οι προοπτικές για τη δημιουργία καινούργιων πρωτότυπων εφαρμογών. Σε αυτή την κατεύθυνση συμβάλει η ραγδαία ανάπτυξη και εξέλιξη του Παγκόσμιου Ιστού κυρίως την τελευταία δεκαετία.

Σίγουρα κανείς δεν θα μπορούσε να φανταστεί πριν από μερικά χρόνια ότι με ένα απλό "κλικ" θα μπορούσε να εξερευνήσει, ακόμα και να διαχειριστεί ,ένα ατελείωτο πλούτο πληροφορίας. Μέχρι το 2004 το διαδίκτυο βασιζόταν στην τεχνολογία του WEB 1.0 το οποίο περιελάμβανε στατικές σελίδες δηλαδή την απλή διακίνηση της πληροφορίας σε παγκόσμιο επίπεδο. Η τεχνολογία αυτή έθετε ένα "αόρατο τείχος" μεταξύ χρήστη και διαδικτύου. Η επικοινωνία ήταν μονόδρομη καθώς ο χρήστης είχε το δικαίωμα μόνο ανάγνωσης και όχι εγγραφής. Με την εξέλιξη του διαδικτύου σε WEB 2.0 αυτό το τείχος "γκρεμίζεται" και δίνεται η δυνατότητα στον χρήστη να συμμετάσχει ενεργά σχολιάζοντας ή εκφέροντας γνώμη πάνω σε οποιαδήποτε θέμα τον αφορούν.

Πιο συγκεκριμένα και για να προσεγγίσουμε το αντικείμενο της παρούσας εργασίας, έρχονται στο προσκήνιο οι ηλεκτρονικές διαβουλεύσεις. Είναι υψίστης σημασίας για τις κυβερνήσεις να "ακούσουν" την γνώμη των πολιτών τους και να αφουγκραστούν την γενικότερη στάση τους απέναντι στο εν λόγω εγχείρημα, κάτι που μπορεί να γίνει άμεσα λόγω της πληθώρας διατυπωθέντων απόψεων στο διαδίκτυο με έμφαση στα κοινωνικά μέσα. Η πρόκληση και ο προβληματισμός πηγάζει από το μέγεθος της διαθέσιμης αυτής πληροφορίας που αναπόφευκτα οδηγεί στην ανάγκη για δημιουργία σύγχρονων λογισμικών που θα έχουν την δυνατότητα να ταξινομούν το εν λόγω υλικό αυτόματα, καθώς ο χρόνος επεξεργασίας κρίνεται πολύτιμος για να σπαταληθεί σε μία εποχή που χαρακτηρίζεται από ανάγκες που αλλάζουν καθημερινά και ανεξέλεγκτα.

Μέσα σε αυτή την φιλοσοφία βρίσκει πρόσφορο έδαφος ο σκοπός της παρούσας διπλωματικής εργασίας καθώς αποσκοπήθηκε η συγκέντρωση απόψεων των Ελλήνων πολιτών σχετικά με το εγχείρημα της ηλεκτρονικής διακυβέρνησης, εκμεταλλεύοντας ουσιαστικά τα πλεονεκτήματα που παρέχει ο WEB 2.0, και μετέπειτα η δημιουργία μιας εφαρμογής που θα είναι σε θέση θεωρητικά να εντοπίζει το συναίσθημα και την υποκειμενική στάση του γράφοντος.

1.2 Στάδια υλοποίησης

Η περίοδος εκπόνησης της παρούσας διπλωματικής εργασίας πραγματοποιήθηκε από το Φεβρουάριο του 2011 και έλαβε τέλος το Σεπτέμβρη του 2011. Για την κατανόησή της θα αναφερθούν τα στάδια υλοποίησης της.

Στάδιο 1:

Επιλογή κατάλληλης ιστοσελίδας που δημοσιεύονται ηλεκτρονικές διαβουλεύσεις. Στην περίπτωσή μας είναι το www.opengov.gr

Στάδιο 2:

Μελέτη και αναζήτηση ηλεκτρονικής διαβούλευσης στην παραπάνω ιστοσελίδα, με κριτήριο το εύρος σχολίων αλλά και το ποσοστό ενδιαφέροντος που παρουσιάζεται για αυτή.

Στάδιο 3:

Επικεντρωθήκαμε στη διαβούλευση της "κάρτας του πολίτη" στο www.opengov.gr γιατί η δημοτικότητα του είναι μεγάλη και περιλαμβάνει τεράστιο όγκο σχολίων συγκριτικά με άλλες.

Στάδιο 4:

Καταγραφή και αποθήκευση των σχολίων σε διαφορετικά text files για περεταίρω μελέτη.

Στάδιο 5:

Αναζήτηση και επιλογή κατάλληλου λογισμικού που παρέχει τη δυνατότητα δημιουργίας εφαρμογής opinion mining.

Στάδιο 6:

Κατηγοριοποίηση μέρους των σχολίων με βάση το ύφος/συναίσθημα σε δύο κλάσεις: θετικά και αρνητικά σχόλια.

Στάδιο 7:

Με την βοήθεια του κατάλληλου λογισμικού αναπτύξαμε, υλοποιήσαμε και εκτελέσαμε το case μας.

Στάδιο 8:

Αξιολόγηση και μελέτη αποτελεσμάτων της εφαρμογής και χρήσιμα συμπεράσματα.

1.3 Γενική επισκόπηση

Η διπλωματική εργασία που υλοποιήσαμε με τίτλο «Εξόρυξη δεδομένων και γνώμης από ηλεκτρονικούς δημόσιους διαλόγους» περιλαμβάνει τα εξής παρακάτω κεφάλαια:

- *Opinion Mining*
- *Text Mning*
- *Αρχιτεκτονική της πλατφόρμας*
- *Συμπεράσματα & Προοπτικές*

Αναλυτικότερα η διάρθρωση της διπλωματικής εργασίας παρουσιάζεται παρακάτω:

Στο πρώτο κεφάλαιο γίνεται μια εισαγωγή στον όρο Εξόρυξη Δεδομένων/Γνώμης πάνω στον οποίο βασίζεται το μεγαλύτερο κομμάτι της διπλωματικής εργασίας. Κατόπιν αναφέρουμε το πεδίο μελέτης μας και τέλος τα κυριότερα πλεονεκτήματα που θα υπάρξουν, δυσκολίες που προέκυψαν αλλά και προοπτικές που παρουσιάστηκαν.

Στο δεύτερο κεφάλαιο « text mining» γίνεται μια εισαγωγή στον όρο « text mining» και παρουσιάζουμε τα κυριότερα εργαλεία με τα οποία μπορούμε να εφαρμόσουμε τεχνικές text mining. Στην συνέχεια κάνουμε μια εισαγωγή στο εργαλείο RapidMiner το οποίο θα χρησιμοποιήσουμε για την εκπόνηση της εργασίας μας.

Στο τρίτο κεφάλαιο γίνεται μια πλήρης αναφορά στο πρόγραμμα Padgets και την συσχέτισή του με το αντικείμενο της διπλωματικής μας. Στην συνέχεια γίνεται μια αναλυτική περιγραφή των επιμέρους φάσεων μελέτης και υλοποίησης της εργασίας. Παράλληλα η ανάπτυξη της εργασίας σε προγραμματιστικό επίπεδο με χρήση της γλώσσας Java και τέλος τα αποτελέσματα τα οποία εξήχθησαν.

Στο τέταρτο και τελευταίο κεφάλαιο παρουσιάζονται τα συμπεράσματα που εξάγαμε αλλά και σημεία τα οποία μας προβλημάτισαν. Τέλος αναφέρονται τυχόν προοπτικές που διαφαίνονται από την ενασχόληση μας πάνω στο εν λόγω θέμα.

Κεφάλαιο 2
Opinion Mining

2.1 *Εξόρυξη Γνώμης-Opinion Mining*

Κάθε φορά που κάποιος (άτομο, εταιρεία, κυβέρνηση κλπ) επιθυμεί να πάρει μια σημαντική απόφαση αναζητά να ακούσει την γενικότερη γνώμη των άλλων. Με την ραγδαία ανάπτυξη του Παγκόσμιου Ιστού και των διάφορων forums επικοινωνίας ο εκάστοτε ενδιαφερόμενος μπορεί να εξάγει συμπεράσματα που θα τον βοηθήσουν στην λήψη μιας απόφασης. Για παράδειγμα, για την αγορά ενός αντικειμένου μπορεί να αναζητήσει κανείς στα υπάρχοντα σχετικά forums την γενικότερη αντίληψη για το προϊόν αυτό.

Η εξόρυξη γνώμης έχει ως στόχο, από ένα έγγραφο είτε από διάφορα σχόλια που αναφέρονται σε ένα συγκεκριμένο ζήτημα, να εξάγει τα κύρια χαρακτηριστικά του αντικειμένου συζήτησης, καθώς επίσης και να διαπιστώσει αν το κείμενο ή αντίστοιχα τα σχόλια είναι θετικά, αρνητικά ή ουδέτερα.

Οι πληροφορίες που μπορούμε να συλλέξουμε από κείμενα μπορούν να ταξινομηθούν σε δύο κατηγορίες, τα γεγονότα και τις γνώμες. Τα γεγονότα είναι αντικειμενικές δηλώσεις σχετικά με κάποιο ζήτημα. Οι γνώμες είναι υποκειμενικές δηλώσεις που αντανακλούν στα συναισθήματα ή τις αντιλήψεις των ανθρώπων σχετικά με το ζήτημα. Ένα μεγάλο μέρος έρευνας έχει επικεντρωθεί στην εξόρυξη και την ανάκτηση πραγματικών πληροφοριών.

Σχετικά με την εξόρυξη γνώμης ,όμως, δεν είχε πραγματοποιηθεί ιδιαιτέρα μεγάλη πρόοδος μέχρι πρόσφατα. Ο κύριος λόγος ήταν γιατί τα διάφορα forums επικοινωνίας και οι ιστοσελίδες στις οποίες μπορεί κάποιος απλός χρήστης να διατυπώσει την άποψή του, δεν ήταν ευρύτερα ανεπτυγμένα και γνωστά. Πλέον κάποιος μπορεί να "ποστάρει" την γνώμη του για ένα ζήτημα σε αντίστοιχες ιστοσελίδες (blogs, forums κλπ) στις οποίες συλλέγονται μεγάλος αριθμός απόψεων. Τώρα εάν κάποιος επιθυμεί να αγοράσει ένα προϊόν δεν χρειάζεται να ανατρέξει στους φίλους και τους συγγενείς του, αλλά αρκεί να αναζητήσει στο διαδίκτυο την γνώμη χρηστών που έχουν αγοράσει ήδη το προϊόν. Το ίδιο ισχύει και για μια εταιρεία η οποία πλέον δεν χρειάζεται να διεξάγει έρευνες και να έχει συμβούλους για θέματα που αφορούν τις απόψεις των καταναλωτών όσον αφορά προϊόντα δικά της ή ανταγωνιστικής εταιρείας.

Ο όρος opinion mining εμφανίζεται για πρώτη φορά σε δημοσίευση των Kushal Dave, Steve Lawrence, David M. Pennock στα πλαίσια του συνεδρίου WWW (WWW conference) κατά το έτος 2003. Η δημοσίευση στο συγκεκριμένο συνέδριο μπορεί εν μέρει να εξηγήσει την δημοτικότητα του όρου opinion mining μεταξύ των κοινοτήτων που είναι προσανατολισμένες στην κατεύθυνση αναζήτησης στο διαδίκτυο (Web search) ή ανάκτησης πληροφορίας (information retrieval). Σύμφωνα με την εν λόγω δημοσίευση, η ιδανική εφαρμογή για εξόρυξη συναισθήματος «θα μπορούσε να επεξεργαστεί ένα σύνολο από δεδομένα αναζήτησης, δημιουργώντας μία λίστα των κύριων χαρακτηριστικών αυτών και συνοψίζοντας τις απόψεις που επικρατούν για κάθε ένα από αυτά τα χαρακτηριστικά σε θετικές, ουδέτερες και αρνητικές». Οι περισσότερες έρευνες που έχουν διεξαχθεί στο πεδίο του opinion mining έρχονται σε συμφωνία

με τον παραπάνω ορισμό, με έμφαση κυρίως στο κομμάτι της εξόρυξης και ανάλυσης κρίσεων πάνω σε διάφορες πλευρές των εκάστοτε δοθέντων αντικειμένων. Ωστόσο, ο εν λόγω όρος πρόσφατα έχει διερμηνευτεί καλύπτοντας μεγαλύτερο εύρος και περιέχοντας πολύ περισσότερους και ταυτόχρονα διαφορετικούς τύπους ανάλυσης των κειμένων προς αξιολόγηση.

Αναφέρουμε ότι το 2004 η Αμερικανική Εταιρεία Τεχνητής Νοημοσύνης (American Association for Artificial Intelligence) οργάνωσε συμπόσιο επί της συγκεκριμένης θεματικής περιοχής που έφερε τον τίτλο «Εξερευνώντας Επιρροή και Στάση σε Κείμενο» («Exploring Affect and Attitude in Text»). Έκτοτε, η ανάλυση γνώμης έχει αποτελέσει αντικείμενο συνεχούς ενδιαφέροντος σε πολλά συνέδρια στις Ηνωμένες Πολιτείες, σε αντίθεση με την Ευρώπη όπου το σχετικό ενδιαφέρον σημείωσε μια συγκριτική υστέρηση.

2.2 Εξόρυξη Δεδομένων

Είναι βέβαιο ότι ζούμε στην κοινωνία της πληροφορίας, όπου η μετατροπή των δεδομένων σε πληροφορία απαιτείται να οδηγεί στη μετατροπή της πληροφορίας σε γνώση. Η συνύπαρξη ετερόκλητων επιστημονικών πεδίων όπως της στατιστικής, της μηχανικής εκμάθησης, της θεωρίας της πληροφορίας και των υπολογιστικών διαδικασιών, έχει δημιουργήσει μια νέα επιστήμη με δυναμικά εργαλεία, η οποία καλείται «Εξόρυξη Δεδομένων».

Η σύγκλιση της προόδου υπολογιστικών συστημάτων και της εξέλιξης στην επικοινωνία έχει οδηγήσει στην δημιουργία μιας κοινωνίας ικανής να παρέχει διαρκώς νέες πληροφορίες. Το υλικό που συγκεντρώνεται καταγράφεται διαρκώς, με αποτέλεσμα τη δημιουργία τεράστιων βάσεων δεδομένων. Το ζήτημα λοιπόν που προκύπτει, είναι εάν μπορούμε να διαχειριστούμε αυτές τις βάσεις δεδομένων. Όλα αυτά τα θέματα προκάλεσαν το ενδιαφέρον και οδήγησαν στη διαδικασία της Εξόρυξης Δεδομένων (Data Mining). Πρόκειται για μία σειρά από τεχνικές που βασίζονται σε ανάπτυξη αλγορίθμων και είναι χρήσιμες σε πολλούς κλάδους όπως οι: οικονομία, βιοστατιστική, δημογραφία, μετεωρολογία και γεωλογία. Υπάρχουν αντικρουόμενες απόψεις γύρω από το ποιος θα μπορούσε να είναι ένας σαφής και περιεκτικός ορισμός για την Εξόρυξη Δεδομένων.

Ωστόσο, ο ορισμός των Hand et al. (2001) θεωρείται ο πιο αξιόλογος: «Εξόρυξη Δεδομένων είναι η ανάλυση – συνήθως τεράστιων – παρατηρούμενων συνόλων δεδομένων, έτσι ώστε να βρεθούν μη παρατηρηθείσες σχέσεις και να συνοψιστούν τα δεδομένα με καινοφανείς τρόπους οι οποίοι να είναι κατανοητοί και χρήσιμοι στον κάτοχο των δεδομένων».

Η δήλωση των σχέσεων και η σύνοψη των στοιχείων στην οποία αναφέρεται ο ορισμός αυτός, συχνά αναφέρεται ως μοντέλο ή πρότυπο. Βασικοί στόχοι της Εξόρυξης Δεδομένων είναι η περιγραφή και η πρόβλεψη. Δηλαδή, η αναγνώριση των προτύπων που επικρατούν σε ένα μεγάλο σύνολο δεδομένων και η δημιουργία προβλέψεων όσον αφορά τη μελλοντική αξία ή συμπεριφορά κάποιων μεταβλητών. Η αναγνώριση των προτύπων γίνεται μέσω γραμμικών εξισώσεων, κανόνων, διάκρισης σε συστάδες, απόδοσης γραφημάτων και δομών σε μορφή δέντρου, καθώς και επαναλαμβανόμενων προτύπων σε μορφή χρονοσειρών.

Ένα σημαντικό σημείο, στο οποίο πρέπει να σταθούμε, είναι ότι ο παραπάνω ορισμός αναφέρεται σε παρατηρούμενα δεδομένα και όχι σε εμπειρικά ή πειραματικά. Για το λόγο αυτό, η Εξόρυξη Δεδομένων αναφέρεται συχνά ως «δευτερογενής» ανάλυση δεδομένων, αφού στην ουσία ασχολείται με δεδομένα που έχουν ήδη συλλεχθεί.

2.3 *Ηλεκτρονικοί Δημόσιοι Διάλογοι*

Μέχρι πρόσφατα, ο παραδοσιακός τρόπος λήψης της γνώμης χρηστών, καταναλωτών ή πολιτών ήταν μέσω ερευνών και δημοσκοπήσεων πάνω σε ένα συγκεκριμένο ζήτημα. Σαφώς, τα προβλήματα που ανακύπτουν από μία τέτοιου είδους προσέγγιση σχετίζονται κυρίως με τη δαπάνη σημαντικού χρόνου καθώς και χρηματικού ποσού για την κατάστρωση των εν λόγω ερευνών/ δημοσκοπήσεων, τη διανομή τους και την τελική αξιολόγηση αυτών. Επιπλέον, οι παραπάνω μέθοδοι στηρίζονται κατά κόρον στην «καλή θέληση» των συμμετεχόντων, γεγονός που πολλές φορές καθιστά την αξιοπιστία των αποτελεσμάτων τους αμφίβολη. Ακόμη, θα μπορούσαμε να ισχυριστούμε ότι με την έλευση της άντλησης των ζητούμενων πληροφοριών με αυτοματοποιημένο τρόπο από διαδικτυακές πηγές η διεξαγωγή κάθε είδους ερευνών μέσω ερωτηματολογίων κρίνεται δυνητικά παρωχημένη.

Σημαντικό παράδειγμα που τεκμηριώνει τον παραπάνω ισχυρισμό είναι η ευρεία ανάπτυξη και διάδοση των ηλεκτρονικών δημόσιων διάλογοι, με κύριο σκοπό την παράθεση σκέψεων, παρουσίαση ιδεών και ανταλλαγή γνώσης, γεγονός που τα καθιστά μια από τις πιο πολύτιμες πηγές εξόρυξης γνώμης. Έτσι λοιπόν, τα διάφορα σχόλια στα άρθρα των προαναφερθέντων ηλεκτρονικών δημόσιων καταλόγων αποτελούν μία πλούσια πηγή πληροφοριών που αντικατοπτρίζει άμεσα τις απόψεις των χρηστών σε μία ευρεία γκάμα θεματικών περιοχών και η αυτοματοποιημένη εξόρυξη και μετέπειτα ταξινόμηση αυτών θα μπορούσε αναμφίβολα να αντικαταστήσει τις πατροπαράδοτες έρευνες και δημοσκοπήσεις, ξεπερνώντας μάλιστα αυτές σε αξιοπιστία.

Στα πλαίσια της διαφάνειας και της συμμετοχικότητας των πολιτών στην λήψη αποφάσεων από την Κυβέρνηση, οργανώθηκαν οι Δημόσιες Ηλεκτρονικές Διαβουλεύσεις που

αφορούν ζητήματα σχετικά με μείζων κοινωνικά ζητήματα. Πιο συγκεκριμένα οι ηλεκτρονικές διαβουλεύσεις είναι μέρος της ηλεκτρονικής διακυβέρνησης (που αναπτύσσεται διεξοδικά στην ενότητα που ακολουθεί) και θεωρείται δημόσιος ηλεκτρονικός κατάλογος. Στο παρών έγγραφο ασχολούμαστε αποκλειστικά με της διαβουλεύσεις του opengov.

2.4 Ηλεκτρονική Διακυβέρνηση

Η «Ηλεκτρονική Διακυβέρνηση» στοχεύει στη δημιουργία πυρήνα πληροφοριακού υλικού σε ηλεκτρονική μορφή, μέσω του οποίου οι χρήστες θα μπορούν να έχουν σφαιρική ενημέρωση σχετικά με τα τεκταινόμενα και τις εξελίξεις σε θέματα Ηλεκτρονικής Διακυβέρνησης και Ηλεκτρονικής Δημοκρατίας (E-Government και E-Democracy) τόσο σε εθνικό όσο και σε διεθνές επίπεδο. Προδιαγράφει τις προϋποθέσεις για την υλοποίηση ενός πλαισίου για την παροχή ηλεκτρονικών υπηρεσιών για πολίτες (G2C), επιχειρήσεις και τα άλλα Νομικά Πρόσωπα (G2B) και φορείς της Δημόσιας Διοίκησης (G2G).

Πιο συγκεκριμένα η ηλεκτρονική διακυβέρνηση:

- Θέτει το θεσμικό νομικό υπόβαθρο για την παροχή ηλεκτρονικών υπηρεσιών από τη δημόσια διοίκηση προς φυσικά και νομικά πρόσωπα.
- Δημιουργεί έναν συνεκτικό ιστό που ενοποιεί παλαιότερες μη αποδοτικές και κατακερματισμένες προσεγγίσεις σε θέματα ηλεκτρονικής διακυβέρνησης.
- Συνιστά μια ενιαία δομημένη μεθοδολογία για την υλοποίηση και αξιοποίηση ηλεκτρονικών υπηρεσιών στους φορείς της δημόσιας διοίκησης.
- Δίνει έμφαση στην ενεργό συμμετοχή των στελεχών του δημοσίου.
- Η οικονομική ωφέλεια εφαρμογής του εκτιμάται στα 4δισεκατομμύρια ευρώ.

Οι βασικοί στόχοι της εφαρμογής αυτής είναι:

- Η άμεση εξυπηρέτηση του πολίτη μέσα από τη χρήση ηλεκτρονικών υπηρεσιών και με την καθιέρωση της ηλεκτρονικής συναλλαγής σε κάθε δημόσιο φορέα.
- Η πλήρη αξιοποίηση των Τεχνολογιών Πληροφορικής και Επικοινωνιών προκειμένου να περιοριστεί δραστικά η γραφειοκρατία.
- Η μείωση εμφάνισης φαινομένων διαφθοράς και η εδραίωση σχέσης εμπιστοσύνης ανάμεσα σε πολίτες, επιχειρήσεις και φορείς του δημόσιου φορέα.
- Η Δημιουργία προϋποθέσεων Ανάπτυξης
- Η Διασφάλιση της ετοιμότητας της Δημόσιας Διοίκησης να υλοποιήσει τους Στόχους του Νόμου της Ηλεκτρονικής Διακυβέρνησης.

- Η Βελτίωση των συνθηκών εργασίας των εργαζομένων με ταυτόχρονη αύξηση της αποδοτικότητάς τους και η επίτευξη του τρίπτυχου ευελιξία-ταχύτητα-ποιότητα με ασφάλεια στην εσωτερική επικοινωνία και λειτουργία των φορέων.

2.5 Συνεισφορά/Πλεονεκτήματα

Τα πλεονεκτήματα που προκύπτουν από την εφαρμογή opinion mining με συνδυασμό ηλεκτρονικών δημόσιων διαλόγων (στην περίπτωσή μας ηλεκτρονικές διαβουλεύσεις) ποικίλουν. Πρώτον, τα άτομα που μοιράζονται τις απόψεις τους συνήθως τείνουν να έχουν περισσότερο παγιωμένες γνώμες από το μέσο όρο, επομένως τέτοιου είδους απόψεις κρίνονται πολύτιμες. Δεύτερων, οι απόψεις των συμμετεχόντων στις ηλεκτρονικές διαβουλεύσεις εξάγονται σε πραγματικό χρόνο και άμεσα, επιτρέποντας έτσι την καλύτερη αξιοποίησή τους. Τρίτων αυτοί που επωφελούνται από μία αυτοματοποιημένη εξόρυξη γνώμης είναι αφενός ένας αναλυτής αφετέρου και διάφοροι οργανισμοί.

Πιο συγκεκριμένα ένας αναλυτής θα επιθυμούσε να έχει στη διάθεσή του μία ανακεφαλαίωση των αποτυπωμένων απόψεων παρά να διαβάζει πολυάριθμα σχόλια και συζητήσεις χρηστών. Επίσης οι διάφοροι οργανισμοί, διατηρούν μία συνεχή εικόνα του πώς οι χρήστες, καταναλωτές ή πολίτες αισθάνονται απέναντι στις υπηρεσίες τους, αποσπώντας συγκεκριμένες πληροφορίες που βρίσκονται «κρυμμένες» μέσα σε ένα κείμενο όπως παράπονα, προβλήματα και γενικότερες παρατηρήσεις. Στην περίπτωσή μας ως αναλυτές θεωρούνται οι πολίτες που παίρνουν μέρος στις ηλεκτρονικές διαβουλεύσεις και ως οργανισμοί θεωρούμε την ίδια την κυβέρνηση η οποία θα συλλέξει όλες τις πληροφορίες και θα δράσει, λαμβάνοντας υπόψη την γνώμη και άποψη των πολιτών της.

2.6 Δυσκολίες

Σύμφωνα με τον ορισμό της Εξόρυξης Γνώμης που διατυπώθηκε στην ενότητα 2.1 της παρούσας εργασίας, η επεξεργασία δεδομένων πραγματοποιείται συνοψίζοντας τις απόψεις που επικρατούν σε μια από τις παρακάτω κλάσεις : θετικές, ουδέτερες και αρνητικές. Αυτή η κατηγοριοποίηση καθιστά την Εξόρυξη Γνώμης ιδιαίτερα περίπλοκη. Όταν λοιπόν ένα άρθρο ή μία κριτική αντίστοιχα γίνονται αντικείμενο συζήτησης η κατηγοριοποίηση άποψης συνολικά μπορεί να γίνει πολύ δύσκολη. Αυτό ισχύει κυρίως όταν πρόκειται για πολιτικές συζητήσεις, όπου οι χρήστες-σχολιαστές προβαίνουν συχνά σε συγκρίσεις μεταξύ προσώπων, πολιτικών τακτικών ή γεγονότων. Έχει παρατηρηθεί ότι τα ποσοστά (περίπου 70%), όσων αφορά πολιτικών συζητήσεων στους αλγόριθμους ταξινόμησης (classification), είναι χαμηλά σε σύγκριση με ποσοστά (αγγίζουν 90%) συζητήσεων-κριτικών για προϊόντα

Οι ιδιαιτερότητες του συγκεκριμένου προβλήματος είναι:

- Περίπλοκοι εκφραστικοί τρόποι (ειρωνεία, ιδιωματισμοί, μεταφορές)
- Διαφορετική σημασιολογική απόχρωση μιας λέξης ανάλογα με τα συμφραζόμενα
- Αντιθετικό σχήμα

2.7 Εξέλιξη

Η εξόρυξη δεδομένων μέσα από τα οποία γίνεται εφικτή και εφαρμόσιμη η εξόρυξη γνώμης βρίσκεται σε πρώιμο στάδιο τόσο σε εγχώριο επίπεδο όσο(ίσως λίγο καλύτερα)και σε παγκόσμια κλίμακα. Αποτελεί μια πολύ πρόσφατη τεχνική που όπως διαφαίνεται θα εφαρμοστεί κατά κόρον σε εφαρμογές Web 2.0 στο άμεσο μέλλον. Αυτό όμως συνεπάγεται ότι υπάρχουν πολλά περιθώρια εξέλιξης έτσι ώστε να επιτευχθεί ακόμα καλύτερη αποδοτικότητα και εγκυρότητα των αποτελεσμάτων που εξάγουν τέτοιου είδους τεχνικές.

Συγκεκριμένα, ένα από τα σημαντικότερα θέματα για να πραγματοποιηθεί καλύτερη αποδοτικότητα του opinion mining αποτελεί η παρακίνηση των πολιτών να συμμετέχουν όλο και περισσότερο σε ηλεκτρονικές διαβουλεύσεις και να εκφράζουν τη γνώμη τους σε οποιοδήποτε θέμα τους αφορά. Εξίσου σημαντικό θέμα είναι να κατασκευαστούν αξιόπιστα ‘εκπαιδευμένα’ μοντέλα τα οποία θα είναι δυνατό και να επαναχρησιμοποιηθούν και να εφαρμοστούν άμεσα σε πιθανή είσοδο νέων δεδομένων προς μελέτη.

Εν κατακλείδι, θα ήταν εξαιρετικά ενδιαφέρον να εφαρμοστούν οι τεχνικές opinion miming σε μια ευρύτερη κλίμακα απόψεων των χρηστών για να εντοπιστεί το τι είναι πραγματικά αυτό που αφορά τους χρήστες πάνω σε ένα θέμα.

Κεφάλαιο 3
Εργαλεία text mining

3.1 Text mining

Η ευρεία χρήση και η ταχύτατη εξάπλωση του Παγκόσμιου Ιστού είχε σαν αποτέλεσμα τη μετατροπή του σε μια τεράστια αποθήκη πληροφοριών με δισεκατομμύρια σελίδες και περισσότερους από 2,110 εκατομμύρια χρήστες παγκοσμίως (Internet World Stats, 2011). Σήμερα θεωρείται ένα από τα πιο σημαντικά μέσα συλλογής, διαμοίρασης και διάδοσης πληροφοριών και υπηρεσιών. Την ίδια όμως στιγμή ο όγκος των διαθέσιμων πληροφοριών προκαλεί αρκετά προβλήματα που σχετίζονται με τη συνεχώς αυξανόμενη δυσκολία αναζήτησης, εύρεσης, οργάνωσης, πρόσβασης και συντήρησης της αιτούμενης πληροφορίας από τους χρήστες.

Μια λύση στο παραπάνω πρόβλημα έρχεται από την επιστημονική περιοχή του Data Knowledge Mining (Εξόρυξη Γνώσης από Δεδομένα) ή απλώς Data Mining (Εξόρυξη Δεδομένων) (Hand et al., 2001), όπως παρουσιάστηκε στο κεφάλαιο 2 του παρόντος εγγράφου. Πράγματι η εξόρυξη γνώσης από μεγάλες αποθήκες δεδομένων έχει εξελιχθεί σε ένα από τα βασικότερα ερευνητικά ζητήματα. Μέχρι πρόσφατα η εξόρυξη γνώσης αφορούσε αποκλειστικά δομημένα δεδομένα δηλαδή δεδομένα που είναι αποθηκευμένα σε βάσεις δεδομένων. Τα τελευταία χρόνια το ενδιαφέρον στράφηκε και σε μη δομημένα δεδομένα π.χ. κείμενα, εικόνες, ιστοσελίδες, κλπ. Αυτό οδήγησε στη δημιουργία νέων κλάδων όπως το Text Mining (Εξόρυξη Γνώσης από Κείμενα).

Η εξόρυξη κειμένου (text mining) είναι ένας νέος ερευνητικός τομέας που προσπαθεί να επιλύσει το πρόβλημα της υπερφόρτωσης πληροφοριών με τη χρησιμοποίηση των τεχνικών από την εξόρυξη από δεδομένα (data mining), την μηχανική μάθηση (machine learning), την επεξεργασία φυσικής γλώσσας (natural language processing), την ανάκτηση πληροφορίας (information retrieval), την εξαγωγή πληροφορίας (information extraction) και τη διαχείριση γνώσης (knowledge management). Ο κύριος στόχος του text mining είναι να βοηθήσει τους χρήστες να εξαγάγουν πληροφορίες από μεγάλους κειμενικούς πόρους. Δύο από τους σημαντικότερους στόχους είναι η κατηγοριοποίηση και η ομαδοποίηση εγγράφων.

Υπάρχει μια αυξανόμενη ανησυχία για την ομαδοποίηση κειμένων λόγω της εκρηκτικής αύξησης του Παγκόσμιου Ιστού, των ψηφιακών βιβλιοθηκών, των ιατρικών δεδομένων, κλπ. Τα κρισιμότερα προβλήματα για την ομαδοποίηση εγγράφων είναι η υψηλή διαστατικότητα του κειμένου φυσικής γλώσσας και η επιλογή των χαρακτηριστικών γνωρισμάτων που χρησιμοποιούνται για να αντιπροσωπεύσουν μια περιοχή. Κατά συνέπεια, ένας αυξανόμενος αριθμός ερευνητών έχει επικεντρωθεί στην έρευνα για τη σχετική αποτελεσματικότητα των διάφορων τεχνικών μείωσης διάστασης και της σχέσης μεταξύ των επιλεγμένων χαρακτηριστικών γνωρισμάτων που χρησιμοποιούνται για να αντιπροσωπεύσουν το κείμενο και την ποιότητα της τελικής ομαδοποίησης.

3.2 Rapid Miner

Για την αξιολόγηση των σχολίων που συγκεντρώθηκαν από διάφορες διαδικτυακές πηγές, κυρίως σε απόψεις που συγκεντρώθηκαν σε ηλεκτρονικούς δημόσιους διαλόγους, και την κατηγοριοποίησή τους σε θετικές και αρνητικές χρησιμοποιήθηκε το λογισμικό RapidMiner/YALE. Οι κύριοι λόγοι για την επιλογή του συγκεκριμένου λογισμικού είναι:

- Το RapidMiner/YALE είναι ένα open-source software που χρησιμοποιείται για την εξόρυξη δεδομένων και κειμένου για την επιστημονική και εμπορική χρήση. Διατίθεται μέσω της επίσημης ιστοσελίδας του δωρεάν (http://rapid-i.com/)
- Είναι ένα λογισμικό ιδιαίτερα αναγνωρισμένο σε παγκόσμια κλίμακα. Μέχρι σήμερα, χιλιάδες εφαρμογές του RapidMiner σε περισσότερες από 30 χώρες δίνουν στους χρήστες τους ένα ανταγωνιστικό πλεονέκτημα. Μεταξύ των χρηστών είναι πολύ γνωστές εταιρείες όπως Ford, Honda, Nokia, Miele, Philips, IBM, Bank of America και πολλές μεσαίες επιχειρήσεις που επωφελούνται από το open-source επιχειρηματικό μοντέλο RapidMiner. Το RapidMiner χρησιμοποιείται και για έρευνα και διεργασίες σε πραγματικής φύσεως δεδομένων παγκόσμια.

Όπως φαίνεται και παρακάτω στον πίνακα στατιστικών με θέμα «Ποιό data mining εργαλείο χρησιμοποιείται το τελευταίο χρόνο για πραγματικά projects», παρατηρούμε ότι το RapidMiner έρχεται πρώτο και με διαφορά περίπου 8% από το δεύτερο. Στην επόμενη ενότητα θα αναλυθούν περεταίρω και άλλα εργαλεία text mining.

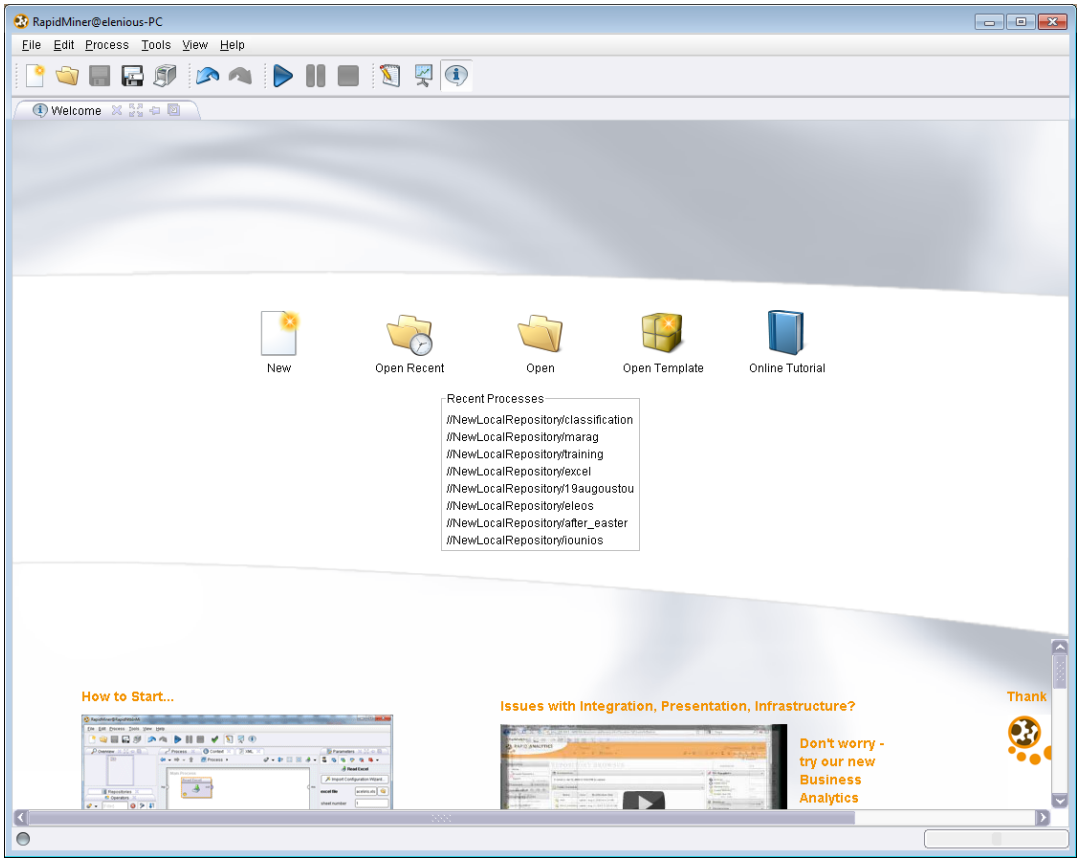
Data Mining / Analytic Tools Used Poll

Which data mining/analytic tools you used in the past 12 months for a real project (not just evaluation) [912 voters]	
RapidMiner (345)	37.8%
R (272)	29.8%
Excel (222)	24.3%
KNIME (175)	19.2%
Your own code (168)	18.4%
Pentaho / Weka (131)	14.3%
SAS (110)	12.0%
MATLAB (84)	9.2%
IBM SPSS Statistics (72)	7.9%
Other free tools (67)	7.3%

Πιο συγκεκριμένα το RapidMiner είναι:

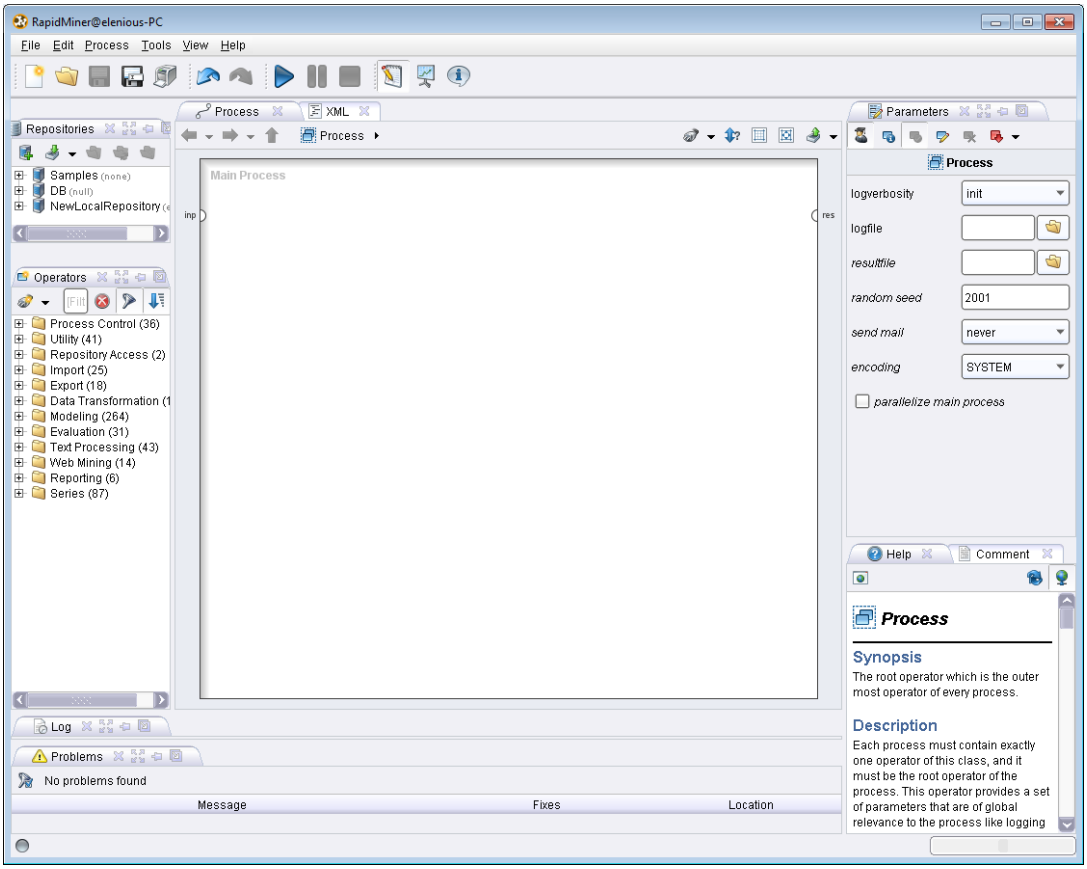
- Γραμμένο στην Java.
- Περιλαμβάνεται μια εσωτερική xml αναπαράσταση ώστε να εξασφαλίζεται η τυποποιημένη μορφή ανταλλαγής εξόρυξη δεδομένων σε διάφορα πειράματα.
- Εξασφαλίζεται η αποτελεσματική διαχείριση των δεδομένων αφού υπάρχει δυνατότητα προβολής αυτών σε πολλά επίπεδα.
- GUI, γραμμή εντολών mode (λειτουργία batch) και Java API για την χρήση του από άλλα προγράμματα
- Περιέχει ποικίλα plugins
- Μια μεγάλη σειρά αναπαράστασης των δεδομένων με λεπτομερή διάσταση

Αφού κατεβάσουμε την κατάλληλη για τον υπολογιστή μας έκδοση από την επίσημη ιστοσελίδα κι εγκαταστήσουμε το RapidMiner σύμφωνα με τις οδηγίες, ανοίγουμε το πρόγραμμα. Με μια πρώτη ματιά βλέπουμε ότι το περιβάλλον της εφαρμογής είναι πολύ κατανοητό για τον χρήστη, αφού πολύ εύκολα μπορεί να περιηγηθεί μέσα στην εφαρμογή χωρίς καμία δυσκολία. Η πρώτη εικόνα που βλέπει ένα χρήστης του RapidMiner είναι η παρακάτω:



Εικόνα 1:Αρχικό περιβάλλον RapidMiner

Μέσα από την αρχική σελίδα η πρώτη κίνηση που κάνουμε είναι να δημιουργήσουμε μία νέα διεργασία ανάλυσης, πατώντας το εικονίδιο «New». Με το πάτημα του εικονιδίου εμφανίζεται στην οθόνη μας η κύρια περιοχή σχεδίασης (Main Process) όπως φαίνεται παρακάτω:



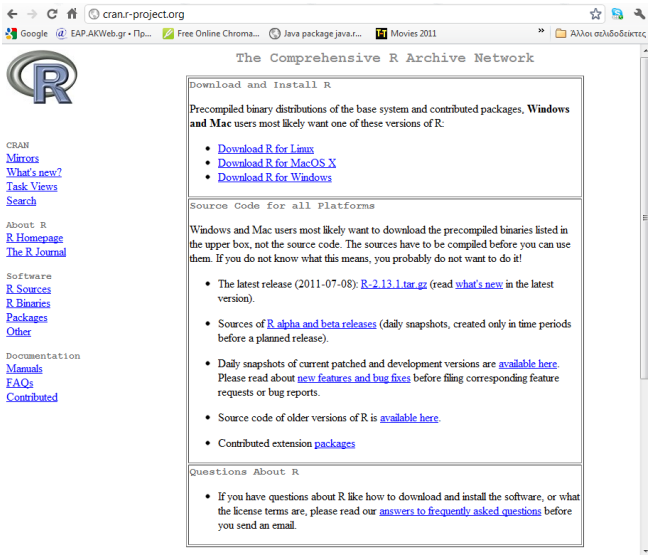
Εικόνα 2:Χώρος εργασίας RapidMiner

Στα αριστερά της οθόνης μπορούμε να διακρίνουμε τους διαθέσιμους Operators του λογισμικού, οι οποίοι παρουσιάζονται σε ομάδες, και είναι αυτοί που μπορούν να συμπεριληφθούν στην εκάστοτε διεργασία μας. Ουσιαστικά, η κάθε διεργασία στο RapidMiner είναι μία κατάλληλη αλληλουχία των διαθέσιμων Operators. Το λογισμικό μας διαθέτει και ένα πλήθος άλλων Operators τους οποίους μπορούμε να εγκαταστήσουμε από το menu «Help» επιλέγοντας «UpdateRapidMiner». Για εφαρμογές κατηγοριοποίησης κειμένου (Document classification/categorization) απαραίτητο είναι να εγκαταστήσουμε τη λίστα: «TextProcessing» που περιέχει ένα σύνολο Operators κατάλληλο για επεξεργασία των επιμέρους στοιχείων ενός κειμένου. Επίσης παρατηρούμε ότι υπάρχει πεδίο με το XML κώδικα (βλ. στο κέντρο της οθόνης), ο οποίος κάθε φορά που εισάγουμε ένα καινούργιο operator τροποποιείται αντίστοιχα. Τέλος στα δεξιά της οθόνης μας υπάρχει το πεδίο Parameters μέσω του οποίο διαλέγουμε τις κατάλληλες παραμέτρους ανάλογα τον operator που χρησιμοποιούμε.

3.3 Άλλα εργαλεία text mining

3.3.1 Το εργαλείο – R

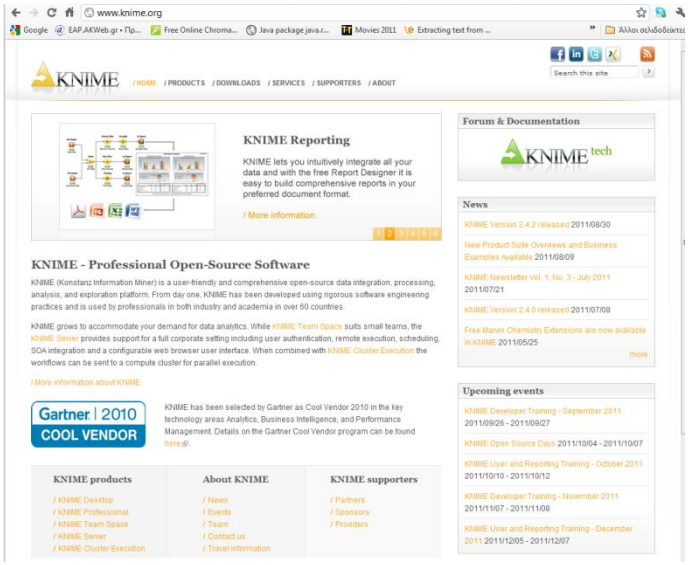
Το εργαλείο R έρχεται δεύτερο σαν επιλογή open-source λογισμικό για data mining ανάλυση, όπως φαίνεται στο πίνακα στατιστικών της προηγούμενης ενότητας (βλ. πίνακας 1). Το συγκεκριμένο λογισμικό σου δίνει την δυνατότητα να αναλύσει κάποιος δομημένα δεδομένα και μπορούμε να το βρούμε στην επίσημη σελίδα http://cran.r-project.org/ (Εικόνα 3). Διατίθεται για λειτουργικά συστήματα Linux, Mac OS, και Windows και είναι εξίσου δωρεάν όπως και το RapidMiner. Επιπλέον υπάρχουν επιπλέον πακέτα που είναι απαραίτητα να τα εγκαταστήσουμε για text mining (tm package).



Εικόνα 3:Το εργαλείο – R

3.3.2 Το εργαλείο – KNIME

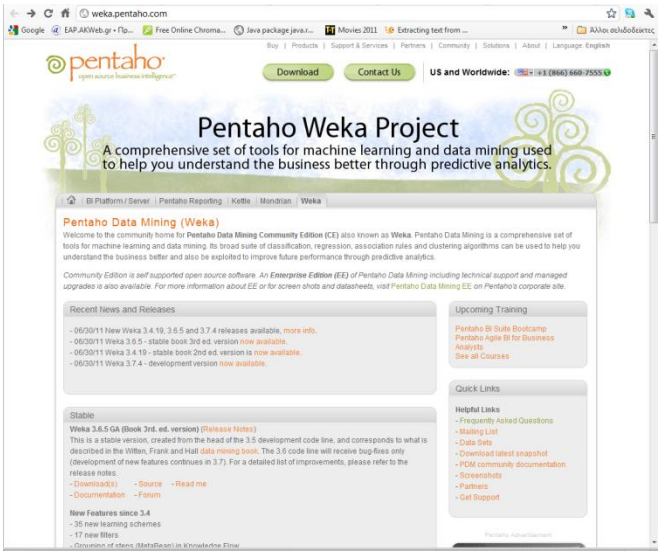
Το KNIME (the Konstanz Information Miner), είναι ένα φιλικό εργαλείο για τον χρήστη που χρησιμοποιείται για open-source ανάλυση δεδομένων. Ενσωματώνει διάφορα στοιχεία για την μηχανική μάθηση και εξόρυξη δεδομένων. Η γραφική διεπαφή χρήστη επιτρέπει τη γρήγορη και εύκολη συναρμολόγηση των κόμβων για την προ-επεξεργασία των δεδομένων, για την ανάλυση και μοντελοποίηση δεδομένων και οπτικοποίησης. Χρησιμοποιείται κατά κόρον από το 2006 στην φαρμακευτική έρευνα αλλά εξίσου και σε άλλους τομείς. Μπορούμε να το βρούμε και να το κατεβάσουμε δωρεάν στην σελίδα http://www.knime.org/ (Εικόνα 4).



Εικόνα 4:Το εργαλείο – KNIME

3.3.3 Το εργαλείο – Pentaho

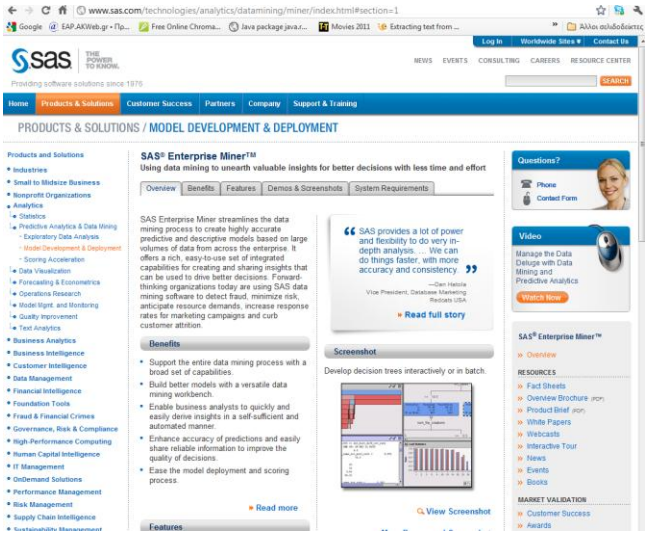
Το εργαλείο Pentaho ή γνωστό και ως WEKA είναι ένα λογισμικό εκμάθησης και εξόρυξης δεδομένων. Προσφέρει αλγόριθμους ταξινόμησης, παλινδρόμησης, ομαδοποίησης και κανόνων συσχέτισης οι οποίοι σε βοηθάνε να καταλάβεις το επιχειρηματικό τομέα και να μπορέσεις να βελτιώσεις τις αποδόσεις μέσω ανάλυσης προγνωστικών. Μπορούμε να το βρούμε και να το κατεβάσουμε με καταβολή χρημάτων στην σελίδα http://weka.pentaho.com/ (Εικόνα5).



Εικόνα 5:Το εργαλείο – Pentaho

3.3.4 Το εργαλείο – SAS Enterprise Miner

Το εργαλείο SAS Enterprise Miner βελτιώνει τη διαδικασία εξόρυξης δεδομένων για την δημιουργεί υψηλής ακρίβειας πρόβλεψης και μοντέλα τα οποία βασίζονται σε μεγάλο όγκο δεδομένων από ολόκληρη επιχείρηση. Χρησιμοποιείται από εταιρείες για την ανίχνευση πόρων και για να αυξήσει τα ποσοστά ανταπόκρισης για τις καμπάνιες μάρκετινγκ. Μπορούμε να το βρούμε και να το κατεβάσουμε δωρεάν στην σελίδα http://www.sas.com/technologies/analytics/datamining/miner/index.html#section=1 (Εικόνα 6)



Εικόνα 6:Το εργαλείο – SAS Enterprise Miner

Μια λίστα από άλλα εργαλεία παρουσιάζεται στο παράρτημα 1.

Κεφάλαιο 4
Αρχιτεκτονική της Πλατφόρμας

4.1 PADGETS

4.1.1 Εισαγωγή

Σήμερα, η ολοένα συνεχής ανάπτυξη του Παγκόσμιου Ιστού, ως μέσο που εκθέτει τη δυνατότητα να προσελκύσει και να διατηρήσει την ανάμιξη της κοινωνίας σε συνδυασμό με την ανάγκη για την δημιουργία μιας πολιτικής που επικεντρώνεται στον πολίτη και την κοινωνία, έχει ανάγκη για νέα εργαλεία τα οποία θα έχουν την ικανότητα να αναλύουν τον ρόλο της κοινωνίας και να προβλέπουν το πιθανό αντίκτυπο στις εκάστοτε πολιτικές. Κατά συνέπεια, η διακυβέρνηση γύρω από τον πολίτη γίνεται πιο πιεστική από ποτέ.

Το πρόγραμμα PADGETS στοχεύει στο συνδυασμό δύο καθιερωμένων μεθόδων, της αρχιτεκτονικής προσέγγισης του Web 2.0 για τη δημιουργία web εφαρμογών (gadgets) και της μεθοδολογίας των system dynamics η οποία αναλύει την σύνθετη συμπεριφορά των συστημάτων.

Ο στόχος είναι να σχεδιάσει, να αναπτύξει και να επεκτείνει ένα σύνολο πρωτότυπων εργαλείων, που θα επιτρέψει στους φορείς χάραξης πολιτικής να δημιουργήσουν γραφικά Web εφαρμογές οι οποίες θα επεκταθούν σε περιβάλλον βασικής γνώσης Web 2.0 media. Για αυτόν τον λόγο, το πρόγραμμα εισάγει την έννοια της πολιτικής συσκευής (Padget) , όπως είδαμε στην προσέγγιση των εφαρμογών συσκευών σε Web 2.0 (gadgets), για να αντιπροσωπεύσει μια μικροεφαρμογή Web, που συνδυάζει ένα πολιτικό μήνυμα το οποίο περιέχει ομαδική γνώση στα κοινωνικά μέσα (υπό μορφή περιεχομένου και δραστηριοτήτων χρηστών), και αλληλεπιδρά με τους τελικούς χρήστες σε δημοφιλείς θέσεις (όπως τα κοινωνικά δίκτυα, blogs, τα φόρουμ, οι περιοχές ειδήσεων, κ.λπ.) προκειμένου να μεταβιβάσουν τα περιεχόμενά του στους φορείς χάραξης πολιτικής.

Μέσω της προτεινόμενης πλατφόρμας οποιαδήποτε πολιτική μπορεί να γίνει μια επαναχρησιμοποιήσιμη και επικοινωνιακή Web εφαρμογή, για χρήση σχετικά με το υπάρχον περιεχόμενο και τις κοινωνικές δραστηριότητες στο Web. Οι φορείς χάραξης πολιτικής θα είναι σε θέση να εγκαταστήσουν τέτοιες εφαρμογές από μόνοι τους και να τις χρησιμοποιούν για να διαβιβάσουν τις πολιτικές τους στο ευρύ κοινό. Οι άνθρωποι μπορούν να χρησιμοποιήσουν αυτές τις εφαρμογές όπως χρησιμοποιούν καθημερινές υπηρεσίες και οι φορείς χάραξης πολιτικής μπορούν να συλλέξουν τα αποτελέσματα αυτής της αλληλεπίδρασης για την καλυτέρευση των πολιτικών τους και για να πάρουν, στη συνέχεια, αποφάσεις που αντιπροσωπεύουν τις πραγματικές φιλοδοξίες της κοινωνίας.

Η εφαρμογή και διαχείριση των PADGETS γίνεται από μια πολυεθνική κοινοπραξία, συντονιζόμενη από το Πανεπιστήμιο Αιγαίου και περιλαμβάνει 11 επιπλέον συνεργάτες από διάφορες χώρες της Ευρώπης.

4.1.2 Περιγραφή Padgets

Το πρόγραμμα PADGETS απευθύνεται και μπορεί να χρησιμοποιηθεί από οποιαδήποτε δημόσια διοίκηση ή φορείς χάραξης πολιτικής. Παρακάτω ακολουθεί ένα πιλοτικό παράδειγμα δραστηριότητας που πραγματοποιείται σε τρεις διαφορετικές χώρες και επικεντρώνεται σε δύο βασικά πολιτικά εκθέματα Ευρωπαϊκού βελληνεκούς. Οι δημόσιες διοικήσεις που περιλαμβάνονται στο παράδειγμα είναι:

- ✓ Το ελληνικό παρατηρητήριο ICT, που εποπτεύεται από το Υπουργείο Οικονομικών, το οποίο είναι η κύρια οργάνωση για τον έλεγχο, την πρόβλεψη και την ανάλυση του αντίκτυπου της ελληνικής ψηφιακής στρατηγικής, σε εθνικό επίπεδο, και είναι αρμόδιο για τη μέτρηση δεικτών i2010, τις διοικητικές μετρικές μείωσης φορτίων, του κυβερνητικού KPI, τον έλεγχο επένδυσης eGovernment, και έχει πρόσβαση σε όλη τη γνώση περί δημόσιου τομέα στην Ελλάδα.

- ✓ Η Piedmont περιοχή, μια από τις πιο δυναμικές και προχωρημένες Ιταλικές περιοχές στον τομέα του προγραμματισμού ICT, της βιομηχανικής ανάπτυξης και της διαχείρισης ΜΜΕ είναι μια περιοχή η οποία αποτελεί σώμα προγραμματισμού και σχεδιασμού που έχει πρωταρχικό ρόλο στην επιδίωξη των καθηκόντων που μεταβιβάζονται από τη διοικητική αποκέντρωση στους στρατηγικούς τομείς των δημόσιων υπηρεσιών (όπως η εργασία, οι μεταφορές, η χρηματοδότηση, η υγεία), στην ανάπτυξη της κοινωνίας των πληροφοριών, στην προώθηση της συνεργασίας μεταξύ των διαφορετικών επιπέδων τοπικής δημόσιας διοίκησης και στο συντονισμό της δράσης στην περιοχή ώστε να επιτευχθούν οι e-government στόχοι σε τοπικό επίπεδο.

- ✓ Το κέντρο για την ανάπτυξη της ηλεκτρονικής διακυβέρνησης στη Σλοβενίας, που ενεργεί ως ενισχυτική αντιπροσωπεία σχεδιασμού τόσο για τη σλοβένικη κυβέρνηση όσο και για ολόκληρη τη δυτική περιοχή των Βαλκανίων, συμπεριλαμβανομένης της Σερβίας, της Κροατίας, του Μαυροβουνίου και της Βοσνίας-Ερζεγοβίνης.

4.1.3 Πολιτικό περιεχόμενο και νομικό πλαίσιο

Η στρατηγική επίδραση του PADGET βρίσκεται στις συνεισφορές του προς την επίτευξη των στόχων της ψηφιακής ατζέντας 2020 για την ανοικτή και συμμετοχική διακυβέρνηση με τη χρήση προηγμένων ICT.

Υιοθετώντας μια διεπιστημονική προσέγγιση, που οδηγεί στο αποδιοργανωτικό αντίκτυπο των ανοικτών κοινωνικών μέσων, της κοινωνικής δικτύωσης και των WEB 2.0 τεχνολογιών, και τις εξελίξεις στις μεθοδολογίες system dynamics στην ανάλυση της συμπεριφοράς σύνθετων συστημάτων, ενώ χρησιμοποιεί με βέλτιστο τρόπο απέραντους πόρους γνώσης δημόσιου τομέα, τα PADGETS θα λειτουργήσουν στον τομέα διαμόρφωσης της πολιτικής βασιζόμενα στην προσομοιωμένη συμπεριφορά και τις επιθυμίες του μεγάλου αριθμού πολιτών.

Αυτός ο συνδυασμός τεχνολογιών και τάσεων χρήσης επιτρέπει την ανάπτυξη και την εφαρμογή νέων πολιτικών και προσομοιώσεων, οι οποίες στα πλαίσια των PADGETS, θα αντιμετωπίσουν τα βασικά κοινωνικά ζητήματα της ευρωπαϊκής ατζέντας μέσω των επιλεγμένων πιλότων στον σχεδιασμό ICT, την διεύθυνση υπηρεσιών και την ανεργία.

Τα αποτελέσματα των PADGETS θα υποστηρίζουν μια νέα προσέγγιση στην διαμόρφωση πολιτικής που περιλαμβάνει την κοινότητα χάραξης πολιτικής, την κοινότητα εμπειρογνωμόνων, και τους πολίτες γενικά συμβάλλοντας έτσι στην υιοθέτηση των ICT για την συμμετοχική ηλεκτρονική διακυβέρνηση, που οδηγεί τελικά στην καλύτερη χάραξη πολιτικής και εφαρμογή.

4.1.4 Αποτελέσματα - Συμπεράσματα

Το πρόγραμμα PADGETS αναμένεται να συμβάλει προς την επίτευξη των ακόλουθων αποτελεσμάτων:

- Βελτίωση της ενδυνάμωσης και της δέσμευσης των ατόμων, των ομάδων και των κοινοτήτων που αφορούν πολιτικές διεργασίες.
- Η εμπιστοσύνη των πολιτών θα αυξηθεί μέσω της διαφάνειας και ανατροφοδότησης των συνεισφορών τους
- Παρέχετε ανατροφοδότηση στα σημαντικά ζητήματα της Ευρωπαϊκής ατζέντας.
- Βελτίωση των προβλέψεων του πολιτικού αντίκτυπου, λόγω της αυξανόμενης συμβολής και συμμετοχής των ατόμων και των κοινοτήτων

- Βελτίωση της ανταγωνιστικής θέσης της Ευρωπαϊκής βιομηχανίας στους τομείς πολιτικής διαμόρφωσης και ICT.

Τα PADGETS είναι μια σημαντική προσέγγιση στο ευρύ κοινό τα οποία θα:

- Παρέχουν εύκολη και διαισθητική πρόσβαση στα Web 2.0 media για επικοινωνία πολιτικών προτάσεων και συλλογή ανατροφοδότησης.
- Ενδυναμώνουν τις δικτυακές επιδράσεις των ήδη υπαρχόντων κοινωνικών μέσων ώστε να περιλαμβάνουν χρήστες και online κοινότητες στη διαδικασία πολιτικής διατύπωσης.
- Αυξάνουν την εμπιστοσύνη και τη διαφάνεια των πολιτών μέσω των δημόσιων και καθιερωμένων κοινωνικών καναλιών
- Βοηθάνε στην πρόβλεψη της δημόσιας απάντησης και του αντίκτυπου των πολιτικών μέτρων.

Μέσω αυτής της μοναδικής προσέγγισης, αναμένονται σημαντικά οφέλη για τους πολίτες και τους φορείς χάραξης πολιτικής με τη συμπλήρωση και την προώθηση των δημοκρατικών τρόπων.

4.2 Εισαγωγή εφαρμογής

Το αντικείμενο της διπλωματικής εργασίας πραγματεύεται την εξόρυξη δεδομένων και γνώμης από ηλεκτρονικούς δημοσίους διάλογους και συγκεκριμένα από το www.opengov.gr. Ανατρέξαμε στο opengov στο οποίο παρέχονται στατιστικά στοιχεία όσον αφορά τις διαβουλεύσεις. Η επιλογή έγινε με βάση το πλήθος των σχολίων και το αντίκτυπο που έχει το θέμα στον πολίτη.

	Αριθμός				
Υπουργείο	**Διαβουλεύσεων**	**Σχολίων**	**Προσκλήσεων**	**Θέσεων**	**Αιτήσεων**
Διοικητικής Μεταρρύθμισης & Ηλεκτρονικής Διακυβέρνησης	0	0	0	0	0
Εσωτερικών	10	14.069	6	45	1.758
Οικονομικών	9	18.713	7	67	3.045
Εξωτερικών	0	0	0	0	0
Εθνικής Άμυνας	0	0	1	2	167
Ανάπτυξης, Ανταγωνιστικότητας & Ναυτιλίας *	28	4.433	23	225	4.079
Περιβάλλοντος, Ενέργειας και Κλιματικής Αλλαγής	33	8.860	11	160	4.517
Παιδείας, Δια Βίου Μάθησης και Θρησκευμάτων	22	10.226	15	581	9.608
Υποδομών, Μεταφορών και Δικτύων	15	2.814	9	72	1.685
Εργασίας και Κοινωνικής Ασφάλισης	8	1.819	6	70	1.842
Υγείας και Κοινωνικής Αλληλεγγύης	2	1.400	2	536	4.556
Αγροτικής Ανάπτυξης και Τροφίμων	8	3.053	5	20	1.508
Δικαιοσύνης	17	1.956	2	2	189
Προστασίας του Πολίτη	5	1.910	0	0	0
Πολιτισμού και Τουρισμού	2	580	23	102	3.161
Υπουργός Επικρατείας & Κυβερνητικός Εκπρόσωπος	2	182	1	1	36
Γενική Γραμματεία της Κυβέρνησης	1	535	1	1	40
OpenGov	5	366	1	1	213
Σύνολα	**167**	**70.916**	**115**	**1.888**	**36.541**

Εικόνα 7:Στοιχεία του opengov

Σύμφωνα με τα στατιστικά επιλέξαμε τη διαβούλευση με θέμα «βέλτιστη αξιοποίηση της Κάρτας Πολίτη». Συγκεκριμένα το μέγεθος των σχολίων είναι 1361 και οι απόψεις που εκφέρονται ποικίλουν. Το θέμα της κάρτας του πολίτη αποτελεί μείζων θέμα και βάζει σε σκέψεις όλους τους πολίτες ανεξαιρέτως επαγγέλματος, πολιτικών πεποιθήσεων και θρησκεύματος. Το εργαλείο που χρησιμοποιήσαμε είναι το RapidMiner, για το οποίο έχει γίνει αναφορά στο κεφάλαιο 4.

Σαν πρώτο βήμα για δική μας ευκολία και αντί να παίρνουμε ένα-ένα τα σχόλια από τη διαβούλευση με τον κλασσικό τρόπο του copy/paste, μέσω μιας διαδικασίας κατάλληλων operators που διαθέτει το εργαλείο RapidMiner καταφέραμε να συλλέξουμε όλο το μέγεθος των σχολίων σε ξεχωριστά txt αρχεία. Αυτή η διαδικασία μας δίνει τη δυνατότητα να επεξεργαστούμε ευκολότερα τα σχόλια στα επόμενα στάδια της εφαρμογής.

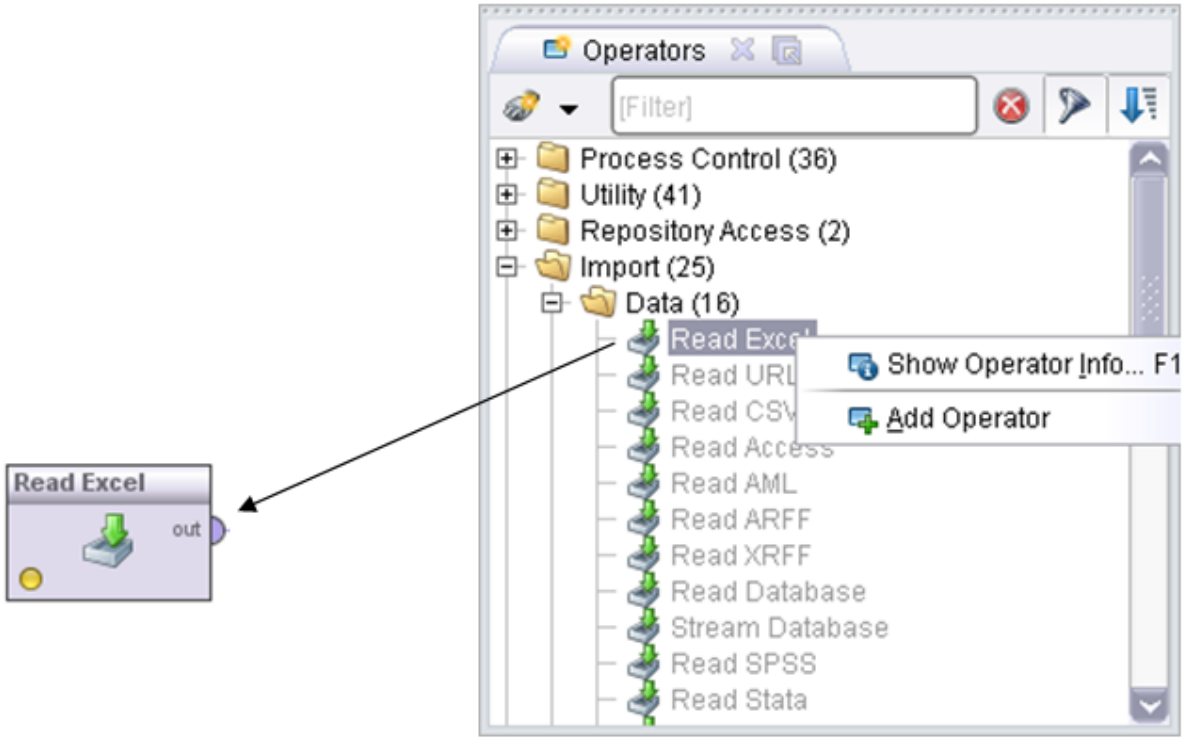
Επόμενο βήμα και το πιο βασικό ήταν να δημιουργήσουμε ένα βαθμό εμπειρίας (Training). Για αυτό το λόγο δημιουργήσαμε 2 διαφορετικές κλάσεις 100 θετικών και 100 αρνητικών σχολίων. Δηλαδή σε 2 διαφορετικούς φάκελους δημιουργήσαμε txt αρχεία, ένα σχόλιο ανά αρχείο, τα οποία ουσιαστικά είναι η είσοδος για γίνει το Training και το Testing(βλ.ενότητα 5.3). Ο αριθμός των σχολίων που επιλέξαμε είναι καθοριστικός γιατί όπως θα δούμε παρακάτω επηρεάζει σημαντικά την αξιοπιστία των αποτελεσμάτων μας.

Στο τελευταίο βήμα γίνεται τυχαία επιλογή ενός σχολίου αγνώστου ύφους/συναισθήματος το οποίο θα αναλυθεί, θα συγκριθεί με τα αποτελέσματα του προηγούμενο βήματος (training-testing) και θα γίνει μια αυτόματη κατηγοριοποίηση.

4.3 *Φάση 1η*

Για την πραγματοποίηση της 1ης φάσης υπάρχει διαθέσιμο excel αρχείο στην σελίδα του opengov με όλα τα σχόλια των χρηστών της διαβούλευσης «κάρτα του πολίτη». Αρχικά δημιουργούμε ένα main process πατώντας το εικονίδιο «New». Πρέπει να σημειωθεί ότι στις γενικές παραμέτρους της διεργασίας επιλέγουμε στο encoding UTF-8 λόγο τον Ελληνικών που θα επεξεργαστούμε (στο εξής σε όποιους καινούργιους operators χρησιμοποιούμε και υπάρχει επιλογή για encoding θα επιλέγουμε UTF-8). Είμαστε πλέον έτοιμοι να επιλέξουμε, να συνδέσουμε με την ορθή σειρά και να εισάγουμε τα αντίστοιχα δεδομένα στο σύνολο των operators που κρίνονται απαραίτητοι για την υλοποίηση του πρώτου σταδίου.

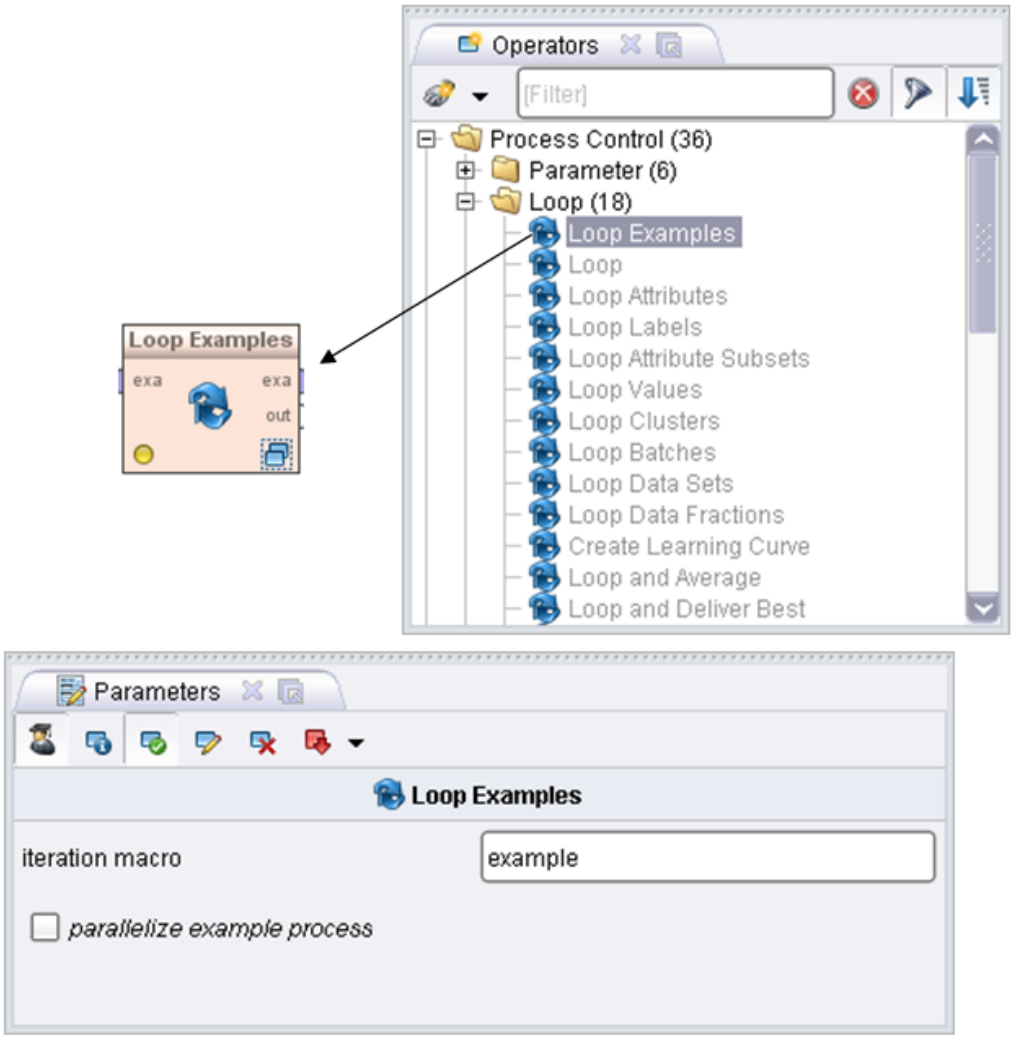
Ο αρχικός μας operator που διαλέγουμε ονομάζεται «Read Excel», και ανήκει στην ομάδα «Import».



Εικόνα 8:Read Excel

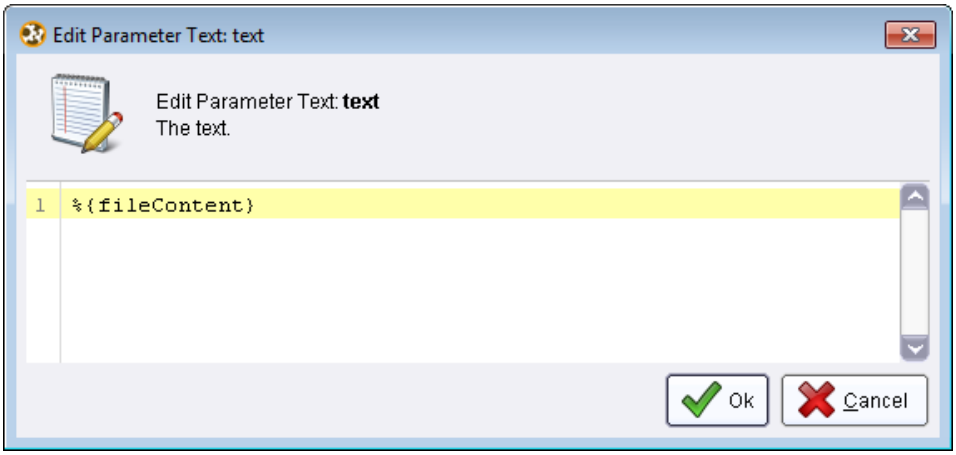
Με δεξί κλικ μπορούμε να τον προσθέσουμε στην διεργασία μας ή να δούμε περισσότερες λεπτομέρειες για την χρήση του ώστε να είμαστε σε θέση να συμπληρώσουμε όσο το δυνατόν ορθότερα τους παραμέτρους. Ο συγκεκριμένος operator χρησιμοποιείται για την ανάγνωση δεδομένων σε μορφή Microsoft Excel (Excel 95, 97, 2000, XP και 2003). Ο χρήστης πρέπει να καθορίσει ποιό λογιστικό φύλλο (spreadsheet) πρέπει να χρησιμοποιηθεί ως πίνακας

στοιχείων. Ο πίνακας πρέπει να έχει μία μορφή έτσι ώστε κάθε γραμμή να είναι ένα παράδειγμα και κάθε στήλη να αντιπροσωπεύει μια ιδιότητα. Στην συνέχεια χρησιμοποιούμε τον operator «Loop Examples» που ανήκει στην ομάδα «Process Control». Με δεξί κλικ μπορούμε να τον προσθέσουμε στην διεργασία μας και την ενώνουμε με την έξοδο του read excel. Με αυτόν τον operator δημιουργούμε ένα loop , όπου ως είσοδο παίρνει ένα σύνολο δεδομένων και εκτελεί τους εσωτερικές διεργασίες. Στις παραμέτρους πρέπει να εισάγουμε μια μακροεντολή ώστε να την χρησιμοποιήσουμε στο εσωτερικό του operator.



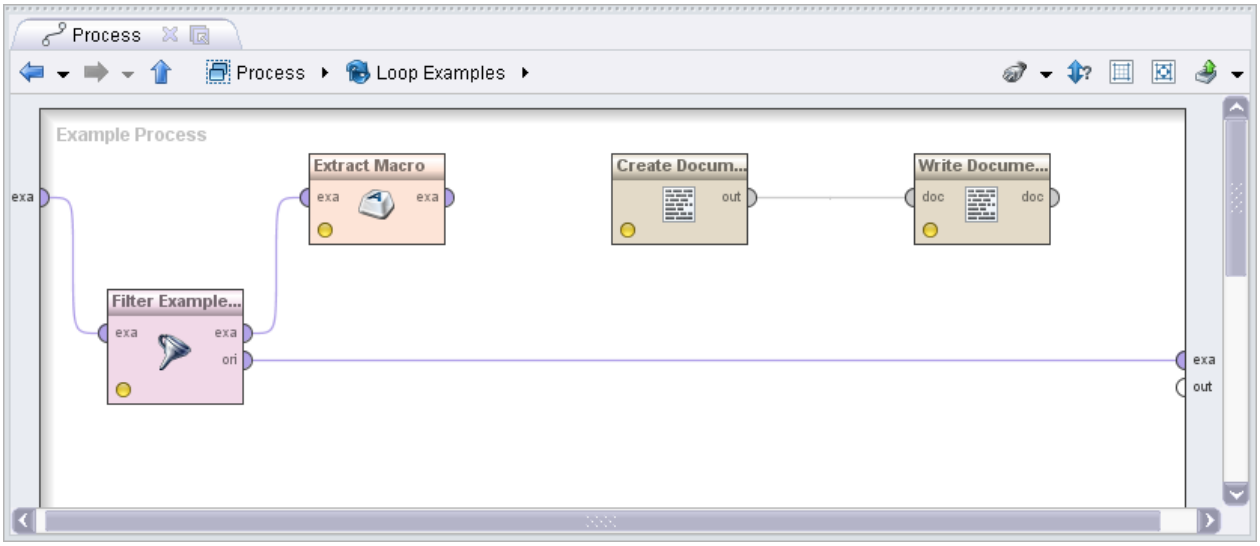
Εικόνα 9:Loop examples

Με διπλό κλικ στο «Loop examples» ανοίγουμε ένα νέο παράθυρο διεργασίας εσωτερικά του loop. Εισάγουμε δηλαδή νέους operator που θα τρέχουν μέσα στο loop. Πιο συγκεκριμένα εισάγουμε τους «Filter Example Range», «Extract Macro», «Create Document», «Write Document ». Ο operator «Filter Example Range» ανήκει στην ομάδα «Import». Αυτός ο operator επιτρέπει μόνο συγκεκριμένα δεδομένα που περνάνε και τα έχουμε διευκρινίσει στους παραμέτρους του «Loop examples». Στην συνέχεια την έξοδό του την κάνουμε είσοδο για στον operator «Extract Macro» που ανήκει στην ομάδα «Utility». Αυτός ο χειριστής καθορίζει μια μακροεντολή για την τρέχουσα διαδικασία (την οποία ονομάζουμε fileContent). Τέλος εισάγουμε στην διεργασία μας τους operators «Create Document» και «Write Document » που ανήκουν στην ομάδα «Text processing», οι οποίοι συνδέονται μεταξύ τους για την δημιουργία νέων αρχείων txt. Για να συνδεθεί το «create document» με την έξοδο του «Extract Macro» , εισάγουμε στους παραμέτρους όπως φαίνεται στην εικόνα.

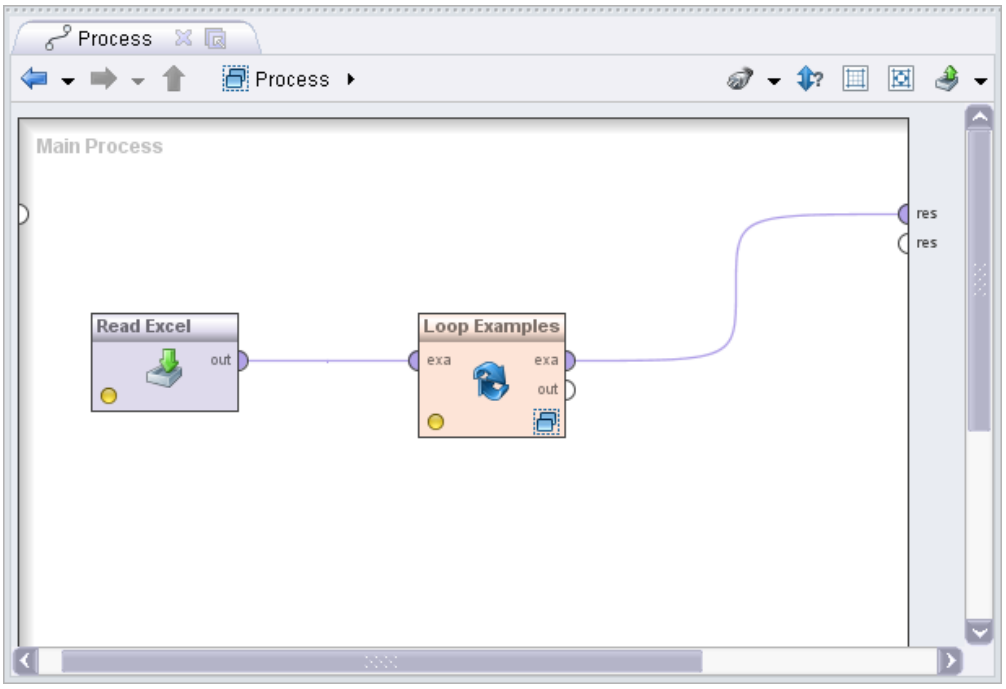


Εικόνα 10:Create Document

Παρακάτω φαίνονται τα print screens της διεργασίας μας. Το πρώτο είναι το κύριος μέρος και το δεύτερο το παράθυρο διεργασίας του loop.



Εικόνα 11:Loop examples

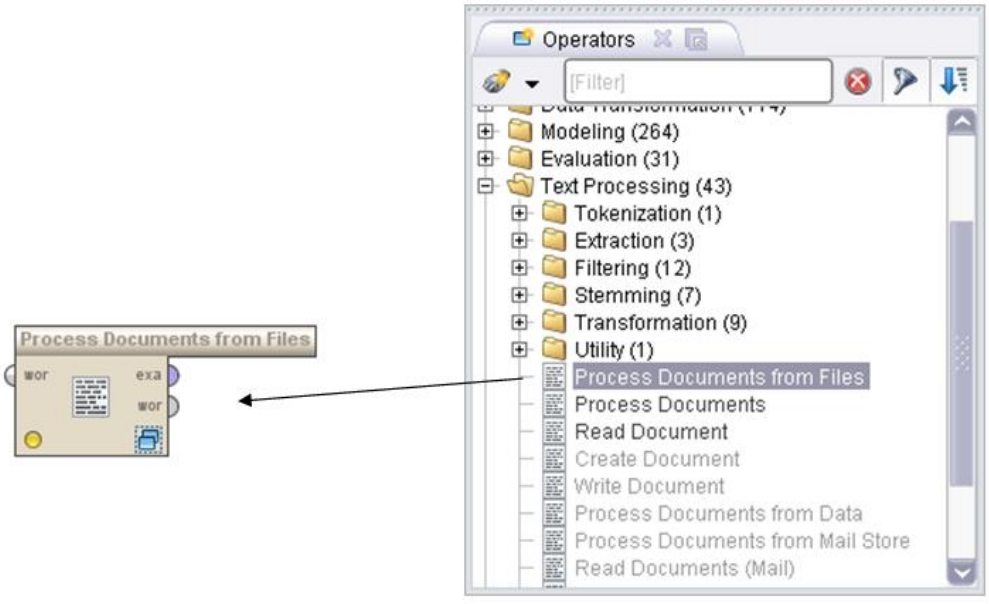


Εικόνα 12:Κύρια διεργασία φάσης 1η

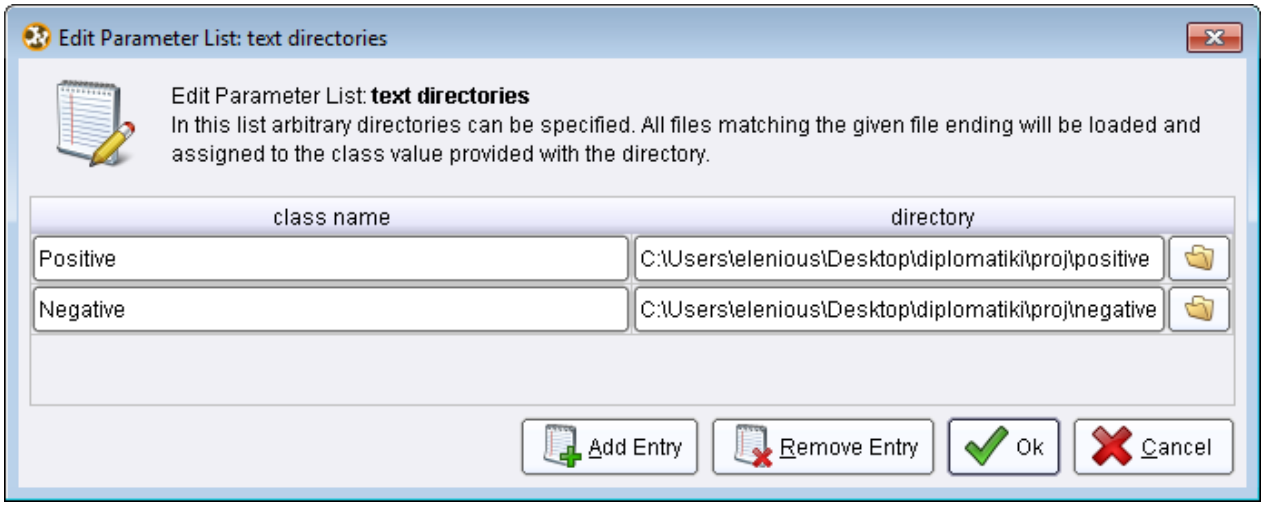
4.4 Φάση 2η

Για την πραγματοποίηση της 2ης φάσης δημιουργούμε ένα main process πατώντας το εικονίδιο «New». Στις βασικές παραμέτρους βάζουμε «Parallelize main process» ώστε να τρέχουν οι operators παράλληλα. Ο αρχικός μας operator είναι «Process Documents from Files» που ανήκουν στην ομάδα «Text processing» και παράγει τα word vectors από μια συλλογή κειμένων που αποθηκεύεται στα directories.

Αφού λοιπόν προσθέσουμε τον συγκεκριμένο Operator στη διεργασία μας, στο πεδίο «text directories» στις παραμέτρους δημιουργούμε τις δύο κλάσεις μας Positive και Negative. Τα σχόλια τα έχουμε κατηγοριοποιήσει προηγουμένως και χειροκίνητα σε θετικά και αρνητικά και αποτελέσουν την βάση για την εκπαίδευση του μοντέλου που θα χρησιμοποιηθεί στη συνέχεια για αυτόματη κατηγοριοποίηση. Για το λόγο αυτό, τα σχόλια χωρίζονται σε δύο διαφορετικά αρχεία προσθέτοντας το κατάλληλο label (ετικέτα, που υποδηλώνει την αντίστοιχη κατηγορία) σε καθένα από αυτά, δηλαδή positive και negative αντίστοιχα όπως φαίνεται παρακάτω.



Εικόνα 13:Process Documents from Files

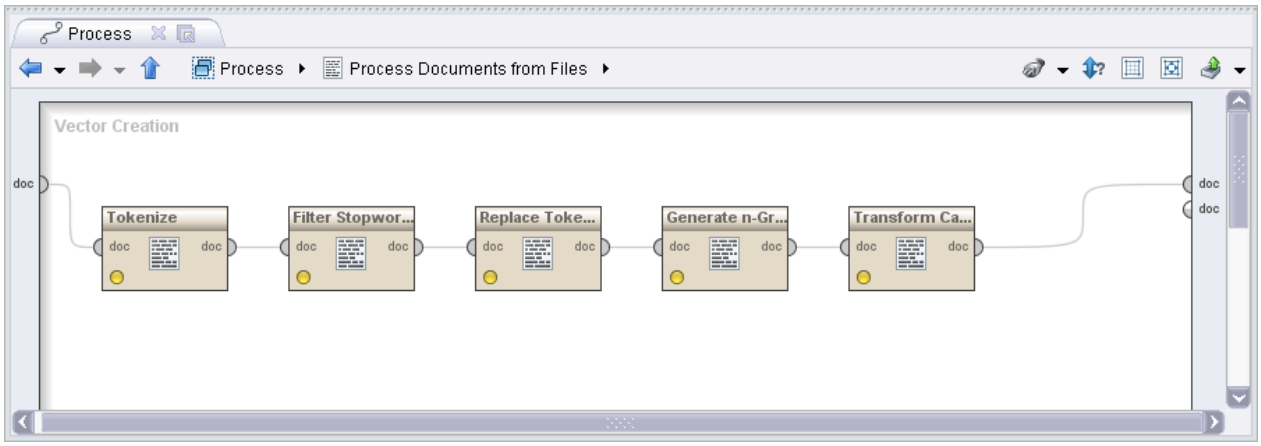


Εικόνα 14:text directories

Σημασία δίνεται ώστε να επιλεχθεί η συμβατή με τα αρχεία μας κωδικοποίηση από τη λίστα encoding όπως επίσης και στην επιλογή του πεδίου «vector creation», όπου εδώ επιλέγουμε από τη διαθέσιμη λίστα την επιλογή «term frequency», ώστε να μετρηθεί η συχνότητα εμφάνισης της κάθε λεκτικής οντότητας. Συνεχίζοντας, με διπλό κλικ στον Operator που έχουμε ήδη προσθέσει δημιουργούμε μία υποδιεργασία (subprocess) εντός αυτού. Εδώ θα επιλεγούν και θα συνδεθούν σε σειρά οι κατάλληλοι Operators από την ομάδα «Text Processing», οι οποίοι και θα βοηθήσουν στην περαιτέρω επεξεργασία των δεδομένων που έχουμε εισάγει. Δυστυχώς, κάποιοι χρήσιμοι Operators δεν είναι προσαρμοσμένοι ώστε να επεξεργάζονται κατάλληλα την ελληνική γλώσσα παρά μόνο την αγγλική κι ένα μικρό σύνολο κάποιων άλλων γλωσσών ευρέως χρησιμοποιούμενων, γεγονός που δυσκόλεψε την εργασία μας. Η υποδιεργασία που αναπτύχθηκε φαίνεται παρακάτω ενώ στην συνέχεια εξηγείται η χρησιμότητα κάθε Operator ξεχωριστά:

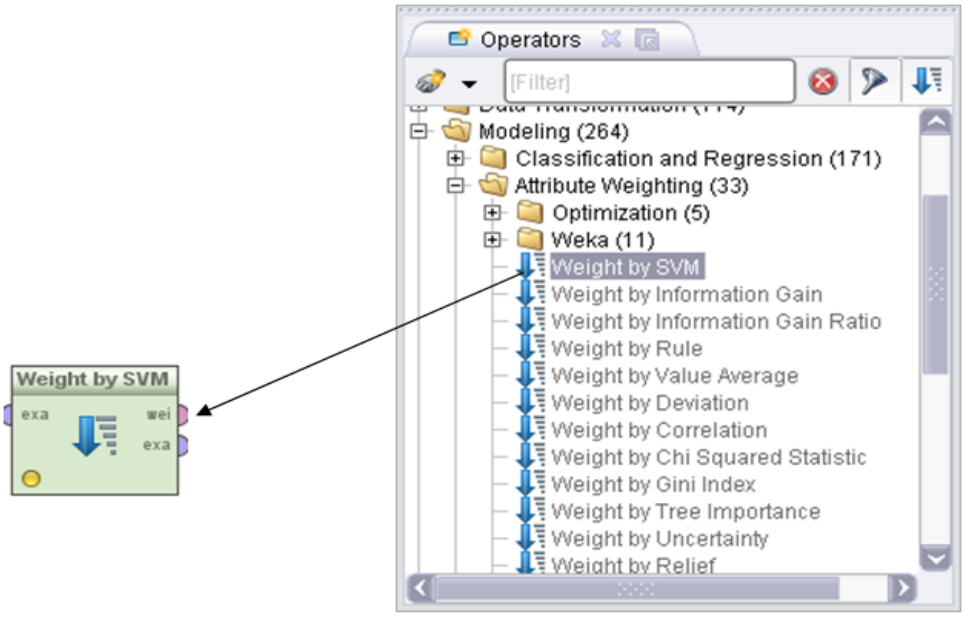
- «Tokenize»: Πατώντας στο συγκεκριμένο Operator κι επιλέγοντας «non letters» χωρίζει το κείμενο προς επεξεργασία σε λεκτικές μονάδες.
- «Filter Stopwords»: Αυτός ο operator αφαιρεί από το αρχείο που έχουμε εισάγει λέξεις που ταυτίζονται με μία λίστα την οποία και έχουμε δημιουργήσει μόνοι μας σε ένα αρχείο αφού δεν υπάρχει αντίστοιχος στα Ελληνικά στο πρόγραμμα.
- «Replace Tokens»: Με τον Operator αυτό αντικαθιστούμε τα ελληνικά γράμματα με αγγλικούς χαρακτήρες. Αυτό το κάναμε γιατί ύστερα από δοκιμές έχει καλύτερα αποτελέσματα (επειδή στα αρχεία υπήρχε πολύ θόρυβος).
- «Generate n-Grams» : Πατώντας στο συγκεκριμένο Operator κι επιλέγοντας «Max length» ίσο με 3, ορίζουμε μια ακολουθία 3 λέξεων.

- «Transform Cases»: Μετατρέπει όλους τους χαρακτήρες ενός κειμένου ή σε κεφαλαία είτε σε μικρά γράμματα αντίστοιχα.



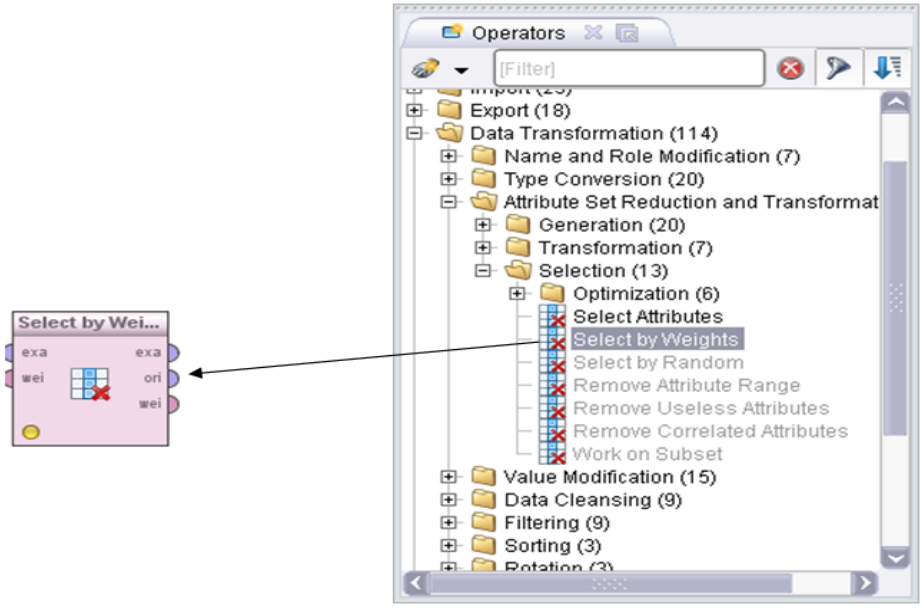
Εικόνα 15Υποδιεργασία Process Documents from files

Στην συνέχεια έχουμε τον operator «Weight by SVM», που ανήκει στην ομάδα «Modeling» και συνδέεται με το «Process Documents from Files». Αυτός ο operator χρησιμοποιεί τους συντελεστές του κανονικού διανύσματος ενός γραμμικού SVM ως βάρη χαρακτηριστικών γνωρισμάτων. Το RapidMiner διαθέτει και άλλους χειριστές βασισμένους στο SVM αλλά διαλέξαμε τον συγκεκριμένο γιατί υποστηρίζει πολλαπλές κατηγορίες. Στην συνέχεια συνδέουμε τον operator «Select by Weights» που ανήκει στην ομάδα «Data Transformation»

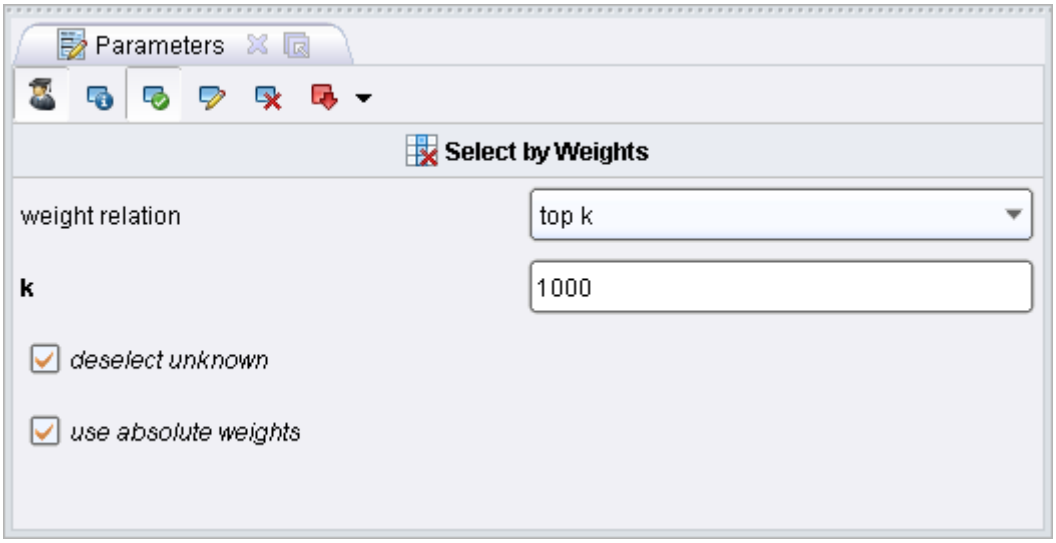


Εικόνα 16:Weight by SVM

Ο operator «Select by Weights» επιλέγει μόνο τις ιδιότητες που τα βάρη εκπληρώνουν μια δεδομένη σχέση, όσον αφορά τα βάρη ιδιοτήτων εισαγωγής. Αυτή την παράμετρο την δημιουργούμε στους παραμέτρους όπου επιλέγουμε «top k» με k ίσο με 1000 (δηλαδή θα επιστρέψει ένα exampleset που περιέχει μόνο τις 100 υψηλότερες σταθμισμένες ιδιότητες).

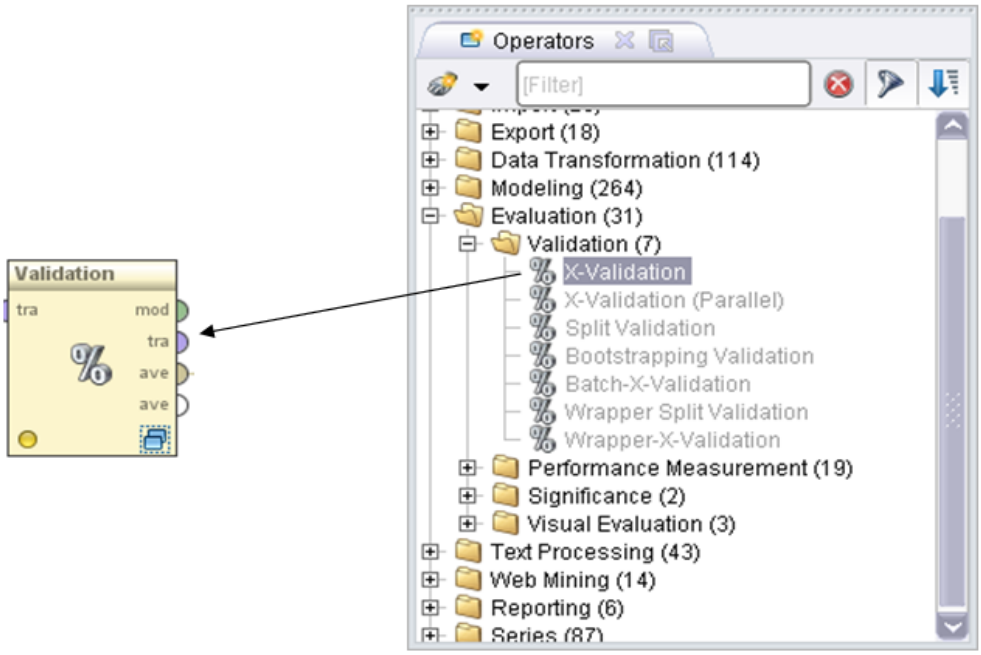


Εικόνα 17:Select by Weights



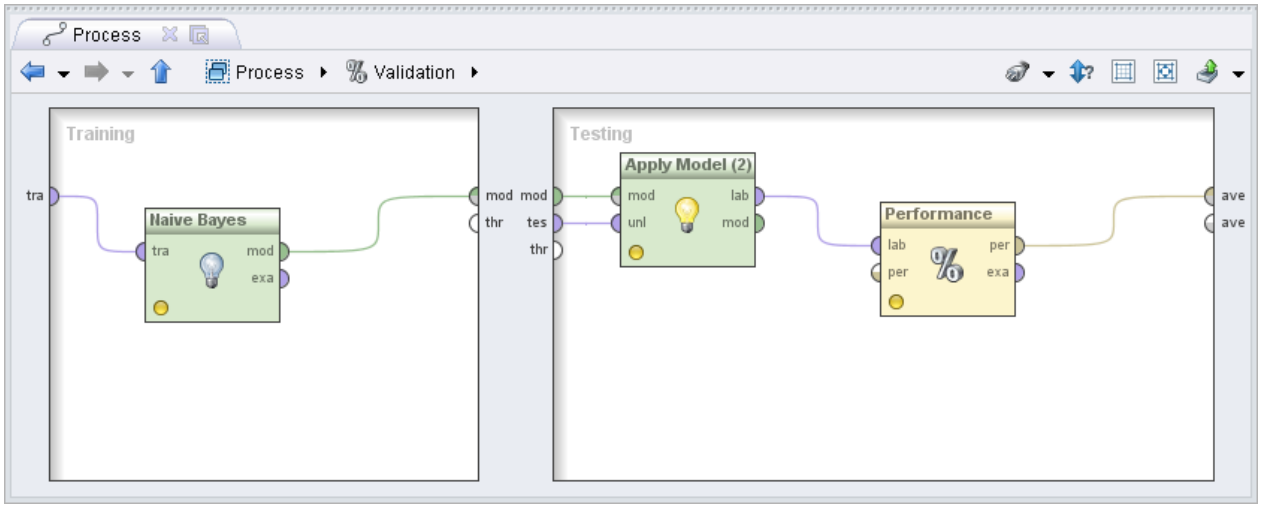
Εικόνα 18:Παράμετροι Select by Weights

Στην συνέχεια συνδέουμε τον operator «X-Validation» που ανήκει στην ομάδα «Evaluation» και ως παραμέτρους βάζουμε «Parallelize training» και «Parallelize testing». Ο συγκεκριμένος Operator εκτιμά την απόδοση του μοντέλου που θα εκπαιδεύσουμε και θα εφαρμόσουμε στη συνέχεια στα μη κατηγοριοποιημένα δεδομένα μας.



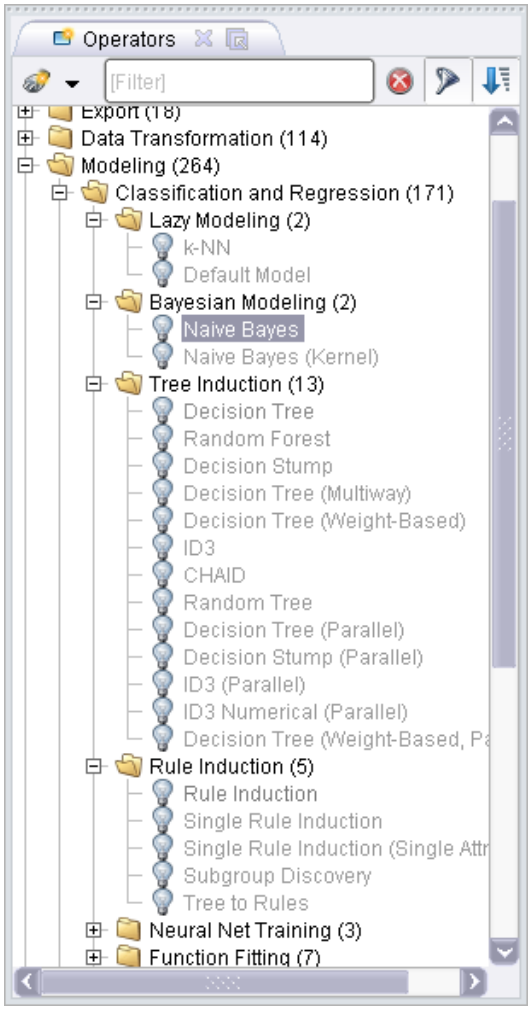
Εικόνα 19:X-Validation

Ο «X-Validation» περιέχει εσωτερικά μία υποδιεργασία, η οποία και επιστρέφει το ζητούμενο αυτό μοντέλο το οποίο εκπαιδεύεται μέσω των δεδομένων που εισαγάγαμε αρχικά στη διεργασία μας, εν προκειμένω τα αρχεία που περιέχουν τις θετικές και αρνητικές απόψεις όπως αυτές έχουν συγκεντρωθεί. Σημειώνεται ότι τα αποτελέσματα που επιστρέφονται αποτελούν απλά μία εκτίμηση της απόδοσης του μοντέλου και δεν πρέπει να ταυτίζονται με τα ακριβή και πραγματικά. Παρακάτω φαίνεται η υποδιεργασία του «X-Validation».



Εικόνα 20:Υποδιεργασία του «X-Validation».

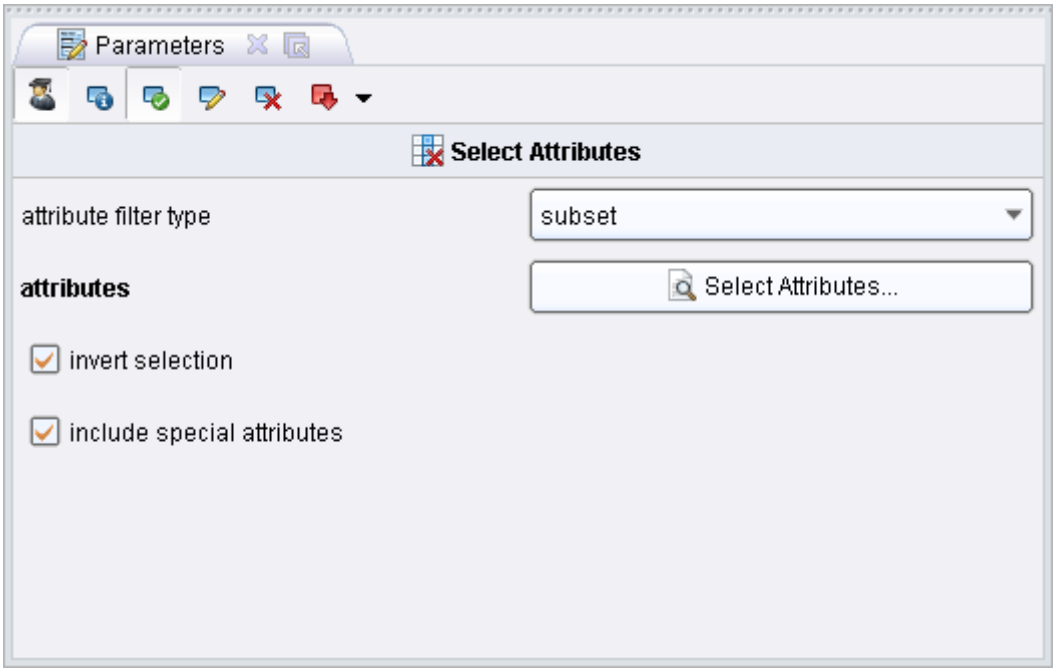
Σχετικά με την υποδιεργασία, όπως φαίνεται και από το παραπάνω σχήμα, αποτελείται από δύο φάσεις, τη φάση της εκπαίδευσης του μοντέλου και τη φάση της εφαρμογής αυτού στο σύνολο των δεδομένων. Αναλυτικότερα, το RapidMiner διαθέτει ένα πλήθος έτοιμων Operators, εξειδικευμένων στην κατηγοριοποίηση δεδομένων, όπως φαίνεται παρακάτω:



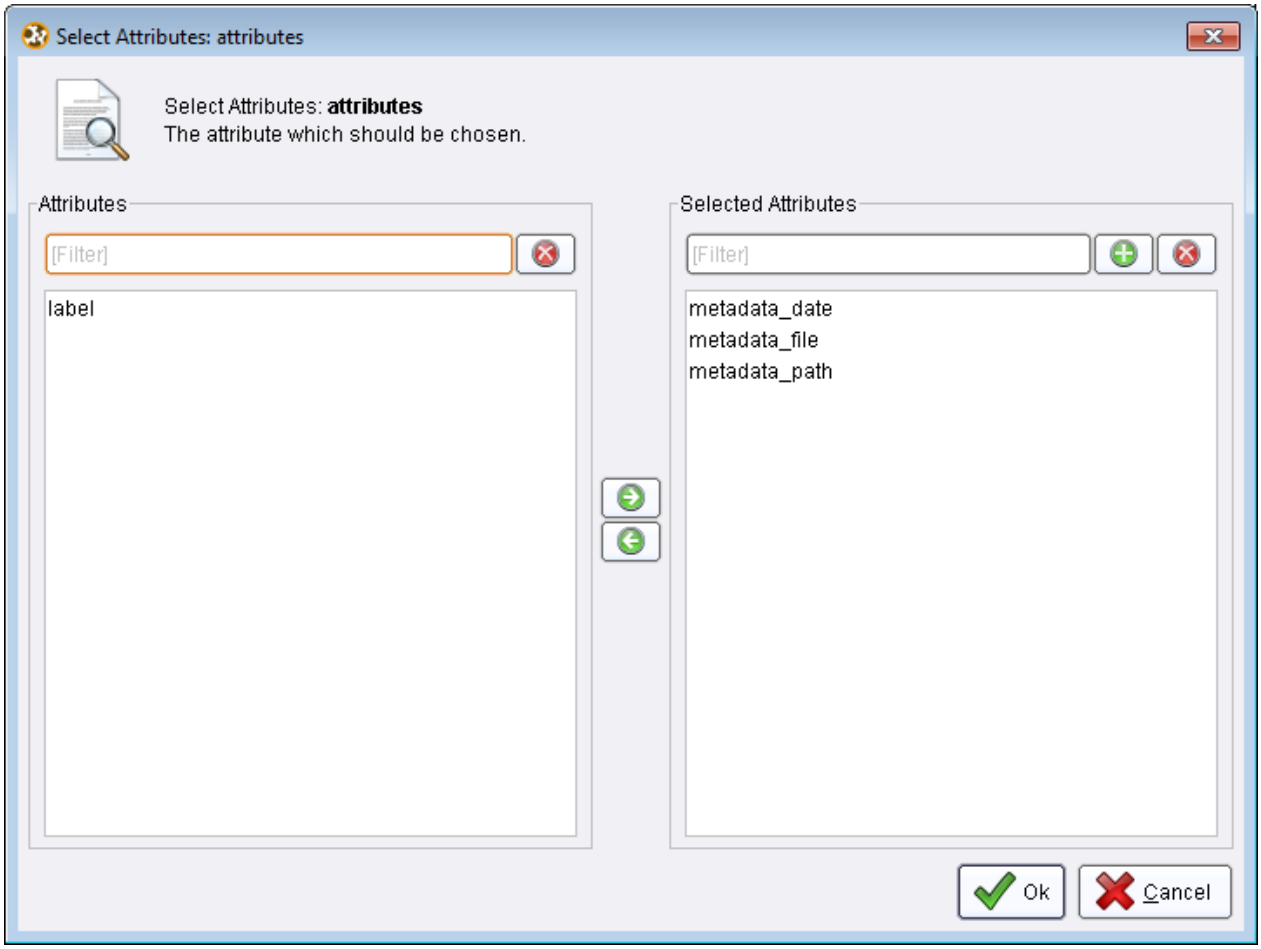
Εικόνα 21:Operators για κατηγοριοποίηση δεδομένων

Από τους διαθέσιμους Operators, έπειτα από δοκιμές, εκλέχθηκε τελικά ο αλγόριθμος «Naïve Bayes», ο οποίος είχε την μεγαλύτερη απόδοση, δηλαδή σωστότερη πρόβλεψη σε σύγκριση με τους υπολοίπους. Ο «Naïve Bayes» αποτελεί ένα μοντέλο που βασίζεται στην αποτίμηση πιθανότητας ώστε να παράγει την ορθότερη πρόβλεψη και συνιστά μία απλή και ιδιαιτέρως δημοφιλή μέθοδο μηχανικής μάθησης που δύναται να εφαρμοστεί και στο πεδίο κατηγοριοποίησης φυσικής γλώσσας. Με τον επόμενο Operator, τον «Apply Model» από την ομάδα «Modeling» - «Model Application», ουσιαστικά εφαρμόζουμε το μοντέλο που δημιουργήθηκε μέσω του «Naïve Bayes» στο σύνολο των δεδομένων του παραδείγματός μας. Ο Naïve Bayes Operator είναι αυτός που περιέχει τις απαραίτητες πληροφορίες και τα δεδομένα πάνω και στα οποία εκπαιδεύτηκε και θα χρησιμοποιηθεί αργότερα για την πρόβλεψη κατηγοριοποίησης της νέας πληροφορίας που θα εισαχθεί σε θετική ή αρνητική. Ο τελευταίος Operator αυτής της υποδιεργασίας είναι ο Operator «Performance» που βρίσκεται αντίστοιχα στην ομάδα: «Evaluation» - «Performance Measurement» και υπολογίζει μία πρώτη εκτίμηση της ακρίβειας της πρόβλεψής μας. Σημειώνεται εδώ ότι η εκτίμηση είναι καθαρά θεωρητική και η πραγματική απόδοση του μοντέλου δύναται να απέχει από αυτή.

Πηγαίνοντας πάλι στην κύρια διεργασία συνδέουμε το «X-Validation» με τον operator «Write model» που βρίσκεται στην ομάδα «Export» και αποθηκεύουμε σε φάκελο της επιλογής μας (naïve_bayes) το μοντέλο που δημιουργήσαμε ώστε μελλοντικά και εδώ συγκεκριμένα στην επόμενη διεργασία μας να είμαστε σε θέση να το εφαρμόσουμε σε νέα δεδομένα. Στην συνέχεια θέλουμε να αποθηκεύσουμε τα στοιχεία μας σε ένα arff αρχείο το οποίο και θα χρησιμοποιήσουμε στην εφαρμογή της java. Αυτό το κάνουμε τον operator «Write Arff» που βρίσκεται στην ομάδα «Export». Όμως θέλουμε να επιλέγονται συγκεκριμένα στοιχεία από το Training, γι' αυτό τον λόγο εισάγουμε τον operator «Select Attributes» που βρίσκεται στην ομάδα «Data Transformation» και στις παραμέτρους του δηλώνουμε τα στοιχεία που χρειαζόμαστε (Label) στο πεδίο «attributes», όπως και φαίνεται παρακάτω

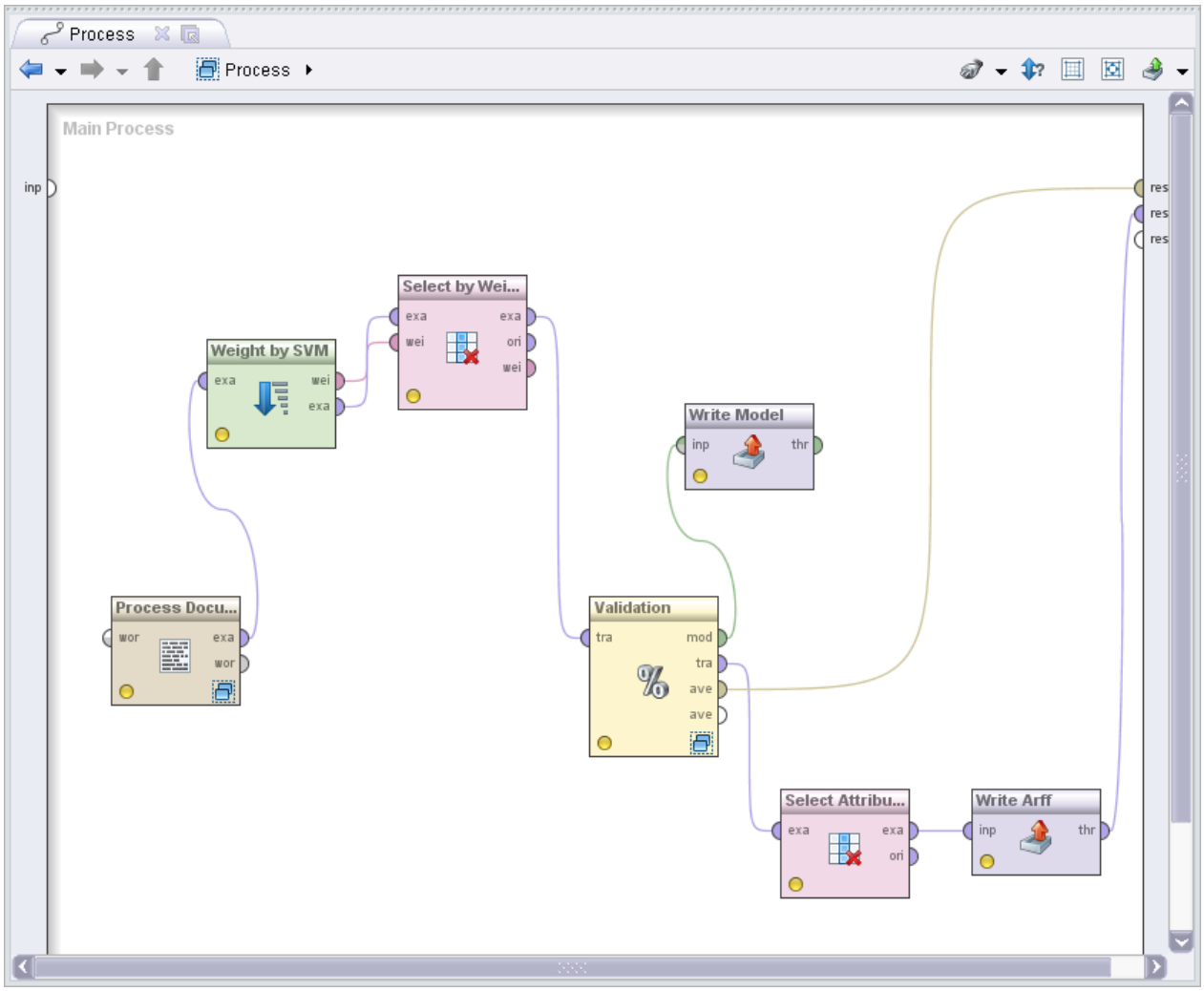


Εικόνα 22:Παράμετροι Select Attributes



Εικόνα 23:Επιλέγεται label

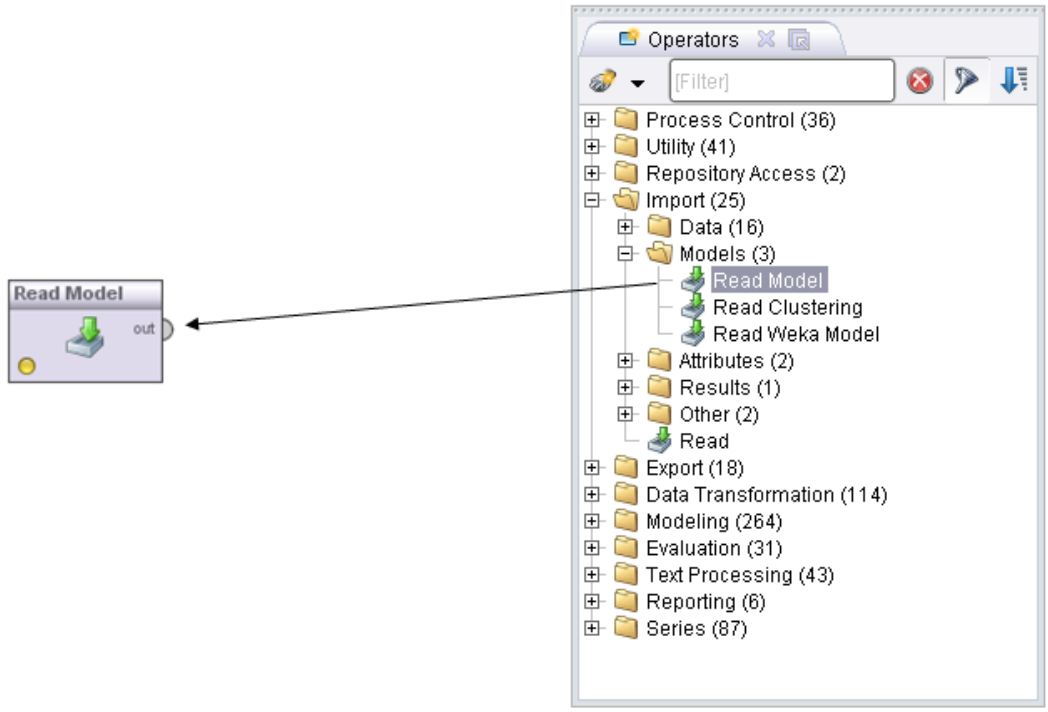
Παρακάτω φαίνεται η ολοκληρωμένη η κύρια διεργασία:



Εικόνα 24:Κύρια διεργασία φάσης 2η

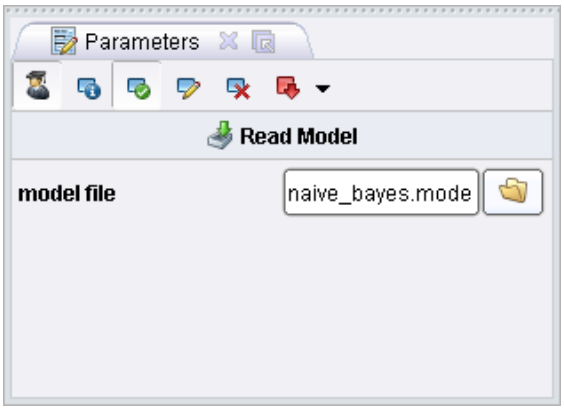
4.5 *Φάση 3η*

Για την πραγματοποίηση της Φάσης 3 δημιουργούμε ένα main process πατώντας στο εικονίδιο «new» .Όπως έχουμε προαναφέρει, στη φάση 3 εξετάζουμε την αξιοπιστία του μοντέλου που έχουμε δημιουργήσει στη Φάση 2. Προσθέτουμε στο main process της Φάσης 3 το operator «read model» το οποίο ανήκει στην κατηγορία «import» όπως φαίνεται παρακάτω. Με τη βοήθεια αυτού του operator κάνουμε import το model file που δημιουργήσαμε στη Φάση 2.



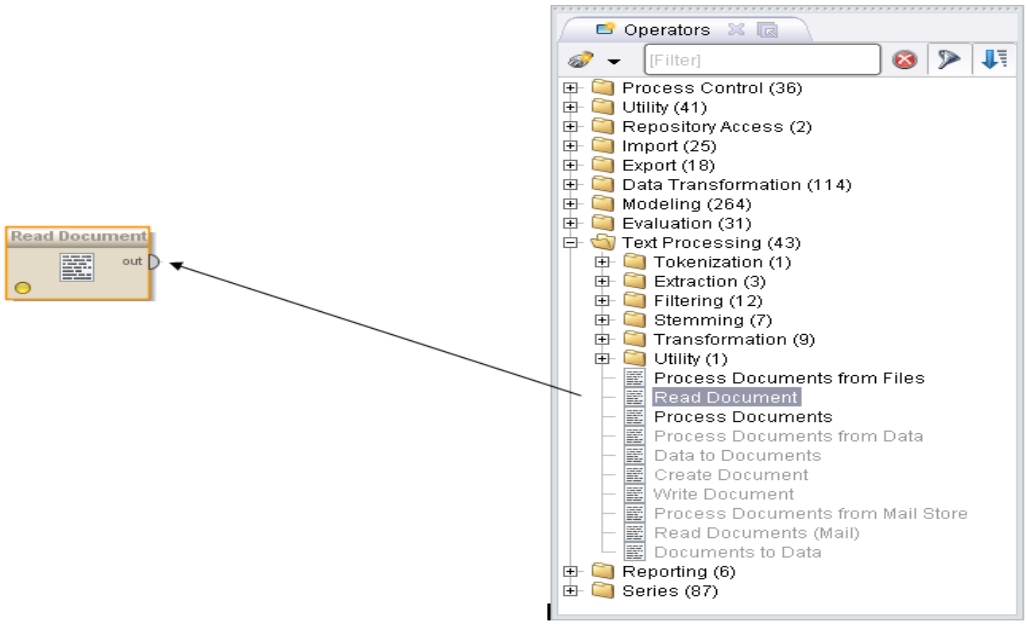
Εικόνα 25:Read Model

Έχουμε την λίστα παραμέτρων του «read model» όπως φαίνεται παρακάτω. Είσοδος είναι το μονοπάτι που έχουμε αποθηκεύσει το model file.



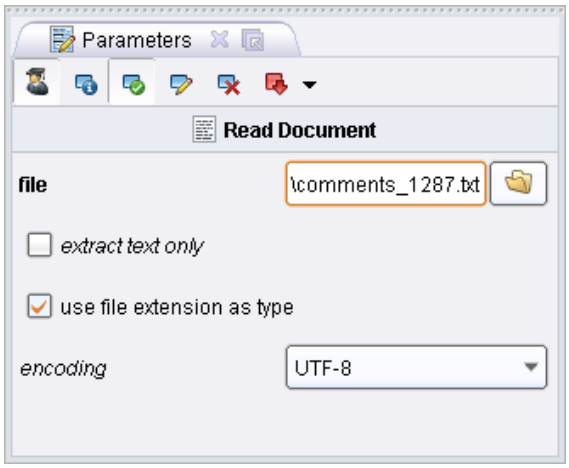
Εικόνα 26:Εισαγωγή του μοντέλου

Όπως έχουμε αναφέρει στη Φάση 1 έχει γίνει καταγραφή και αποθήκευση όλου του πλήθους των σχολίων. Από το σύνολο των σχολίων αυτών επιλέγουμε ένα με τυχαίο τρόπο. Το εισάγουμε ως είσοδο σε ένα operator το οποίο μπορεί να διαβάσει αρχεία κειμένου «read document» το οποίο βρίσκεται στην κατηγορία «text processing».



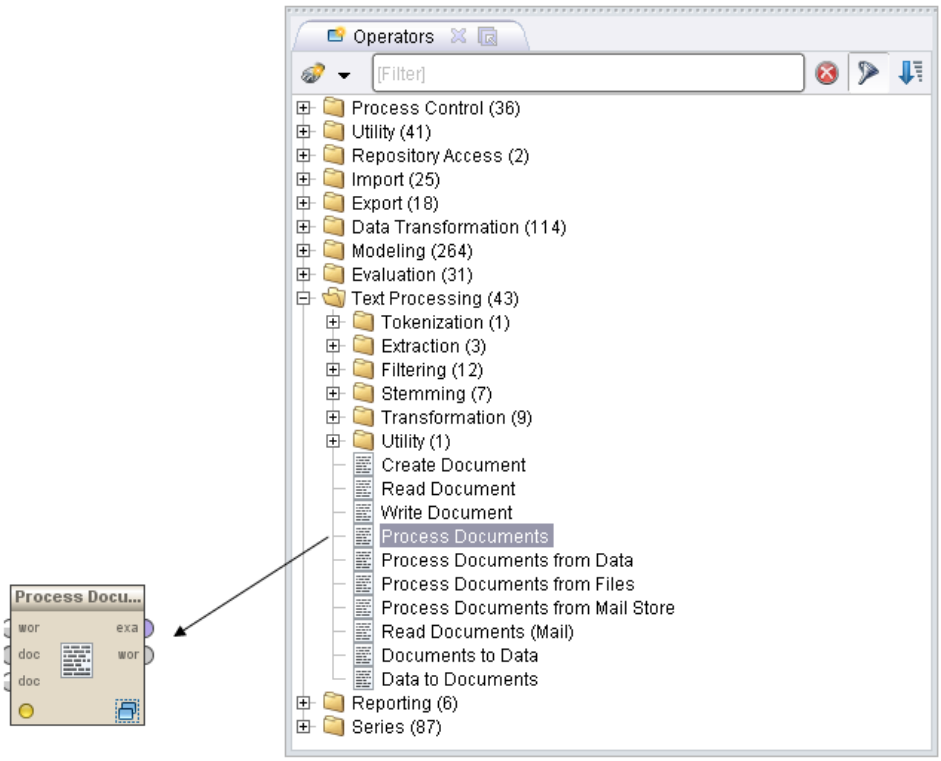
Εικόνα 27:Read Document

Παρακάτω εμφανίζεται η λίστα παραμέτρων που του operator «read document» .Όπως παρατηρούμε σαν είσοδο έχουμε το μονοπάτι που είναι αποθηκευμένο το σχόλιο μας. Επίσης επιλέγουμε κωδικοποίηση UTF-8 για να μπορεί να 'διαβάσει' σωστά το κείμενο το RapidMiner.



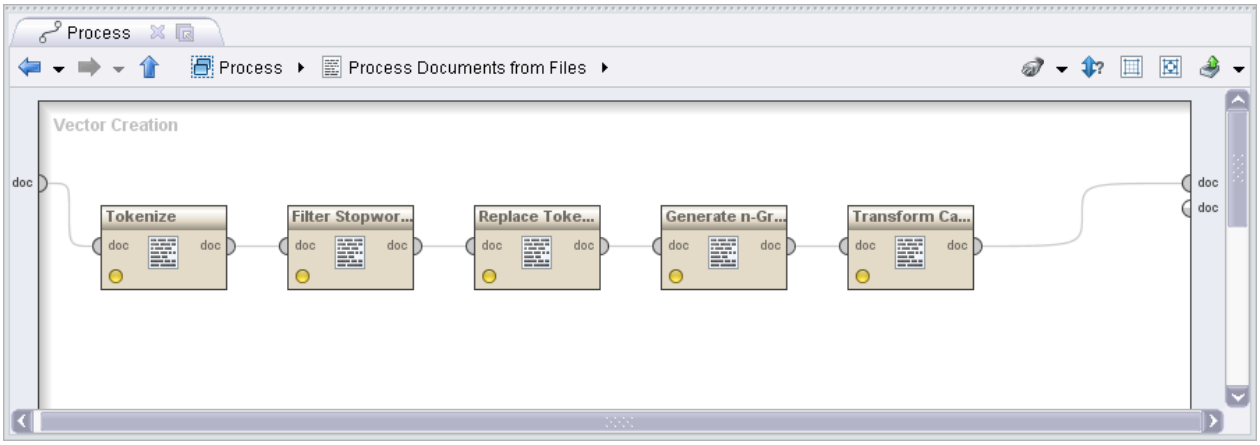
Εικόνα 28:Εισαγωγή τυχαίου σχολίου

Στη main process έχουμε βάλει το «read document». Επόμενο operator που πρέπει να εισάγουμε είναι το «process documents» που συνδέεται με το «read document». Η λειτουργία του είναι για να γίνει text mining στο σχόλιο που έχουμε εισάγει .



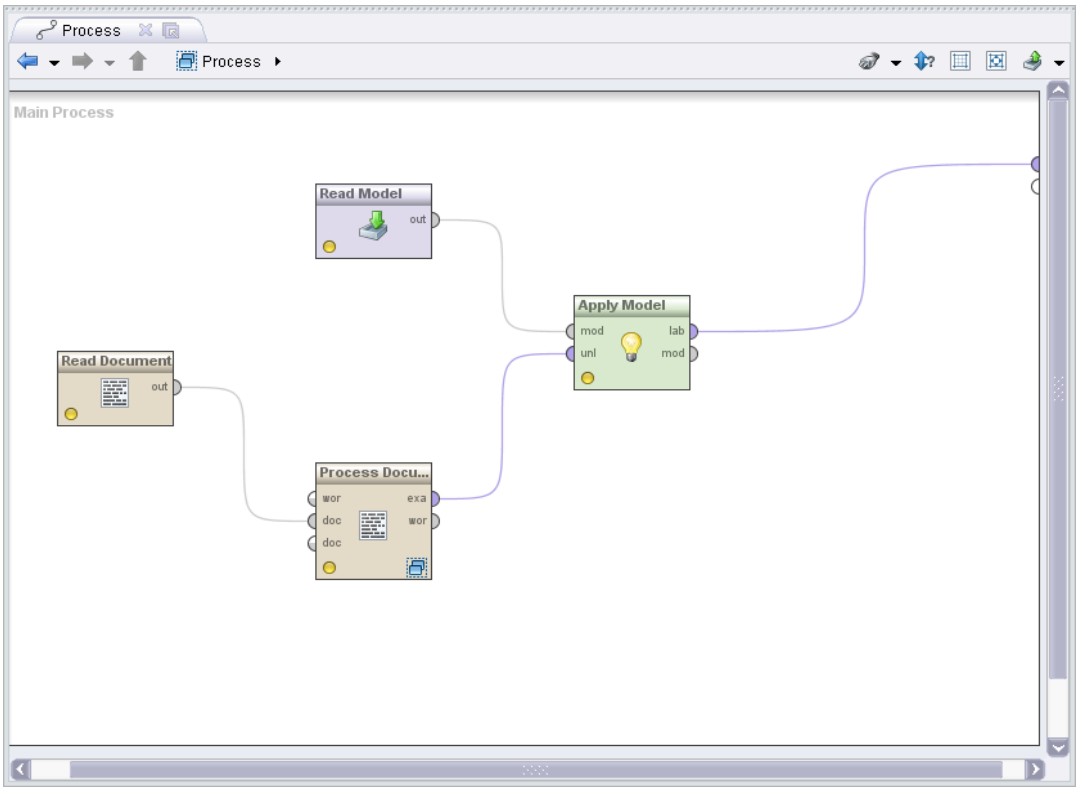
Εικόνα 29:Process Documents

Όμοια με τη Φάση 2 μέσα στο «process documents» με διπλό κλικ στον Operator προσθέτουμε μία υποδιεργασία (subprocess) εισάγοντας τα κατάλληλα operators τα οποία χρησιμοποιούνται για να γίνει text processing στο σχόλιο (τα οποία και αναλύσαμε στην φάση 2η).



Εικόνα 30:Υποδιεργασία του Process Documents

Στην συνέχεια εισάγουμε τον operator «Apply Model» τον οποίο έχουμε αναπτύξει στην φάση 2η ο οποίος έχει δύο εισόδους: το output του read model και του «process documents» , όπως φαίνεται στο main process της τελικής μας φάσης

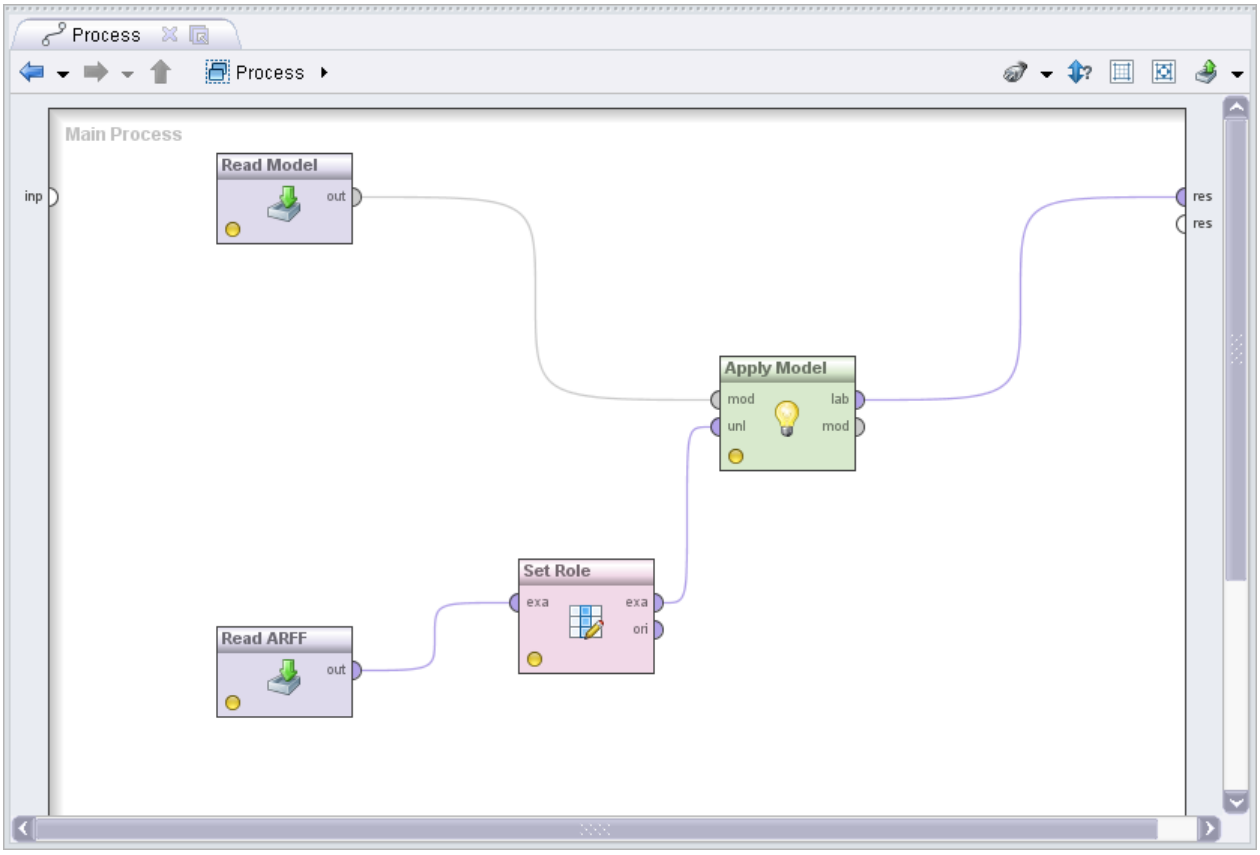


Εικόνα 31:Κύρια διεργασία

4.6 *Εφαρμογή σε java*

Για την καλύτερη διευκόλυνση των χρηστών δημιουργήσαμε μια εφαρμογή στην java ώστε ο εκάστοτε χρήστης να έχει την δυνατότητα να χρησιμοποιήσει τις παραπάνω φάσεις για την εξόρυξης γνώμης, χωρίς να χρειαστεί να εγκαταστήσει το λογισμικό RapidMiner. Στην πορεία διαπιστώθηκε ότι η φάση 3 είναι πιο αποτελεσματική να γίνει στην java χρησιμοποιώντας όμως τη φάσης 3 του RapidMiner με λίγες τροποποιήσεις.

Η τροποποίηση που έγινε είναι η αφαίρεση των operators «read document» και «process documents» και ως εκ τούτου και των operators που είναι ενσωματωμένοι στον «process documents». Αντί για αυτούς τους operators εισάγαμε τον «Read arff» ο οποίος διαβάζει αρχεία arff γνωστό από την βιβλιοθήκη μηχανικής μάθησης Weka. Επίσης ένας νέος operator είναι ο «Set Role» στον οποίο επιλέγουμε το label. Το παρακάτω project το δημιουργούμε μέσα στο πρόγραμμα java με την βοήθεια της βιβλιοθήκης του RapidMiner που την εισάγουμε στο πρόγραμμά μας.



Εικόνα 32:Νέα φάση 3

Αξίζει να σημειωθεί πως σε αυτήν την εφαρμογή δημιουργούμε την φάση 2 στην java για καλύτερα αποτελέσματα. Πολλές φορές αν απλά το εισάγαμε από το αρχείο που δημιουργήθηκε από το rapidminer δεν λειτουργούσε και έβγαζε πολλά σφάλματα. Γι' αυτό και το ενσωματώσαμε μέσα στον κώδικα και ξαναδημιουργήθηκε σε αυτόν. Το πρόγραμμα χωρίζεται σε τρεις κλάσεις:Textmining.class,choosefile.class και About.class. Η κύρια διεργασία γίνεται στη Textmining.class ενώ οι υπόλοιπες είναι βοηθητικές. Σε επόμενη ενότητα θα παρουσιαστούν εικόνες της εφαρμογής και θα αναλυθούν οι διεργασίες που γίνονται παράλληλα με τις εικόνες.

Εν κατακλείδι, σε αυτήν την εφαρμογή που δημιουργήθηκε στα πλαίσια της παρούσας διπλωματικής εργασίας ,διαβάζει και δημιουργεί τις φάσεις της παραπάνω ενότητας του Rapidminer σε μια εφαρμογή φιλική προς τον χρήστη και εύκολη στην χρήση.

4.7 Κώδικας Εκτέλεσης στο RapidMiner

4.7.1 Xml – φάσης 1η

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<process version="5.1.012">
 <context>
  <input/>
  <output/>
  <macros/>
 </context>
 <operator activated="true" class="process" compatibility="5.1.012" expanded="true" name="Process">
  <parameter key="logverbosity" value="init"/>
  <parameter key="random_seed" value="2001"/>
  <parameter key="send_mail" value="never"/>
  <parameter key="notification_email" value=""/>
  <parameter key="process_duration_for_mail" value="30"/>
  <parameter key="encoding" value="UTF-8"/>
  <parameter key="parallelize_main_process" value="false"/>
  <process expanded="true" height="476" width="547">
   <operator activated="true" class="read_excel" compatibility="5.1.012" expanded="true" height="60"
name="Read Excel" width="90" x="112" y="210">
    <parameter key="excel_file" value="C:\Users\elenious\Desktop\RapidMiner\diavouleusi.xls"/>
    <parameter key="sheet_number" value="1"/>
    <parameter key="imported_cell_range" value="G1:G1362"/>
    <parameter key="first_row_as_names" value="false"/>
    <list key="annotations">
     <parameter key="0" value="Name"/>
    </list>
    <parameter key="date_format" value=""/>
```

```
      <parameter key="time_zone" value="SYSTEM"/>
      <parameter key="locale" value="English (United States)"/>
      <list key="data_set_meta_data_information">
        <parameter key="0" value="Σχόλιο.true.polynominal.attribute"/>
      </list>
      <parameter key="read_not_matching_values_as_missings" value="true"/>
      <parameter key="datamanagement" value="double_array"/>
    </operator>
    <operator activated="false" class="select_attributes" compatibility="5.1.012" expanded="true"
height="76" name="Select Attributes" width="90" x="179" y="390">
      <parameter key="attribute_filter_type" value="single"/>
      <parameter key="attribute" value="comments"/>
      <parameter key="attributes" value=""/>
      <parameter key="use_except_expression" value="false"/>
      <parameter key="value_type" value="attribute_value"/>
      <parameter key="use_value_type_exception" value="false"/>
      <parameter key="except_value_type" value="time"/>
      <parameter key="block_type" value="attribute_block"/>
      <parameter key="use_block_type_exception" value="false"/>
      <parameter key="except_block_type" value="value_matrix_row_start"/>
      <parameter key="invert_selection" value="false"/>
      <parameter key="include_special_attributes" value="false"/>
    </operator>
    <operator activated="true" class="loop_examples" compatibility="5.1.012" expanded="true"
height="76" name="Loop Examples" width="90" x="447" y="210">
      <parameter key="iteration_macro" value="example"/>
      <parameter key="parallelize_example_process" value="false"/>
      <process expanded="true" height="607" width="761">
        <operator activated="true" class="filter_example_range" compatibility="5.1.012" expanded="true"
height="76" name="Filter Example Range" width="90" x="45" y="120">
          <parameter key="first_example" value="%{example}"/>
          <parameter key="last_example" value="%{example}"/>
          <parameter key="invert_filter" value="false"/>
        </operator>
        <operator activated="true" class="extract_macro" compatibility="5.1.012" expanded="true"
height="60" name="Extract Macro" width="90" x="179" y="30">
          <parameter key="macro" value="fileContent"/>
          <parameter key="macro_type" value="data_value"/>
          <parameter key="statistics" value="average"/>
          <parameter key="attribute_name" value="Σχόλιο"/>
          <parameter key="example_index" value="1"/>
        </operator>
        <operator activated="true" class="text:create_document" compatibility="5.1.003"
expanded="true" height="60" name="Create Document" width="90" x="313" y="30">
          <parameter key="text" value="%{fileContent}"/>
          <parameter key="add label" value="false"/>
          <parameter key="label_type" value="nominal"/>
        </operator>
```

```
      <operator activated="true" class="text:write_document" compatibility="5.1.003" expanded="true"
height="60" name="Write Document" width="90" x="447" y="30">
        <parameter key="file" value="C:\Users\elenious\Desktop\sxolia\comments_%{example}.txt"/>
        <parameter key="overwrite" value="true"/>
      </operator>
      <connect from_port="example set" to_op="Filter Example Range" to_port="example set input"/>
      <connect from_op="Filter Example Range" from_port="example set output" to_op="Extract
Macro" to_port="example set"/>
      <connect from_op="Filter Example Range" from_port="original" to_port="example set"/>
      <connect from_op="Create Document" from_port="output" to_op="Write Document"
to_port="document"/>
      <portSpacing port="source_example set" spacing="0"/>
      <portSpacing port="sink_example set" spacing="108"/>
      <portSpacing port="sink_output 1" spacing="0"/>
     </process>
   </operator>
   <connect from_op="Read Excel" from_port="output" to_op="Loop Examples" to_port="example
set"/>
   <connect from_op="Loop Examples" from_port="example set" to_port="result 1"/>
   <portSpacing port="source_input 1" spacing="0"/>
   <portSpacing port="sink_result 1" spacing="0"/>
   <portSpacing port="sink_result 2" spacing="0"/>
  </process>
 </operator>
</process>
```

4.7.2 Xml – φάσης 2η

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<process version="5.1.012">
  <context>
    <input/>
    <output/>
    <macros/>
  </context>
  <operator activated="true" class="process" compatibility="5.1.012" expanded="true"
name="Process">
    <parameter key="logverbosity" value="init"/>
    <parameter key="random_seed" value="2001"/>
    <parameter key="send_mail" value="never"/>
    <parameter key="notification_email" value=""/>
    <parameter key="process_duration_for_mail" value="30"/>
    <parameter key="encoding" value="SYSTEM"/>
    <parameter key="parallelize_main_process" value="true"/>
    <process expanded="true">
```

```
    <operator activated="true" class="text:process_document_from_file"
compatibility="5.1.003" expanded="true" name="Process Documents from Files">
      <list key="text_directories">
        <parameter key="Positive" value="C:/Users/elenious/Desktop/RapidMiner/positive"/>
        <parameter key="Negative" value="C:/Users/elenious/Desktop/RapidMiner/negative"/>
      </list>
      <parameter key="file_pattern" value="*"/>
      <parameter key="extract_text_only" value="true"/>
      <parameter key="use_file_extension_as_type" value="true"/>
      <parameter key="content_type" value="txt"/>
      <parameter key="encoding" value="UTF-8"/>
      <parameter key="create_word_vector" value="true"/>
      <parameter key="vector_creation" value="TF-IDF"/>
      <parameter key="add_meta_information" value="true"/>
      <parameter key="keep_text" value="false"/>
      <parameter key="prune_method" value="absolute"/>
      <parameter key="prunde_below_percent" value="3.0"/>
      <parameter key="prune_above_percent" value="30.0"/>
      <parameter key="prune_below_absolute" value="2"/>
      <parameter key="prune_above_absolute" value="999"/>
      <parameter key="prune_below_rank" value="5.0"/>
      <parameter key="prune_above_rank" value="5.0"/>
      <parameter key="datamanagement" value="double_sparse_array"/>
      <parameter key="parallelize_vector_creation" value="true"/>
      <process expanded="true">
        <operator activated="true" class="text:tokenize" compatibility="5.1.003" expanded="true"
name="Tokenize">
          <parameter key="mode" value="non letters"/>
          <parameter key="characters" value=".:"/>
          <parameter key="language" value="English"/>
        </operator>
        <operator activated="true" class="text:replace_tokens" compatibility="5.1.003"
expanded="true" name="Replace Tokens">
          <list key="replace_dictionary">
            <parameter key="ά" value="a"/>
            <parameter key="α" value="a"/>
            <parameter key="β" value="b"/>
            <parameter key="γ" value="g"/>
            <parameter key="δ" value="d"/>
            <parameter key="ζ" value="z"/>
            <parameter key="η" value="h"/>
            <parameter key="ή" value="h"/>
            <parameter key="θ" value="th"/>
            <parameter key="ι" value="i"/>
            <parameter key="ί" value="i"/>
            <parameter key="κ" value="k"/>
            <parameter key="λ" value="l"/>
```

```
            <parameter key="μ" value="m"/>
            <parameter key="ν" value="n"/>
            <parameter key="ξ" value="ks"/>
            <parameter key="ο" value="o"/>
            <parameter key="ό" value="o"/>
            <parameter key="π" value="p"/>
            <parameter key="ρ" value="r"/>
            <parameter key="σ" value="s"/>
            <parameter key="τ" value="t"/>
            <parameter key="υ" value="u"/>
            <parameter key="φ" value="f"/>
            <parameter key="χ" value="x"/>
            <parameter key="ψ" value="ps"/>
            <parameter key="ω" value="o"/>
            <parameter key="ς" value="s"/>
            <parameter key="ώ" value="o"/>
            <parameter key="ύ" value="u"/>
            <parameter key="ε" value="e"/>
            <parameter key="έ" value="e"/>
          </list>
        </operator>
        <operator activated="true" class="text:filter_stopwords_dictionary"
compatibility="5.1.003" expanded="true" name="Filter Stopwords (Dictionary)">
          <parameter key="file" value="C:/Users/elenious/Desktop/RapidMiner/stopwords.txt"/>
          <parameter key="case_sensitive" value="false"/>
          <parameter key="encoding" value="UTF-8"/>
        </operator>
        <operator activated="true" class="text:generate_n_grams_terms" compatibility="5.1.003"
expanded="true" name="Generate n-Grams (Terms)">
          <parameter key="max_length" value="2"/>
        </operator>
        <operator activated="true" class="text:transform_cases" compatibility="5.1.003"
expanded="true" name="Transform Cases">
          <parameter key="transform_to" value="lower case"/>
        </operator>
        <connect from_port="document" to_op="Tokenize" to_port="document"/>
        <connect from_op="Tokenize" from_port="document" to_op="Filter Stopwords
(Dictionary)" to_port="document"/>
        <connect from_op="Replace Tokens" from_port="document" to_op="Transform Cases"
to_port="document"/>
        <connect from_op="Filter Stopwords (Dictionary)" from_port="document"
to_op="Replace Tokens" to_port="document"/>
        <connect from_op="Generate n-Grams (Terms)" from_port="document"
to_port="document 1"/>
        <connect from_op="Transform Cases" from_port="document" to_op="Generate n-Grams
(Terms)" to_port="document"/>
      </process>
```

```
    </operator>
    <operator activated="true" class="weight_by_svm" compatibility="5.1.012" expanded="true"
name="Weight by SVM">
      <parameter key="normalize_weights" value="true"/>
      <parameter key="sort_weights" value="true"/>
      <parameter key="sort_direction" value="ascending"/>
      <parameter key="C" value="0.0"/>
    </operator>
    <operator activated="true" class="select_by_weights" compatibility="5.1.012"
expanded="true" name="Select by Weights">
      <parameter key="weight_relation" value="top k"/>
      <parameter key="weight" value="1.0"/>
      <parameter key="k" value="1000"/>
      <parameter key="p" value="0.5"/>
      <parameter key="deselect_unknown" value="true"/>
      <parameter key="use_absolute_weights" value="true"/>
    </operator>
    <operator activated="true" class="x_validation" compatibility="5.1.012" expanded="true"
name="Validation">
      <parameter key="create_complete_model" value="false"/>
      <parameter key="average_performances_only" value="true"/>
      <parameter key="leave_one_out" value="false"/>
      <parameter key="number_of_validations" value="10"/>
      <parameter key="sampling_type" value="stratified sampling"/>
      <parameter key="use_local_random_seed" value="false"/>
      <parameter key="local_random_seed" value="1992"/>
      <parameter key="parallelize_training" value="true"/>
      <parameter key="parallelize_testing" value="true"/>
      <process expanded="true">
        <operator activated="true" class="naive_bayes" compatibility="5.1.012" expanded="true"
name="Naive Bayes">
          <parameter key="laplace_correction" value="true"/>
        </operator>
        <connect from_port="training" to_op="Naive Bayes" to_port="training set"/>
        <connect from_op="Naive Bayes" from_port="model" to_port="model"/>
      </process>
      <process expanded="true">
        <operator activated="true" class="apply_model" compatibility="5.1.012" expanded="true"
name="Apply Model">
          <list key="application_parameters"/>
          <parameter key="create_view" value="false"/>
        </operator>
        <operator activated="true" class="performance_binominal_classification"
compatibility="5.1.012" expanded="true" name="Performance">
          <parameter key="main_criterion" value="first"/>
          <parameter key="accuracy" value="true"/>
          <parameter key="classification_error" value="false"/>
```

```
          <parameter key="kappa" value="false"/>
          <parameter key="AUC (optimistic)" value="false"/>
          <parameter key="AUC" value="false"/>
          <parameter key="AUC (pessimistic)" value="false"/>
          <parameter key="precision" value="false"/>
          <parameter key="recall" value="false"/>
          <parameter key="lift" value="false"/>
          <parameter key="fallout" value="false"/>
          <parameter key="f_measure" value="false"/>
          <parameter key="false_positive" value="false"/>
          <parameter key="false_negative" value="false"/>
          <parameter key="true_positive" value="false"/>
          <parameter key="true_negative" value="false"/>
          <parameter key="sensitivity" value="false"/>
          <parameter key="specificity" value="false"/>
          <parameter key="youden" value="false"/>
          <parameter key="positive_predictive_value" value="false"/>
          <parameter key="negative_predictive_value" value="false"/>
          <parameter key="psep" value="false"/>
          <parameter key="skip_undefined_labels" value="true"/>
          <parameter key="use_example_weights" value="true"/>
        </operator>
        <connect from_port="model" to_op="Apply Model" to_port="model"/>
        <connect from_port="test set" to_op="Apply Model" to_port="unlabelled data"/>
        <connect from_op="Apply Model" from_port="labelled data" to_op="Performance"
to_port="labelled data"/>
        <connect from_op="Performance" from_port="performance" to_port="averagable 1"/>
      </process>
    </operator>
    <operator activated="true" class="select_attributes" compatibility="5.1.012"
expanded="true" name="Select Attributes">
      <parameter key="attribute_filter_type" value="subset"/>
      <parameter key="attribute" value=""/>
      <parameter key="attributes" value="metadata_date|metadata_file|metadata_path|"/>
      <parameter key="use_except_expression" value="false"/>
      <parameter key="value_type" value="attribute_value"/>
      <parameter key="use_value_type_exception" value="false"/>
      <parameter key="except_value_type" value="time"/>
      <parameter key="block_type" value="attribute_block"/>
      <parameter key="use_block_type_exception" value="false"/>
      <parameter key="except_block_type" value="value_matrix_row_start"/>
      <parameter key="invert_selection" value="true"/>
      <parameter key="include_special_attributes" value="true"/>
    </operator>
    <operator activated="true" class="write_model" compatibility="5.1.012" expanded="true"
name="Write Model">
```

```
      <parameter key="model_file"
value="C:/Users/elenious/Desktop/RapidMiner/naivebayes.mod"/>
      <parameter key="overwrite_existing_file" value="true"/>
      <parameter key="output_type" value="Binary"/>
    </operator>
    <operator activated="true" class="write_arff" compatibility="5.1.012" expanded="true"
name="Write Arff">
      <parameter key="example_set_file"
value="C:/Users/elenious/Desktop/RapidMiner/training_arff.arff"/>
      <parameter key="encoding" value="SYSTEM"/>
    </operator>
    <connect from_op="Process Documents from Files" from_port="example set"
to_op="Weight by SVM" to_port="example set"/>
    <connect from_op="Weight by SVM" from_port="weights" to_op="Select by Weights"
to_port="weights"/>
    <connect from_op="Weight by SVM" from_port="example set" to_op="Select by Weights"
to_port="example set input"/>
    <connect from_op="Select by Weights" from_port="example set output" to_op="Validation"
to_port="training"/>
    <connect from_op="Validation" from_port="model" to_op="Write Model" to_port="input"/>
    <connect from_op="Validation" from_port="training" to_op="Select Attributes"
to_port="example set input"/>
    <connect from_op="Validation" from_port="averagable 1" to_port="result 1"/>
    <connect from_op="Select Attributes" from_port="example set output" to_op="Write Arff"
to_port="input"/>
  </process>
 </operator>
</process>
```

4.7.3 Xml – φάσης 3η

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<process version="5.1.008">
  <context>
    <input/>
    <output/>
    <macros/>
  </context>
  <operator activated="true" class="process" compatibility="5.1.008" expanded="true"
name="Process">
    <parameter key="encoding" value="UTF-8"/>
    <process expanded="true" height="404" width="809">
      <operator activated="true" class="text:read_document" compatibility="5.1.002"
expanded="true" height="60" name="Read Document" width="90" x="45" y="210">
        <parameter key="file" value="C:\Users\elenious\Desktop\sxolia\comments_1005.txt"/>
        <parameter key="encoding" value="UTF-8"/>
      </operator>
      <operator activated="true" class="text:process_documents" compatibility="5.1.002"
expanded="true" height="94" name="Process Documents" width="90" x="246" y="255">
        <process expanded="true" height="438" width="1004">
          <operator activated="true" class="text:tokenize" compatibility="5.1.002" expanded="true"
height="60" name="Tokenize (2)" width="90" x="45" y="30"/>
          <operator activated="true" class="text:filter_stopwords_dictionary"
compatibility="5.1.002" expanded="true" height="60" name="Filter Stopwords (2)" width="90"
x="179" y="75">
            <parameter key="file"
value="C:\Users\elenious\Desktop\diplomatiki\proj\stopwords.txt"/>
            <parameter key="encoding" value="UTF-8"/>
          </operator>
          <operator activated="true" class="text:replace_tokens" compatibility="5.1.002"
expanded="true" height="60" name="Replace Tokens (2)" width="90" x="313" y="120">
            <list key="replace_dictionary">
              <parameter key="α" value="a"/>
              <parameter key="ά" value="a"/>
              <parameter key="β" value="b"/>
              <parameter key="γ" value="g"/>
              <parameter key="δ" value="d"/>
              <parameter key="ε" value="e"/>
              <parameter key="έ" value="e"/>
              <parameter key="ζ" value="z"/>
              <parameter key="η" value="h"/>
              <parameter key="ή" value="h"/>
              <parameter key="θ" value="th"/>
              <parameter key="ι" value="i"/>
```

<parameter key="ί" value="i"/>
<parameter key="κ" value="k"/>
<parameter key="λ" value="l"/>
<parameter key="μ" value="m"/>
<parameter key="ν" value="n"/>
<parameter key="ξ" value="ks"/>
<parameter key="ο" value="o"/>
<parameter key="ό" value="o"/>
<parameter key="π" value="p"/>
<parameter key="ρ" value="r"/>
<parameter key="σ" value="s"/>
<parameter key="τ" value="t"/>
<parameter key="υ" value="u"/>
<parameter key="φ" value="f"/>
<parameter key="χ" value="x"/>
<parameter key="ψ" value="ps"/>
<parameter key="ω" value="w"/>
</list>
</operator>
<operator activated="true" class="text:generate_n_grams_terms" compatibility="5.1.002" expanded="true" height="60" name="Generate n-Grams (2)" width="90" x="514" y="120">
<parameter key="max_length" value="3"/>
</operator>
<operator activated="true" class="text:transform_cases" compatibility="5.1.002" expanded="true" height="60" name="Transform Cases" width="90" x="648" y="30"/>
<connect from_port="document" to_op="Tokenize (2)" to_port="document"/>
<connect from_op="Tokenize (2)" from_port="document" to_op="Filter Stopwords (2)" to_port="document"/>
<connect from_op="Filter Stopwords (2)" from_port="document" to_op="Replace Tokens (2)" to_port="document"/>
<connect from_op="Replace Tokens (2)" from_port="document" to_op="Generate n-Grams (2)" to_port="document"/>
<connect from_op="Generate n-Grams (2)" from_port="document" to_op="Transform Cases" to_port="document"/>
<connect from_op="Transform Cases" from_port="document" to_port="document 1"/>
<portSpacing port="source_document" spacing="0"/>
<portSpacing port="sink_document 1" spacing="0"/>
<portSpacing port="sink_document 2" spacing="0"/>
</process>
</operator>
<operator activated="true" class="read_model" compatibility="5.1.008" expanded="true" height="60" name="Read Model" width="90" x="246" y="75">
<parameter key="model_file" value="C:\Users\elenious\Desktop\diplomatiki\proj\naive_bayes"/>
</operator>
<operator activated="true" class="apply_model" compatibility="5.1.008" expanded="true" height="76" name="Apply Model" width="90" x="447" y="165">

```
      <list key="application_parameters"/>
    </operator>
    <connect from_op="Read Document" from_port="output" to_op="Process Documents"
to_port="documents 1"/>
    <connect from_op="Process Documents" from_port="example set" to_op="Apply Model"
to_port="unlabelled data"/>
    <connect from_op="Process Documents" from_port="word list" to_port="result 1"/>
    <connect from_op="Read Model" from_port="output" to_op="Apply Model"
to_port="model"/>
    <connect from_op="Apply Model" from_port="labelled data" to_port="result 2"/>
    <portSpacing port="source_input 1" spacing="0"/>
    <portSpacing port="sink_result 1" spacing="0"/>
    <portSpacing port="sink_result 2" spacing="0"/>
    <portSpacing port="sink_result 3" spacing="0"/>
   </process>
  </operator>
</process>
```

4.8 Κώδικας Εκτέλεσης στη Java

4.8.1 Κλάση Textmining.class

```
import com.rapidminer.operator.features.selection.AttributeWeightSelection;
import com.rapidminer.operator.features.weighting.SVMWeighting;
import com.rapidminer.operator.io.ArffExampleSetWriter;
import com.rapidminer.operator.io.ModelWriter;
import com.rapidminer.operator.learner.bayes.NaiveBayes;
import com.rapidminer.operator.performance.BinominalClassificationPerformanceEvaluator;
import com.rapidminer.operator.text.io.FileDocumentInputOperator;
import com.rapidminer.operator.text.io.tokenizer.StringTokenizerOperator;
import com.rapidminer.operator.text.io.tokenizer.TermNGramGeneratorOperator;
import com.rapidminer.operator.text.io.transformer.CaseTransformationOperator;
import com.rapidminer.operator.text.io.transformer.TokenReplaceOperator;
import com.rapidminer.operator.text.io.wordfilter.StopwordFilterOperator;
import com.rapidminer.operator.validation.XValidation;
import com.rapidminer.Process;
import com.rapidminer.RapidMiner;
import com.rapidminer.RapidMiner.ExecutionMode;
import com.rapidminer.example.Example;
import com.rapidminer.operator.OperatorCreationException;
import com.rapidminer.operator.OperatorException;
import com.rapidminer.operator.ExecutionUnit;
```

```
import com.rapidminer.operator.IOContainer;
import com.rapidminer.operator.ModelApplier;
import com.rapidminer.operator.Operator;
import com.rapidminer.operator.io.ModelLoader;
import com.rapidminer.operator.io.ArffExampleSource;
import com.rapidminer.tools.OperatorService;
import java.io.*;
import com.rapidminer.example.ExampleSet;
import com.rapidminer.operator.preprocessing.filter.ChangeAttributeRole;
import java.util.LinkedList;
import java.util.List;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import weka.core.FastVector;
import weka.core.Instances;
import weka.core.Attribute;
import com.rapidminer.operator.preprocessing.filter.attributes.AttributeFilter;
import java.net.InetAddress;

public class TextMining extends javax.swing.JFrame {

    static String str="text";
    private static String computername="";
    private static String cname[]=null;
        File x;

  /* Δημιουργία NewJFrame */
  public TextMining() {
    initComponents();
    //Καλούμε την μέθοδο CentralizedFrame() η οποία υλοποιείται παρακάτω
    //και βάζει το JFrame στο κέντρο της οθόνης
    CentralizedFrame();
  }


  @SuppressWarnings("unchecked")
  // <editor-fold defaultstate="collapsed" desc="Generated Code">
  private void initComponents() {

    jPanel1 = new javax.swing.JPanel();
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    Training = new javax.swing.JButton();
    excel = new javax.swing.JButton();
    jLabel4 = new javax.swing.JLabel();
    jLabel6 = new javax.swing.JLabel();
```

```
jLabel7 = new javax.swing.JLabel();
classification = new javax.swing.JButton();
jLabel8 = new javax.swing.JLabel();
sxolio = new javax.swing.JTextField();
jLabel9 = new javax.swing.JLabel();
Classification = new javax.swing.JButton();
about = new javax.swing.JButton();
jSeparator1 = new javax.swing.JSeparator();
jSeparator2 = new javax.swing.JSeparator();
jLabel5 = new javax.swing.JLabel();
Exit = new javax.swing.JButton();
Clear = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

jPanel1.setBackground(new java.awt.Color(204, 204, 204));
jPanel1.setBorder(new javax.swing.border.MatteBorder(null));
jPanel1.setDebugGraphicsOptions(javax.swing.DebugGraphics.NONE_OPTION);

jLabel1.setText("Stage1");

jLabel2.setText("Select an excel file:");

jLabel3.setText("Stage 2");

Training.setText("Training");
Training.addActionListener(new java.awt.event.ActionListener() {
   public void actionPerformed(java.awt.event.ActionEvent evt) {
      TrainingActionPerformed(evt);
   }
});

excel.setText("jButton1");
excel.addActionListener(new java.awt.event.ActionListener() {
   public void actionPerformed(java.awt.event.ActionEvent evt) {
      excelActionPerformed(evt);
   }
});

jLabel4.setText("For training push the button:");

jLabel6.setText("Stage 3");

jLabel7.setText("Select a txt file with the comment:");

classification.setText("jButton1");
classification.addActionListener(new java.awt.event.ActionListener() {
   public void actionPerformed(java.awt.event.ActionEvent evt) {
```

```
            classificationActionPerformed(evt);
        }
    });

    jLabel8.setText("or wtite the comment here (in greek):");

    sxolio.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            sxolioActionPerformed(evt);
        }
    });

    jLabel9.setText("and press the button for classification");

    Classification.setText("Classification");
    Classification.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            ClassificationActionPerformed(evt);
        }
    });

    about.setForeground(new java.awt.Color(0, 0, 102));
    about.setText("Edit");
    about.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            aboutActionPerformed(evt);
        }
    });

    jLabel5.setIcon(new javax.swing.ImageIcon(getClass().getResource("/textmining/PADGETS.jpg"))); //
NOI18N

    Exit.setText("Exit");
    Exit.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            ExitActionPerformed(evt);
        }
    });

    Clear.setText("Clear");
    Clear.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            ClearActionPerformed(evt);
        }
    });

    javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
    jPanel1.setLayout(jPanel1Layout);
```

```
      jPanel1Layout.setHorizontalGroup(
         jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
         .addGroup(jPanel1Layout.createSequentialGroup()
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING,
false)
               .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanel1Layout.createSequentialGroup()
                  .addGap(23, 23, 23)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                     .addGroup(jPanel1Layout.createSequentialGroup()
                        .addComponent(jLabel4)
                        .addGap(18, 18, 18)
                        .addComponent(Training))
                     .addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                           .addComponent(jLabel7)
                           .addComponent(jLabel8)
                           .addComponent(jLabel9))
                        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                           .addGroup(jPanel1Layout.createSequentialGroup()
                              .addGap(18, 18, 18)
                              .addComponent(classification, javax.swing.GroupLayout.PREFERRED_SIZE, 30,
javax.swing.GroupLayout.PREFERRED_SIZE))
                           .addGroup(jPanel1Layout.createSequentialGroup()
                              .addGap(40, 40, 40)
                              .addComponent(Classification))
                           .addGroup(jPanel1Layout.createSequentialGroup()
                              .addGap(18, 18, 18)
                              .addComponent(sxolio, javax.swing.GroupLayout.PREFERRED_SIZE, 227,
javax.swing.GroupLayout.PREFERRED_SIZE)))))
                  .addGap(18, 18, 18))
               .addGroup(jPanel1Layout.createSequentialGroup()
                  .addGap(28, 28, 28)
                  .addComponent(jLabel2)
                  .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 79,
Short.MAX_VALUE)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                     .addComponent(jLabel1)
                     .addGroup(jPanel1Layout.createSequentialGroup()
                        .addComponent(excel, javax.swing.GroupLayout.PREFERRED_SIZE, 30,
javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addGap(172, 172, 172)
                        .addComponent(jLabel5)))
```

```
                .addGap(46, 46, 46)))
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        .addComponent(jSeparator1, javax.swing.GroupLayout.DEFAULT_SIZE, 521, Short.MAX_VALUE)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jSeparator2, javax.swing.GroupLayout.DEFAULT_SIZE, 501, Short.MAX_VALUE)
            .addContainerGap())
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(198, 198, 198)
            .addComponent(jLabel3)
            .addContainerGap(286, Short.MAX_VALUE))
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(202, 202, 202)
            .addComponent(jLabel6)
            .addContainerGap(282, Short.MAX_VALUE))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()
            .addContainerGap(314, Short.MAX_VALUE)
            .addComponent(Exit)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(Clear)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(about)
            .addGap(36, 36, 36))
    );
    jPanel1Layout.setVerticalGroup(
        jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addContainerGap()
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                .addGroup(jPanel1Layout.createSequentialGroup()
                    .addComponent(jLabel1)
                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                        .addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE, 69,
Short.MAX_VALUE)
                        .addComponent(excel)))
                .addComponent(jLabel5))
            .addGap(18, 18, 18)
            .addComponent(jSeparator1, javax.swing.GroupLayout.PREFERRED_SIZE, 10,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(4, 4, 4)
            .addComponent(jLabel3)
            .addGap(18, 18, 18)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel4)
                .addComponent(Training))
            .addGap(45, 45, 45)
```

```
            .addComponent(jSeparator2, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)
            .addComponent(jLabel6)
            .addGap(18, 18, 18)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
              .addComponent(jLabel7)
              .addComponent(classification))
            .addGap(35, 35, 35)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
              .addComponent(jLabel8)
              .addComponent(sxolio, javax.swing.GroupLayout.PREFERRED_SIZE, 38,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(23, 23, 23)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
              .addComponent(jLabel9)
              .addComponent(Classification))
            .addGap(51, 51, 51)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
              .addComponent(about)
              .addComponent(Clear)
              .addComponent(Exit))
            .addContainerGap())
      );

      javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
      getContentPane().setLayout(layout);
      layout.setHorizontalGroup(
         layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
         .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE, 497,
javax.swing.GroupLayout.PREFERRED_SIZE)
      );
      layout.setVerticalGroup(
         layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
         .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
      );

      pack();
   }// </editor-fold>

   private void TrainingActionPerformed(java.awt.event.ActionEvent evt) {
  //Με το που πατιέται το κουμπί training δημιουργούμε το trainig.rmp με την βοήθεια της βιβλιοθήκης
RM
          try {

              RapidMiner.setExecutionMode(ExecutionMode.COMMAND_LINE);
              RapidMiner.init();
```

```
//βρίσκουμε το hostname του υπολογιστή και κάνουμε split στην - για να πάρουμε μόνο το name
            computername=InetAddress.getLocalHost().getHostName();
            cname = computername.split("-");

            // Create a process
            final Process process = new Process();

            // Set the parameters of process
            process.getRootOperator().setParameter("parallelize_main_process", "true");

            //create writemodel with the method createModelWriter()
             final Operator writemodel = createModelWriter();

            //create Process documents from files
             final FileDocumentInputOperator processdocumentfromfiles = OperatorService
                   .createOperator(FileDocumentInputOperator.class);

             //set textlist of Process documents from files
              List<String[]> textList = new LinkedList<String[]>();
              textList.add(new String[]
{"Positive","C:/Users/"+cname[0]+"/Desktop/RapidMiner/positive"});
              textList.add(new String[]
{"Negative","C:/Users/"+cname[0]+"/Desktop/RapidMiner/negative"});
               // Set the parameters of Process documents from files
              processdocumentfromfiles.setListParameter("text_directories", textList);
              processdocumentfromfiles.setParameter("encoding", "UTF-8");
              processdocumentfromfiles.setParameter("vector_creation", "TF-IDF");
              processdocumentfromfiles.setParameter("prune_method", "absolute");
              processdocumentfromfiles.setParameter("prune_below_absolute", "2");
              processdocumentfromfiles.setParameter("prune_above_absolute", "999");
              processdocumentfromfiles.setParameter("parallelize_vector_creation","true");

            //create X validation
            final XValidation xvalidation = OperatorService
                        .createOperator(XValidation.class);

            // Set the parameters of X validation
            xvalidation.setParameter(XValidation.PARAMETER_NUMBER_OF_VALIDATIONS,
                        Integer.valueOf(10).toString());
             xvalidation.setParameter("parallelize_training", "true");
             xvalidation.setParameter("parallelize_testing", "true");

            //Operators inside the validation
            final NaiveBayes naivebayes = OperatorService.createOperator(NaiveBayes.class);

            final Operator modelApplier = OperatorService
                        .createOperator(ModelApplier.class);
```

```
final BinominalClassificationPerformanceEvaluator performance = OperatorService
        .createOperator(BinominalClassificationPerformanceEvaluator.class);

//back to main process
//create the svm operator
 final SVMWeighting svm = OperatorService
        .createOperator(SVMWeighting.class);
 //create the weightbysvm operator
 final AttributeWeightSelection weight = OperatorService
        .createOperator(AttributeWeightSelection.class);
 // Set the parameters
 weight.setParameter("weight_relation","top k");
 weight.setParameter("k","1000");

 //create select attribute operator
  final AttributeFilter select_attributes = OperatorService
     .createOperator(AttributeFilter.class);

 // Set the parameters
 select_attributes.setParameter("invert_selection", "true");
 select_attributes.setParameter("include_special_attributes", "true");
 select_attributes.setParameter("attribute_filter_type", "subset");
 select_attributes.setParameter("include_special_attributes","true");

select_attributes.setParameter("attributes","metadata_date|metadata_file|metadata_path|");

 //create write arff operator
 final ArffExampleSetWriter Write_arff = OperatorService
        .createOperator(ArffExampleSetWriter.class);

// Set the parameters

Write_arff.setParameter("example_set_file","C:/Users/"+cname[0]+"/Desktop/RapidMiner/training_arff
.arff");


// add operators to the main process and connect them

 process.getRootOperator().getSubprocess(0).addOperator(processdocumentfromfiles);
 process.getRootOperator().getSubprocess(0).addOperator(svm);
 connect(processdocumentfromfiles,"example set",svm,"example set");
 process.getRootOperator().getSubprocess(0).addOperator(weight);
 connect(svm,"example set",weight,"example set input");
 connect(svm,"weights",weight,"weights");
 process.getRootOperator().getSubprocess(0).addOperator(xvalidation);
 process.getRootOperator().getSubprocess(0).addOperator(select_attributes);
 connect(weight,"example set output", xvalidation, "training");
 process.getRootOperator().getSubprocess(0).addOperator(writemodel);
```

```
 process.getRootOperator().getSubprocess(0).addOperator(Write_arff);
 connect(xvalidation,"model", writemodel, "input");
 connect(xvalidation,"training", select_attributes, "example set input");
 connect(select_attributes, "example set output",Write_arff,"input");
 connect(xvalidation,"averagable 1",process.getRootOperator().getSubprocess(0),"result 1");

//operators  inside the process documents from files
//create tokenize operator
final StringTokenizerOperator tokenize=OperatorService
        .createOperator(StringTokenizerOperator.class);
//create replace tokens operator
final TokenReplaceOperator replacetokens=OperatorService
        .createOperator(TokenReplaceOperator.class);
// Set the parameters
List<String[]> replaceList = new LinkedList<String[]>();

  replaceList.add(new String[] {"ά","a"});
  replaceList.add(new String[] {"α","a"});
  replaceList.add(new String[] {"β","b"});
  replaceList.add(new String[] {"γ","g"});
  replaceList.add(new String[] {"δ","d"});
  replaceList.add(new String[] {"ζ","z"});
  replaceList.add(new String[] {"η","h"});
  replaceList.add(new String[] {"ή","h"});
  replaceList.add(new String[] {"θ","th"});
  replaceList.add(new String[] {"ι","i"});
  replaceList.add(new String[] {"ί","i"});
  replaceList.add(new String[] {"κ","k"});
  replaceList.add(new String[] {"λ","l"});
  replaceList.add(new String[] {"μ","m"});
  replaceList.add(new String[] {"ν","n"});
  replaceList.add(new String[] {"ξ","ks"});
  replaceList.add(new String[] {"ο","o"});
  replaceList.add(new String[] {"ό","o"});
  replaceList.add(new String[] {"π","p"});
  replaceList.add(new String[] {"ρ","r"});
  replaceList.add(new String[] {"σ","s"});
  replaceList.add(new String[] {"τ","t"});
  replaceList.add(new String[] {"υ","u"});
  replaceList.add(new String[] {"φ","f"});
  replaceList.add(new String[] {"χ","x"});
  replaceList.add(new String[] {"ψ","ps"});
  replaceList.add(new String[] {"ω","o"});
  replaceList.add(new String[] {"ς","s"});
  replaceList.add(new String[] {"ώ","o"});
  replaceList.add(new String[] {"ύ","u"});
  replaceList.add(new String[] {"ε","e"});
  replaceList.add(new String[] {"έ","e"});
```

```
replacetokens.setListParameter("replace_dictionary", replaceList);

//create stopwords operator
final StopwordFilterOperator stopwords=OperatorService
        .createOperator(StopwordFilterOperator.class);

// Set the parameters of stopwords

stopwords.setParameter("file","C:/Users/"+cname[0]+"/Desktop/RapidMiner/stopwords.txt");
stopwords.setParameter("encoding", "UTF-8");

//create tranformcases operator
final CaseTransformationOperator tranformcases=OperatorService
        .createOperator(CaseTransformationOperator.class);

//create generarate n-grams operator
final TermNGramGeneratorOperator ngram=OperatorService
        .createOperator(TermNGramGeneratorOperator.class);
// Set the parameters
ngram.setParameter("max_length", "2");


// add operators to the process documents from files and connect them
processdocumentfromfiles.getSubprocess(0).addOperator(tokenize);
processdocumentfromfiles.getSubprocess(0).addOperator(replacetokens);
processdocumentfromfiles.getSubprocess(0).addOperator(stopwords);
processdocumentfromfiles.getSubprocess(0).addOperator(ngram);
processdocumentfromfiles.getSubprocess(0).addOperator(tranformcases);

connect(processdocumentfromfiles.getSubprocess(0),"document",tokenize,"document");
connect(tokenize,"document",stopwords,"document");
connect(stopwords,"document",replacetokens,"document");
connect(replacetokens,"document",tranformcases,"document");
connect(tranformcases,"document",ngram,"document");
connect(ngram,"document",processdocumentfromfiles.getSubprocess(0), "document 1");

// xvalidation
// training part of xvalidation
xvalidation.getSubprocess(0).addOperator(naivebayes);

// create connection within training process: from left to right ...
connect(xvalidation.getSubprocess(0), "training", naivebayes,"training set");
connect(naivebayes,"model",xvalidation.getSubprocess(0), "model");

// testing part of xvalidation
xvalidation.getSubprocess(1).addOperator(modelApplier);
xvalidation.getSubprocess(1).addOperator(performance);
```

```
            // create connection within testing process: from left to right ...
            connect(xvalidation.getSubprocess(1), "model", modelApplier, "model");
            connect(xvalidation.getSubprocess(1), "test set", modelApplier,"unlabelled data");
            connect(modelApplier, "labelled data", performance, "labelled data");
            connect(performance, "performance", xvalidation.getSubprocess(1),"averagable 1");

             x=new File ("C:/Users/"+cname[0]+"/Desktop/RapidMiner/training.rmp");
             //save the process
             process.save(x);
             // perform process

             IOContainer ioResult=process.run();

             JOptionPane.showMessageDialog(null, ioResult.toString(), "ACCURACY",
JOptionPane.INFORMATION_MESSAGE);

      } catch (Exception ex) {
         Logger.getLogger(TextMining.class.getName()).log(Level.SEVERE, null, ex);
      }


   }

   private void excelActionPerformed(java.awt.event.ActionEvent evt) {
      //Όταν πατηθεί το κουμπί excel δημιουργείται αντικείμενο choosefile με παράμετρο excel
      //και καλείται η main
      choosefile cf1=new choosefile("excel");
      cf1.main();

   }

   private void classificationActionPerformed(java.awt.event.ActionEvent evt) {
   //Όταν πατηθεί το κουμπί classification δημιουργείται αντικείμενο choosefile με παράμετρο
classification
      //και καλείται η main
      choosefile cf2=new choosefile("classification");
      cf2.main();
      //βοηθητική μεταβλητή
      str="file";
   }

   private void aboutActionPerformed(java.awt.event.ActionEvent evt) {
     //Όταν πατηθεί το κουμπί classification δημιουργείται αντικείμενο About
      //και καλείται η main
      About ab=new About();
      ab.main();
```

```
    }

private void sxolioActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
}

private void ClassificationActionPerformed(java.awt.event.ActionEvent evt) {
    //Όταν πατηθεί το κουμπί classification ελέγχουμε την βοηθητική μας μεταβλητή
    //Αν είναι file έχει πατηθεί το κουμπί να επιλέξει αρχείο έτοιμο
    //η μέθοδος classification θα καλεστεί και απο τις δύο if
    if(str=="file"){
        str="text";
        classification();
    }
    //αν είναι text γράφει ο χρήστης στο Jtextfield
    else if(str=="text"){
          try {
             //βρίσκουμε το όνομα του υπολογιστή
             computername=InetAddress.getLocalHost().getHostName();
             cname = computername.split("-");

             //Δημιουργούμε ένα καινούργιο txt αρχείο που θα του βάλουμε αυτό που έγραψε ο χρήστης

               Writer out = null;
               FileOutputStream outputFile = new
FileOutputStream("C:/Users/"+cname[0]+"/Desktop/RapidMiner/testing.txt");
               out = new OutputStreamWriter(outputFile,"UTF8");

               //μετατρέπουμε τα ελληνικά σε αγγλικά
               String rtokens=sxolio.getText();

                    rtokens=rtokens.replace('ά','a');
                    rtokens=rtokens.replace('α','a');
                    rtokens=rtokens.replace ('β','b');
                    rtokens=rtokens.replace ('γ','g');
                    rtokens=rtokens.replace ('δ','d');
                    rtokens=rtokens.replace ('ζ','z');
                    rtokens=rtokens.replace ('η','h');
                    rtokens=rtokens.replace ('ή','h');
                    rtokens=rtokens.replace ("θ","th");
                    rtokens=rtokens.replace ('ι','i');
                    rtokens=rtokens.replace ('ί','i');
                    rtokens=rtokens.replace ('κ','k');
                    rtokens=rtokens.replace ('λ','l');
                    rtokens=rtokens.replace ('μ','m');
                    rtokens=rtokens.replace ('ν','n');
                    rtokens=rtokens.replace ("ξ","ks");
                    rtokens=rtokens.replace ('ο','o');
```

```
                rtokens=rtokens.replace  ('ό','o');
                rtokens=rtokens.replace ('π','p');
                rtokens=rtokens.replace ('ρ','r');
                rtokens=rtokens.replace ('σ','s');
                rtokens=rtokens.replace ('τ','t');
                rtokens=rtokens.replace ('υ','u');
                rtokens=rtokens.replace ('φ','f');
                rtokens=rtokens.replace ('χ','x');
                rtokens=rtokens.replace ("ψ","ps");
                rtokens=rtokens.replace ('ω','o');
                rtokens=rtokens.replace ('ς','s');
                rtokens=rtokens.replace ('ώ','o');
                rtokens=rtokens.replace ('ύ','u');
                rtokens=rtokens.replace ('ε','e');
                rtokens=rtokens.replace ('έ','e');


                //κάνουμε n-grams
                String[] tokens = rtokens.split(" ");
                String extraContent = "";
                for (int i=0;i<tokens.length-1;i++){
                   extraContent += tokens[i] + "_" + tokens[i+1] + " ";
                }
                out.write(rtokens + " " + extraContent);
                out.close();

    //διαβάζει το arff που βγάζει το training(training_arff.arff) και ένα νέο test.txt και φτιάχνει το test με μορφή arff
            File f = new File("C:/Users/"+cname[0]+"/Desktop/RapidMiner/test.arff");
               if (f.exists()) {
                f.delete();
                }

            BufferedReader reader = null;
            reader = new BufferedReader(new FileReader("C:/Users/"+cname[0]+"/Desktop/RapidMiner/training_arff.arff"));
            Instances data = new Instances(reader);
            Attribute [] attributes;
            FastVector attributeInfo = new FastVector();
            attributes =  new Attribute[data.numAttributes()];
            for (int i = 0; i < attributes.length; i++) {
               attributes[i] =  data.attribute(i);
               attributeInfo.addElement(attributes[i]);
            }
            saveToFile("C:/Users/"+cname[0]+"/Desktop/RapidMiner/test.arff", "@relation " + attributes[attributes.length - 4].name() + "\n\n", true);
            for (int i = 0; i < attributes.length; i++) {
```

```
                saveToFile("C:/Users/"+cname[0]+"/Desktop/RapidMiner/test.arff", "@attribute " +
attributes[i].name() + "\treal" + "\n", true);
            }
            saveToFile("C:/Users/"+cname[0]+"/Desktop/RapidMiner/test.arff", "\n@data" + "\n", true);
            String dataFile = "C:/Users/"+cname[0]+"/Desktop/RapidMiner/testing.txt";//filepath;
            BufferedReader br = new BufferedReader(new FileReader(dataFile));
            String s = null;
            String contents = "";
            while (null != (s = br.readLine())) {
               contents += s + "\n";
            }
            String results = "";
            for (int j = 0; j < attributes.length; j++) {
               if (contents.contains(attributes[j].name())) {
                  results += "1" + ",";
               } else {
                  results += "0" + ",";
               }
            }
            saveToFile("C:/Users/"+cname[0]+"/Desktop/RapidMiner/test.arff", results.substring(0,
results.length() - 1) + "\n", true);

            out.close();
            //καλείται η μέθοδος classification
            classification();


         } catch (IOException ex) {
            Logger.getLogger(TextMining.class.getName()).log(Level.SEVERE, null, ex);
         }


   }
}

private void ExitActionPerformed(java.awt.event.ActionEvent evt) {
    //κουμπί exit κάνει dispose το παράθυρο
   this.dispose();
}

private void ClearActionPerformed(java.awt.event.ActionEvent evt) {
   //κουμπί clear κάνει το jtextfield άδειο πάλι
   sxolio.setText(null);
}

//μέθοδο savetofile
private void saveToFile(String fileName, String content, boolean append) {
      try {
```

```
            File outputFile = new File(fileName);
            FileWriter out = new FileWriter(outputFile, append);
            out.write(content);
            out.close();
        } catch (Exception ex) {
            ex.printStackTrace();
        }
    }
    //ακολουθούν οι μέθοδοι connect που συνδέουμε τους operator μεταξύ τους και υπάρχουν 3 τύποι
    private static void connect(Operator from, String fromPortName,
                    Operator to, String toPortName) {
                from.getOutputPorts().getPortByName(fromPortName).connectTo(
                        to.getInputPorts().getPortByName(toPortName));
        }

    private static void connect(ExecutionUnit from, String fromPortName,
                    Operator to, String toPortName) {
                from.getInnerSources().getPortByName(fromPortName).connectTo(
                        to.getInputPorts().getPortByName(toPortName));
        }

    private static void connect(Operator from, String fromPortName,
                    ExecutionUnit to, String toPortName) {
                from.getOutputPorts().getPortByName(fromPortName).connectTo(
                        to.getInnerSinks().getPortByName(toPortName));
        }

    // δημιουργία του operator modelwriter
    private static Operator createModelWriter() throws Exception {

                Operator writemodel = OperatorService
                        .createOperator(ModelWriter.class);
                // Set the parameters for model writer
                writemodel.setParameter("model_file",
    "C:/Users/"+cname[0]+"/Desktop/RapidMiner/naivebayes.mod");

                writemodel.setParameter("output_type", "Binary");

                return writemodel;
        }
    //μέθοδο classification όπου δημιουργείται το classification.rmp και μας δίνει τα αποτελέσματα
    private static void classification(){

        try {
            //αποθήκευση του ονόματος του υπολογιστή
             computername=InetAddress.getLocalHost().getHostName();
             cname = computername.split("-");
```

```
        // init rapidminer
        RapidMiner.setExecutionMode(ExecutionMode.COMMAND_LINE);
        RapidMiner.init();

        // Create a process
        final Process process = new Process();

        // Set the parameters of process
       process.getRootOperator().setParameter("parallelize_main_process", "true");

        // all operators from "left to right

        // create read model
        final ModelLoader readmodel = OperatorService
                .createOperator(ModelLoader.class);
        // Set the parameters of read model

readmodel.setParameter("model_file","C:/Users/"+cname[0]+"/Desktop/RapidMiner/naivebayes.mod");
        // create set role
        final Operator setrole = OperatorService
                .createOperator(ChangeAttributeRole.class);
         // Set the parameters
        setrole.setParameter("name","label");
        setrole.setParameter("target_role","label");

        //create apply model
        final Operator modelApplier = OperatorService
                .createOperator(ModelApplier.class);
        //create read arff
        final ArffExampleSource readarff = OperatorService
                .createOperator(ArffExampleSource.class);
        // Set the parameters
        readarff.setParameter("data_file","C:/Users/"+cname[0]+"/Desktop/RapidMiner/test.arff");

        // add operators to the main process and connect them
        process.getRootOperator().getSubprocess(0).addOperator(readarff);
        process.getRootOperator().getSubprocess(0).addOperator(readmodel);
        process.getRootOperator().getSubprocess(0).addOperator(setrole);
        process.getRootOperator().getSubprocess(0).addOperator(modelApplier);

        connect(readarff,"output",setrole,"example set input");
        connect(readmodel,"output",modelApplier,"model");
        connect(setrole,"example set output",modelApplier,"unlabelled data");
        connect(modelApplier,"labelled data",process.getRootOperator().getSubprocess(0),"result 1");

        //αποθήκευση του προγράμματος
        File x=new File ("C:/Users/"+cname[0]+"/Desktop/RapidMiner/Classification.rmp");
```

```
            process.save(x);

            // perform process
            //run το πρόγραμμα
            process.run();
            IOContainer ioResult=process.run();

           //εμφάνιση μόνο το predict label
           ExampleSet resultSet = (ExampleSet)ioResult.getElementAt(0);
           Example example=resultSet.getExample(0);

           String resultString = example.getValueAsString(example.getAttributes().getPredictedLabel());
           JOptionPane.showMessageDialog(null,"The comment is "+ resultString, "Classification",
JOptionPane.INFORMATION_MESSAGE);

      } catch (OperatorException ex) {
         Logger.getLogger(TextMining.class.getName()).log(Level.SEVERE, null, ex);
      } catch (IOException ex) {
         Logger.getLogger(TextMining.class.getName()).log(Level.SEVERE, null, ex);
      } catch (OperatorCreationException ex) {
         Logger.getLogger(TextMining.class.getName()).log(Level.SEVERE, null, ex);
      }
 }
//main
public static void main(String args[]) {
      /* Set the Nimbus look and feel */
      //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
      /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
       * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
       */
      try {
         for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
               javax.swing.UIManager.setLookAndFeel(info.getClassName());
               break;
            }
         }
      } catch (ClassNotFoundException ex) {
         java.util.logging.Logger.getLogger(TextMining.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
      } catch (InstantiationException ex) {
         java.util.logging.Logger.getLogger(TextMining.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
      } catch (IllegalAccessException ex) {
         java.util.logging.Logger.getLogger(TextMining.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
```

```
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {
            java.util.logging.Logger.getLogger(TextMining.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {

            public void run() {
                JOptionPane.showMessageDialog(null,"Hello user,\nBefore you use this software you must
copy \nthe folder RapidMiner to your Desktop \n", "Attention", JOptionPane.INFORMATION_MESSAGE);
                new TextMining().setVisible(true);

            }
        });
    }
//CentralizedFrame μέθοδο να βάζει το jframe στο κέντρο της οθόνης
private void CentralizedFrame(){
        java.awt.Dimension screenSizes = java.awt.Toolkit.getDefaultToolkit().getScreenSize() ;
        int wScreen = screenSizes.width ;
        int hScreen = screenSizes.height ;
        int w = this.getWidth() ;
        int h = this.getHeight() ;
        this.setLocation((wScreen-w)/2, (hScreen-h)/2) ;
        this.setAlwaysOnTop(false);
        this.setResizable(false);
        this.setFocusable(true);
}
    // Variables declaration - do not modify
    private javax.swing.JButton Classification;
    private javax.swing.JButton Clear;
    private javax.swing.JButton Exit;
    private javax.swing.JButton Training;
    private javax.swing.JButton about;
    private javax.swing.JButton classification;
    private javax.swing.JButton excel;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JLabel jLabel5;
    private javax.swing.JLabel jLabel6;
    private javax.swing.JLabel jLabel7;
    private javax.swing.JLabel jLabel8;
    private javax.swing.JLabel jLabel9;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JSeparator jSeparator1;
```

```
    private javax.swing.JSeparator jSeparator2;
    private javax.swing.JTextField sxolio;
    // End of variables declaration
}
```

4.8.2 Κλάση choosefile.class

```
import java.net.UnknownHostException;
import javax.swing.JFileChooser;
import com.rapidminer.Process;
import com.rapidminer.RapidMiner;
import com.rapidminer.RapidMiner.ExecutionMode;
import com.rapidminer.operator.OperatorException;
import com.rapidminer.operator.ExecutionUnit;
import com.rapidminer.operator.IOContainer;
import com.rapidminer.operator.Operator;
import com.rapidminer.tools.XMLException;
import java.io.*;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import weka.core.FastVector;
import weka.core.Instances;
import weka.core.Attribute;
import java.net.InetAddress;
import java.nio.channels.FileChannel;


public class choosefile extends javax.swing.JFrame {


    static String x="C:/Users/elenious/Desktop/RapidMiner/excel.rmp";
    private static String idiotita="";
    private static String computername="";
    private static String cname[]=null;
    private String str="";

   //κατασκευαστής που καλείται με την ιδιότητα, ανάλογα αν πάτης το κουμπί
   //για excel η το κουμπι για να πατηθεί το txt αρχείο
    public choosefile(String id) {
        this.idiotita=id;
        initComponents();
        CentralizedFrame();
    }


    @SuppressWarnings("unchecked")
```

```
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jFileChooser1 = new javax.swing.JFileChooser();

        setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);

        jFileChooser1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jFileChooser1ActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jFileChooser1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap())
        );
        layout.setVerticalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jFileChooser1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        );

        pack();
    }// </editor-fold>

    private void jFileChooser1ActionPerformed(java.awt.event.ActionEvent evt) {
        try {
    //βρίσκουμε το hostname του υπολογιστή και κάνουμε split στην - για να πάρουμε μόνο το name
            computername=InetAddress.getLocalHost().getHostName();
            cname = computername.split("-");

        //αν πατηθεί το open ανάλογα με την ιδιότητα γίνοται και ξεχωριστά πράγματα
             if (JFileChooser.APPROVE_SELECTION.equals(evt.getActionCommand())) {
                if(idiotita=="excel"){
                 try {
                    //παίρνουμε το path του αρχείου που επιλέχθηκε
                   File  selectedPfile = jFileChooser1.getSelectedFile();
                   String tmpString = selectedPfile.getAbsolutePath();
```

```
            tmpString  = tmpString.replace( '\\', '/' );

            //το μεταφέρουμε εκεί που θέλουμε και το μετονομάζουμε
            File source = new File(tmpString);
            File destination = new File("C:/Users/"+cname[0]+"/Desktop/RapidMiner/diavouleusi.xls");
            copyFile(source, destination);
            this.dispose();
           //τρέχουμε το έτοιμο process excel.rmp που το διαβάζουμε με την readFileAsString
            RapidMiner.setExecutionMode(ExecutionMode.COMMAND_LINE);
            RapidMiner.init();
            Process process = new Process(readFileAsString(x));
            IOContainer ioResult = process.run();

            //εμφάνιση αποτελέσματος
            JOptionPane.showMessageDialog(null, ioResult.getElementAt(0).toString(), "Comments
saved...", JOptionPane.INFORMATION_MESSAGE);


          } catch (OperatorException ex) {
             Logger.getLogger(choosefile.class.getName()).log(Level.SEVERE, null, ex);
          } catch (IOException ex) {
             Logger.getLogger(choosefile.class.getName()).log(Level.SEVERE, null, ex);
          } catch (XMLException ex) {
             Logger.getLogger(choosefile.class.getName()).log(Level.SEVERE, null, ex);
          }
        }
        else if(idiotita=="classification"){
            FileInputStream fstream = null;
           try {
              //παίρνουμε το path του αρχείου που επιλέχθηκε
              File  selectedPfile = jFileChooser1.getSelectedFile();
              String tmpString = selectedPfile.getAbsolutePath();
              tmpString  = tmpString.replace( '\\', '/' );
              fstream = new FileInputStream(tmpString);

               //Δημιουργούμε ένα καινούργιο txt αρχείο που θα του βάλουμε το τροποποιημένο μας
txt file
              DataInputStream in = new DataInputStream(fstream);
              BufferedReader br2 = new BufferedReader(new InputStreamReader(in));
              Writer out = null;
              FileOutputStream outputFile = new
FileOutputStream("C:/Users/"+cname[0]+"/Desktop/RapidMiner/testing.txt");
              out = new OutputStreamWriter(outputFile,"UTF8");

              String strLine;

              //Καθώς το διαβάζουμε κάνουμε replace tokens
              while ((strLine = br2.readLine()) != null)   {
```

```
                strLine=strLine.replace("α","a");
                strLine=strLine.replace("ά","a");
                strLine=strLine.replace("β","b");
                strLine=strLine.replace("γ","g");
                strLine=strLine.replace("δ","d");
                strLine=strLine.replace("ε","e");
                strLine=strLine.replace("έ","e");
                strLine=strLine.replace("ζ","z");
                strLine=strLine.replace("η","h");
                strLine=strLine.replace("ή","h");
                strLine=strLine.replace("θ","th");
                strLine=strLine.replace("ι","i");
                strLine=strLine.replace("ί","i");
                strLine=strLine.replace("κ","k");
                strLine=strLine.replace("λ","l");
                strLine=strLine.replace("μ","m");
                strLine=strLine.replace("ν","n");
                strLine=strLine.replace("ξ","ks");
                strLine=strLine.replace("ο","o");
                strLine=strLine.replace("ό","o");
                strLine=strLine.replace("π","p");
                strLine=strLine.replace("ρ","r");
                strLine=strLine.replace("σ","s");
                strLine=strLine.replace("τ","t");
                strLine=strLine.replace("υ","u");
                strLine=strLine.replace("ύ","u");
                strLine=strLine.replace("φ","f");
                strLine=strLine.replace("χ","x");
                strLine=strLine.replace("ψ","ps");
                strLine=strLine.replace("ω","w");
                strLine=strLine.replace("ώ","w");
                strLine=strLine.replace("ς","s");
                //και παράλληλα ngrams
                 String[] tokens = strLine.split(" ");
                 String extraContent = "";
                    for (int i=0;i<tokens.length-1;i++){
                       extraContent += tokens[i] + "_" + tokens[i+1] + " ";
                    }
                    out.write(strLine + " " + extraContent);
             }

            in.close();
            out.close();

    //διαβάζει το arff που βγάζει το training(training_arff.arff) και ένα νέο test.txt και φτιάχνει το test με μορφή arff
             File f = new File("C:/Users/"+cname[0]+"/Desktop/RapidMiner/test.arff");
```

```
                if (f.exists()) {
                   f.delete();
                }
                BufferedReader reader = null;
                reader = new BufferedReader(new
FileReader("C:/Users/"+cname[0]+"/Desktop/RapidMiner/training_arff.arff"));
                Instances data = new Instances(reader);
                Attribute [] attributes;
                FastVector attributeInfo = new FastVector();
                attributes =  new Attribute[data.numAttributes()];
                for (int i = 0; i < attributes.length; i++) {
                   attributes[i] =  data.attribute(i);
                   attributeInfo.addElement(attributes[i]);
                }

                saveToFile("C:/Users/"+cname[0]+"/Desktop/RapidMiner/test.arff", "@relation " +
attributes[attributes.length - 4].name() + "\n\n", true);
                for (int i = 0; i < attributes.length; i++) {
                  saveToFile("C:/Users/"+cname[0]+"/Desktop/RapidMiner/test.arff", "@attribute " +
attributes[i].name() + "\treal" + "\n", true);
                }

                saveToFile("C:/Users/"+cname[0]+"/Desktop/RapidMiner/test.arff", "\n@data" + "\n",
true);
                String dataFile = "C:/Users/"+cname[0]+"/Desktop/RapidMiner/testing.txt";//filepath;
                BufferedReader br = new BufferedReader(new FileReader(dataFile));
                String s = null;
                String contents = "";

                while (null != (s = br.readLine())) {
                   contents += s + "\n";
                }
                String results = "";
                for (int j = 0; j < attributes.length; j++) {

                   if (contents.contains(attributes[j].name())) {
                      results += "1" + ",";
                   } else {
                      results += "0" + ",";
                   }
                }
                saveToFile("C:/Users/"+cname[0]+"/Desktop/RapidMiner/test.arff", results.substring(0,
results.length() - 1) + "\n", true);

                this.dispose();

             } catch (IOException ex) {
                Logger.getLogger(choosefile.class.getName()).log(Level.SEVERE, null, ex);
```

```
            }
          }
        }
        if (JFileChooser.CANCEL_SELECTION.equals(evt.getActionCommand()))  {
              //αν πατηθεί το cancel κλείνει το παράθυρο
              this.dispose();
            }
      } catch (UnknownHostException ex) {
         Logger.getLogger(choosefile.class.getName()).log(Level.SEVERE, null, ex);
      }
   }

  //διαβάζει το xml αρχείο
   private static String readFileAsString(String filePath) throws java.io.IOException{
                StringBuffer fileData = new StringBuffer(1000);
                BufferedReader reader = new BufferedReader(new FileReader(filePath));
                char[] buf = new char[1024];
                int numRead=0;
                while((numRead=reader.read(buf)) != -1){
                   String readData = String.valueOf(buf, 0, numRead);
                   fileData.append(readData);
                   buf = new char[1024];
                }
                reader.close();
                return fileData.toString();
           }
//βάζει το αρχείο όπου θέλουμε εμείς
      public static void copyFile(File sourceFile, File destFile) throws IOException {

         if(!destFile.exists()) {
              destFile.createNewFile();
             }

             FileChannel source = null;
             FileChannel destination = null;
             try {
              source = new RandomAccessFile(sourceFile,"rw").getChannel();
              destination = new RandomAccessFile(destFile,"rw").getChannel();

              long position = 0;
              long count    = source.size();

              source.transferTo(position, count, destination);
             }
             finally {
              if(source != null) {
               source.close();
              }
```

```
            if(destination != null) {
             destination.close();
            }
           }
         }
//μέθοδο savetofile
  private void saveToFile(String fileName, String content, boolean append) {

      try {
         File outputFile = new File(fileName);
         FileWriter out = new FileWriter(outputFile, append);
         out.write(content);
         out.close();
      } catch (Exception ex) {
         ex.printStackTrace();
      }
  }
 //ακολουθούν οι μέθοδοι connect που συνδέουμε τους operator μεταξύ τους και υπάρχουν 3 τύποι
private static void connect(Operator from, String fromPortName,
                                Operator to, String toPortName) {
                        from.getOutputPorts().getPortByName(fromPortName).connectTo(
                                        to.getInputPorts().getPortByName(toPortName));
}


private static void connect(ExecutionUnit from, String fromPortName,
                                Operator to, String toPortName) {
                        from.getInnerSources().getPortByName(fromPortName).connectTo(
                                        to.getInputPorts().getPortByName(toPortName));
}


 private static void connect(Operator from, String fromPortName,
                                ExecutionUnit to, String toPortName) {
                        from.getOutputPorts().getPortByName(fromPortName).connectTo(
                                        to.getInnerSinks().getPortByName(toPortName));
}

//main
 public void main() {

      java.awt.EventQueue.invokeLater(new Runnable() {

         public void run() {
            new choosefile(idiotita).setVisible(true);

         }
      });
```

```
}
 //CentralizedFrame μέθοδο να βάζει το jframe στο κέντρο της οθόνης
   private void CentralizedFrame(){
      java.awt.Dimension screenSizes = java.awt.Toolkit.getDefaultToolkit().getScreenSize() ;
      int wScreen = screenSizes.width ;
      int hScreen = screenSizes.height ;
      int w = this.getWidth() ;
      int h = this.getHeight() ;
      this.setLocation((wScreen-w)/2, (hScreen-h)/2) ;
      this.setAlwaysOnTop(true);
      this.setResizable(false);
      this.setFocusable(true);
}
   // Variables declaration - do not modify
   private javax.swing.JFileChooser jFileChooser1;
   // End of variables declaration
}
```

4.8.3 Κλάση About.class

```
public class About extends javax.swing.JFrame {
//Αυτή η κλάση είναι ενημερωτική για εμάς
   public About() {
      initComponents();
      CentralizedFrame();
   }


   @SuppressWarnings("unchecked")
   // <editor-fold defaultstate="collapsed" desc="Generated Code">
   private void initComponents() {

      jPanel1 = new javax.swing.JPanel();
      ok = new javax.swing.JButton();
      jScrollPane1 = new javax.swing.JScrollPane();
      jTextArea1 = new javax.swing.JTextArea();

      setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

      jPanel1.setBackground(new java.awt.Color(204, 204, 204));

      ok.setText("OK");
      ok.addActionListener(new java.awt.event.ActionListener() {
         public void actionPerformed(java.awt.event.ActionEvent evt) {
            okActionPerformed(evt);
         }
      });
```

```
        jTextArea1.setColumns(20);
        jTextArea1.setRows(5);
        jTextArea1.setText("\n This software was developed at the University of the Aegean \n in the
department of Information and Communication Systems Engineering\n located on the Greek island of
Samos at town Karlobasi. \n\n This is a thesis of Helen Vourou and George Dimitrakis with supervision \n
of the assistant professors Charalabidis Ioannis,Loukis Euripidis and the\n lecturer Maragkoudakis
Emmanouil\n \n \n\n Thank you for using it!!!\n Enjoy !!!\n\n \n http://www.icsd.aegean.gr/");
        jScrollPane1.setViewportView(jTextArea1);

        javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
        jPanel1.setLayout(jPanel1Layout);
        jPanel1Layout.setHorizontalGroup(
            jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGap(220, 220, 220)
                .addComponent(ok)
                .addContainerGap(264, Short.MAX_VALUE))
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT_SIZE, 500, Short.MAX_VALUE)
                .addGap(21, 21, 21))
        );
        jPanel1Layout.setVerticalGroup(
            jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()
                .addGap(29, 29, 29)
                .addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT_SIZE, 282, Short.MAX_VALUE)
                .addGap(18, 18, 18)
                .addComponent(ok)
                .addGap(47, 47, 47))
        );

        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        );
        layout.setVerticalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        );

        pack();
    }// </editor-fold>
```

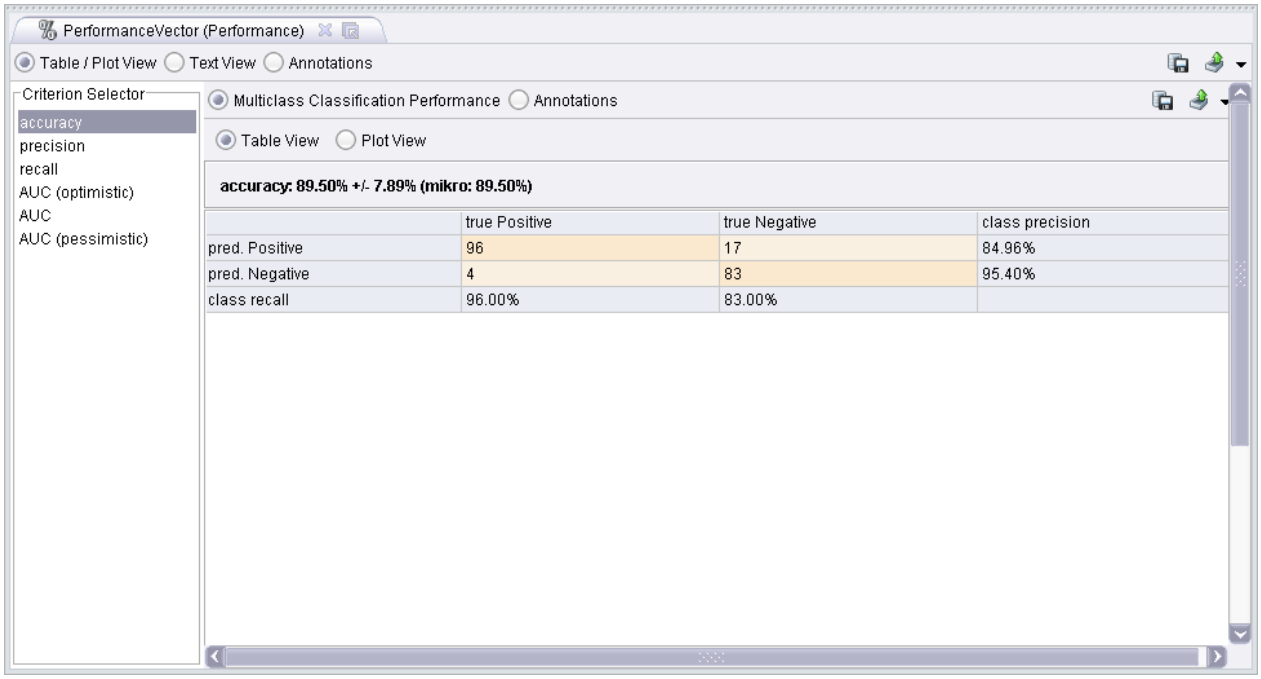
```
    private void okActionPerformed(java.awt.event.ActionEvent evt) {
        this.dispose();
    }


    public void main() {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
         * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {
            java.util.logging.Logger.getLogger(About.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
        } catch (InstantiationException ex) {
            java.util.logging.Logger.getLogger(About.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
        } catch (IllegalAccessException ex) {
            java.util.logging.Logger.getLogger(About.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {
            java.util.logging.Logger.getLogger(About.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {

            public void run() {
                new About().setVisible(true);
            }
        });
    }
//CentralizedFrame μέθοδο να βάζει το jframe στο κέντρο της οθόνης
private void CentralizedFrame(){
        java.awt.Dimension screenSizes = java.awt.Toolkit.getDefaultToolkit().getScreenSize() ;
        int wScreen = screenSizes.width ;
        int hScreen = screenSizes.height ;
```

```
        int w = this.getWidth() ;
        int h = this.getHeight() ;
        this.setLocation((wScreen-w)/2, (hScreen-h)/2) ;
        this.setAlwaysOnTop(true);
        this.setResizable(false);
        this.setFocusable(true);
    }
    // Variables declaration - do not modify
    private javax.swing.JPanel jPanel1;
    private javax.swing.JScrollPane jScrollPane1;
    private javax.swing.JTextArea jTextArea1;
    private javax.swing.JButton ok;
    // End of variables declaration
}
```

4.9 *Εκτέλεση και Αποτελέσματα*

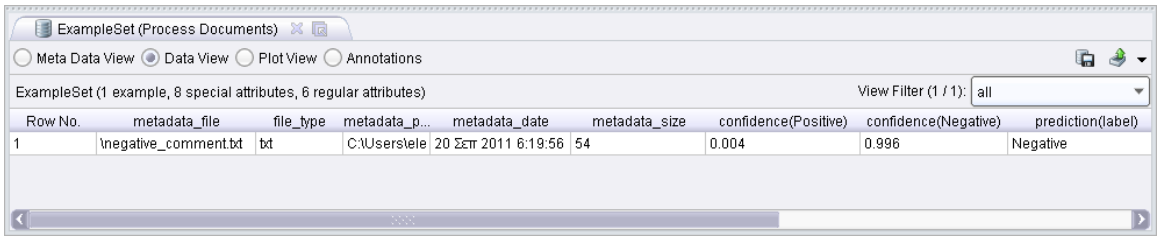
4.9.1 Αποτελέσματα στο RapidMiner

Παρακάτω παρατηρούμε το accuracy να είναι 89,50% το οποίο είναι το αποτέλεσμα του «Perfomance». Το 89,50% σημαίνει ότι υπάρχει πιθανότητα 89,50% να προβλέψει το ύφος του σχολίου σωστά.

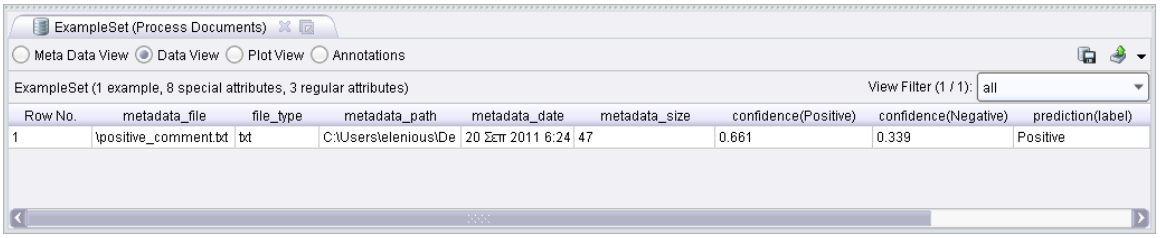


Εικόνα 33:Accuracy 89,50%

Παρακάτω παρουσιάζουμε δύο παραδείγματα εισαγωγής ενός θετικού και ενός αρνητικού σχολίου και την πρόβλεψη η οποία είναι σωστή στις συγκεκριμένες περίπτωσεις.



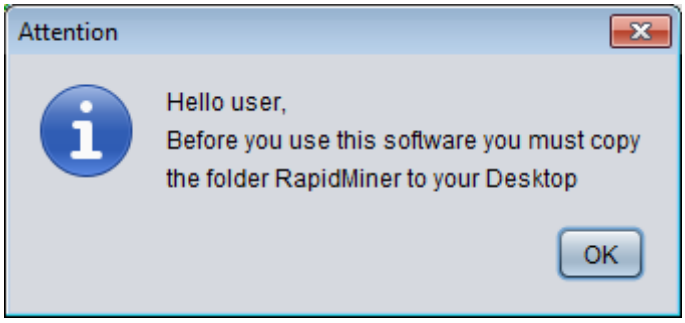
Εικόνα 34:Εισαγωγή θετικού σχολίου



Εικόνα 35:Εισαγωγή αρνητικού σχολίου

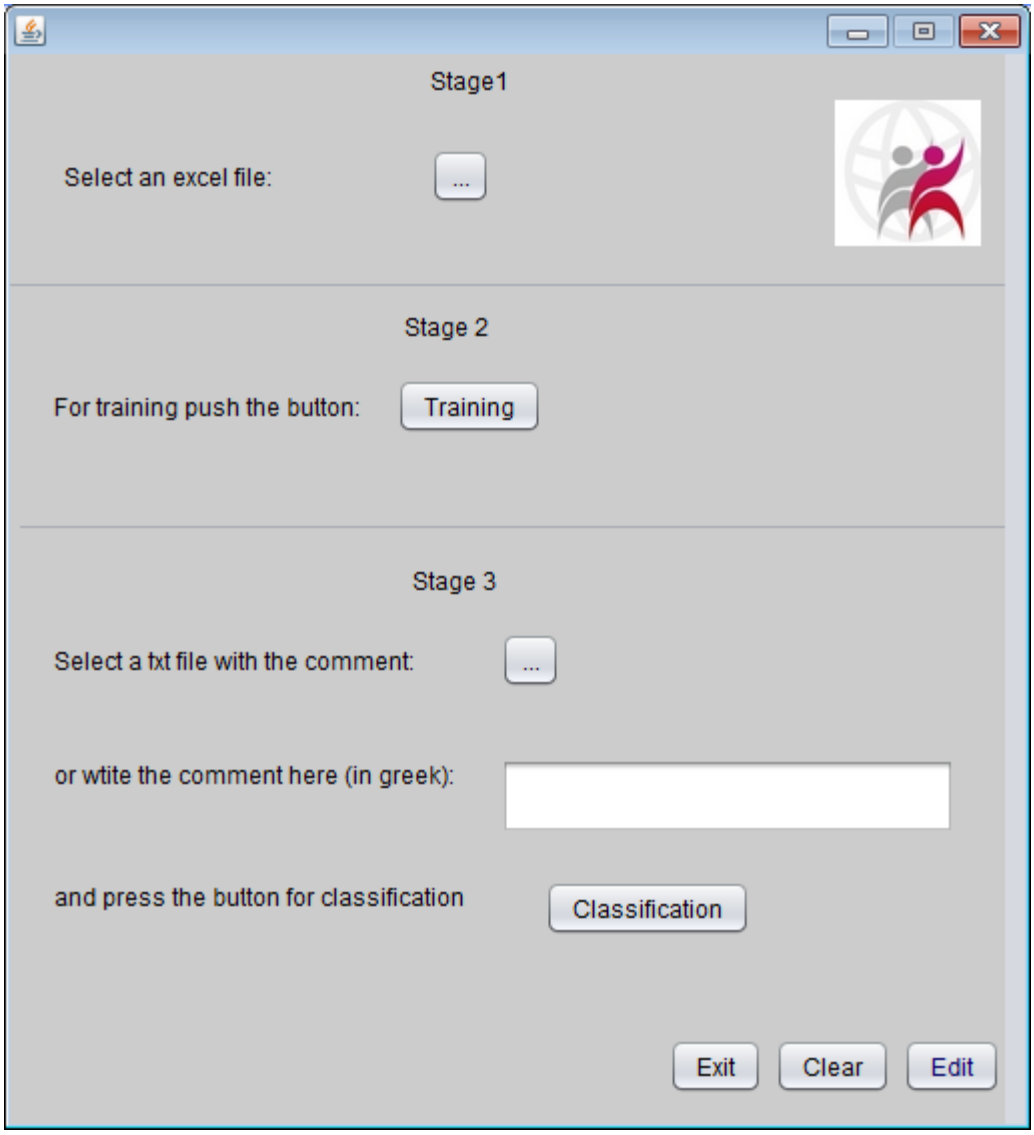
4.9.3 Αποτελέσματα εφαρμογής

Σε αυτή την ενότητα θα παρουσιαστεί η εφαρμογή που δημιουργήθηκε στα πλαίσια της παρούσας διπλωματικής εργασίας. Πρέπει να σημειωθεί ότι μαζί με την εφαρμογή υπάρχει ένας φάκελος με όνομα Rapidminer ,ο οποίος περιέχει αρχεία απαραίτητα για να «τρέξει» η εφαρμογή. Ο φάκελος αυτός θα πρέπει να αντιγραφεί στην επιφάνεια εργασίας του κάθε χρήστη. Αυτό έγινε με σκοπό η εφαρμογή να μπορεί να «τρέξει» σε όλους τους υπολογιστές που έχουν λογισμικό Windows. Οπότε με το πάτημα της εφαρμογής ενημερώνουμε τον χρήστη για τον συγκεκριμένο φάκελο, όπως φαίνεται παρακάτω.



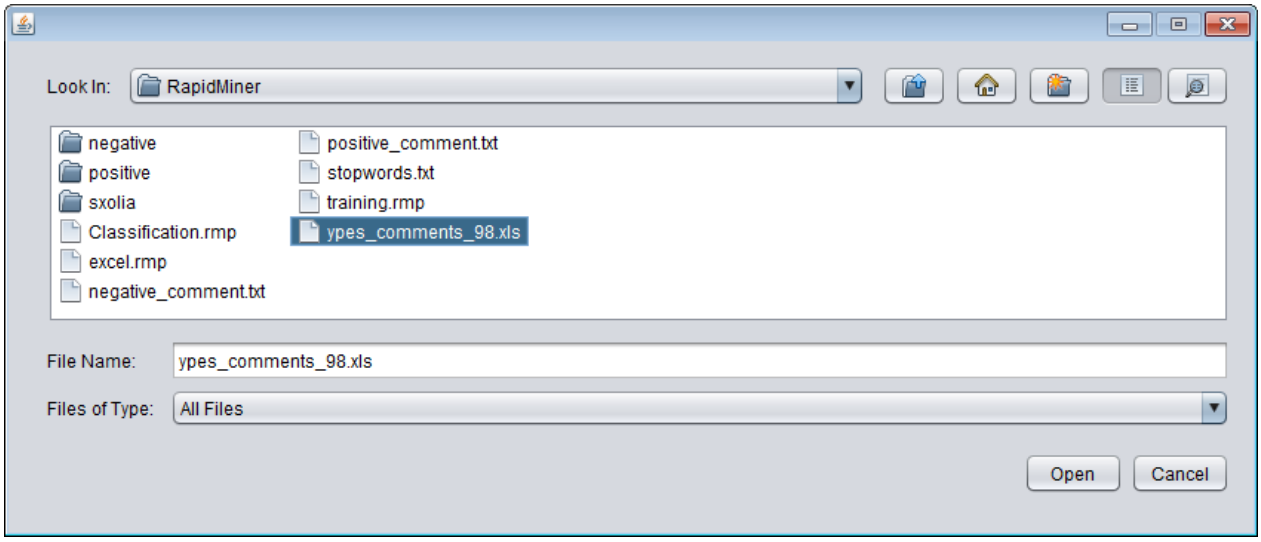
Εικόνα 36:Ενημερωτικό παράθυρο

Με το πάτημα του κουμπιού «Οκ» ο χρήστης είναι έτοιμος να λειτουργήσει την εφαρμογή. Στην επόμενη εικόνα παρουσιάζεται η κύρια εφαρμογή.



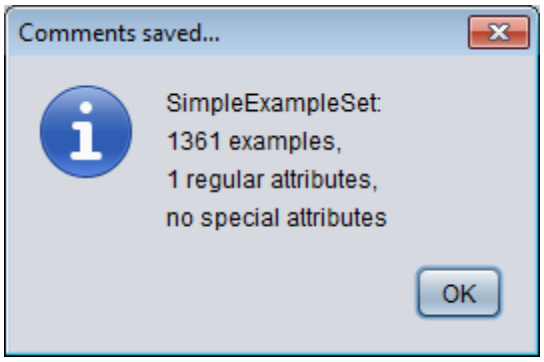
Εικόνα 37:Κύριο παράθυρο εφαρμογής

Όπως παρατηρούμε είναι χωρισμένο σε φάσεις, οι οποίες και είναι οι φάσεις που δημιουργήθηκαν και στο Rapidminer και αναλύονται στις ενότητες 4.3,4.4.4.5. Στην φάση 1 ο χρήστης έχει την δυνατότητα να εισάγει στο σύστημα ένα excel αρχείο το οποίο μπορεί να βρει ο χρήστης στην ιστοσελίδα opengov. Με το πάτημα του κουμπιού δίπλα στο «Select an excel file» εμφανίζεται στον χρήστη ένα παράθυρο επιλογής του αρχείου και με το πάτημα «Open» τρέχει στο background το αρχείο excel που δημιουργήσαμε στην φάση 1 του Rapidminer. Δηλαδή αποθηκεύονται σε ξεχωριστά txt αρχεία τα σχόλια της διαβούλευσης που βρίσκονται στο excel αρχείο που του εισάγαμε. Παρακάτω φαίνεται η επιλογή του excel αρχείου.



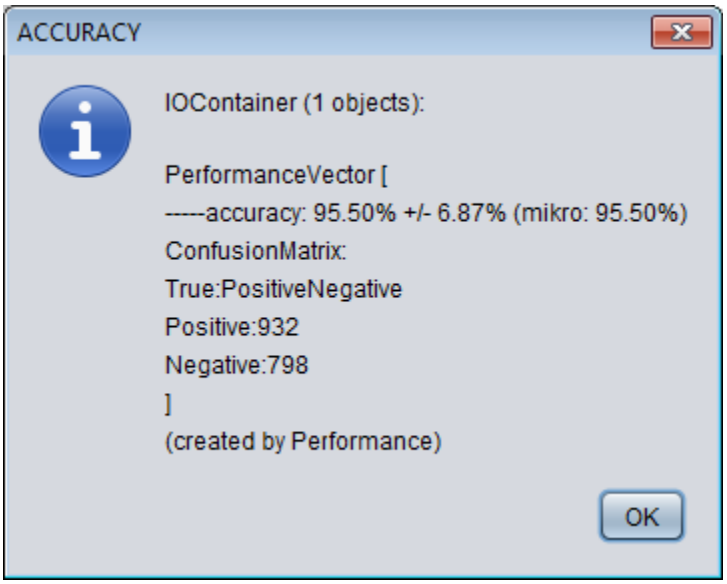
Εικόνα 38:Επιλογή excel αρχείου

Ως αποτέλεσμα το πρόγραμμα μας εμφανίζει ενημερωτικό παράθυρο για ον αριθμό των αρχείων που αποθηκεύτηκαν.



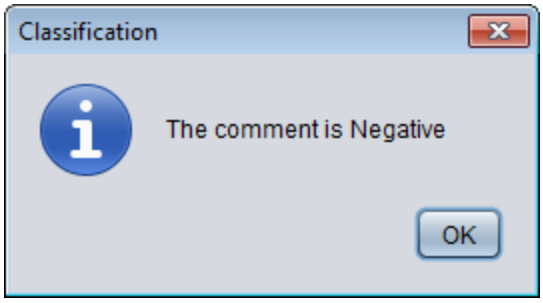
Εικόνα 39:Ενημέρωση αποθήκευσης αρχείων

Στην συνέχεια προχωράμε στη φάση 2, η οποία είναι το training και είναι πολύ βασική και απαραίτητη για την εφαρμογή μας. Έτσι με το πάτημα του κουμπιού «training» δημιουργείται το αρχείο rapidminer και παράλληλα τρέχει και μας δίνει το εξής αποτέλεσμα. Όπως παρατηρούμε παρακάτω έχουμε 95,50% accuracy καλύτερη από αυτή του Rapidminer.



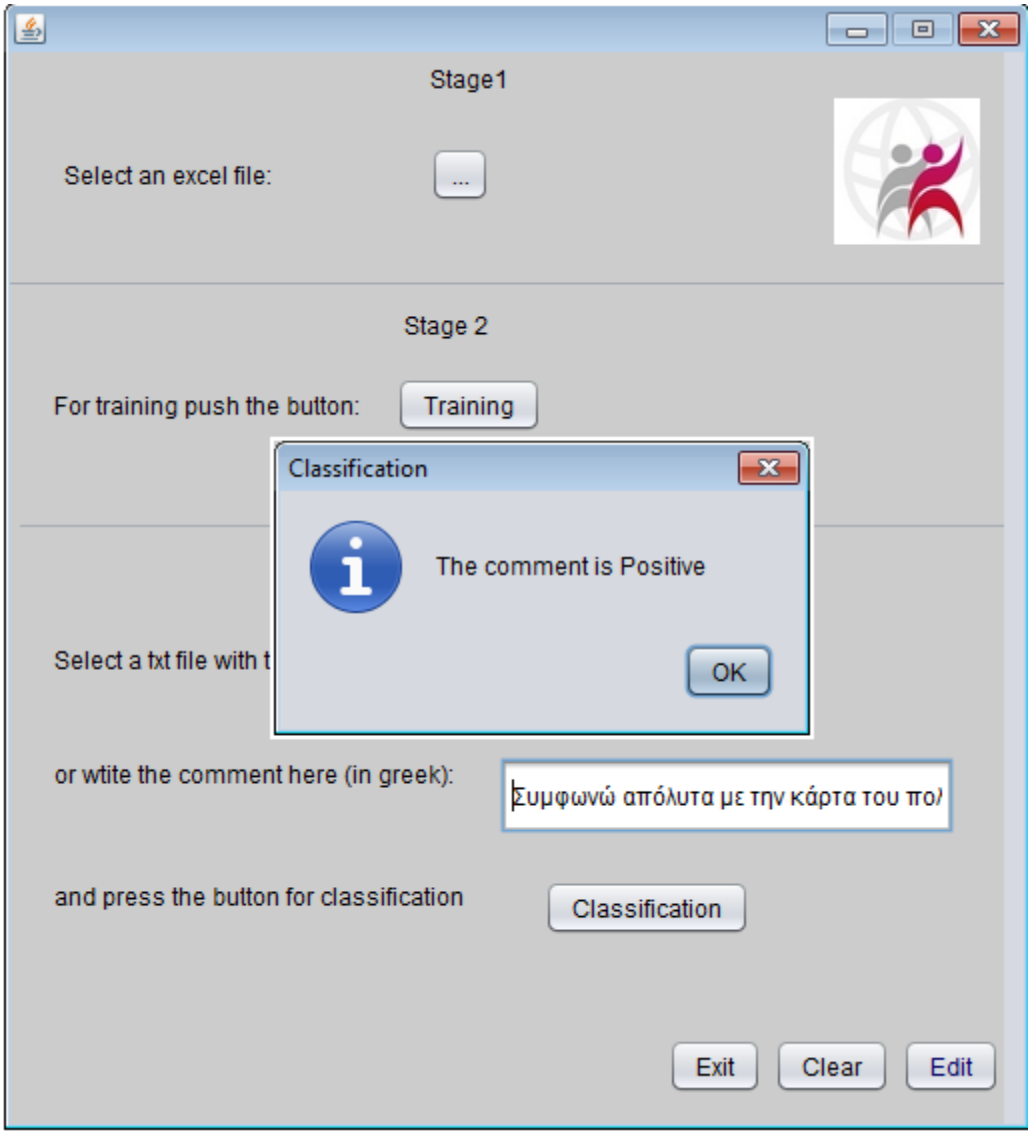
Εικόνα 40:Accuracy

Επόμενο βήμα είναι είτε να εισάγουμε ένα νέο σχόλιο επιλέγοντας το από τον υπολογιστή, είτε να εισάγουμε εμείς ένα σχόλιο στο πεδίο που δίνεται. Αμέσως μετά εμφανίζεται το αποτέλεσμα, δηλαδή η πρόβλεψη αν το σχόλιο είναι θετικού ύφους ή αρνητικού. Παρακάτω παρουσιάζονται δύο παραδείγματα. Αρχικά εισάγουμε ένα αρχείο με ένα αρνητικό σχόλιο και πατώντας το κουμπί «Classification» εμφανίζεται το παρακάτω αποτέλεσμα.



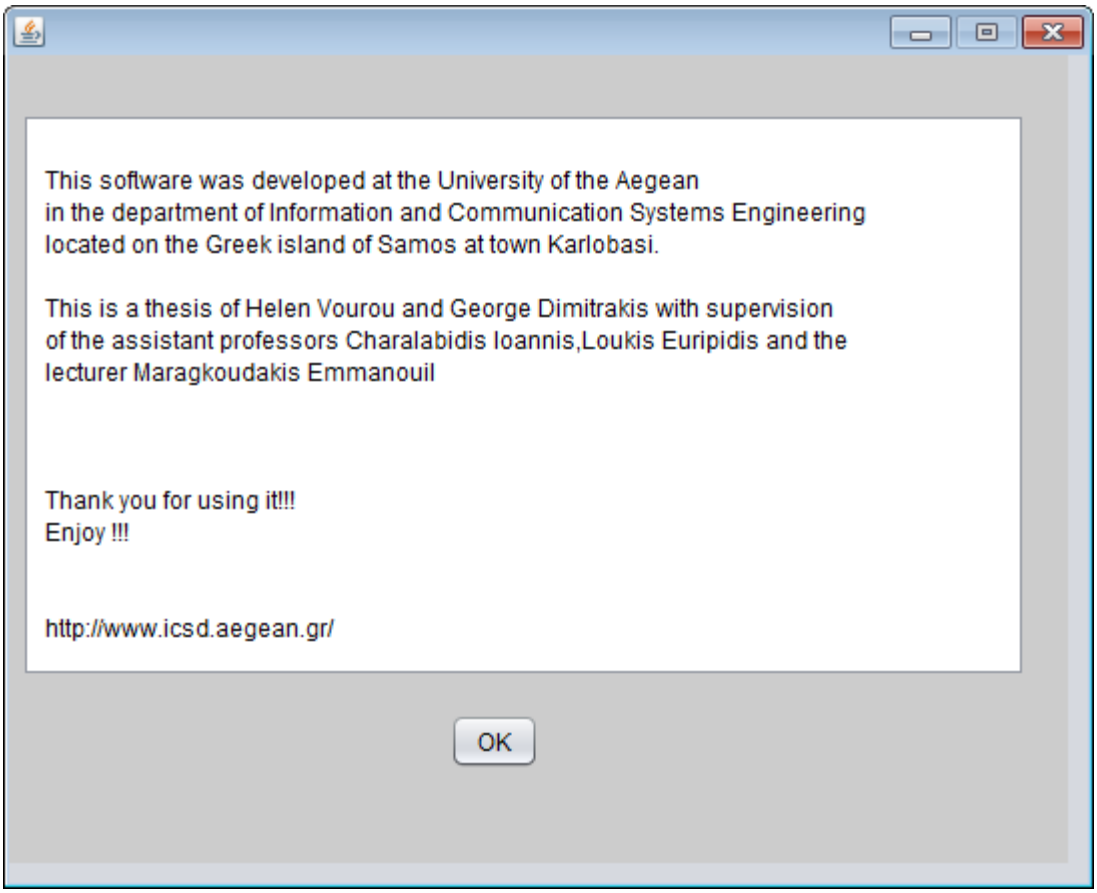
Εικόνα 41:Prediction of comment from txt file

Επόμενο παράδειγμα να γράψουμε εμείς ένα σχόλιο στο πεδίο που μας δίνεται για παράδειγμα «Συμφωνώ απόλυτα με την κάρτα του πολίτη». Επίσης υπάρχει το πεδίο «Clear» με το οποίο κάνουμε το πεδίο πάλι άδειο.



Εικόνα 42:Prediction θετικού σχολίου από το πεδίο

Τέλος υπάρχει το κουμπί «Edit» το οποίο εμφανίζει παράθυρο για τους δημιουργούς της εφαρμογής. Επίσης με το πάτημα του κουμπιού «Exit» στην κεντρική εφαρμογή ο χρήστης βγαίνει από την εφαρμογή.



Εικόνα 43:Edit

Κεφάλαιο 6
Συμπεράσματα - Προοπτικές

6.1 *Συμπεράσματα*

Όπως προκύπτει μέσα από τις σελίδες της παρούσας διπλωματικής εργασίας, η μετάβαση στην εποχή της ηλεκτρονικής διακυβέρνησης αποτελεί ένα από τα πιο σύγχρονα και σπουδαία εγχειρήματα της εποχής μας. Η συνειδητοποίηση τόσο από πλευράς κυβερνήσεων όσο από πλευράς πολιτών της αναγκαιότητας της εν λόγω μετάβασης είναι καθοριστική καθώς μόνο οφέλη μπορούν να αποκομισθούν. Η προσπάθεια για την βελτιστοποίηση και την μέγιστη δυνατή επιτυχία της ηλεκτρονικής διακυβέρνησης πρέπει να καταβληθεί αμφίδρομα και από τις δύο πλευρές, στα πλαίσια μιας υγιούς σχέσης συνεργασίας, συμμετοχής και εν τέλει διαδραστικότητας.

Και σε αυτή την ηλεκτρονική συμμετοχή και διαδραστικότητα βρίσκει πάτημα η τεχνική της αυτόματης εξόρυξης γνώμης/ συναισθήματος μέσω διαδικτυακών πηγών. Όμως η εξόρυξης γνώμης/ συναισθήματος είναι μια δύσκολη διαδικασία γιατί στηρίζεται σε ποιοτικά δεδομένα και όχι σε ποσοτικά. Συνεπάγεται ότι η ακρίβεια της διαδικασίας κυμαίνεται αναμφίβολα σε χαμηλότερα επίπεδα.

Με την ανάπτυξη του Παγκόσμιου Ιστού είναι διαθέσιμη ελεύθερα στο διαδίκτυο μια μεγάλη ποσότητα πληροφοριών και δεδομένων που πρέπει να αξιοποιηθεί ώστε να οδηγηθούμε αρχικά σε υπηρεσίες ηλεκτρονικής διακυβέρνησης κατάλληλα προσαρμοσμένες στις ανάγκες και τις απαιτήσεις του χρήστη και εν τέλει στη βέλτιστη λειτουργία του κρατικού μηχανισμού αλλά και την εξοικονόμηση πόρων. Η συνεισφορά της εξόρυξης γνώμης είναι τεράστια γιατί μέσω κατάλληλων αλγορίθμων, εξάγουμε αξιόπιστα συμπεράσματα σχετικά με αντιλήψεις, πεποιθήσεις και προτιμήσεις του μέσου χρήστη πάνω σε ένα θέμα. Αυτοί οι αλγόριθμοι και τα διάφορα λογισμικά προγράμματα ξεπερνούν τον ανθρώπινο νου σε ταχύτητα και αποτέλεσμα.

Με τη δημιουργία εφαρμογής κατηγοριοποίησης γνώμης, στα πλαίσια της εργασίας, προσπαθήσαμε να προσεγγίσουμε όσο το δυνατόν ακριβέστερα και πληρέστερα τον ζητούμενο στόχο. Θεωρητικά, πρόκειται για μία διαδικασία που ακολουθεί τα απαραίτητα βήματα και περιέχει τους κατάλληλους αλγορίθμους για τον σκοπό αυτό, στην πράξη όμως η ακρίβεια απέχει από τα επιθυμητά επίπεδα για λόγους που αναλύθηκαν διεξοδικά.

6.2 Προβλήματα

Ολοκληρώνοντας την παρούσα διπλωματική εργασία, θα θέλαμε να παρουσιάσουμε κάποια προβλήματα που αντιμετωπίσαμε και τα οποία θα μπορούσαν να συνεισφέρουν σε περαιτέρω μελέτη.

Το πιο κύριο πρόβλημα που αντιμετωπίσαμε ήταν η Ελληνική γλώσσα. Πιο συγκεκριμένα, το πρόγραμμα του RapidMiner δεν μπορούσε να διαβάσει τα Ελληνικά με αποτέλεσμα να βγάζει ακαταλαβίστικες λέξεις. Ως λύση βρήκαμε την μετατροπή των txt αρχείων σε encoding UTF-8. Επίσης δεν υπήρχε έτοιμο λεξικό stopwords λέξεων σε Ελληνικά κείμενα, το οποίο και δημιουργήσαμε οι ίδιοι. Επιπλέον καταφύγαμε στην χρησιμοποίηση του operator «Replace Token», όπως αναφέραμε στο κεφάλαιο 5, γιατί είχαμε μεγαλύτερα ποσοστά. Τέλος το RapidMiner δεν διαθέτει stem, δηλαδή να εντοπίζεται και να αφαιρείται από κάθε λέξη η κατάληξή της.

Σαν δεύτερο πρόβλημα είχαμε να αντιμετωπίσουμε μεγάλες απαιτήσεις του λογισμικού RapidMiner σε υπολογιστικούς πόρους. Υπήρχαν πολλές περιπτώσεις που εμφανιζόταν μήνυμα μη διαθέσιμης μνήμης.

Ένα τρίτο πρόβλημα ήταν ότι το RapidMiner δεν υποστηρίζει δημιουργία arff αρχείου ενός txt αρχείου δοθέντος το αρχείο arff του training που έγινε στην φάση 2. Γι' αυτό το λόγο και καταφύγαμε στην δημιουργία java εφαρμογής για να χρησιμοποιούμε αυτή την διαδικασία που θεωρείται πιο αποτελεσματική.

Τέλος ένα άλλο πρόβλημα που αντιμετωπίσαμε ήταν η ανεπιτυχής σύνδεση java με RapidMiner στην δημιουργία εφαρμογής το οποίο όμως στην πορεία λύθηκε.

6.3 Προοπτικές

Η ενασχόληση μας με το text mining μας δημιούργησε ορισμένες σκέψεις οι οποίες ίσως μπορούν να μεταφραστούν και προοπτικές οι οποίες θα μπορούσαν να βελτιώσουν τις τεχνικές opinion mining στο τομέα της Ηλεκτρονικής Διακυβέρνησης.

Αρχικά, πρέπει να αναφέρουμε ότι θα ήταν πιο συνετό να είχαμε γνώση όσον αφορά τις απαιτήσεις των πολιτών σχετικά με κάποια υπηρεσία πριν ακόμα αρχίσει η υλοποίηση της. Αυτό συνεπάγεται την ανάγκη για ανάπτυξη κατάλληλων Πληροφοριακών Συστημάτων τα οποία θα έχουν την δυνατότητα να επεξεργαστούν μεγάλο όγκο πληροφοριών δηλαδή τις γνώμες των

ίδιων των πολιτών για να υπάρξουν επαρκή στοιχεία/δεδομένα για το πόσο χρήσιμη είναι μια υπηρεσία.

Όπως προαναφέραμε και σε προηγούμενο κεφάλαιο οι διαδικτυακές ηλεκτρονικές υπηρεσίες υστερούν κατά πολύ συγκριτικά με τα αντίστοιχα ποσοστά που παρουσιάζονται στην υπόλοιπη ΕΕ. Πρακτικά αυτό σημαίνει ότι είναι μείζον σημασίας να εκσυγχρονιστούν οι διαδικτυακοί μας πόροι έτσι ώστε να γίνεται ακόμα απλούστερη η συμμετοχή του μέσου πολίτη σε τέτοιου είδους υπηρεσίες.

Παράλληλα με τα παραπάνω, θετικό βήμα θα αποτελούσε και η συνεργασία σε ευρωπαϊκό επίπεδο. Όσο μεγαλύτερος όγκος δεδομένων υπάρχει τόσο καλύτερα συμπεράσματα θα μπορούσαν να εξαχθούν.

Βιβλιογραφία

[1] Μ.Χαλκίδη – Μ.Βαρζογιάννη, «ΕΞΟΡΥΞΗ ΓΝΩΣΗΣ ΑΠΟ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΤΟΝ ΠΑΓΚΟΣΜΙΟ ΙΣΤΟ», Γιώργος Δαρδάνος,Αθήνα 2005.

[2] Weiss, Sholom M. «Text Mining : Predictive Methods for Analyzing Unstructured Information», New York :Springer, c2005.

[3] Konchady, Manu, «Text mining application programming», Boston, Mass :Charles River Media, c2006.

[4] Groth, Robert, «Data mining : a hands-on approach for business professionals», Upper Saddle River, NJ. :Prentice Hall, c1998.

[5] Bigus, Joseph P., «Data mining with neural networks : solving business problems from application development to decision support», New York, San Francisco :McGraw Hill, c1996.

[6] Westphal Christophe R.Blaxton, Teresa, « Data mining solutions : methods and tools for solving real-world problems », New York ; Chichester :John Wiley, c1998

[7] Michalski, Ryszard Stanislaw, 1937-, 340,Bratko, Ivan, 340,Kubat, Miroslav, 340, « Machine learning and data mining : methods and applications», Chichester :, New York :John Wiley, c1998

[8] R.Cadenhead,L.Lemay, «Πλήρες Εγχειρίδιο της Java 2»,Μ.Γκιούρδας,Αθήνα 2003.

[9] Γ.Λιακέας, «Εισαγωγή στη Java 2»,Κλειδάριθμος,Αθήνα 2003.

[10] Kushal Dave, Steve Lawrence, and David M. Pennock. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In Proceedings of WWW

[11] Hand et al., 2001Hand2001 Hand, D., Mannila, H., & Smyth, P. (2001)Principles of data mining. Adaptive Computation and Machine Learning Series. MIT Press.

[12] Internet World Stats (2011). Internet Usage Statistics - The Big Picture. World Internet Users and Population Stats. Retrieved from: http://www.internetworldstats.com/emarketing.htm

[13] http://www.wordle.net/create

[14] www.padgets.eu

[15] http://www.epractice.eu/en/cases/padgets

[16] http://www-958.ibm.com/software/data/cognos/manyeyes/visualizations

[17] http://en.wikipedia.org/wiki/Data_mining

[18]http://www.opengov.gr/ypes/?p=863

[19] http://people.ischool.berkeley.edu/~hearst/text-mining.html

[20] http://www.cs.uic.edu/~liub/FBS/opinion-mining.pdf

[21]http://www.opengov.gr/ypes/wp-content/uploads/downloads/2011/04/egovroadmapV0.3-2.pdf

[22] http://openarchives.gr/search/Text%20mining

[23] http://www.kdnuggets.com/polls/2011/tools-analytics-data-mining.html

[24] http://www.slideshare.net/rapidminercontent/rapidminer-introduction-to-rapidminer

[25] http://en.wikipedia.org/wiki/KNIME

[26] http://tech.knime.org/files/KNIME-TextProcessing-HowTo.pdf

[27] http://cran.r-project.org/

[28] http://www.knime.org/

[29] http://www.kdd.org/explorations/issues/11-1-2009-07/p2V11n1.pdf

[30] http://weka.pentaho.com/

[31]http://www.hstathome.com/tjziyuan/SAS%20Data%20Mining%20Using%20Sas%20Enterprise%20Miner%20-%20A%20Case%20Study%20Appro.pdf

[32] http://support.sas.com/resources/papers/proceedings11/160-2011.pdf

[33] http://en.wikipedia.org/wiki/Sentiment_analysis

[34] http://en.wikipedia.org/wiki/Machine_learning

[35] http://en.wikipedia.org/wiki/List_of_machine_learning_algorithms

[36] http://www.dataminingtools.net/viewtutorials.php?id=3667258

[37] http://en.wikipedia.org/wiki/Operator_(programming)

[38] http://en.wikipedia.org/wiki/Document_classification

[39] http://rapid-i.com/wiki/index.php?title=Text:Process_Documents_from_Files

[40] http://www.slideshare.net/rapidminercontent/rapidminer-word-vector-tool-and-rapid-miner-3667261

[41] http://www.dataminingtools.net/viewtutorials.php?id=3627412

[42] http://rapid-i.com/api/rapidminer-5.1/com/rapidminer/operator/validation/XValidation.html

[43] http://www.dataminingtools.net/viewtutorials.php?id=3667253

[44] http://www.slideshare.net/rapidminercontent/rapidminer-advanced-processes-and-operators-3667255

[45] http://www.statsoft.com/textbook/naive-bayes-classifier/

[46] http://www.cs.cmu.edu/~tom/mlbook/NBayesLogReg.pdf

[47] http://en.wikipedia.org/wiki/Naive_Bayes_classifier

[48] http://rapid-i.com/wiki/index.php?title=Apply_Model&oldid=2772

[49] http://rapid-i.com/wiki/index.php?title=Performance_(Regression)

[50] http://rapid-i.com/wiki/index.php?title=Write_Model&oldid=2646

[51] http://rapid-i.com/wiki/index.php?title=Select_Attributes

[52] http://en.wikipedia.org/wiki/Web_2.0

[53] http://en.wikipedia.org/wiki/Social_media

Παράρτημα 1

Συγκεκριμένα Παραδείγματα λογισμικού και εφαρμογών:

- AeroText - provides a suite of text mining applications for content analysis.

  Content used can be in multiple languages.

- AlchemyAPI - SaaS-based text mining platform that supports 6+ languages.

  Includes named entity extraction, keyword extraction, document categorization, etc.

- Autonomy - suite of text mining, clustering and categorization solutions for a variety of industries.
- Endeca Technologies - provides software to analyze and cluster unstructured text.
- Expert System S.p.A. - suite of semantic technologies and products for developers and knowledge managers.
- Fair Isaac - leading provider of decision management solutions powered by advanced analytics (includes text analytics).
- Inxight - provider of text analytics, search, and unstructured visualization technologies. (Inxight was bought by Business Objects that was bought by SAP AG in 2008)
- Nstein - text mining solution that creates rich metadata to allow publishers to increase page views, increase site stickiness, optimize SEO, automate tagging, improve search experience, increase editorial productivity, decrease operational publishing costs, increase online revenues
- Pervasive Data Integrator - includes Extract Schema Designer that allows the user to point and click identify structure patterns in reports, html, emails, etc. for extraction into any database
- SPSS - provider of SPSS Text Analysis for Surveys, Text Mining for Clementine, LexiQuest Mine and LexiQuest Categorize, commercial text analytics software that can be used in conjunction with SPSS Predictive Analytics Solutions.
- Thomson Data Analyzer - Enables complex analysis on patent information, scientific publications and news.
- LexisNexis - LexisNexis is a provider of business intelligence solutions based on an extensive news and company information content set. Through the recent acquisition of Datops LexisNexis is leveraging its search and retrieval expertise to become a player in the text and data mining field.
- LanguageWare - Text Analysis libraries and customization tooling from IBM