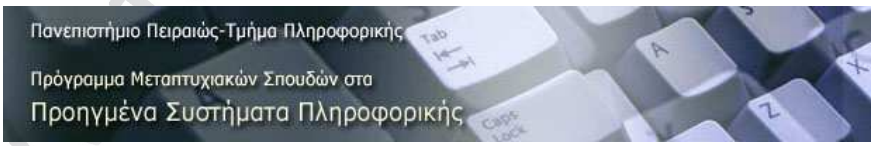
Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής

Πρόγραμμα Μεταπτυχιακών Σπουδών

«Προηγμένα Συστήματα Πληροφορικής»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	**Εξόρυξη Γνώσης & Ανάκτηση Δεδομένων Εικόνας με Χρήση Υποδομών Σχεσιακών Βάσεων Δεδομένων**
Ονοματεπώνυμο Φοιτητή	**Λεπίδας Αλέξανδρος του Ευαγγέλου**
Αριθμός Μητρώου	**ΜΠΣΠ 07002**
Κατεύθυνση	**Συστήματα Υποστήριξης Αποφάσεων (ΣΥΑ)**
Επιβλέπων	**Θεοδωρίδης Γιάννης, Αναπληρωτής Καθηγητής**



Ημερομηνία Παράδοσης **Απρίλιος 2011**

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)	(υπογραφή)	(υπογραφή)
Γιάννης Θεοδωρίδης Αναπληρωτής Καθηγητής	Γιάννης Σίσκος Καθηγητής	Νίκος Πελέκης Λέκτορας

Περίληψη

Οι οργανισμοί χρησιμοποιούν σήμερα τα συστήματα ψηφιακής απεικόνισης, περιμένοντας να βελτιωθεί η αποτελεσματικότητά τους. Η ψηφιακή εικόνα επιτρέπει την αποθήκευση, ανάκτηση και ανταλλαγή τεράστιου αριθμό εγγραφών σε υποδομή δικτύου και το Internet γενικότερα. Οι χρήστες μπορούν να βρουν ένα αρχείο με ένα σύστημα ψηφιακής απεικόνισης γρηγορότερα από ό, τι μπορούν να βρουν την έντυπη έκδοση ή την αντίστοιχη μικροφίλμ. Μπορούν επίσης να μοιράζονται αρχεία εύκολα με τη χρήση διαφόρων υποδομών, όπως e-mail και άμεσων μηνυμάτων. Από την άλλη πλευρά, υπάρχει η ανάγκη για αυξημένη αποθηκευτικό χώρο για εικόνες και φωτογραφίες. Αν και αυτό το τελευταίο σημείο είναι ότι κατά κύριο λόγο τονίζει το πραγματικό πλεονέκτημα της ψηφιακής απεικόνισης είναι η ηλεκτρονική πρόσβαση στα αρχεία και την ανταλλαγή των σχετικών πληροφοριών. Η απόφαση για το αν και πώς να εφαρμόζουν ένα σύστημα απεικόνισης είναι πολύπλοκη. Πολλοί παράγοντες πρέπει να λαμβάνονται υπόψη. Κατά κύριο λόγο, ποιο είναι το επιθυμητό αποτέλεσμα; Θα καλύψει τις πραγματικές ανάγκες και πώς να ενσωματώσουν την υπάρχουσα υποδομή; Υπάρχουν επαρκείς οικονομικοί πόροι για τη στήριξη των συστημάτων;

Η μελέτη επικεντρώθηκε στην αποθήκευση και ανάκτηση καθώς και στην επιτυχή κατηγοριοποίηση. Ειδικότερα, υπάρχει μια διαδικασία εισαγωγής των εικόνων σε μια σχεσιακή βάση δεδομένων (Oracle Corporation RDBMS), μια επιτυχημένη μετατροπή σε κατάλληλη μορφή για την αναζήτηση σε αυτές τις εικόνες, τη δημιουργία της εξόρυξης δεδομένων και τη διεξαγωγή πειραμάτων για να καταλήξουμε σε σχετικές διαπιστώσεις. Το κλειδί για την επιτυχημένη σχεδίαση, ανάπτυξη και εφαρμογή ενός συστήματος για την επεξεργασία δεδομένων εικόνας είναι η σωστή ανάλυση. Οι τέσσερις κύριες φάσεις είναι οι εξής: α) Σχεδιασμός και ανάλυση απαιτήσεων, β) Ανάλυση της τεχνολογίας που επιλέγεται, γ) διαδικασία εφαρμογής, γ)Ανάλυση των αποτελεσμάτων. Σε αυτό το πλαίσιο, μελετήσαμε τέσσερις διαφορετικούς αλγορίθμους (decision trees, naive bayes, support vector machines and logistic regression), για ένα σύνολο περίπου χιλίων εικόνων. Τα αποτελέσματα είναι ικανοποιητικά, αν και υπάρχει πάντα περιθώριο για βελτίωση. Εκτός αυτού, ένας αλγόριθμος εξόρυξης δεδομένων που επιτυγχάνει πολύ υψηλό ποσοστό επιτυχίας για ένα συγκεκριμένο πρόβλημα που δημιουργείται από την πιθανότητα να είναι over optimized για το συγκεκριμένο πρόβλημα, καθιστώντας ενδεχομένως ακατάλληλο για ένα ευρύ φάσμα προβλημάτων.

Abstract

Organizations use nowadays digital imaging systems expecting to improve their effectiveness and multimedia presentation business and operation-wise. The digital image allows the capture, storage, retrieval and sharing of a huge number of recordings to network infrastructure and the Internet in general. Users can typically find a file with a digital imaging system faster than they can find the printed version or the respective microfilm. They can also share files easily using various infrastructures such as e-mail and instant messaging. On the other hand there is the need for increased storage space for images and photos. Although this last point is that primarily emphasizes the real advantage of digital imaging is the online access to files and the exchange of relevant information. The decision on whether and how to implement an imaging system is complex. Many factors must be considered. Primarily, what is the desired result? How will the display resolution of user problems? Will cover the real needs and how to integrate the existing infrastructure and are there sufficient financial resources to support systems over time?

This study focused on the concepts of digital imaging and in particular in the storage and successful categorized search / retrieval. In particular there is a study of the import images process into a relational database (in specific Oracle Corporation RDBMS), a successful conversion to suitable format for the search patterns in these images, creating the data mining

and conducting experiments to conclude to related findings. The decision to implement an image-processing system should be based on needs arising from the specific application requested. The key to successful design, development and implementation of a system for processing image data to find what is the correct analysis. The four main phases are: a) planning and analysis requirements, b) analysis of the technology chosen, c) process implementation, d) analysis of results.

Observing that the process of creating, training and applying data mining algorithms is not a standardized procedure, which cannot comprise a uniform solution to all problems that require searching and processing large data sets.For this reason, it is understood that continuous study and research development in this sector is essential to the broad and heterogeneous range of problems requiring data mining. More specifically, applications that can only extract data sets of images - either medical or different categories as in our case - is perceived to be too many. Since applications for registration, search, image processing on the Internet, management of databases with medical images in large health facilities, and automatic comparison of these images to suggest for diagnosis / indications of certain pathogenicity. In this context, we studied four different algorithms (decision trees, naive bayes, support vector machines and logistic regression) for a total of approximately one thousand images, and a corresponding classification of those. The results are satisfactory, although there is always room for improvement. Besides, a data mining algorithm that achieves very high success rate for a specific problem posed by the chance to be over-optimized for the specific problem, making it potentially unsuitable for a wide range of problems. Customizing also showed that for the algorithms applied increasing the percentage of the available data in a training set of algorithms performance improved the performance of the algorithm significantly.

ΠΕΡΙΕΧΟΜΕΝΑ

ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ

ΕΥΡΕΤΗΡΙΟ ΠΙΝΑΚΩΝ

1. Εισαγωγή

1.1. Γενικά

Οι οργανισμοί χρησιμοποιούν ψηφιακά συστήματα απεικόνισης αποσκοπώντας στη βελτίωση της αποτελεσματικότητάς τους. Η ψηφιακή εικόνα τους δίνει τη δυνατότητα σύλληψης, αποθήκευσης, και ανάκτησης και διαμοιρασμού ενός τεράστιου αριθμού εγγραφών μέσω δικτυακών υποδομών αλλά και του διαδικτύου ειδικότερα. Οι χρήστες μπορούν να βρουν τυπικά ένα αρχείο με ένα σύστημα ψηφιακής απεικόνισης γρηγορότερα από ό, τι μπορούν να βρουν την εκτυπωμένη έκδοση ή την έκδοση σε μικροφίλμ. Μπορούν επίσης να μοιραστούν εύκολα τα αρχεία κάνοντας χρήση διάφορων υποδομών όπως το ηλεκτρονικό ταχυδρομείο.

Παράλληλα δημιουργείται μειωμένη ανάγκη για αρχειοθήκες και χώρο αποθήκευσης εικόνων και φωτογραφιών. Αν και αυτό το τελευταίο σημείο είναι αυτό που κατά κύριο λόγο τονίζεται, το πραγματικό πλεονέκτημα της ψηφιακής απεικόνισης βρίσκεται στην online πρόσβαση σε αρχεία και στην ανταλλαγή χρήσιμων πληροφοριών.

Η απόφαση για το αν θα εφαρμοστεί ένα σύστημα απεικόνισης είναι πολύπλοκη. Πολλοί παράγοντες πρέπει να ληφθούν υπόψη. Κατά κύριο λόγο, ποιο είναι το επιθυμητό αποτέλεσμα; Πώς θα αποτελέσει η απεικόνιση επίλυση των προβλημάτων του χρήστη; Θα καλύψει τις πραγματικές ανάγκες; Πώς θα ενσωματωθεί στην υπάρχουσα υποδομή; Υπάρχουν επαρκείς οικονομικοί πόροι για τη στήριξη των συστημάτων με την πάροδο του χρόνου;

1.2. Σκοπός της Εργασίας

Σκοπός της εργασίας είναι η επεξεργασία και εισαγωγή εικόνων σε σχεσιακή βάση δεδομένων, με σκοπό την εξόρυξη δεδομένων βασιζόμενοι στη χρήση αλγορίθμων και την εξαγωγή προτύπων (patterns).

Η απόφαση για την εφαρμογή ενός συστήματος επεξεργασίας εικόνας πρέπει να βασίζεται στις ανάγκες που προκύπτουν από την ανάλυση της συγκεκριμένης εφαρμογής που ζητείται. Το κλειδί για τον επιτυχή σχεδιασμό, την ανάπτυξη, και την εφαρμογή ενός συστήματος επεξεργασίας εικόνας και αναζήτησης δεδομένων σε αυτό είναι η σωστή ανάλυση. Η τέσσερις σημαντικότερες φάσεις είναι οι ακόλουθες:

- Σχεδιασμός και ανάλυση απαιτήσεων.
- Ανάλυση της τεχνολογίας που επιλέγεται.
- Διαδικασία εφαρμογής.
- Ανάλυση αποτελεσμάτων

Οι ενότητες που προαναφέρθηκαν παρουσιάζονται συνοπτικά με έμφαση στην ερευνητική περιοχή που πραγματεύεται η παρούσα εργασία, καθώς μέσα από αυτές σκιαγραφείται ο σκοπός και η πορεία της παρούσας εργασίας.

- **Εκτίμηση των αναγκών.** Στο στάδιο της εκτίμησης των αναγκών για να καθοριστεί ποια, ενδεχομένως, τα οφέλη που θα κερδίσουμε χρησιμοποιώντας ένα σύστημα ψηφιακής αποθήκευσης και ανάκτησης πολυμέσων με σχεσιακή βάση δεδομένων. Στο στάδιο αυτό πρέπει να εκτιμηθούν οι διαθέσιμες εναλλακτικές τεχνολογίες και να γίνει η σχετική αξιολόγηση.
- **Ανάλυση χρησιμοποιούμενης τεχνολογίας.** Στο στάδιο της ανάλυσης της τεχνολογίας που επελέγη (Oracle RDBMS και πακέτο Oracle Multimedia) παρουσιάζονται οι υποδομές που χρησιμοποιήθηκαν για την τροποποίηση και εισαγωγή των δεδομένων στη βάση, καθώς και την ανάλυσή τους.

- **Διαδικασία εφαρμογής τεχνολογιών / αλγορίθμων.** Στο στάδιο αυτό αναλύονται και εφαρμόζονται αλγόριθμοι εξόρυξης δεδομένων, για την επίλυση του προβλήματος που πραγματεύεται η παρούσα εργασία. Πιο συγκεκριμένα δημιουργείται η κατάλληλη υποδομή (μορφοποίηση δεδομένων και εισαγωγή στη βάση) ώστε να εφαρμοστούν μετά τη διαδικασία εκπαίδευσης οι διάφοροι αλγόριθμοι για την εξόρυξη δεδομένων. Στο στάδιο αυτό θα γίνουν δοκιμές και για την εναλλακτική παραμετροποίηση των μοντέλων.
- **Ανάλυση αποτελεσμάτων εξαγωγή συμπερασμάτων.** Αφού εφαρμοστούν οι εναλλακτικοί αλγόριθμοι κατηγοριοποίησης τα αποτελέσματα θα αναλυθούν και θα εκτιμηθεί η αποτελεσματικότητα των αλγορίθμων.

1.3. Δομή της Εργασίας

Η δομή της παρούσας εργασίας περιγράφεται στη συνέχεια.

Στο παρόν πρώτο κεφάλαιο γίνεται μια γενική εισαγωγή στην ερευνητική περιοχή την οποία πραγματεύεται η παρούσα εργασία. Γίνεται συνοπτική αναφορά στην ανάγκη για χρήση ψηφιακής εικόνας αλλά και στο σκοπό του παρόντος πονήματος.

Στο δεύτερο κεφάλαιο γίνεται μια αναλυτική παρουσίαση των υπαρχουσών προσεγγίσεων, τόσο σε επίπεδο επεξεργασίας εικόνας, όσο και σε επίπεδο data mining και αλγορίθμων κατηγοριοποίησης.

Στο τρίτο κεφάλαιο παρουσιάζεται η σχεσιακή βάση δεδομένων της Oracle, εστιάζοντας περισσότερο στις υποδομές και στην αρχιτεκτονική που θα χρησιμοποιηθούν στα πλαίσια της εργασίας αυτής.

Στο τέταρτο κεφάλαιο γίνεται παρουσίαση της δουλειάς που έγινε στα πλαίσια της εργασίας αυτής. Έτσι παρουσιάζεται τόσο το αρχιτεκτονικό μοντέλο στο οποίο βασίστηκε η σχετική πρότυπη υλοποίηση, όσο και το θεωρητικό υπόβαθρο στο οποίο στηρίχτηκε η ανάπτυξη του προαναφερθέντος αρχιτεκτονικού μοντέλου.

Στο πέμπτο κεφάλαιο γίνεται παρουσίαση των πειραματικών εφαρμογών όσων αναπτύχθηκαν στα πλαίσια της εργασίας, καθώς και τα σχετικά συμπεράσματα.

Τέλος, στο έκτο κεφάλαιο γίνεται μια σύνοψη της εργασίας καταλήγοντας στα συνολικά συμπεράσματα και δίνοντας πιθανές κατευθύνσεις για μελλοντικές επεκτάσεις.

1.4. Αντικείμενο Εργασίας

Η παρούσα εργασία επικεντρώθηκε στις έννοιες της ψηφιακής απεικόνισης και ειδικότερα στο τμήμα της αποθήκευσης και επιτυχούς κατηγοριοποιημένης αναζήτησης / ανάκτησης. Πιο συγκεκριμένα γίνεται μελέτη της διαδικασίας εισαγωγής εικόνων σε σχεσιακή βάση δεδομένων (σε τεχνολογίες της Oracle Corporation), η επιτυχής μετατροπή σε κατάλληλη μορφή για την αναζήτηση προτύπων (patterns) στις εικόνες αυτές, η δημιουργία διαδικασιών data mining και η διεξαγωγή πειραμάτων και των σχετικών συμπερασμάτων. Οι παραπάνω ενότητες περιγράφονται στις επόμενες παραγράφους

2. Υπάρχουσες Προσεγγίσεις - Προκλήσεις

2.1. Εξόρυξη Δεδομένων / Αλγόριθμοι Κατηγοριοποίησης

Η εξόρυξη με τη χρήση προτύπων σε δεδομένα είναι ένα πρόβλημα που απασχολεί την ερευνητική κοινότητα αιώνες. Οι πρώτες μέθοδοι αναγνώρισης προτύπων σε δεδομένα περιλαμβάνουν το θεώρημα του Bayes (1700) και την ανάλυση παλινδρόμησης (1800). Η εξέλιξη, η πανταχού παρουσία και αυξανόμενη δύναμη της τεχνολογίας των υπολογιστών έχει αυξήσει τη συλλογή δεδομένων, την αποθήκευση και την επεξεργασία τους. Με τη διόγκωση των συνόλων δεδομένων και την αύξηση της πολυπλοκότητάς τους, η ανάλυση δεδομένων όλο και περισσότερο έχουν προταθεί είτε με έμμεσες είτε αυτόματες διεργασίες επεξεργασίας δεδομένων. Αυτό έχει υποβοήθηση από άλλες εξελίξεις στην επιστήμη των υπολογιστών, όπως τα νευρωνικά δίκτυα, το clustering, οι γενετικοί αλγόριθμοι (δεκαετία 1950), δένδρα αποφάσεων (δεκαετία 1960) και την υποστήριξη διανυσματικών μηχανές (δεκαετία 1980). Η εξόρυξη δεδομένων είναι η διαδικασία της εφαρμογής των μεθόδων αυτών στα δεδομένα με σκοπό την αποκάλυψη αρχικά «κρυμμένων» προτύπων [6]. Έχει χρησιμοποιηθεί εδώ και πολλά χρόνια από τις επιχειρήσεις, τους επιστήμονες και τις κυβερνήσεις για τη διερεύνηση μεγάλου όγκου δεδομένων όπως αρχεία αεροπορικών εταιρειών για τους επιβάτες τους, τα στοιχεία της απογραφής και τα δεδομένα του σαρωτή σούπερ μάρκετ ώστε να εκπονηθούν εκθέσεις έρευνας αγοράς. (Σημειώστε, ωστόσο, ότι μια αναφορά δεν θεωρείται πάντα εξόρυξη δεδομένων.)

Ένας κύριος λόγος για τη χρήση του data mining είναι να βοηθήσει στην ανάλυση των συλλογών των παρατηρήσεων συμπεριφοράς. Τα δεδομένα αυτά είναι ευάλωτα σε συγγραμμικότητας λόγω άγνωστων συσχετισμών. Ένα αναπόφευκτο γεγονός της εξόρυξης δεδομένων είναι ότι τα σύνολα και υποσύνολα των στοιχείων που αναλύθηκαν δεν μπορεί να είναι αντιπροσωπευτικά του συνόλου των δεδομένων, και ως εκ τούτου δεν επιτρέπεται να περιέχουν παραδείγματα ορισμένων κρίσιμων σχέσεων και συμπεριφορών που υπάρχουν σε άλλα τμήματα των δεδομένων. Για την αντιμετώπιση αυτού του προβλήματος, η ανάλυση μπορεί να αυξηθεί με βάση το πείραμα και άλλες προσεγγίσεις, όπως τα μοντέλα επιλογής που προορίζονται για δεδομένα που δημιουργούνται / εισάγονται από ανθρώπους και όχι μηχανές. Σε αυτές τις περιπτώσεις, οι συσχετίσεις μπορούν είτε να ελέγχονται είτε να αφαιρούνται εντελώς, κατά την κατασκευή του πειραματικού μοντέλου.

2.2. Υπάρχουσες Προσεγγίσεις

Οι υπάρχουσες προσεγγίσεις στον τομέα της ανάλυσης και εξόρυξης δεδομένων μπορούν να κατηγοριοποιηθούν στις τεχνικές που βασίζονται στα δεδομένα και στις τεχνικές που βασίζονται στις διαδικασίες. Στις λύσεις που βασίζονται στα δεδομένα, η ιδέα είναι να εξεταστεί μόνο ένα υποσύνολο του συνόλου των δεδομένων ή τη μετατροπή των δεδομένων κάθετα ή οριζόντια σε μία κατά προσέγγιση μικρότερη αντιπροσώπευση του συνολικού μεγέθους των δεδομένων. Από την άλλη πλευρά, στις λύσεις με βάση τις διαδικασίες, χρησιμοποιούνται υπολογιστικές θεωρίες για την επίτευξη αποδοτικών λύσεων χρονικά και χωρικά. Στην ενότητα παρουσιάζονται οι δύο προσεγγίσεις.

2.2.1 Τεχνικές βασισμένες στα δεδομένα

Οι τεχνικές με βάση τα Δεδομένα αναφέρονται συνοπτικά στο σύνολο δεδομένων ή επιλέγοντας ένα υποσύνολο της εισερχόμενης προς ανάλυση ροής δεδομένων. Η δειγματοληψία, η απόρριψη φορτίου και η σκιαγράφηση είναι οι τεχνικές που απαρτίζουν αυτήν την κατηγορία. Εδώ παρουσιάζεται μια περίληψη από τα βασικά των τεχνικών αυτών με αναφορές σε εφαρμογές τους στο πλαίσιο της ανάλυσης δεδομένων.

Δειγματοληψία

Η δειγματοληψία αναφέρεται στη διαδικασία της πιθανοτικής επιλογής ενός στοιχείου δεδομένων προς επεξεργασία ή μη. Τα όρια του ποσοστού σφαλμάτων του υπολογισμού δίνονται ως συνάρτηση του ρυθμού δειγματοληψίας. Η τεχνική Very Fast Machine Learning [39] που χρησιμοποιούν το όριο Hoeffding για τη μέτρηση του μεγέθους του δείγματος.

Το πρόβλημα στη χρήση της δειγματοληψίας στην ανάλυση ροής δεδομένων είναι το άγνωστο μέγεθος του συνόλου δεδομένων. Έτσι, η εφαρμογή στη ροή δεδομένων θα πρέπει να ακολουθήσει μια ειδική ανάλυση για να βρεθούν τα όρια σφάλματος. Άλλο πρόβλημα στη δειγματοληψία είναι ότι θα ήταν σημαντικό να ελεγχθούν για ανωμαλίες σαν εφαρμογή στην εξόρυξη δεδομένων. Η δειγματοληψία μπορεί να μην είναι η σωστή επιλογή για μια τέτοια εφαρμογή καθώς δεν αντιμετωπίζει το πρόβλημα των διακυμάνσεων των ρυθμών δεδομένων. Θα ήταν χρήσιμη η διερεύνηση της σχέσης μεταξύ των τριών παραμέτρων: ρυθμού δεδομένων, ρυθμού δειγματοληψίας και όρια λαθών.

Απώλεια φορτίων

Η απώλεια φορτίων αναφέρεται [35, 50] στη διαδικασία της διαγραφής της ακολουθίας των ροών δεδομένων. Η απώλεια των φορτίων έχει χρησιμοποιηθεί με επιτυχία σε επερωτήσεις σε ροές δεδομένων. Έχει τα ίδια προβλήματα της δειγματοληψίας. Η απώλεια φορτίων είναι δύσκολο να χρησιμοποιηθεί με αλγορίθμους εξόρυξης, διότι «πετάει» κομμάτια των ροών δεδομένων που θα μπορούσαν να χρησιμοποιηθούν για τη διάρθρωση των παραγόμενων μοντέλων ή θα μπορούσαν να αντιπροσωπεύουν ένα ενδιαφέρον πρότυπο για την ανάλυση των δεδομένων.

Σκιαγράφηση

Η σκιαγράφηση [34, 48] είναι η διαδικασία της τυχαίας προβολής ενός υποσυνόλου των χαρακτηριστικών γνωρισμάτων. Είναι η διαδικασία της δειγματοληψίας της εισερχόμενης ροής δεδομένων. Η σκιαγράφηση έχει εφαρμοστεί σε σύγκριση διαφορετικών ροών δεδομένων και συνολικά ερωτήματα. Το βασικό μειονέκτημα της σκιαγράφησης είναι αυτό της ακρίβειας. Η Ανάλυση Κύριων Συνιστωσών (Principal Component Analysis - PCA) θα ήταν μια καλύτερη λύση σε εφαρμογές ροών δεδομένων [49].

Σύνοψη Δομών Δεδομένων

Η δημιουργία σύνοψης των δεδομένων αναφέρεται στη διαδικασία της εφαρμογής των τεχνικών περιλήψεων που είναι σε θέση να συνοψίζουν την εισερχόμενη ροή για περαιτέρω ανάλυση. Η ανάλυση κυματομορφών [45], ιστογράμματα, στιγμιαίες συχνότητες [34], έχουν προταθεί ως μορφές σύνοψης δεδομένων. Δεδομένου ότι η σύνοψη των δεδομένων δεν αντιπροσωπεύει το σύνολο των χαρακτηριστικών του συνόλου των δεδομένων, οι απαντήσεις παράγονται κατά προσέγγιση.

Συνυπολογισμός

Ο συνυπολογισμός είναι η διαδικασία υπολογισμού στατιστικών μέτρων, όπως τα μέσα και διακύμανση που συνοψίζουν την εισερχόμενη ροή δεδομένων. Η συγκεντρωτική αντιμετώπιση των δεδομένων θα μπορούσε να χρησιμοποιηθεί από τον αλγόριθμο εξόρυξης. Το πρόβλημα με τον συνυπολογισμό, είναι ότι δεν αποδίδει καλά με έντονα μεταβαλλόμενα δεδομένα. Η παράλληλη χρήση Online και Offline συνυπολογισμού μελετάται [31, 32, 33].

2.2.2 Τεχνικές που βασίζονται στις διαδικασίες

Οι τεχνικές που βασίζονται στις διαδικασίες είναι εκείνες οι μέθοδοι που τροποποιούν τις υπάρχουσες τεχνικές ή εφεύρουν νέες, προκειμένου να αντιμετωπιστούν οι υπολογιστικές προκλήσεις της επεξεργασίας ροών δεδομένων. Οι προσεγγιστικοί αλγόριθμοι, συρόμενο παράθυρο και ο αλγόριθμος διακριτότητας εξόδου απαρτίζουν αυτή την κατηγορία. Στη

συνέχεια παρουσιάζονται αυτές τις τεχνικές και η εφαρμογή της στο πλαίσιο της ανάλυσης δεδομένων.

Προσεγγιστικοί αλγόριθμοι

Οι προσεγγιστικοί αλγόριθμοι [48] έχουν τις ρίζες τους στο σχεδιασμό αλγορίθμων. Χρησιμοποιούνται σε σχεδίαση αλγορίθμων για δύσκολα υπολογιστικά προβλήματα. Οι αλγόριθμοι μπορούν να οδηγήσουν σε μια κατά προσέγγιση λύση μέσα σε κάποια όρια σφάλματος. Η ιδέα είναι ότι οι αλγόριθμοι εξόρυξης θεωρείται δύσκολο υπολογιστικό προβλήμα λόγω των χαρακτηριστικών του, της συνέχειας και της ταχύτητας και το περιβάλλον εφαρμογής που χαρακτηρίζεται από τους περιορισμένους πόρους. Οι προσεγγιστικοί αλγόριθμοι έχουν προσελκύσει ερευνητές ως άμεση λύση στα προβλήματα εξόρυξης δεδομένων. Ωστόσο, το πρόβλημα των ποσοστών των δεδομένων σε σχέση με τους διαθέσιμους πόρους δεν θα μπορούσε να επιλυθεί με τη χρησιμοποίηση προσεγγιστικών αλγορίθμων. Άλλα εργαλεία θα πρέπει να χρησιμοποιούνται μαζί με αυτούς τους αλγόριθμους προκειμένου να προσαρμοστούν στους διαθέσιμους υπολογιστικούς πόρους. Οι προσεγγιστικοί αλγόριθμοι έχουν χρησιμοποιηθεί σε διάφορες περιπτώσεις [36] .

Κυλιόμενο παράθυρο

Η έμπνευση πίσω από το κυλιόμενο παράθυρο είναι ότι ο χρήστης ασχολείται περισσότερο με την ανάλυση των πιο πρόσφατων δεδομένων. Έτσι, η λεπτομερής ανάλυση γίνεται για τα πιο πρόσφατα στοιχεία δεδομένων και συνοψίζονται οι εκδόσεις των παλαιών. Η ιδέα αυτή έχει υιοθετηθεί σε πολλές τεχνικές εξόρυξης δεδομένων [40].

Αλγόριθμος διακριτής εξόδου (AOG)

Ο Αλγόριθμος διακριτής εξόδου (Algorithm Output Granularity - AOG) [42, 43, 44], εισάγει την πρώτη προσέγγιση ανάλυση δεδομένων που λαμβάνει υπόψη τους διαθέσιμους πόρους και μπορεί να αντιμετωπίσει διακυμάνσεις σε πολύ υψηλές ταχύτητες δεδομένων, ανάλογα με τη διαθέσιμη μνήμη και την ταχύτητα επεξεργασίας που αποτελούν τους χρονικούς περιορισμούς. Ο αλγόριθμος AOG εκτελεί τοπική ανάλυση των δεδομένων σε συσκευές περιορισμένων πόρων που παράγουν ή να λαμβάνουν ροές δεδομένων. Ο αλγόριθμος AOG έχει χρησιμοποιηθεί σε ομαδοποίηση, ταξινόμηση και μέτρηση συχνότητα [42].

2.2.3. Τεχνικές Εξόρυξης Δεδομένων - Ταξινόμηση

Ένας αριθμός των αλγορίθμων έχουν προταθεί για την εξαγωγή γνώσης από μια συνεχή ροή πληροφοριών. Στην ενότητα αυτή, εξετάζουμε μόνο την ταξινόμηση, καθώς αυτή αντιπροσωπεύει την ερευνητική περιοχή στην οποία επικεντρώνεται η εργασία αυτή, η οποία στοχεύει στην ταξινόμηση των εικόνων βάσει εξαγωγής προτύπων.

Ταξινόμηση

Ο Wang και οι συνεργάτες [51] έχουν προτείνει ένα γενικό πλαίσιο για την έννοια της εξόρυξης σε ροές δεδομένων. Έχουν παρατηρήσει ότι οι αλγόριθμοι εξόρυξης δεδομένων που έχουν προτείνει μέχρι στιγμής δεν έχουν ασχοληθεί με την έννοια της περιπλάνησης κατά την εξέλιξη των δεδομένων. Η προτεινόμενη τεχνική χρησιμοποιεί έναν σταθμισμένο ταξινομητή που χρησιμοποιείται σε συνδυασμό με τα δεδομένα.

Η λήξη της ισχύος παλαιών δεδομένων στο εκάστοτε μοντέλο εξαρτάται από την κατανομή των δεδομένων. Γίνεται χρήση σύνθετων και πραγματικών δεδομένων για τη δοκιμή του αλγορίθμου και η σύγκριση μεταξύ ενός μεμονωμένου ταξινομητή και ενός συνδεδεμένου με τα δεδομένα ταξινομητή. Ο προτεινόμενος αλγόριθμος συνδυάζει πολλαπλούς σταθμισμένους ταξινομητές με εκτίμηση της αναμενόμενης ακρίβειας πρόβλεψής τους. Επίσης, η επιλογή ενός αριθμού ταξινομητών αντί να χρησιμοποιηθούν όλοι είναι μια επιλογή στο προτεινόμενο πλαίσιο χωρίς να χάνει την ακρίβεια κατά τη διαδικασία ταξινόμησης.

Ο Ganti και οι ομάδα του [41] έχουν αναπτύξει αναλυτικά έναν αλγόριθμο για τη συντήρηση του μοντέλου κατά την εισαγωγή και διαγραφή μπλοκ δεδομένων. Ο αλγόριθμος αυτός μπορεί να εφαρμοστεί σε κάθε στοιχειώδες μοντέλο εξόρυξης δεδομένων. Έχει περιγραφεί επίσης ένα γενικό πλαίσιο για την ανίχνευση των αλλαγών μεταξύ δύο συνόλων δεδομένων σχετικά με τα αποτελέσματα εξόρυξης δεδομένων που δίνουν. Οι παραπάνω δύο τεχνικές μορφοποιούνται σε δύο γενικούς αλγορίθμους: τον GEMM και τον FOCUS.

Οι αλγόριθμοι εφαρμόστηκαν σε μοντέλα δέντρων αποφάσεων και των μοντέλο συχνών στοιχειοσυνόλων. Ο αλγόριθμος GEMM δέχεται μια κατηγορία μοντέλων και ένα μοντέλο συντήρησης αλγορίθμων για την απεριόριστη επιλογή παράθυρου, έχει ως αποτέλεσμα ένα μοντέλο συντήρησης των δεδομένων, τόσο σε σχέση όσο και χωρίς εξάρτηση από το παράθυρο. Το πλαίσιο / αλγόριθμος FOCUS χρησιμοποιεί τη διαφορά μεταξύ των μοντέλων εξόρυξης ως την απόκλιση στα σύνολα δεδομένων. Ο Domingos και οι συνεργάτες του [38] έχουν αναπτύξει τον VFDT. Είναι ένα σύστημα μάθησης με δέντρα απόφασης που βασίζεται σε δέντρα Hoeffding. Χωρίζει το δένδρο με το τρέχον βέλτιστο χαρακτηριστικό λαμβάνοντας υπόψη ότι ο αριθμός των εξετασθέντων δεδομένων ικανοποιεί ένα στατιστικό μέτρο το οποίο είναι το όριο Hoeffding. Ο αλγόριθμος απενεργοποιεί επίσης το λιγότερο πιθανά φύλλα του δέντρου και απορρίπτει τις μη πιθανές ιδιότητες.

Ο Παπαδημητρίου και οι συνεργάτες του [49], πρότειναν το μοντέλο AWSOM (Arbitrary Window Stream mOdeling Method) για το πρόβλημα εύρεσης προτύπων από αισθητήρες. Ανέπτυξαν ένα αλγόριθμο μοναδικής εκτέλεσης για να ενημερώνει σταδιακά τα πρότυπα. Η μέθοδος αυτή απαιτεί μόνο O (log N) μνήμης, όπου N είναι το μήκος της ακολουθίας. Τα πειράματα διεξάγονται με πραγματικά και σύνθετα σύνολα δεδομένων. Ο Aggarwal και οι συνεργάτες του έχουν υιοθετήσει την ιδέα της μικρο-ομάδας που ορίστηκε στο CluStream στην κατηγοριοποίηση On-Demand [33], και δείχνει μεγάλη ακρίβεια. Η τεχνική αυτή χρησιμοποιεί την ομαδοποίηση των αποτελεσμάτων για την ταξινόμηση των δεδομένων με στατιστικά στοιχεία της ταξικής κατανομής σε κάθε ομάδα. Τελευταία [47] πρότειναν ένα online σύστημα ταξινόμησης το οποίο μπορεί να προσαρμόζεται σε μεταβαλλόμενα δεδομένα. Το σύστημα ξαναχτίζει το μοντέλο ταξινόμησης με τα πιο πρόσφατα παραδείγματα / δεδομένα. Χρησιμοποιώντας το ποσοστό σφάλματος ως οδηγός για την έννοια μετατόπιση, η συχνότητα αναδημιουργίας του μοντέλου και το μέγεθος του παραθύρου αναπροσαρμόζονται.

Ο Ding και οι συνεργάτες του [37] έχουν αναπτύξει ένα δέντρο απόφασης που βασίζεται στη δομή Peano. Έχει αποδειχθεί πειραματικά ότι είναι ένας γρήγορος στη δημιουργία αλγόριθμος το οποίο τον καθιστά κατάλληλο για εφαρμογές ροής (streaming). Ο Gaber και οι συνεργάτες του [42] έχουν αναπτύξει έναν αλγόριθμο «ελαφριάς ταξινόμησης» τον LWClass. Πρόκειται για μια παραλλαγή του LWC. Είναι κι αυτή μια τεχνική που βασίζεται στον AOG. Η ιδέα είναι να χρησιμοποιήσετε K-κοντινότερους γείτονες με επικαιροποίηση της συχνότητας εμφάνισης της κατηγορίας λόγω των χαρακτηριστικών των δεδομένων. Σε περίπτωση αντίφασης μεταξύ του εισερχόμενου ρεύματος και των αποθηκευμένων περιλήψεων των υποθέσεων, η συχνότητα μειώνεται. Σε περίπτωση που η συχνότητα γίνει μηδέν, όλες οι περιπτώσεις που βρίσκονται σε αυτή την κατηγορία αφαιρούνται από τη μνήμη.

Παράλληλα υπάρχουν αρκετές προσπάθειες για τον καθορισμό προτύπων για την εξόρυξη δεδομένων, όπως για παράδειγμα το 1999 στο συνέδριο European Cross Industry Standard Process for Data Mining (CRISP-DM 1.0) και του προτύπου του 2004 Java Data Mining (JDM 1.0). Αυτά τα πρότυπα εξελίσσονται και νεότερες εκδόσεις αυτών των προτύπων είναι υπό ανάπτυξη. Ανεξάρτητα από αυτές τις προσπάθειες τυποποίησης, ελεύθερα διαθέσιμα είναι και συστήματα λογισμικού ανοικτού κώδικα, όπως το R Project, Weka, KNIME, RapidMiner, jHepWork και άλλα έχουν γίνει άτυπα πρότυπα για τον ορισμό των data-mining διαδικασιών. Αξίζει να σημειωθεί ότι σε όλα τα συστήματα αυτά είναι δυνατή η εισαγωγή και εξαγωγή μοντέλων PMML (Predictive Model Markup Language) το οποίο παρέχει ένα πρότυπο τρόπο να εκπροσωπούν τα δεδομένα μοντέλων εξόρυξης, έτσι ώστε αυτά να μπορούν να

διαμοιραστούν μεταξύ των διαφόρων στατιστικών εφαρμογών [7]. Η PMML είναι μια βασισμένη σε XML γλώσσα που αναπτύχθηκε από το Data Mining Group (DMG), [3], μια ανεξάρτητη ομάδα που αποτελείται από πολλές εταιρείες εξόρυξης δεδομένων.Η PMML έκδοση 4.0 κυκλοφόρησε τον Ιούνιο του 2009 [3] [4] [5].

Εκτός από τη βιομηχανία με γνώμονα τη ζήτηση για τα πρότυπα και τη διαλειτουργικότητα, η επαγγελματική και η ακαδημαϊκή κοινότητα έχουν επίσης καταβάλει σημαντικές προσπάθειες στην εξέλιξη και την ακρίβεια των μεθόδων και των μοντέλων εξόρυξης δεδομένων. Ένα άρθρο που δημοσιεύθηκε σε μια έκδοση του 2008 του διεθνούς περιοδικού «*International Journal of Information Technology and Decision Making*» συνοψίζει τις αποτελέσματα μιας βιβλιογραφικής έρευνας που μελετά και αναλύει την εξέλιξη αυτή [8].

Ο κύριος επαγγελματικός φορέας στον τομέα αυτό είναι η Special Interest Group on Knowledge discovery and Data Mining (SIGKDD) [8]. Από το 1989 έχουν φιλοξενηθεί ετήσια διεθνή συνέδρια και πολλές δημοσιευμένες εργασίες της, [9] και από το1999 εκδίδει κάθε δύο χρόνια ένα διεθνές ακαδημαϊκό περιοδικό με τίτλο "SIGKDD Explorations» [10]. Άλλες διεθνή συνέδρια σχετικά με τον ερευνητικό τομέα της εξόρυξης δεδομένων περιλαμβάνουν τα παρακάτω:

- SDM - SIAM International Conference on Data Mining.
- EDM - International Conference on Educational Data Mining.
- ECDM - European Conference on Data Mining.
- PAKDD - The annual Pacific-Asia Conference on Knowledge Discovery and Data Mining.
- DMIN - International Conference on Data Mining.
- DMKD - Research Issues on Data Mining and Knowledge Discovery.
- ECML-PKDD - European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases.
- ICDM - IEEE International Conference on Data Mining.
- MLDM - Machine Learning and Data Mining in Pattern Recognition.

2.2.4. Αλγόριθμοι Κατηγοριοποίησης

Τύποι αλγορίθμων εξόρυξης δεδομένων

Η εξόρυξη δεδομένων περιλαμβάνει τους ακόλουθους τύπους αλγορίθμων:

- Αλγόριθμοι ταξινόμησης πρόβλεψης μίας ή περισσότερων διακριτών μεταβλητών, με βάση διάφορα χαρακτηριστικά στο σύνολο δεδομένων.
- Αλγόριθμοι παλινδρόμησης που προβλέπουν μία ή περισσότερες συνεχείς μεταβλητές, όπως κέρδη ή ζημίες, βάσει άλλων χαρακτηριστικών στο σύνολο δεδομένων.
- Αλγόριθμοι τμηματοποίησης που χωρίζουν τα δεδομένα σε ομάδες, ή υποομάδες, των στοιχείων που έχουν παρόμοιες ιδιότητες.
- Αλγόριθμοι συσχέτισης οι οποίοι βρίσκουν συσχετίσεις μεταξύ των διαφορετικών ιδιοτήτων σε ένα σύνολο δεδομένων. Η πιο κοινή εφαρμογή αυτού του είδους του αλγορίθμου είναι για τη δημιουργία κανόνων συσχέτισης, και μπορούν να χρησιμοποιηθούν σε μια ανάλυση καλαθιού της αγοράς.
- Αλγόριθμοι ακολουθιακής ανάλυσης οι οποίοι συνοψίζουν συχνές ακολουθίες ή επεισόδια σε δεδομένα, όπως μια ακολουθία διαδικασιών στο διαδίκτυο.

Εφαρμογή Αλγορίθμων

Η επιλογή του καλύτερου αλγορίθμου που θα χρησιμοποιηθεί για μια συγκεκριμένη εργασία μπορεί να είναι μια ξεχωριστή πρόκληση. Ενώ μπορεί να χρησιμοποιηθούν διαφορετικοί αλγόριθμοι για την εκτέλεση της ίδιας εργασίας, κάθε αλγόριθμος παράγει ένα διαφορετικό αποτέλεσμα, και μερικοί αλγόριθμοι μπορούν να παράγουν περισσότερους από έναν τύπους του αποτελέσματος. Για παράδειγμα, μπορείτε να χρησιμοποιήσετε τον αλγόριθμο δένδρων απόφασης όχι μόνο για την πρόβλεψη, αλλά και σαν ένα τρόπο για να μειωθεί ο αριθμός των στηλών σε ένα σύνολο δεδομένων, επειδή το δέντρο αποφάσεων μπορεί να εντοπίσει τις στήλες που δεν επηρεάζουν το τελικό μοντέλο εξόρυξης.

Είναι εφικτό, επίσης, να μην χρειάζεται να χρησιμοποιηθούν αλγόριθμοι ανεξάρτητα. Σε μια ενιαία λύση εξόρυξης δεδομένων μπορεί να χρησιμοποιηθεί κάποιος αλγόριθμος για την εξερεύνηση δεδομένων, και στη συνέχεια να χρησιμοποιηθούν άλλοι αλγόριθμοι για να προβλεφθεί ένα συγκεκριμένο αποτέλεσμα με βάση αυτά τα δεδομένα. Για παράδειγμα, μπορεί να χρησιμοποιηθεί ένας αλγόριθμος ο οποίος αναγνωρίζει πρότυπα, για να σπάσει τα δεδομένα σε ομάδες που είναι περισσότερο ή λιγότερο ομοιογενείς και, στη συνέχεια να χρησιμοποιηθούν τα αποτελέσματα για τη δημιουργία ενός καλύτερου μοντέλου δέντρων απόφασης.

Τα μοντέλα εξόρυξης μπορούν να προβλέψουν τις τιμές, την παραγωγή αναφορών για τα δεδομένα, και να βρουν κρυμμένες συσχετίσεις. Για να διευκολυνθεί η επιλογή αλγορίθμου για την εξόρυξη στα δεδομένα, ο ακόλουθος πίνακας παρέχει προτάσεις για τους αλγόριθμους που θα μπορούσαν να χρησιμοποιηθούν για συγκεκριμένες εργασίες.

Πρόβλεψη διακριτών χαρακτηριστικών. Για παράδειγμα, η πρόβλεψη του κατά πόσον ο αποδέκτης μιας διαφημιστικής καμπάνιας μέσω αλληλογραφίας θα αγοράσει ένα προϊόν.	Decision Trees Αλγόριθμος Naive Bayes Αλγόριθμος Clustering Αλγόριθμος Neural Network Αλγόριθμος
Πρόβλεψη συνεχών χαρακτηριστικών. Για παράδειγμα, η πρόβλεψη των πωλήσεων του επόμενου έτους.	Decision Trees Αλγόριθμος Time Series Αλγόριθμος
Πρόβλεψη μιας ακολουθίας. Για παράδειγμα, η εκτέλεση μιας ανάλυσης ακολουθιών διαδρομών στο Web site μιας εταιρείας.	Sequence Clustering Αλγόριθμος
Εύρεση ομάδων κοινών στοιχείων σε συναλλαγές. Για παράδειγμα, χρήση της ανάλυσης ενός καλαθιού αγοράς για την πρόταση επιπλέον προϊόντα για αγορά σε έναν πελάτη.	Association Αλγόριθμος Decision Trees Αλγόριθμος
Εύρεση ομάδων από παρόμοια στοιχεία. Για παράδειγμα, επεξεργασία στοιχείων από το τμήμα δημογραφικών δεδομένων και οργάνωση σε ομάδες για να κατανοηθούν καλύτερα οι σχέσεις μεταξύ των διάφορων χαρακτηριστικών.	Clustering Αλγόριθμος Sequence Clustering Αλγόριθμος

3. Βάση Δεδομένων Oracle 11g / Σχετικά Εργαλεία

Στη συνέχεια παραθέτονται οι διαθέσιμες υποδομές και λειτουργικότητα από την σχεσιακή βάση δεδομένων Oracle στην τελευταία της έκδοση (11g).

3.1. Υποδομές – Λειτουργικότητα

Η Oracle Database (που συνήθως αναφέρονται ως σχεσιακή βάση δεδομένων Oracle RDBMS ή απλώς ως Oracle) είναι ένα αντικείμενο - σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων(ORDBMS) [1], που παράγονται και διατίθενται στο εμπόριο από την εταιρεία Oracle Corporation.

3.2. Χρησιμοποιούμενα Εργαλεία

Από την πλειάδα διαθέσιμων υποδομών που παρέχονται από την σχεσιακή βάση δεδομένων Oracle 11g έγινε χρήση ενός μέρους για τις ανάγκες της παρούσας εργασίας.

Έτσι μπορούμε να αναφερθούμε στις παρακάτω λειτουργικές ενότητες που παρέχονται από την Oracle και οι οποίες χρησιμοποιήθηκαν για την επεξεργασία εικόνων και την εξόρυξη δεδομένων από αυτές.

- PL/SQL: Η γλώσσα προγραμματισμού που παρέχει συνδυαστικές με την απλή SQL δυνατότητες. Η γλώσσα αυτή προγραμματισμού αποτελεί μια επέκταση στη λογική της SQL και κάνει δυνατή την υλοποίηση αρκετών tasks που χρησιμοποιούνται στα πλαίσια της εργασίας.
- Oracle Multimedia: Το πακέτο της Oracle που παρέχει εξειδικευμένες λειτουργίες και δυνατότητες σχετικά με την επεξεργασία, εισαγωγή και κατηγοριοποίηση πολυμεσικού περιεχομένου. Λόγω της μεγάλης σημασίας που παίζει το εν λόγω πακέτο στην εργασία περιγράφεται αναλυτικότερα στην επόμενη ενότητα.
- SQL Loader. Εργαλείο για τη μαζική εισαγωγή και μεταφορά δεδομένων μεταξύ μορφών. Είναι δυνατή η χρήση του για αντιγραφή δεδομένων από βάση σε βάση ή από οποιαδήποτε δομημένη μορφή σε κάποια Oracle βάση δεδομένων. Η δυνατότητα αυτή χρησιμοποιείται για τη φόρτωση των εικόνων στη βάση.
- Πρόσβαση σε φυσικά αρχεία. Η Oracle δίνει τη δυνατότητα για πρόσβαση σε φυσικά αρχεία και φόρτωσή τους με κατάλληλη μορφή σε κατάλληλα σχεδιασμένους πίνακες. Η δυνατότητα αυτή χρησιμοποιείται για τη φόρτωση των εικόνων στη βάση.
- Γραφικό περιβάλλον ανάπτυξης TOAD – παρεχόμενο από την εταιρεία Quest ειδικά για τη βάση Oracle. Αποτελεί το γραφικό περιβάλλον που χρησιμοποιήθηκε για την ανάπτυξη όλων των PL/SQL διαδικασιών αλλά και την σχεδίαση/υλοποίηση του σχήματος της βάσης δεδομένων που χρησιμοποιήθηκε στα πλαίσια αυτής της εργασίας.

3.3. Το πακέτο Oracle Multimedia

Το πακέτο Oracle Multimedia (πρώην InterMedia Oracle) είναι μια επιπρόσθετη λειτουργικότητα της σχεσιακής βάσης δεδομένων Oracle Standard Edition και Enterprise Edition. Παρέχει μια πλατφόρμα για ένα ευρύ φάσμα εφαρμογών πολυμέσων - διαχείριση νοσοκομειακής εικόνας για νοσοκομεία και ερευνητικούς οργανισμούς, για τις φαρμακευτικές εταιρείες, διαχείρισης πολυμέσων σχετικών με την τέχνη και τα μουσεία, πολλές εταιρίες και το δημόσιο τομέα, χρηματοπιστωτικά ιδρύματα, διαχείριση πόρων για δημοσίευση στο διαδίκτυο, ασφάλειας βίντεο

και πολλές άλλες πολυμεσικές εφαρμογές με υψηλές απαιτήσεις. Οι δυνατότητες που κάνουν τη διαφορά είναι οι ακόλουθες:

Η Oracle Multimedia επιτρέπει την αποτελεσματική διαχείριση και ανάκτηση των πολυμέσων (εικόνας, ήχου, και βίντεο) δεδομένων σε Oracle Database με:

- Αποθήκευση υψηλής απόδοσης και ανάκτηση με χρήση Oracle SecureFiles.
- Ασφαλής, κλιμακούμενη διαχείριση των πληροφοριών.
- Υποστήριξη για τις πιο δημοφιλείς μορφές πολυμέσων με ενσωματωμένη εξαγωγή metadata και βασική επεξεργασία εικόνας.
- Προγραμματιστικές διεπαφές (Application Programming Interfaces – APIs), Java Server Pages - JSPs, Servlets, PL / SQL διαδικασίες και λειτουργίες για να απλοποιηθεί η ανάπτυξη εφαρμογών.

Η ολοκληρωμένη υποστήριξη Oracle πολυμέσων για το περιεχόμενο DICOM, το ευρέως υιοθετηθεί πρότυπο για ιατρικές εικόνες, βίντεο και δομημένες εκθέσεις παρέχει τις παρακάτω υπηρεσίες:

- Διευκολύνει την ανάπτυξη των μεγάλων αρχείων με DICOM (CTS, MRIs, X-Rays, Υπερηχογραφήματα, παθολογικές εικόνες) τα οποία διαχειρίζονται ασφαλώς χρησιμοποιώντας τα εργαλεία της Oracle Database.
- Επιτρέπει την ανάπτυξη της εικόνας με δυνατότητα EMRs (ηλεκτρονικά αρχεία).
- Περιλαμβάνει ένα πλούσιο σύνολο της ασφάλειας, της προστασίας της ιδιωτικής ζωής, τη συμμόρφωση και δυνατότητες επικύρωσης ώστε να είναι ανοικτή και επεκτάσιμη η πρόσβαση σε αρχεία και εφαρμογές στην υγειονομική περίθαλψη και τις βιοεπιστήμες.

Το πακέτο Oracle Multimedia (με προηγούμενη ονομασία Oracle InterMedia) επιτρέπει στην σχεσιακή βάση δεδομένων της Oracle την αποθήκευση, διαχείριση και ανάκτηση των εικόνων, σε DICOM μορφή ιατρικών εικόνων και άλλων αντικειμένων ήχου, βίντεο ή άλλα ετερογενή δεδομένα πολυμεσικής επικοινωνίας στο πλαίσιο μιας ολοκληρωμένης αντιμετώπισης με άλλες πληροφορίες των επιχειρήσεων. Το πακέτο Oracle Multimedia επεκτείνει την αξιοπιστία της βάσης δεδομένων Oracle, τη διαθεσιμότητα και τη διαχείριση δεδομένων για το περιεχόμενο των πολυμέσων στις κλασσικές εφαρμογές, στην ιατρική, στο Διαδίκτυο, στο ηλεκτρονικό εμπόριο, και στα μέσα ενημέρωσης όπου δίνεται η δυνατότητα για ανάπτυξη πλούσιων εφαρμογών.

Η παράγραφος αυτή περιλαμβάνει τις εξής ενότητες:
- Αντικείμενα πολυμέσων
- Αποθήκευση πολυμέσων

3.4. Αντικείμενα Πολυμέσων

Το πακέτο Oracle Multimedia παρέχει τους εξής τύπους αντικειμένων πολυμέσων:

- ORDAudio,
- ORDDoc,
- ORDImage,
- ORDVideo και
- SI_StillImage

και μεθόδους για:

- Εξαγωγή metadata και ιδιότητες από δεδομένα πολυμέσων Ενσωμάτωση metadata που δημιουργούνται από τις εφαρμογές σε αρχεία εικόνας.
- Λήψη και διαχείριση δεδομένων πολυμέσων από την Oracle Multimedia, Web servers, συστήματα αρχείων, καθώς και άλλους servers.
- Εκτέλεση εργασιών χειρισμού των δεδομένων εικόναςν Oracle Multimedia με τη δυνατότητα πρόβλεψης του τύπου του αντικειμένου ORDDicom και μεθόδους για την αποθήκευση, διαχείριση και η επεξεργασία της DICOM μορφή ιατρικών εικόνων και άλλων δεδομένων.
- Σύνταξη SQL για σύνθετα αντικείμενα.

Η σύνταξη αντικειμένων για την πρόσβαση σε χαρακτηριστικά μέσα σε ένα πολύπλοκο αντικείμενο ακολουθεί τη λογική της σύνταξης με τη διαδρομή αντικειμένων χωρισμένων με τελεία (dot syntax) :

variable.data_attribute

Η σύνταξη για την κλήση των μεθόδων ενός σύνθετου αντικειμένου ακολουθεί επίσης την λογική dot syntax:

variable.function(παράμετρος_1, παράμετρος_2, ...)

Σύμφωνα με τις συνιστώμενες πρακτικές προγραμματισμού, ένα πλήρες σύνολο μεθόδων πρόσβασης στις ιδιότητες και τα χαρακτηριστικά αντικειμένων πολυμέσων (getters μέθοδοι) καθώς και οι μέθοδοι καθορισμού των προαναφερθέντων ιδιοτήτων και χαρακτηριστικών (setters μέθοδοι) προβλέπονται για κάθε τύπο μέσου

3.5. Αποθήκευση Πολυμέσων

Το πακέτο Oracle Multimedia παρέχει τον τύπο αντικειμένου ORDSource και των αντίστοιχων μεθόδων για την διαχείριση και αποθήκευση δεδομένων πολυμέσων και της αντίστοιχης πηγής δεδομένων (data source).

Οι παρακάτω τύποι αντικειμένων αποτελούν εξειδικεύσεις του προαναφερθέντος τύπου ORDSource για τους αντίστοιχους τύπους πολυμέσων αντίστοιχα

- ORDAudio, για τη διαχείριση πολυμέσων σχετικών με ήχο
- ORDDoc, για τη διαχείριση πολυμεσικών εγγράφων
- ORDImage, για τη διαχείριση πολυμέσων εικόνας, και
- ORDVideo για τη διαχείριση πολυμέσων εικόνας.

4. Επεξεργασία Εικόνας με Χρήση Σχεσιακής Βάσης Δεδομένων

Στο προηγούμενο κεφάλαιο ασχοληθήκαμε με την ανάλυση και τη σχεδίαση του συστήματος. Σε αυτό το κεφάλαιο θα περιγράψουμε την υλοποιημένη εφαρμογή.

4.1. Γενικά

Στα κεφάλαιο αυτό γίνεται παρουσίαση τόσο του σταδίου ανάλυσης και σχεδιασμού που έγινε για το πρόβλημα της επεξεργασίας εικόνας με τη χρήση σχεσιακής βάσης δεδομένων, όσο και του σταδίου υλοποίησης όσων προέκυψαν από το στάδιο της ανάλυσης.

Έτσι στην επόμενη ενότητα ξεκινάμε από την περιγραφή της αρχιτεκτονικής και των απαιτούμενων ενεργειών / προϋποθέσεων για την διεξαγωγή επεξεργασίας και αναζήτησης εικόνων στη βάση δεδομένων. Στο δεύτερο μέρος γίνεται αναλυτική τεχνική παρουσίαση όσων γίνονται καθώς και των αλγορίθμων/ τεχνολογιών που χρησιμοποιήθηκαν στην όλη διαδικασία, ενώ το πειραματικό μέρος, η εφαρμογή δηλαδή όσων υλοποιήθηκαν αναλύεται διεξοδικά στο επόμενο κεφάλαιο.

4.2. Περιγραφή Αρχιτεκτονικής

Στο ενότητα αυτό θα περιγραφεί η προσέγγιση που ακολουθήθηκε στα πλαίσια της παρούσας εργασίας σε επίπεδο αρχιτεκτονικής.

Πιο συγκεκριμένα, καθεμία από τις ακόλουθες ενότητες περιγράφει τα διαδοχικά στάδια σε επίπεδο αρχιτεκτονικής και αναλύει το θεωρητικό υπόβαθρο, καθώς και την αντίστοιχη υποδομή που σχεδιάζεται να χρησιμοποιηθεί στα πλαίσια της παρούσας εργασίας.

4.2.1. Ανάκτηση Ψηφιακών Εικόνων

Βασική προϋπόθεση για την διεξαγωγή πειραματικών εφαρμογών διαδικασιών εξόρυξης δεδομένων σε πολυμεσικό περιεχόμενο είναι η ύπαρξη του περιεχομένου. Έτσι θα γίνει μαζική ανάκτηση εικόνων για την εισαγωγή τους στη βάση δεδομένων και την περαιτέρω επεξεργασία / μετατροπή / διαχείρισή τους με χρήση του πακέτου Oracle Multimedia.

Το πολυμεσικό αυτό περιεχόμενο θέλουμε να έχει τα εξής χαρακτηριστικά:

- **Υποστηριζόμενη από το πακέτο Oracle Multimedia μορφή.**

Τα υποστηριζόμενα από την Oracle format εικόνων φαίνονται στον ακόλουθο πίνακα.

Τύπος Εικόνας	Επέκταση Αρχείου
BMPF	.bmp
CALS	.cal
FPIX	.fpx
GIFF	.gif
JFIF	.jpg / .jpeg
PBMF, PGMF, PPMF, και PNMF	.pbm, .pgm, .ppm, .pnm
PCXF	.pcx
PICT	.pct

PNGF	.png
PRIX	.rpx
RASF	.ras
TGAF	.tga
TIFF	.tif
WBMP	.wbmp

Για τις ανάγκες της παρούσας εργασίας και δεδομένου ότι είναι ο ευρύτερα διαδεδομένος τύπος εικόνων στο διαδίκτυο θα επιλεγεί ο τύπος JFIF.

- **Ικανό όγκο δεδομένων**

Οι εικόνες πρέπει να έχουν έναν ικανό αριθμό για να μπορέσουν να τρέξουν αποτελεσματικά οι αλγόριθμοι εξόρυξης δεδομένων. Ο λόγος είναι ότι πρέπει το σύνολο των διαθέσιμων δεδομένων να μοιραστεί στο λεγόμενο σύνολο δεδομένων εκπαίδευσης (training data) και στα δεδομένα εφαρμογής του αλγορίθμου (test data). Ως μια ικανή τάξη μεγέθους ορίζονται τα 1000 δείγματα ώστε να μπορούμε να έχουμε ικανό δείγμα τόσο για εκπαίδευση όσο και για δοκιμές [11][12]

- **Σαφή κατηγοριοποίηση**

Τα δεδομένα (εικόνες) πρέπει να επιλεγούν με τέτοιο τρόπο ώστε να καθίσταται δυνατή η κατηγοριοποίησή τους με βάση κάποιο χαρακτηριστικό. Σαν παράδειγμα αναφέρεται η διαφοροποίηση θεματικού περιεχομένου των εικόνων σε 4 διαφορετικές ενότητες. [13]

- **Περιορισμένο μέγεθος**

Για την εξοικονόμηση υπολογιστικών πόρων αλλά και τη δυνατότητα σχετικά ταχέων εκτελέσεων των πειραμάτων θα επιλεγεί σχετικά μικρός μέσος όγκος ανά εικόνα (μικρότερη του 1MByte).

Για την ανάκτηση των προαναφερθέντων εικόνων θα ανατρέξουμε στο διαδίκτυο και με αντίστοιχες υπολογιστικές υποδομές θα βρούμε κατάλληλο περιεχόμενο για τα σχετικά πειράματα.

4.2.2. Εισαγωγή Ψηφιακών Εικόνων σε Σχεσιακή Βάση Δεδομένων

Για την εισαγωγή των εικόνων που θα ανακτηθούν είναι απαραίτητο να αναπτυχθεί αυτοματοποιημένη διαδικασία.

Η αυτοματοποιημένη αυτή διαδικασία θα πρέπει να μπορεί να διαβάσει μαζικά εικόνες ανά κατηγορία και να τις εισάγει στη βάση δεδομένων σε κατάλληλη μορφή για επεξεργασία. Παράλληλα πρέπει να υπάρχει κατάλληλη υποδομή για τον χαρακτηρισμό αναφοράς των εικόνων, ήτοι την αρχική κατηγοριοποίηση των εικόνων στο στάδιο του φορτώματος ώστε να είναι εφικτός ο έλεγχος της όλης διαδικασίας κατηγοριοποίησης με τους αλγόριθμους εξόρυξης δεδομένων.

Δεδομένου ότι θα γίνει χρήση του πακέτου Oracle Multimedia πρέπει να ληφθεί υπόψη ο τρόπος ανάγνωσης ώστε να καθίσταται δυνατή η επόμενη φάση, αυτή της μετατροπής και επεξεργασίας των προς φόρτωμα εικόνων.

4.2.3. Μετατροπή και Επεξεργασία Ψηφιακών Εικόνων

Εξαγωγή Χαρακτηριστικών

Μετά το στάδιο της εισαγωγής των πρωτογενών δεδομένων εικόνας στη βάση δεδομένων προδιαγράφεται η εξαγωγή των χαρακτηριστικών της εικόνας αυτής όπως προδιαγράφεται και από το πακέτο Oracle Multimedia.

Πιο συγκεκριμένα παρέχεται ο τύπος δεδομένων ORDImageSignature ο οποίος χρησιμοποιείται για την εξαγωγή της «ψηφιακής υπογραφής» κάθε εικόνας. Στη συνέχεια δίνεται η δυνατότητα κατηγοριοποίησης / ομαδοποίησης των εικόνων βάσει των χαρακτηριστικών – «ψηφιακής υπογραφής».

Επεξεργασία και Μετατροπή Εικόνων

Μετά την εισαγωγή σε κατάλληλη μορφή δεδομένων των αρχικών εικόνων ανά κατηγορία προδιαγράφεται η επεξεργασία τους και η μετατροπή τους στον τύπο που παρέχεται από το πακέτο Oracle Multimedia.

Πιο συγκεκριμένα μετά την μετατροπή της εικόνας σε BLOB (τύπος δεδομένων μεγάλου όγκου ψηφιακών – binary δεδομένων) πρέπει η υποδομή που θα αναπτυχθεί να μετατρέπει τις εικόνες σε ORDImage για την εκμετάλλευση των διαθέσιμων μεθόδων που παρέχονται από την Oracle.

Πράγματι όπως φαίνεται και στον Πίνακα 1 γίνεται μετατροπή από BLOB σε ORDImage και παράλληλη εξαγωγή της ψηφιακής υπογραφής (ORDImageSignature) που περιγράφηκε στο προηγούμενο κεφάλαιο.

4.2.4. Αναζήτηση Προτύπων / Data Mining

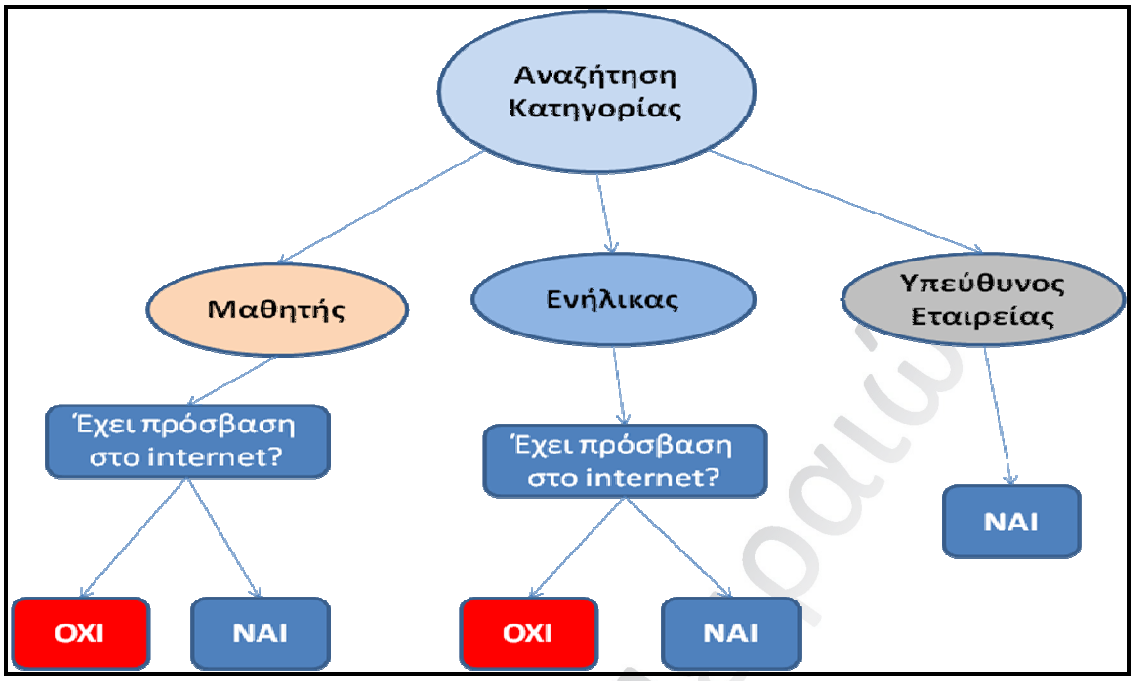
Για την εξόρυξη δεδομένων προδιαγράφονται στη βιβλιογραφία αρκετοί αλγόριθμοι αναζήτησης προτύπων, τα οποία πρότυπα θα χρησιμοποιηθούν στη συνέχεια για την αναζήτηση και κατηγοριοποίηση του περιεχομένου. Στα πλαίσια της παρούσας εργασίας θα γίνει χρήση των παρακάτω κυρίων αλγορίθμων, οι οποίοι και θα συγκριθούν για την εξαγωγή συμπερασμάτων:

- Αλγόριθμος Decision Trees
- Αλγόριθμος Naïve Bayes
- Αλγόριθμος Support Vector Machines
- Αλγόριθμος Logistic Regression

Στη συνέχεια περιγράφονται οι προαναφερθέντες αλγόριθμοι.

Αλγόριθμος Decision Trees

Τα δέντρα απόφασης (decision trees - DT) [15] είναι ισχυρά και δημοφιλή εργαλεία για την ταξινόμηση και πρόβλεψη. Η ελκυστικότητα των δέντρων απόφασης οφείλεται στο γεγονός ότι, σε αντίθεση με τα νευρωνικά δίκτυα, τα δέντρα απόφασης αντιπροσωπεύονται επίσης από κανόνες. Οι κανόνες μπορούν εύκολα να εκφραστούν έτσι ώστε οι άνθρωποι να μπορούν να τους κατανοήσουν ή ακόμη και να τους χρησιμοποιήσουν απευθείας σε μια γλώσσα σχετική με βάσεις δεδομένων όπως η SQL (Structured Query Language) , έτσι ώστε τα αρχεία που εμπίπτουν σε μια συγκεκριμένη κατηγορία να μπορούν να ανακτηθούν.



Εικόνα 4.1. Αλγόριθμος Decision Tree

Σε ορισμένες εφαρμογές, η ακρίβεια της κατάταξης ή πρόβλεψης είναι το μόνο πράγμα που έχει σημασία. Σε τέτοιες καταστάσεις δεν ενδιαφερόμαστε πώς ή γιατί το μοντέλο λειτουργεί. Σε άλλες περιπτώσεις, η ικανότητα να αιτιολογηθεί ο λόγος που πάρθηκε μια απόφαση, είναι ζωτικής σημασίας. Στο μάρκετινγκ κάποιος περιγράφει τις κατηγορίες πελατών, ώστε να μπορέσουν οι επαγγελματίες του μάρκετινγκ να χρησιμοποιήσουν αυτή τη γνώση για την έναρξη μιας επιτυχούς διαφημιστικής καμπάνιας. Οι ειδικοί του χώρου πρέπει να αναγνωρίσουν και να εγκρίνουν μια τέτοια γνώση, και για αυτό χρειαζόμαστε καλές περιγραφές. Υπάρχει μια ποικιλία από αλγόριθμους για την κατασκευή δέντρων αποφάσεων που μοιράζονται την επιθυμητή ιδιότητα της «περιγραψιμότητας».

Το δέντρο αποφάσεων είναι ένας ταξινομητής με τη μορφή μιας δομής δέντρου (Εικόνα 4.1), όπου κάθε κόμβος είναι είτε:

- Ένας κόμβος-φύλλο – που δείχνει την τιμή του χαρακτηριστικού στόχου (τάξη) των παραδειγμάτων, ή
- Ένας κόμβος απόφασης – που προσδιορίζει κάποια δοκιμή που θα πραγματοποιηθεί σε ένα μόνο χαρακτηριστικό υπό εξέταση, με το ένα σκέλος και το αντίστοιχο υπο-δέντρο για κάθε πιθανό αποτέλεσμα της δοκιμής.

Ένα δέντρο απόφασης μπορεί να χρησιμοποιηθεί για να χαρακτηρισθεί ένα ενδεχόμενο/συμβάν ξεκινώντας από τη ρίζα του δέντρου και να κινούμενο μέσα σε αυτό μέχρι έναν κόμβο-φύλλο, το οποίο προβλέπει την ταξινόμηση του ενδεχόμενου/συμβάντος.

Στην Εικόνα 4.1 φαίνεται ένα δέντρο απόφασης για την στόχευση ανά τις εκάστοτε –ενδεικτικές- κατηγορίες πιθανών πελατών της εκτέλεσης ή μη της προαναφερθείσας διαφημιστικής καμπάνιας.

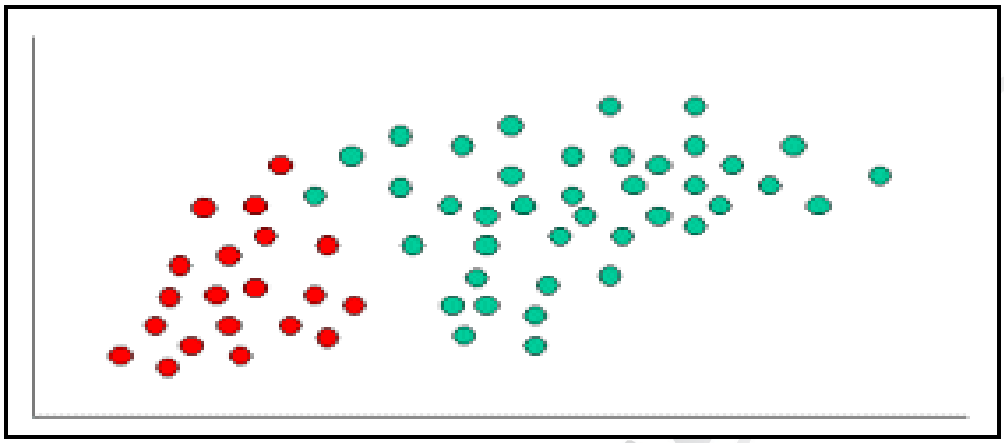
Αλγόριθμος Naïve Bayes

Η τεχνική ταξινόμησης Naive Bayes βασίζεται στη Bayesian Theorem (θεώρημα Bayes) και ενδείκνυται ιδιαίτερα όταν η πυκνότητα των δεδομένων είναι μεγάλη. Παρά την απλότητά του, ο αλγόριθμος Naive Bayes μπορεί να ξεπεράσει συχνά άλλες, πιο εξελιγμένες μεθόδους ταξινόμησης. Το θεώρημα Bayes αναφέρει λεκτικά ότι

Πιθανότητα (Β δεδομένου του Α) =

Πιθανότητα(Α και Β)/Πιθανότητα (Α) (1)

Για να γίνει κατανοητή την έννοια του Naive Bayes Ταξινόμηση, παρουσιάζεται το παράδειγμα που εμφανίζεται στην εικόνα 4.2. Όπως αναφέρεται, τα αντικείμενα μπορούν να ταξινομηθούν είτε ως ΠΡΑΣΙΝΑ είτε ως ΚΟΚΚΙΝΑ. Ο αλγόριθμος καλείται να χαρακτηρίσει τα νέα αντικείμενα που φθάνουν, δηλαδή, να αποφασίσει προς σε ποια κατηγορία ανήκουν, με βάση τα υπάρχοντα αντικείμενα.

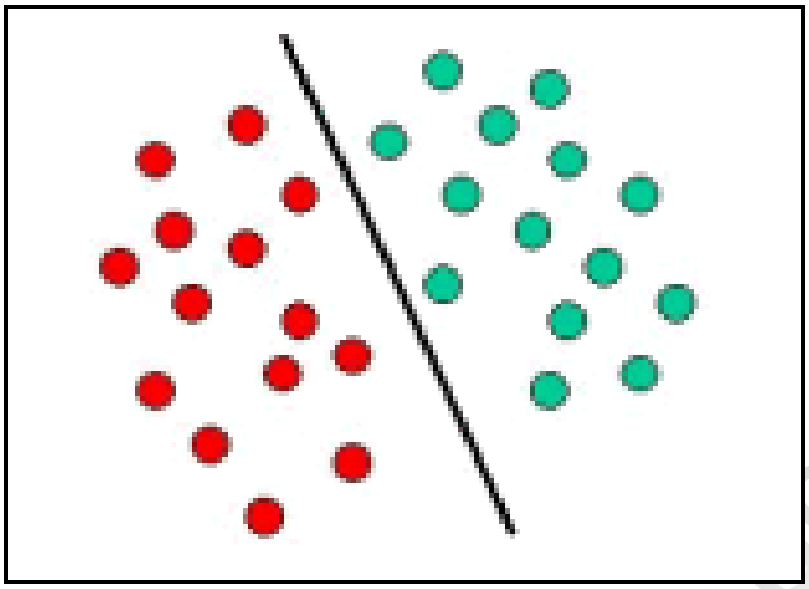


Εικόνα 4.2. Διάταξη για Αλγόριθμο Naïve Bayes

Δεδμένου ότι υπάρχουν διπλάσια ΠΡΑΣΙΝΑ αντικείμενα σε σχέση με τα ΚΟΚΚΙΝΑ, είναι λογικό να πιστεύουμε ότι ένα νέο αντικείμενο (το οποία δεν έχει παρατηρηθεί ακόμη) είναι δύο φορές πιο πιθανό να έχει χαρακτηριστικά της «ΠΡΑΣΙΝΗΣ» ομάδας αντί της «ΚΟΚΚΙΝΗΣ». Στη Bayesian ανάλυση, αυτή η πεποίθηση είναι γνωστή ως εκ των προτέρων πιθανότητα. Οι εκ των προτέρων πιθανότητες βασίζονται στην προηγούμενη εμπειρία, στην δυνατότητα δηλαδή το ποσοστό αυτό πράσινων και κόκκινων υπαρχόντων αντικείμενα, να χρησιμοποιείται για να προβλέψει τα επερχόμενα αντικείμενα / γεγονότα πριν πραγματικά αυτά καταφθάσουν / συμβούν.

Αλγόριθμος Support Vector Machines

Ο αλγόριθμος Support Vector Machines [14][16] βασίζονται στην έννοια των χώρων απόφασης που καθορίζουν τα αντίστοιχα όρια απόφασης. Ένας χώρος απόφασης είναι αυτός που χωρίζει το σύνολο των αντικειμένων που έχουν κάποια συγκεκριμένη ομάδα ιδιοτήτων (class membership). Ένα σχηματικό παράδειγμα φαίνεται στην εικόνα 4.3. Σε αυτό το παράδειγμα, τα αντικείμενα ανήκουν είτε στην ΠΡΑΣΙΝΗ τάξη είτε στην ΚΟΚΚΙΝΗ. Η διαχωριστική γραμμή ορίζει ένα όριο στη δεξιά πλευρά της οποίας όλα τα αντικείμενα είναι πράσινα και στα αριστερά της οποίας είναι κόκκινα όλα τα αντικείμενα. Κάθε νέο αντικείμενο, που εισάγεται στα δεξιά επισημαίνεται, δηλαδή ταξινομείται ως ΠΡΑΣΙΝΟ (ή για να ταξινομηθεί ως ΚΟΚΚΙΝΟ θα πρέπει να τοποθετηθεί στα αριστερά της διαχωριστικής γραμμής).



Εικόνα 4.3. Support Vector Machines

Αλγόριθμος Logistic Regression

Ο αλγόριθμος Logistic Regression (λογιστική παλινδρόμηση) [17] είναι μια προσέγγιση για την πρόβλεψη και ταξινόμηση αντίστοιχη με την παλινδρόμηση ελαχίστων τετραγώνων (Ordinary Least Squares - OLS). Ωστόσο, με τη λογιστική παλινδρόμηση, ο ερευνητής στοχεύει σε αποτέλεσμα διχοτόμησης. Αυτή η κατάσταση δημιουργεί προβλήματα για συνδυαζόμενη με την υπόθεση ότι οι αποκλίσεις και τα σφάλματα ακολουθούν ομοιόμορφη κατανομή. Αντί αυτού, είναι πιο πιθανό τα σφάλματα να ακολουθούν μια λογιστική κατανομή η οποία προσιδιάζει στην ομοιόμορφη κατανομή αλλά έχει μεγαλύτερες συγκεντρώσεις πιθανότητας σε συγκεκριμένες περιοχές.

Με τον αλγόριθμο Logistic Regression (λογιστική παλινδρόμηση), δεν υπάρχει κάποια προτυποποιημένη λύση. Και κάτι που κάνει τα πράγματα ακόμα πιο περίπλοκα, η έλλειψη προτυποποιημένης λύσης δεν είναι τόσο σαφώς ορισμένη όπως συμβαίνει με τον αλγόριθμο OLS.

Ο αλγόριθμος λογιστικής παλινδρόμησης μπορεί να χρησιμοποιηθεί για τις παρακάτω διαδικασίες:

- για την πρόβλεψη μιας εξαρτημένης μεταβλητής με βάση την συνεχή και / ή κατηγορηματική ανεξαρτησία και να προσδιοριστεί το μέγεθος της επίδρασης των ανεξάρτητων στις εξαρτημένες μεταβλητές,
- για να ταξινομήσει τη σχετική σημασία των ανεξάρτητων μεταβλητών,
- για να εκτιμήσει τις αλληλεπιδράσεις μεταβλητών και να κατανοηθεί η επίδραση των διαφόρων μεταβλητών.

4.3. Υλοποίηση

4.3.1. Βάση Δεδομένων / Σχετικές Υπηρεσίες

Για την υλοποίηση της παρούσας εργασίας χρησιμοποιήθηκε η τελευταία έκδοση της σχεσιακής βάσης δεδομένων Oracle 11g.

Πιο συγκεκριμένα εγκαταστάθηκε η έκδοση 11g Release 2. μέσω της αντίστοιχης διεπαφής (Εικόνα 4.4) .

Εικόνα 4.4.Εγκατάσταση Oracle 11g release 2

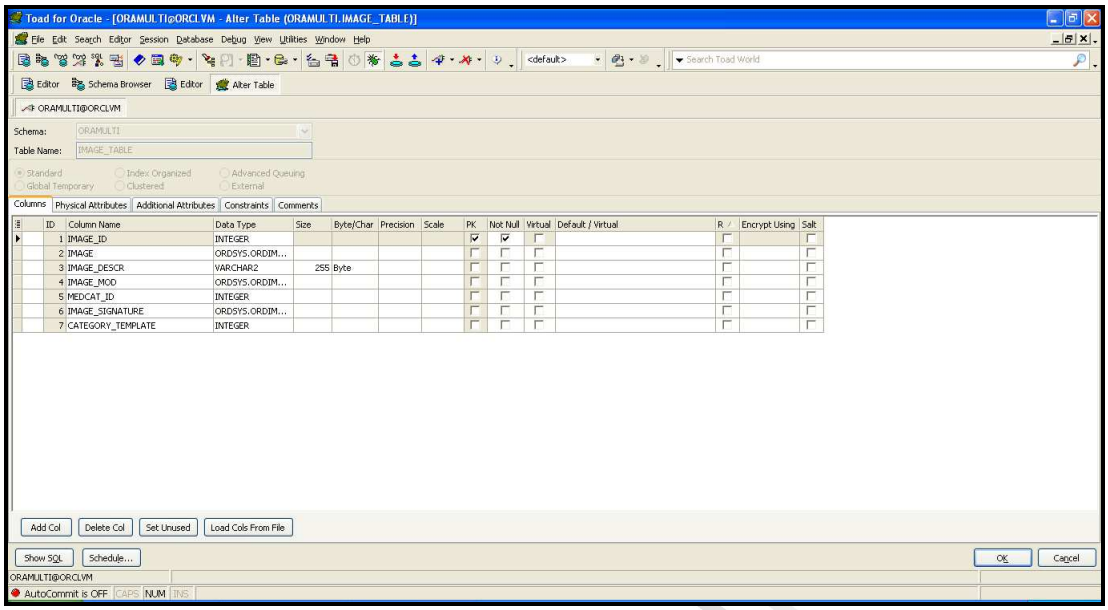
Στη συνέχεια δημιουργήθηκαν και ενεργοποιήθηκαν οι αντίστοιχες υπηρεσίες για την δυνατότητα (πιθανά απομακρυσμένης) σύνδεσης στη βάση δεδομένων.

Περιβάλλον Ανάπτυξης Εφαρμογής

Το περιβάλλον (IDE) που επελέγη τόσο για την –σε φάση ανάπτυξης εφαρμογών- διαχείρισης της βάσης δεδομένων (προσθήκη, τροποποίηση πινάκων, διεργασιών, κανόνων, εφαρμογών) όσο και για αυτή καθαυτή την ανάπτυξη της εφαρμογής και της υλοποίησης όσων πραγματεύεται αυτή η εργασία ήταν το TOAD for Oracle της εταιρείας Quest Software.

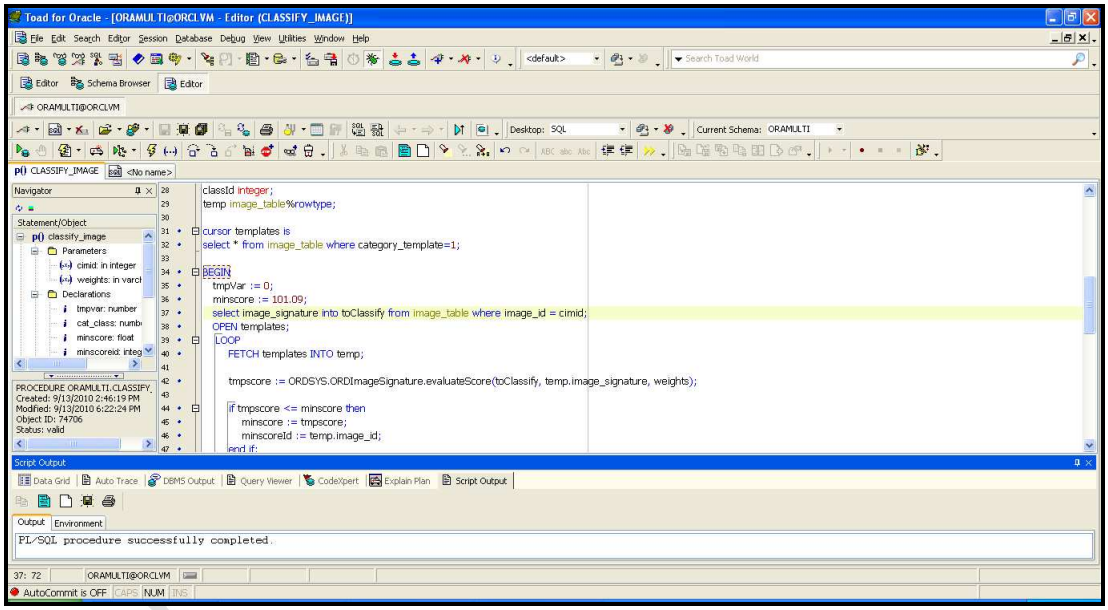
Ο λόγος που επελέγη το περιβάλλον ανάπτυξης αυτό είναι ότι συνεργάζεται άψογα με την συγκεκριμένη βάση δεδομένων, παρέχει πολλά εργαλεία ανάπτυξης, αλλά και συνοψίζει λειτουργίες ανάπτυξης εφαρμογής και διαχείρισης/διάρθρωσης βάσης δεδομένων σε ένα ενιαίο περιβάλλον με αποδοτικό τρόπο.

Στην Εικόνα 4.5 φαίνεται μια τυπική διαδικασία μορφοποίησης / επεξεργασίας / δημιουργίας καινούριου πίνακα στη βάση δεδομένων. Πιο συγκεκριμένα για τον τύπο κολώνας που απαιτείται στα πλαίσια της συγκεκριμένης εργασίας (Oracle Multimedia – ORDImage, ORDImageSignature) η δημιουργία / προσαρμογή των αντίστοιχων κολώνων δεν γίνεται από το υποφαινόμενο εργαλείο, αλλά μέσω εντολών SQL και PL/SQL που εκτελούνται από το αντίστοιχο εργαλείου του *Toad for Oracle* IDE (SQL Editor).

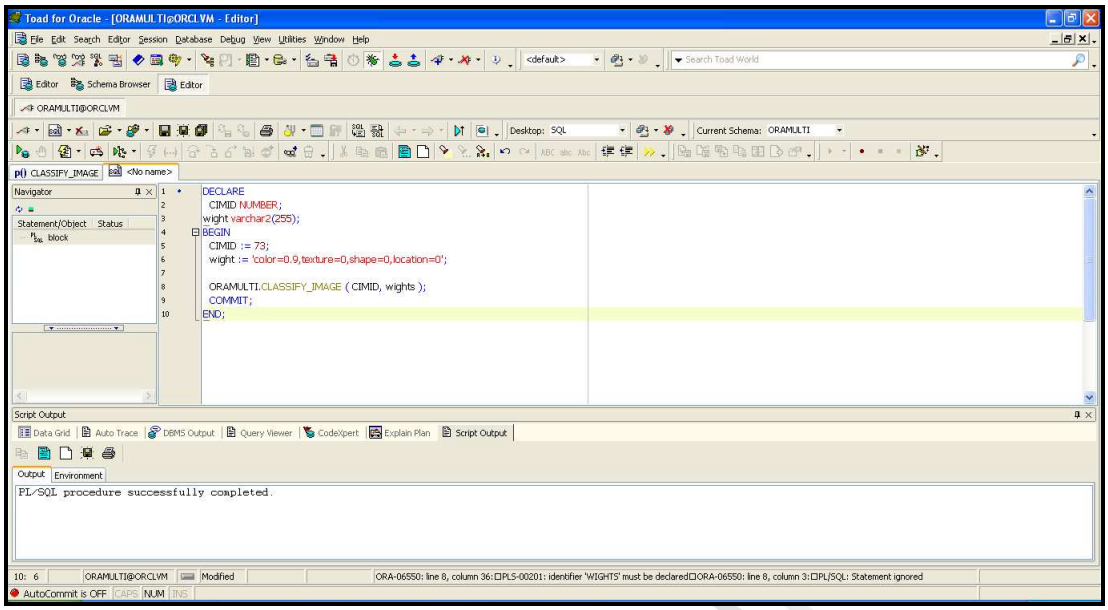


Εικόνα 4.5. Διαχείριση Πίνακα Βάσης Δεδομένων

Στην Εικόνα 4.6 φαίνεται το περιβάλλον ανάπτυξης διαδικασιών / εφαρμογής (procedures, functions κλπ), ενώ στην Εικόνα 4.7 φαίνεται η οθόνη εκτέλεσης των εφαρμογών/διαδικασιών που έχουν δημιουργηθεί.



Εικόνα 4.6. Ανάπτυξη Διαδικασιών (Procedures) / Εφαρμογής



Εικόνα 4.7. Εκτέλεση Διαδικασιών (Procedures)

4.3.2. Σχήμα Βάσης Δεδομένων

Στα πλαίσια της εργασίας αυτής τα δεδομένα θα πρέπει να διαμορφωθούν κατάλληλα και στη συνέχεια να φορτωθούν στη βάση της οποίας ο σχεδιασμός περιγράφεται στη συνέχεια και λαμβάνει ειδική μέριμνα για την υποδοχή των παραπάνω δεδομένων. Πρέπει να σημειωθεί ότι η δημιουργία της παραπάνω βάσης δεν είναι δημιουργία ενός απλού σχήματος μιας σχεσιακής βάσης δεδομένων, καθώς αποτελείται από ειδικούς τύπους αντικειμένων/δεδομένων που θα φορτωθούν στους πίνακες που θα δημιουργηθούν.

Μετά το τέλος της φόρτωσης των δεδομένων στη βάση (και της αντίστοιχης δημιουργίας υποδομής για ανάκτησή τους) θα πρέπει να τρέξουμε τη διαδικασία της ομαδοποίησης και της συσταδοποίησης των εικόνων με όλους τους αλγορίθμους που περιγράφονται στην παράγραφο 2.3. Η διαδικασία αυτή περιλαμβάνει την εκπαίδευση των αλγορίθμων, καθώς και την λήψη αποφάσεων σχετικά με την αφαίρεση κάποιων δεδομένων (outliers) τα οποία πιθανόν να προκαλούν κάποια ανισσόροπα αποτελέσματα.

Σχεδιασμός

Στα πλαίσια του σχεδιασμού ενός σχήματος της βάσης δεδομένων που θα καλύπτει τις προαναφερθείσες απαιτήσεις χρησιμοποιήθηκε το πακέτο Oracle Multimedia, το οποίο παρέχει API για τη διαχείριση πολυμεσικών οντοτήτων (εικόνες, ήχο, βίντεο αλλά και έγγραφα).

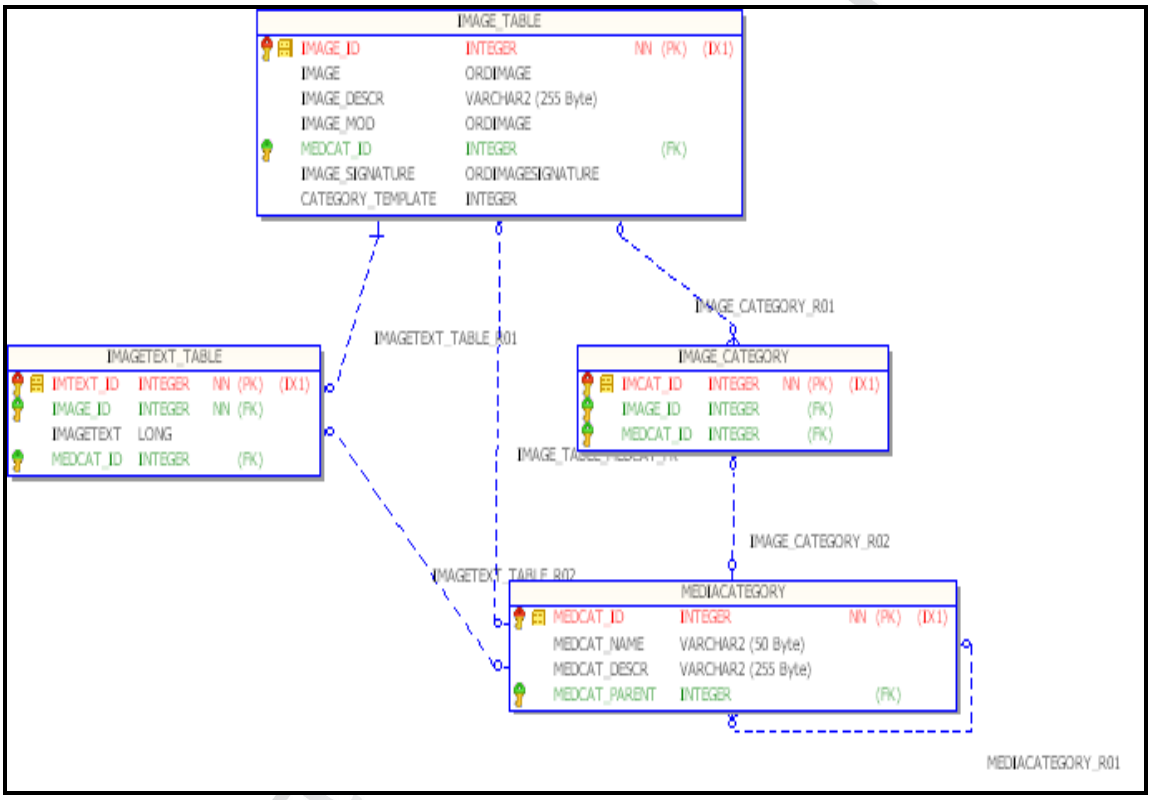
Έτσι δημιουργήθηκαν οι παρακάτω πίνακες:

- Ένας πίνακας ο οποίος αποτελεί τον «υποδοχέα» των εικόνων, με την μορφή ORDImage και ORDImageSignature του πακέτου Oracle Multimedia για εύκολη διαχείριση.
- Ένας πίνακας ορισμού και περιγραφής των κατηγοριών των εικόνων.
- Ένας πίνακας στον οποίο φορτώνονται μορφοποιημένα τα δεδομένα που ανακτώνται από τις εικόνες με τέτοιο τρόπο ώστε να καθίσταται δυνατό το data mining στα προαναφερθέντα δεδομένα.

- Ένας πίνακας «δεσίματος» εγγραφών εικόνας και κατηγορίας, ο οποίος δύναται για λόγους απλότητας (ένα-προς-ένα αντιστοίχιση εικόνων/κατηγοριών) να παραληφθεί και να γίνει απευθείας σύνδεση των 2 πινάκων εικόνων και κατηγοριών, όπως θα περιγραφεί στην επόμενη παράγραφο.

Περιγραφή / Απεικόνιση Σχήματος ORAMULTI

Στην Εικόνα 4.8 φαίνεται το διάγραμμα ER (Entity Relationship, διάγραμμα σχέσεων οντοτήτων), το οποίο απεικονίζει τους χρησιμοποιούμενους πίνακες που δημιουργήθηκαν στο σχήμα ORAMULTI για τις ανάγκες της εργασίας αυτής όπως περιγράφηκε και στις προηγούμενες παραγράφους.



Εικόνα 4.8. Διάγραμμα ER σχήματος ORAMULTI

Όπως φαίνεται και στην παραπάνω εικόνα έχουμε τους παρακάτω πίνακες με τα αντίστοιχα πεδία:

- Πίνακας **IMAGE_TABLE**: Είναι ο πίνακας ο οποίος αποτελεί τον «υποδοχέα» των εικόνων. Τα πεδία του είναι τα εξής:
  - IMAGE_ID. Τύπος INTEGER, είναι το Primary Key (PK) του συγκεκριμένου πίνακα.
  - IMAGE. Τύπος ORDSYS.ORDImage, είναι ο υποδοχέας των εικόνων στη μορφή που παρέχεται από το framework Oracle Multimedia.
  - IMAGE_DESCR. Τύπος VARCHAR2. Είναι ένα πεδίο περιγραφής για την εικόνα που εισάγεται στην αντίστοιχη εγγραφή.

- IMAGE_MOD. Τύπος ORDSYS.ORDImage, είναι ένας επιπρόσθετος υποδοχέας εικόνων στη μορφή που παρέχεται από το framework Oracle Multimedia για πιθανές τροποποιήσεις και μελλοντική χρήση.
- MEDCAT_ID. Τύπος INTEGER, είναι Foreign Key (FK) προς το πεδίο MEDCAT_ID (PK) του πίνακα MEDIACATEGORY.
- IMAGE_SIGNATURE. Τύπος ORDImageSignature , είναι η ψηφιακή υπογραφή της αντίστοιχης εικόνας στη μορφή που παρέχεται από το framework Oracle Multimedia και περιλαμβάνει στοιχεία που αφορούν την εικόνα (χρώμα, texture και άλλα).
- CATEGORY_TEMPLATE. Τύπος INTEGER, είναι ένας σημαφόρος (flag) που υποδεικνύει αν η συγκεκριμένη εγγραφή θεωρείται πρότυπο κατηγορίας για την κατηγοριοποίηση των εικόνων με βάση τις υποδομές του framework Oracle Multimedia.

- Πίνακας **IMAGE_CATEGORY**: Είναι ο πίνακας «δεσίματος» εγγραφών εικόνας και κατηγορίας. Τα πεδία του είναι τα εξής:
  - IMCAT_ID. Τύπος INTEGER, είναι το Primary Key (PK) του συγκεκριμένου πίνακα.
  - IMAGE_ID. Τύπος INTEGER, είναι Foreign Key (FK) προς το πεδίο IMAGE_ID (PK) του πίνακα IMAGE_TABLE.
  - MEDCAT_ID. Τύπος INTEGER, είναι Foreign Key (FK) προς το πεδίο MEDCAT_ID (PK) του πίνακα MEDIACATEGORY.
- Πίνακας **MEDIACATEGORY**: Είναι ο πίνακας ορισμού και περιγραφής των κατηγοριών των εικόνων. Τα πεδία του είναι τα εξής:
  - MEDCAT_ID. Τύπος INTEGER, είναι το Primary Key (PK) του συγκεκριμένου πίνακα.
  - MEDCAT_NAME. Τύπος VARCHAR2. Είναι ένα πεδίο περιγραφής της ονομασίας της κατηγορίας που εισάγεται στην αντίστοιχη εγγραφή.
  - MEDCAT_DESCR. Τύπος VARCHAR2. Είναι ένα πεδίο περιγραφής για την κατηγορία που εισάγεται στην αντίστοιχη εγγραφή.
  - MEDCAT_PARENT. Τύπος INTEGER, είναι το ένα Foreign Key (FK) προς το PK του ίδιου πίνακα (self reference).
- Πίνακας **IMAGETEXT_TABLE**: Είναι ο πίνακας στον οποίο φορτώνονται μορφοποιημένα τα δεδομένα που ανακτώνται από τις εικόνες με τέτοιο τρόπο ώστε να καθίσταται δυνατό το data mining στα προαναφερθέντα δεδομένα. Τα πεδία του είναι τα εξής:
  - IMTEXT_ID. Τύπος INTEGER, είναι το Primary Key (PK) του συγκεκριμένου πίνακα.
  - IMAGE_ID. Τύπος INTEGER, είναι Foreign Key (FK) προς το πεδίο IMAGE_ID (PK) του πίνακα IMAGE_TABLE.
  - IMAGETEXT. Τύπος LONG, είναι το πεδίο που αποθηκεύεται η τροποποιημένη φωτογραφία σε μορφή text με τη βοήθεια του framework Oracle Multimedia.

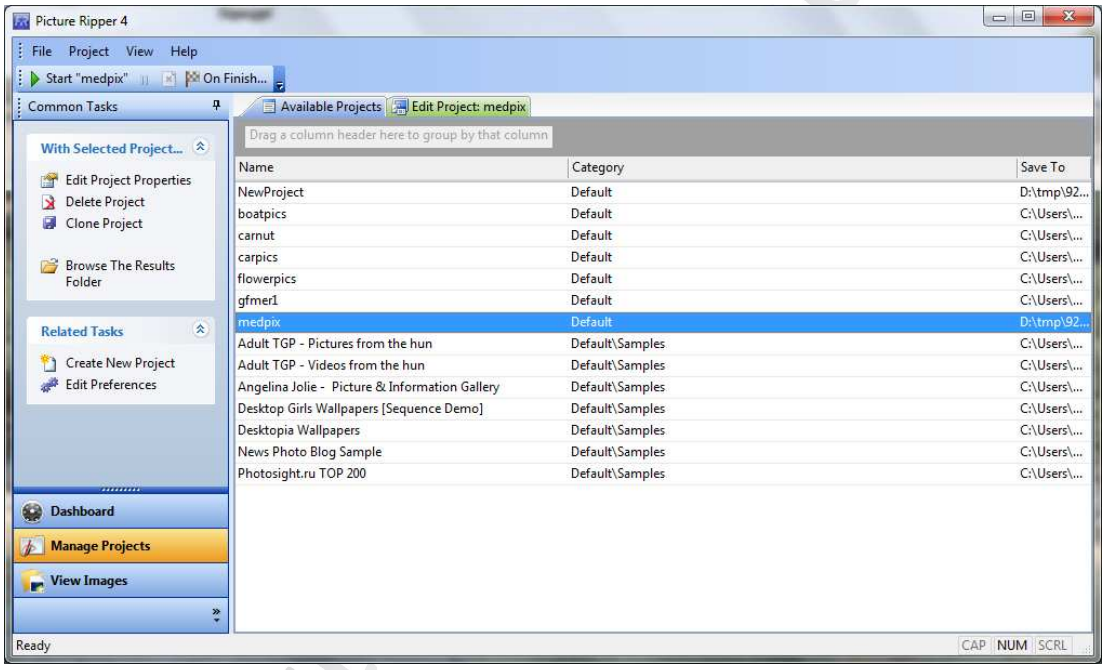
- MEDCAT_ID. Τύπος INTEGER, είναι Foreign Key (FK) προς το πεδίο MEDCAT_ID (PK) του πίνακα MEDIACATEGORY.

4.3.3. Υλοποίηση Αρχιτεκτονικής

Στα πλαίσια της επεξεργασίας των εικόνων ώστε μέσα από μια διαδικασία Data Mining να μπορέσουμε να καταλήξουμε σε κατηγοριοποίηση των εικόνων σε κάποιες προκαθορισμένες κατηγορίες βάσει του περιεχομένου τους ακολουθήθηκε μια διαδικασία από 8 στάδια.

Στάδιο 1. Συλλογή πρωτογενών δεδομένων από το διαδίκτυο.

Η πρώτη φάση αφορούσε την συγκέντρωση πρωτογενών δεδομένων για την εργασία μας από το διαδίκτυο. Πιο συγκεκριμένα αναζητήθηκαν ιστότοποι που έχουν δημοσιευμένες φωτογραφίες από διάφορες θεματικές ενότητες. Στη συνέχεια εκτελέσαμε μια επαναλαμβανόμενη διαδικασία μαζικής ανάκτησης δεδομένων από τους συγκεκριμένους ιστοτόπους (crawling) με σκοπό να συγκεντρώσουμε ικανό αριθμό φωτογραφιών.



Εικόνα 4.9. Εργαλείο PictureRipper για Ανάκτηση Εικόνων από το Διαδίκτυο

Αυτό έγινε σε μεγάλο αριθμό ιστοτόπων και με τη χρήση του εργαλείου PictureRipper [xxx] και είχε ως αποτέλεσμα την συγκέντρωση περίπου χιλίων φωτογραφιών από τις ακόλουθες εννοιολογικές κατηγορίες:

- Ιατρικές Εικόνες
- Αυτοκίνητα
- Ιστιοφόρα Σκάφη
- Λουλούδια

Στη συνέχεια προχωρήσαμε με το στάδιο 2 της διαδικασίας που ακολουθήθηκε στα πλαίσια αυτής της εργασίας.

Στάδιο 2. Φόρτωση πρωτογενών δεδομένων στη βάση.

Το πρώτο στάδιο αφορά την φόρτωση των πρωτογενών δεδομένων (εικόνες με μορφή JPEG) στη βάση δεδομένων που σχεδιάστηκε. Στο στάδιο αυτό γίνεται μαζική φόρτωση των εικόνων που ανακτήθηκαν από τον παγκόσμιο ιστό στον σχετικό πίνακα της βάσης (IMAGE_TABLE).

Στη συνέχεια προχωρήσαμε με το στάδιο 3 της διαδικασίας που ακολουθήθηκε στα πλαίσια αυτής της εργασίας.

Στάδιο 3. Μετατροπή Δεδομένων Εικόνας στην μορφή του πακέτου Oracle Multimedia.

Στο στάδιο αυτό έγινε η μετατροπή των δεδομένων εικόνας από την πρωτογενή τους μορφή (εικόνες JPEG) στην μορφή που υποστηρίζεται και παρέχεται από το πακέτο Oracle Multimedia (ORDImage). Η εκτέλεση των σταδίων 2 και 3 περιγράφεται αναλυτικά στην επεξήγηση του αντίστοιχου προγράμματος σε PL/SQL που αναπτύχθηκε και ακολουθεί.

Στον Πίνακα 3 φαίνεται ο PL/SQL κώδικας που εκτελέστηκε για την μαζική εισαγωγή των αρχείων στη βάση. Επειδή πρόσβαση στο σύστημα αρχείων (filesystem) έχουν μόνο οι χρήστες με ρόλο / επίπεδο ασφαλείας “Διαχειριστές βάσης” (database administrators) η εν λόγω διαδικασία εκτελείται από χρήστη με ανάλογα δικαιώματα. Η διαδικασία που αναπτύχθηκε σε PL/SQL παίρνει ως παράμετρο μια διαδρομή στο δίσκο (διαδρομή που δείχνει στο φάκελο που περιέχει ένα σύνολο εικόνων μιας κατηγορίας), καθώς και την κατηγορία των εικόνων που φορτώνονται. Στη συνέχεια η διαδικασία διαβάζει τα περιεχόμενα του φακέλου που δόθηκε και για κάθε έγκυρο αρχείο εικόνας κάνει την ανάλογη μετατροπή στη μορφή που παρέχει το πακέτο Oracle Multimedia (ORDImage), εξάγοντας παράλληλα και κάποια χαρακτηριστικά με χρήση μεθόδων του προαναφερθέντος πακέτου (ORDImageSignature). Η κατηγορία που δίνεται σαν παράμετρος περνιέται σε ένα πεδίο text για αναφορά της σωστής κατηγοριοποίησης της εν λόγω φωτογραφίας. Αντίστοιχα στον πίνακα 1 φαίνεται ο κώδικας που εκτελείται για την εισαγωγή και μετατροπή σε ORDImage της κάθε φωτογραφίας που βρίσκεται στο φάκελο που δίνεται παραμετρικά στην συνάρτηση μαζικής εισαγωγής εικόνων.

Στη συνέχεια προχωρήσαμε με το στάδιο 4 της διαδικασίας που ακολουθήθηκε στα πλαίσια αυτής της εργασίας.

Στάδιο 4. Κατηγοριοποίηση των φορτωμένων εικόνων με τη βοήθεια του πακέτου Oracle Multimedia.

Με τη βοήθεια μεθόδων του πακέτου Oracle Multimedia και κάνοντας χρήση των χαρακτηριστικών που εξήχθησαν κατά την μετατροπή των εικόνων σε μορφή ORDImage έγινε κατηγοριοποίηση των εικόνων (classification).

Πιο συγκεκριμένα, όπως φαίνεται και στην εφαρμογή που παρατίθεται στον Πίνακα 2 καλείται η μέθοδος ***classify_image*** και διαβάζοντας τα στοιχεία για την κάθε εικόνα που έχουν εξαχθεί (ORDImageSignature) και κάνοντας αναφορά σε μια εικόνα – πρότυπο ανά κατηγορία (σήμανση μεταβλητής ***category_template***) γίνεται απόφαση σε ποια κατηγορία ανήκει η εκάστοτε φωτογραφία.

Η κατηγοριοποίηση γίνεται κάνοντας πλήρη χρήση των δυνατοτήτων που παρέχει το πακέτο Oracle Multimedia, και πιο συγκεκριμένα ο τύπος ORDImageSignature. Ο τύπος αυτός αναλαμβάνει την εξαγωγή κάποιων χαρακτηριστικών από κάθε εικόνα και τα αποθηκεύει στον αντίστοιχο τύπο. Στην διάταξη που υλοποιήθηκε κατά το στάδιο της εισαγωγής των εικόνων στη βάση δεδομένων γίνεται η εξαγωγή αυτών των χαρακτηριστικών και η αποθήκευσή τους σε μια ξεχωριστή κολώνα (τύπου ORDImageSignature).

Στη συνέχεια, η όλη διαδικασία βασίζεται στην ιδέα να χαρακτηριστούν «φωτογραφίες-αντιπρόσωποι» από κάθε κατηγορία φωτογραφιών που θέλουμε τελικά να αναγνωρίσουμε και να γίνεται σύγκριση των προς κατηγοριοποίηση τυχαίων φωτογραφιών με την εκάστοτε «φωτογραφία - αντιπρόσωπο». Αυτό γίνεται όπως φαίνεται και στον Πίνακα 2 με τη χρήση της μεθόδου evaluateScore που παρέχει το πακέτο OracleMultimedia και η οποία επιστρέφει την ομοιότητα των εξαχθεισών χαρακτηριστικών από τις συγκρινόμενες φωτογραφίες.

Ορίζοντας μια τιμή-στάθμης (threshold) πάνω από την οποία η ομοιότητα σημαίνει ότι η τυχαία φωτογραφία πρέπει να κατηγοριοποιηθεί στην κατηγορία που αντιπροσωπεύεται από την «φωτογραφία-αντιπρόσωπο».

Στη συνέχεια προχωρήσαμε με το στάδιο 5 της διαδικασίας που ακολουθήθηκε στα πλαίσια αυτής της εργασίας.

Στάδιο 5. Μετατροπή των φορτωμένων εικόνων σε κείμενο για να είναι δυνατή η αναζήτηση προτύπων (patterns).

Στο στάδιο αυτό μετετράπησαν οι εικόνες από τη μορφή του πακέτου Oracle Multimedia (ORDimage) στη μορφή πολλαπλών πεδίων κειμένου VARCHAR2 ώστε να μπορεί να συντεθεί μια διαδικασία αναζήτησης προτύπων πάνω στα δεδομένα κειμένου αυτά. Επισημαίνεται ότι οι διαδικασίες data mining υποστηρίζονται μόνο για τους παρακάτω τύπους δεδομένων:

- INTEGER
- NUMBER
- FLOAT
- VARCHAR2
- CHAR
- DM_NESTED_NUMERICALS (εμφωλευμένη κολώνα)
- DM_NESTED_CATEGORICALS (εμφωλευμένη κολώνα)

Για την μεταφορά και μετατροπή των δεδομένων ακολουθήθηκε η εξής διαδικασία:

Εξαγωγή δεδομένων τύπου ORDImage μέσω του εργαλείου TOAD της Quest Software σε μορφή αρχείου κειμένου χωρισμένου από ειδικό χαρακτήρα (κολώνα – pipe |).

Χρήση του εργαλείου SQL Loader της Oracle για την εισαγωγή των δεδομένων σε δεύτερο πίνακα, χωρισμένα σε 50 πεδία κειμένου 4KByte το καθένα.

Στον Πίνακα 4 φαίνεται η περιγραφή του τρόπου κατάτμησης / μετατροπής των δεδομένων για τον SQL Loader.

Στη συνέχεια προχωρήσαμε με το στάδιο 6 της διαδικασίας που ακολουθήθηκε στα πλαίσια αυτής της εργασίας.

Στάδιο 6. Αποκατάστασης ομογένειας μεγέθους εικόνων.

Για την ομογενοποίηση των δεδομένων αποκόπηκαν από τα προς εξέταση δεδομένα οι πολύ μεγάλες (>200KB) και οι πολύ μικρές φωτογραφίες (<30KB). Για το λόγο αυτό υπάρχει αποθηκευμένη διαδικασία η οποία εκτελείται πριν την δημιουργία και εφαρμογή των αλγορίθμων εξόρυξης δεδομένων. Στη συνέχεια προχωρήσαμε με το στάδιο 7 της διαδικασίας που ακολουθήθηκε στα πλαίσια αυτής της εργασίας.

Στάδιο 7. Χρήση αλγορίθμων data mining.

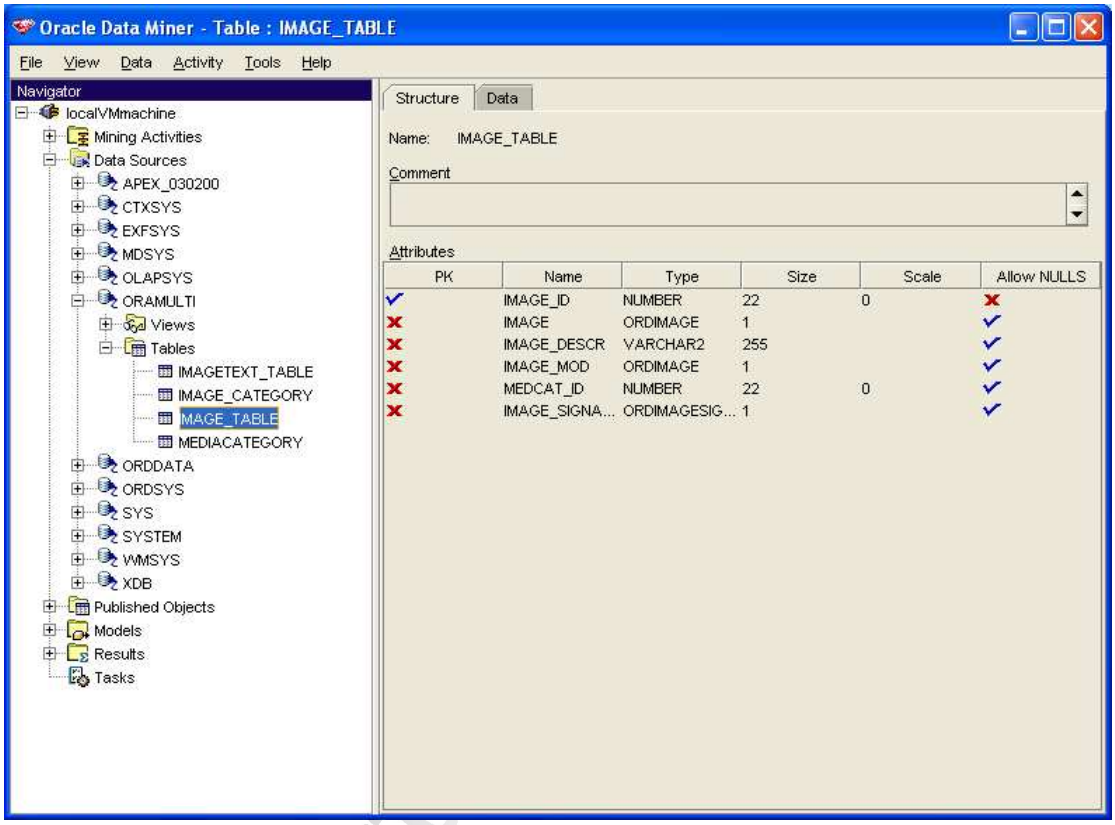
Όπως περιγράφηκε και στο προηγούμενο κεφάλαιο για την διαδικασία της εξόρυξης δεδομένων που θα εφαρμοστεί στο σύνολο δεδομένων που έχουν εισαχθεί στα προηγούμενα στάδια, θα χρησιμοποιηθούν τέσσερις διαφορετικοί αλγόριθμοι. Συγκεκριμένα θα αναπτυχθούν τέσσερις διαδικασίες κατηγοριοποίησης PL/SQL μία για κάθε αλγόριθμο από τους Decision Trees (DT), Naïve Bayes, Support Vector Machines και Logistic Regression.

Για τα στάδια του ορισμού και εκπαίδευσης ενός αλγορίθμου πρέπει να οριστούν τα παρακάτω:

- Ο χρησιμοποιούμενος αλγόριθμος ,
- Τα δεδομένα εισόδου και εξόδου (κολώνες που περιέχουν τα στοιχεία των εικόνων όπως αυτά περιγράφονται από το πακέτο Oracle Multimedia – τύπος ORDImage, και η αντίστοιχη κολώνα που περιέχει την κατηγορία της κάθε εικόνας),
- Τον διαχωρισμό των δεδομένων σε δεδομένα εκπαίδευσης και σε δεδομένα δοκιμής,
- Την παραμετροποίηση του αλγορίθμου,

- Την εκπαίδευση του αλγορίθμου.

Για την εκτέλεση των προαναφερθέντων σταδίων καθώς και του σταδίου εφαρμογής του αλγορίθμου και της εκτέλεσης των πειραμάτων χρησιμοποιήθηκε το εργαλείο DataMiner της Oracle. (Εικόνα 4.10)



Εικόνα 4.10. Εργαλείο Δημιουργίας / Παραμετροποίησης Αλγορίθμων Data Mining

Στη συνέχεια προχωρήσαμε με το στάδιο 8 της διαδικασίας που ακολουθήθηκε στα πλαίσια αυτής της εργασίας.

Στάδιο 8. Εκτέλεση αλγορίθμων data mining.

Το στάδιο αυτό περιγράφεται αναλυτικά στη συνέχεια στο πέμπτο κεφάλαιο.

5. Πειράματα – Δοκιμαστικές Εφαρμογές

5.1. Πειραματική Διάταξη

Τα δεδομένα που εισήχθησαν στη βάση στα προηγούμενα στάδια χρησιμοποιήθηκαν τόσο για την εκπαίδευση όσο και για την δοκιμή της εφαρμογής των τεσσάρων προαναφερθέντων αλγορίθμων εξόρυξης δεδομένων.

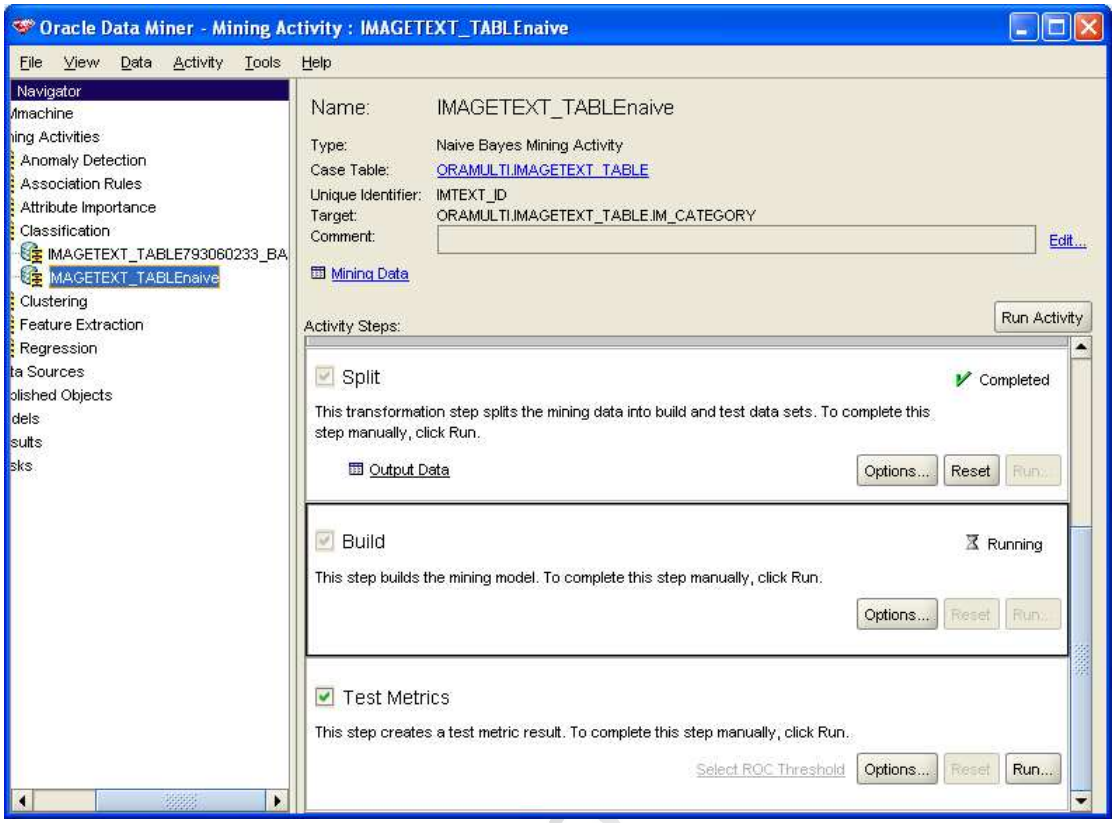
Στη συνέχεια περιγράφονται τα δύο βασικά μέρη της πειραματικής διάταξης, των τμημάτων δηλαδή των δεδομένων που χρησιμοποιήθηκαν για την τελική εφαρμογή των αλγορίθμων Decision Trees, Naïve Bayes, Support Vector Machine και Logistic Regression.

5.1.1. Εκπαίδευση Αλγορίθμων

Για την επιτυχή ολοκλήρωση της εκπαίδευσης ώστε ο αλγόριθμος να είναι εφαρμόσιμος στα δεδομένα, χωρίζονται τα δεδομένα σε 2 μέρη: τα δεδομένα εκπαίδευσης (train data) και τα δεδομένα δοκιμής (test data). Στη συνέχεια εκτελείται ο αλγόριθμος στα δεδομένα εκπαίδευσης στα δεδομένα εκπαίδευσης, έχοντας ως δεδομένο το αποτέλεσμα (κατηγορία στην οποία ανήκει η εικόνα).

Με τον τρόπο αυτό είναι δυνατή η εξαγωγή προτύπων (patterns) από τον εκάστοτε αλγόριθμο, τα οποία πρότυπα θα χρησιμοποιηθούν σε δεύτερο χρόνο κατά την εφαρμογή του στα δεδομένα δοκιμής.

Στην Εικόνα 5.1 φαίνεται η εκτέλεση της διαδικασίας εκπαίδευσης του αλγορίθμου (συγκεκριμένα του Decision Trees) μέσα από το εργαλείο Dataminer της Oracle. Σημειώνεται ότι ακολουθείται πανομοιότυπη διαδικασία και στους υπόλοιπους αλγορίθμους, και συγκεκριμένα λαμβάνεται το 60% των δεδομένων (δεδομένα που αντιστοιχούν σε περίπου 600 εικόνες) ως δεδομένα εκπαίδευσης και τα υπόλοιπα (περίπου 300 εικόνες) ως δεδομένα δοκιμής.



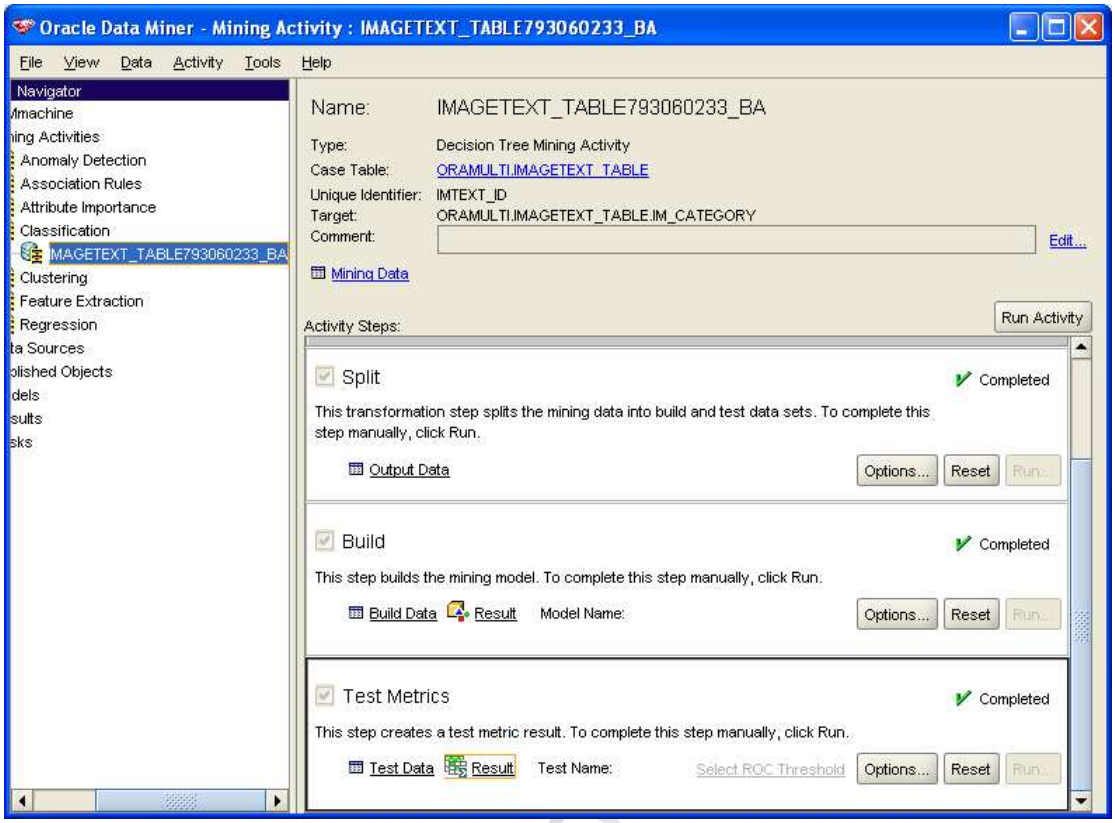
Εικόνα 11. Εκπαίδευση Μοντέλου Data Mining

5.2. Εκτέλεση Πειραμάτων

5.2.1. Εξόρυξη Δεδομένων με τη Χρήση των Αλγορίθμων

Ο κάθε αλγόριθμος εφαρμόζεται στο προαναφερθέν δείγμα των 350 εικόνων χωρίς να λαμβάνει υπόψη την a-priori κατηγοριοποίηση. Στη συνέχεια εξάγεται ένα συμπέρασμα όσον αφορά την κατηγορία στην οποία ανήκει κάθε εικόνα από το σύνολο των test data.

Εισάγοντας τα test data στον αλγόριθμο και πάλι μέσω του γραφικού περιβάλλοντος του Oracle Dataminer γίνεται η εκτέλεση του αλγορίθμου για την εύρεση των εικόνων που ανήκουν σε μια συγκεκριμένη κατηγορία από αυτές που αρχικά εισήχθησαν στη βάση δεδομένων. Για την εύρεση των εικόνων που ανήκουν σε όλες τις κατηγορίες απαιτείται επαναλαμβανόμενη εκτέλεση του αλγορίθμου. Στην Εικόνα 5.2 φαίνεται το παράθυρο εκτέλεσης του αλγορίθμου.



Εικόνα 12. Ολοκλήρωση Πειράματος

Το γραφικό εργαλείο Dataminer εκτελεί κατά την εφαρμογή των αλγορίθμων τις διαδικασίες PL/SQL που φαίνονται στον Πίνακα 5.

Εναλλακτικά, τρέξαμε και τη διαδικασία κατηγοριοποίησης που παρέχεται από το πακέτο Oracle Multimedia. Πιο συγκεκριμένα ορίσαμε ανά κατηγορία μια εγγραφή (εικόνα) ως αναφορά, και με βάση τη σύγκριση των χαρακτηριστικών των διαθέσιμων εγγραφών αναφοράς με την εκάστοτε εγγραφή έγινε η ταξινόμηση ανά κατηγορία με μία κλήση της μεθόδου που φαίνεται στον Πίνακα 2.

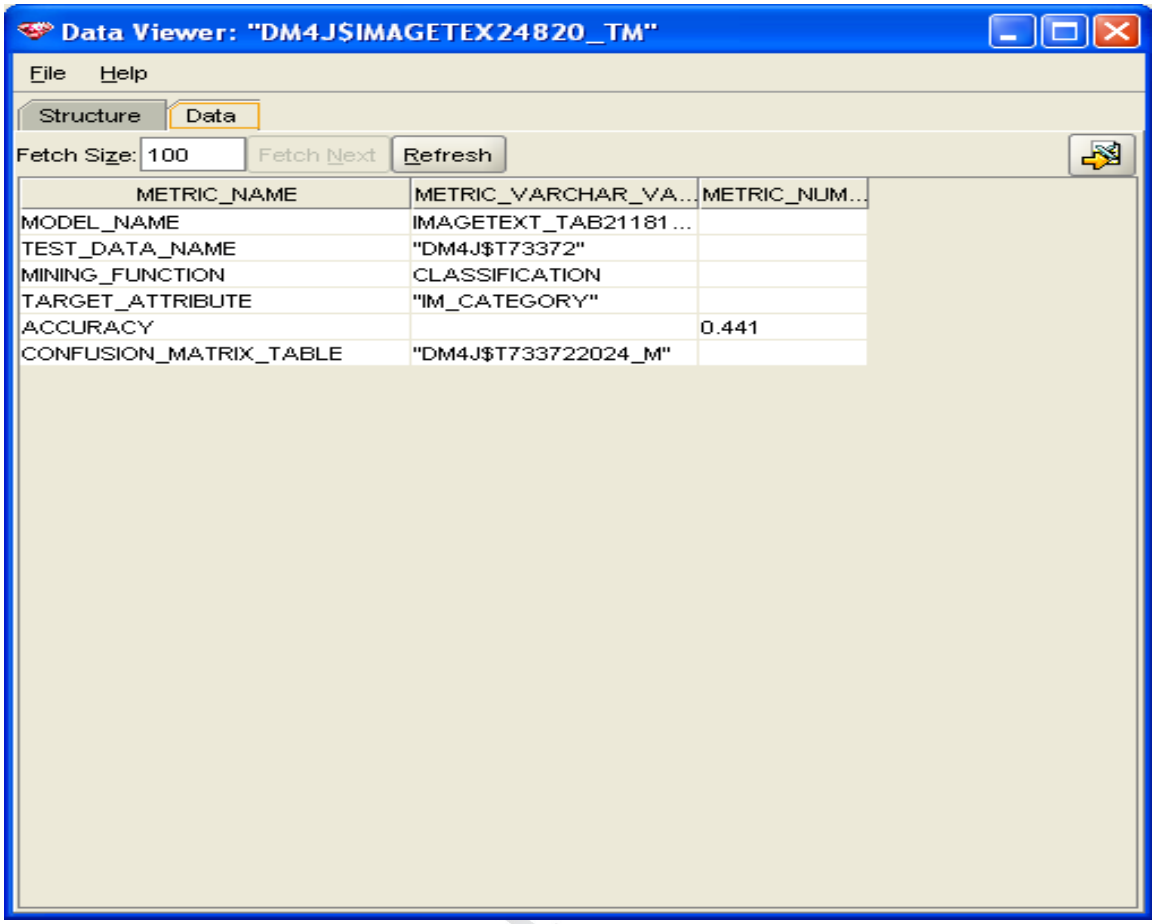
Λόγω του περιορισμένου αριθμού δεδομένων (περίπου χίλιες εικόνες οι οποίες καλούνται να αποτελέσουν τόσο το σύνολο εκπαίδευσης όσο και το σύνολο δοκιμών) μια σημαντική παράμετρος η οποία τροποποιούμενη αναμένεται να δώσει διαφορετικά αποτελέσματα. Έτσι εκπαιδεύτηκαν και εφαρμόστηκαν οι αλγόριθμοι με σύνολο εκπαίδευσης διαδοχικά το 60%, το 65%, το 70%, 75%, το 80 και το 85% του συνόλου των εικόνων.

Στην επόμενη ενότητα περιγράφεται η επιρροή που είχε η προαναφερόμενη διακύμανση στο σύνολο εκπαίδευσης στα αποτελέσματα που παρουσίασε ο κάθε αλγόριθμος.

5.2.2. Αποτελέσματα – Συμπεράσματα

Στην Εικόνα 5.3 φαίνεται το παράθυρο εμφάνισης αποτελεσμάτων του γραφικού εργαλείου εξόρυξης δεδομένων Oracle Dataminer (ενδεικτικά αποτελέσματα εκτέλεσης αλγορίθμου Decision Trees για την κατηγορία εικόνων «Ιστιοφόρα Σκάφη»).

Οι 4 αλγόριθμοι εκτελέστηκαν επαναλαμβανόμενα (2 φορές για κάθε μία από τις 4 κατηγορίες εικόνων) και εξάχθηκε ο μέσος όρος ποσοστού επιτυχίας πρόβλεψης.



Εικόνα 13 Παράθυρο Αποτελεσμάτων Εκτέλεσης Αλγορίθμου

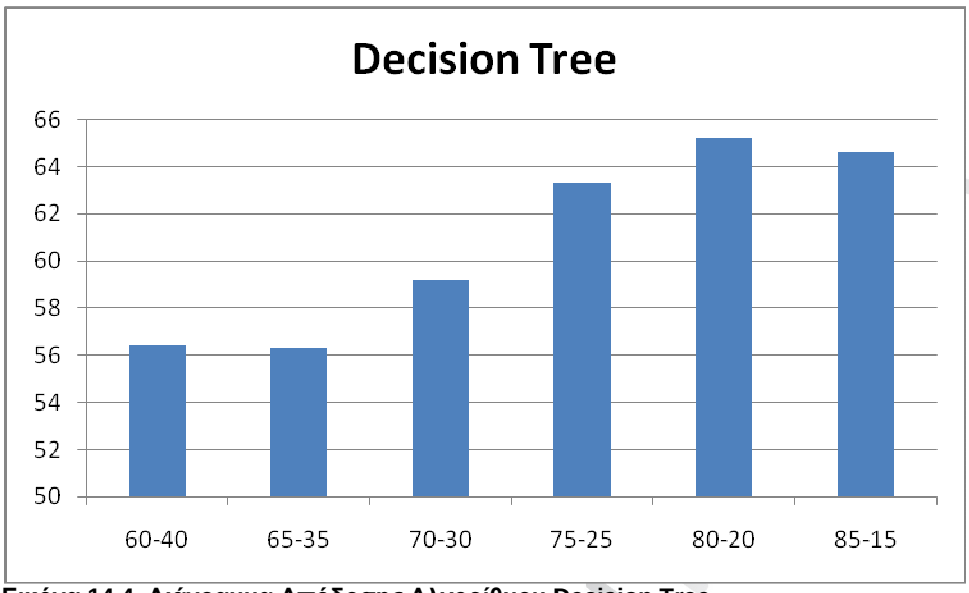
Στον παρακάτω πίνακα παρουσιάζονται οι προαναφερθέντες μέσοι όροι ποσοστού επιτυχίας για τους 4 διαφορετικούς αλγορίθμους που εφαρμόστηκαν.

Αλγόριθμος Εξόρυξης Δεδομένων	**Ποσοστό Επιτυχίας (Σύνολο εκπαίδευσης 60%)**
Decision Tree	56,4%
Naïve Bayes	64,1%
Support Vector Machine (SVM)	65,2%
Logistic Regression	62%

Όπως παρατηρούμε ο καλύτερος αλγόριθμος για το συγκεκριμένο πρόβλημα (κατηγοριοποίηση εικόνων που αναπαρίστανται με τη μορφή που παρέχει το πακέτο Oracle Multimedia) είναι ο Support Vector Machines. Παρόλα αυτά παρατηρούμε ότι τα ποσοστά επιτυχίας παρότι κυμαίνονται σε αποδεκτά επίπεδα, δεν είναι ιδιαιτέρως υψηλά. Αυτό οφείλεται κυρίως στην υψηλή ανομοιογένεια των εικόνων, κάτι που συνεπάγεται και ανομοιογενή δεδομένα εκπαίδευσης. Επίσης οι περίπου 1000 εικόνες που εισήχθησαν στη βάση δεδομένων είναι ένας οριακός αριθμός για την επιτυχή εκτέλεση εκπαίδευσης και δοκιμών αλγορίθμων εξόρυξης δεδομένων και αυτό επηρέασε αρνητικά τα αποτελέσματα όσον αφορά στο ποσοστό επιτυχίας.

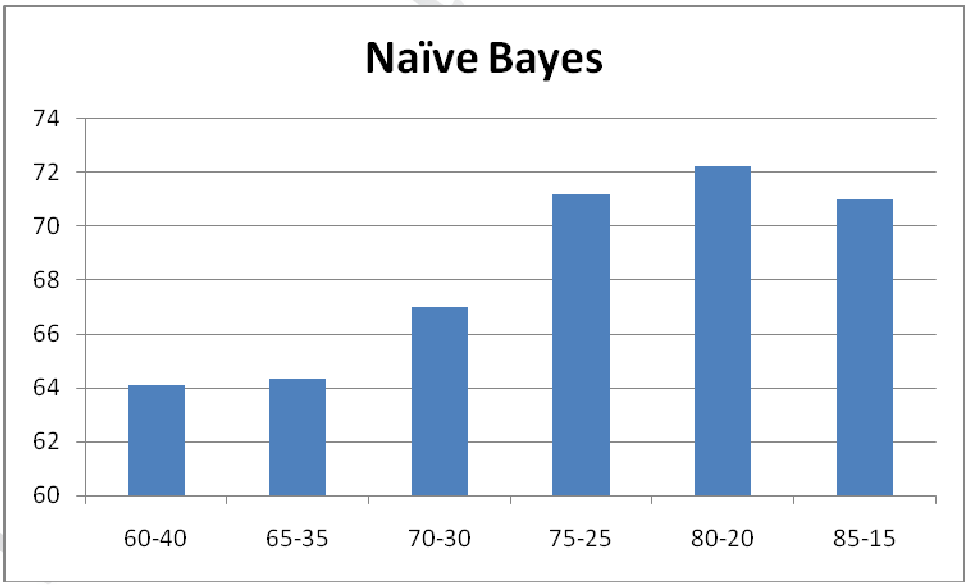
Στη συνέχεια παρουσιάζονται οι διακυμάνσεις της απόδοσης των αλγορίθμων καθώς μεταβάλλεται η ποσόστωση (και παράλληλα ο απόλυτος αριθμός) δειγμάτων εκπαίδευσης επί των συνολικών δειγμάτων. Στην εικόνα 5.4 φαίνεται η διακύμανση του αλγορίθμου Decision

Tree με αναλογία συνόλου εκπαίδευσης-δοκιμών 60%-40%, 65%-35%, 70%-30%, 75%-25%, 80%-20% και 85%-25%.

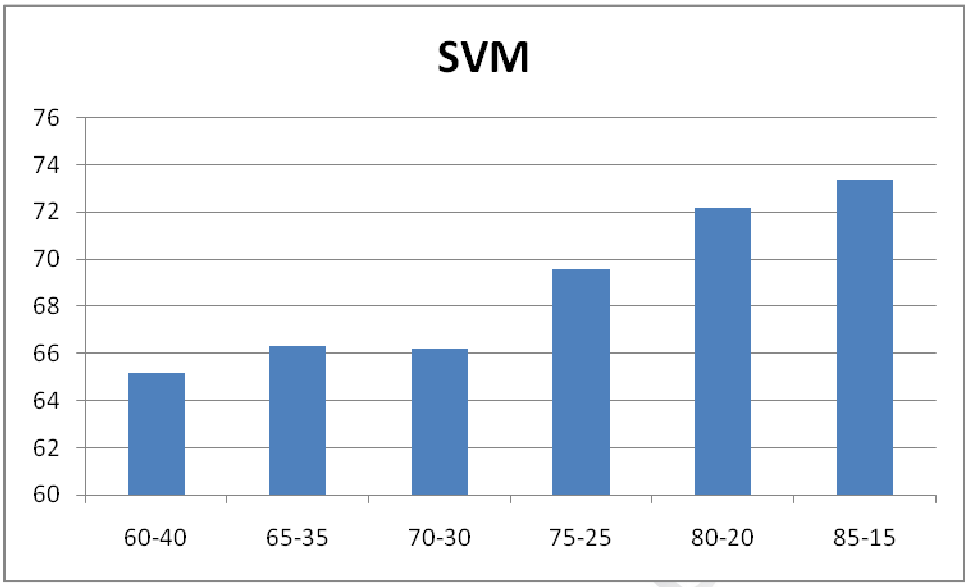


Εικόνα 14.4. Διάγραμμα Απόδοσης Αλγορίθμου Decision Tree

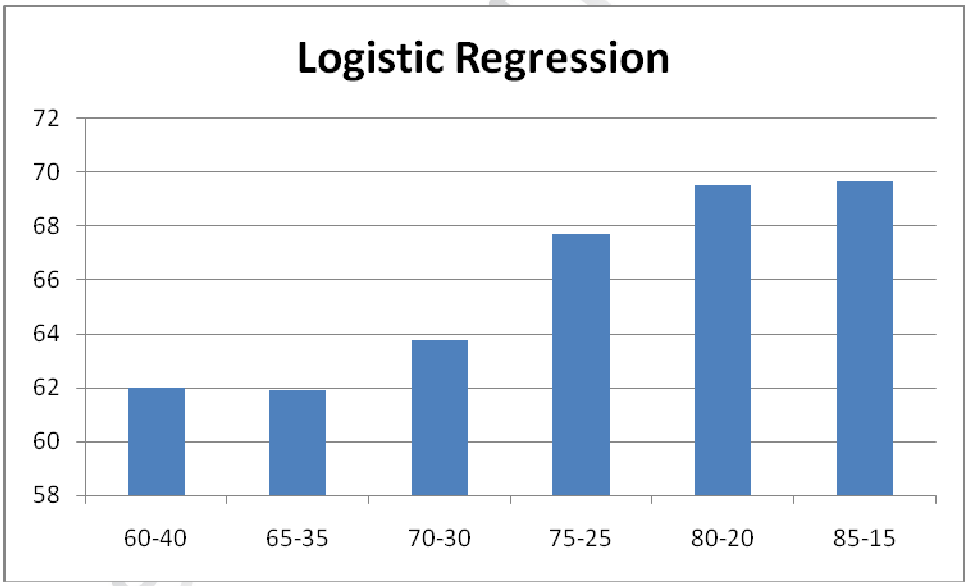
Στις εικόνες 5.5, 5.6 και 5.7 φαίνονται τα αντίστοιχα διαγράμματα για τους αλγόριθμους Naïve Bayes, SVM και Logistic Rregression.



Εικόνα 15. Διάγραμμα Απόδοσης Αλγορίθμου Naive Bayes



Εικόνα 16. Διάγραμμα Απόδοσης Αλγορίθμου Support Vector Machines



Εικόνα 17. Διάγραμμα Απόδοσης Αλγορίθμου Logistic Regression

6. Συμπεράσματα – Περαιτέρω Έρευνα

6.1. Σύνοψη / Συμπεράσματα

Στη συγκεκριμένη εργασία έγινε μια αναλυτική μελέτη του θέματος της εξόρυξης δεδομένων, και πιο συγκεκριμένα στον τομέα των εικόνων. Αφού έγινε η μελέτη των υπαρχουσών προσεγγίσεων στον ερευνητικό χώρο της εξόρυξης δεδομένων και η εύρεση ικανού όγκου εικόνων κατηγοριοποιημένων, αυτές εισήχθησαν σε μια σχεσιακή βάση δεδομένων της εταιρείας Oracle με τη μορφή που παρέχει το πακέτο Oracle Multimedia. Στη συνέχεια έγινε ανάπτυξη, εκπαίδευση και εφαρμογή αλγορίθμων για εξόρυξη δεδομένων πάνω στις προαναφερθείσες εικόνες και εξήχθησαν τα σχετικά συμπεράσματα.

Παρατηρείται ότι η διαδικασία δημιουργίας, εκπαίδευσης και εφαρμογής αλγορίθμων εξόρυξης δεν είναι μια προτυποποιημένη διαδικασία, η οποία μπορεί να αποτελέσει πανάκεια σε όλα τα προβλήματα που απαιτείται αναζήτηση και επεξεργασία σε μεγάλα σύνολα δεδομένων. Για το λόγο αυτό γίνεται αντιληπτό ότι η διαρκής μελέτη και εξέλιξη στον ερευνητικό αυτό τομέα είναι απαραίτητη για την ευρεία και ετερογενή γκάμα προβλημάτων που απαιτούν data mining.

Πιο συγκεκριμένα οι εφαρμογές στις οποίες μπορεί απλώς να χρησιμεύσει έως και είναι απαραίτητη η εξόρυξη δεδομένων σε σύνολα εικόνων – είτε ιατρικών, είτε διάφορων κατηγοριών όπως στην περίπτωσή μας – γίνεται αντιληπτό ότι είναι πάρα πολλές. Από εφαρμογές καταχώρησης, αναζήτησης, επεξεργασίας εικόνων στο διαδίκτυο, μέχρι διαχείριση βάσεων με ιατρικές εικόνες σε μεγάλες μονάδες υγείας, και αυτόματη σύγκριση αυτών για εξαγωγή ενδεικτικής διάγνωσης / ενδείξεων παθογένειας.

Στα πλαίσια αυτά έγινε μελέτη τεσσάρων διαφορετικών αλγορίθμων (decision trees, naïve bayes, support vector machines και logistic regression) για ένα σύνολο χιλίων περίπου εικόνων, και αντίστοιχη κατηγοριοποίηση αυτών. Τα αποτελέσματα κρίνονται ικανοποιητικά, αν και πάντα υπάρχει περιθώριο βελτίωσης. Άλλωστε ένας αλγόριθμος εξόρυξης δεδομένων που επιτυγχάνει πολύ μεγάλα ποσοστά επιτυχίας για ένα συγκεκριμένο πρόβλημα εγκυμονεί την πιθανότητα να είναι υπερ-βελτιωμένος (overoptimized) για το συγκεκριμένο πρόβλημα, κάνοντάς τον πιθανώς ακατάλληλο για μια ευρύτερη γκάμα σχετικών προβλημάτων.

Επίσης παρατηρήσαμε ότι παραμετροποιώντας τους χρησιμοποιούμενους αλγορίθμους, και πιο συγκεκριμένα αυξάνοντας το ποσοστό από τα διαθέσιμα δεδομένα που αποτελούσε το σύνολο εκπαίδευσης των αλγορίθμων οι επιδόσεις βελτιώθηκαν αισθητά, κυμαινόμενες πλέον σε αποδεκτά επίπεδα, λαμβάνοντας πάντα υπόψη τους υπαρκτούς περιορισμούς που αναφέρθηκαν προηγουμένως.

6.2. Περαιτέρω Έρευνα

Ενδεχόμενη μελλοντική προσπάθεια στην εναρμονισμένη με την κατεύθυνση της παρούσας εργασίας θα έβρισκε πρόσφορο έδαφος τόσο στον τομέα της βελτίωσης του παράγοντα φιλικότητας προς τον χρήστη, όσο και στον τομέα των επιδόσεων των αλγορίθμων. Πράγματι η διαδικασία εισαγωγής μεγάλου αριθμού εικόνων στη βάση, φροντίζοντας παράλληλα για

- την εξαγωγή των χαρακτηριστικών τους η οποία είναι απαραίτητη για το στάδιο της κατηγοριοποίησης με τη χρήση του πακέτου Oracle Multimedia
- την μετατροπή των εικόνων σε μορφή που υποστηρίζει το πακέτο Oracle Multimedia
- προετοιμασία των δεδομένων για εφαρμογή αλγορίθμων εξόρυξης δεδομένων είναι αρκετά πολύπλοκη και απαιτεί γνώση των χρησιμοποιούμενων υποδομών από τον διαχειριστή που επιχειρεί την προαναφερθείσα εισαγωγή. Έτσι η ανάπτυξη μιας ενιαίας – πιθανώς γραφικής- διεπαφής για την απλοποίηση των διαδικαστικών ενεργειών εισαγωγής/μετατροπής των δεδομένων θα διευκόλυνε την ευρύτερη εφαρμογή όσων μελετήθηκαν στα πλαίσια αυτής της εργασίας.

Ακόμη, πεδίο περαιτέρω έρευνας στην κατεύθυνση που κινήθηκε η συγκεκριμένη εργασία μπορεί να αποτελέσει η μελέτη περισσότερων αλγορίθμων καθώς και η εφαρμογή υβριδικών μεθόδων/αλγορίθμων για την επίτευξη βελτιωμένων αποτελεσμάτων.

Σε κάθε περίπτωση δύσκολα ένα κομμάτι ερευνητικής προσπάθειας μπορεί να χαρακτηριστεί πλήρες, καθώς πάντα παρουσιάζονται πολλές παράμετροι που χρήζουν περαιτέρω μελέτης και ανάπτυξης για την επίτευξη ενός ακόμα καλύτερου αποτελέσματος.

Παράρτημα Α – Κώδικας Λειτουργιών /Αλγορίθμων

Πίνακας 1. Διαδικασία Εισαγωγής Εικόνων στη Βάση Δεδομένων

```
CREATE OR REPLACE PROCEDURE ORAMULTI.loadImageToDB(in_dir VARCHAR2,
                                in_fname VARCHAR2,
                                image_descr VARCHAR2,
                                in_mod_fname VARCHAR2)
AS
  objIm ORDIMAGE;
  objImMod ORDIMAGE;
  imsig ORDImageSignature;
  newId INTEGER;
  ctx RAW(64) := NULL;
BEGIN
dbms_output.put_line('Input directory: ' || in_dir);
dbms_output.put_line('Image filename: ' || in_fname);
dbms_output.put_line('Image description: ' || image_descr);
dbms_output.put_line('Mod Image filename: ' || in_mod_fname);
  EXECUTE IMMEDIATE 'CREATE OR REPLACE DIRECTORY MEDIA_DIR AS ''' || in_dir ||'''';
  COMMIT;
  INSERT INTO ORAMULTI.image_table(image, image_descr, image_mod,
IMAGE_SIGNATURE) VALUES
     (ORDImage.init('FILE', 'MEDIA_DIR', in_fname), image_descr, null,
ORDSYS.ORDImageSignature.init())
     RETURNING image, image_mod, image_id, image_signature INTO objIm, objImMod,
newId, imsig;

  dbms_output.put_line('Image source: ' || objIm.getSource());
    objIm.import(ctx);
  dbms_output.put_line('Mod Image source: ' || objIm.getSource());
  --objImMod := objIm;
```

```
  imsig.generateSignature(objIm);

  UPDATE ORAMULTI.image_table SET image = objIm, image_mod = objImMod,
image_signature = imsig  WHERE image_id = newId;


  --EXECUTE IMMEDIATE 'drop directory media_dir';

  COMMIT;

 END;

/
```

Πίνακας 2. Διαδικασία Κατηγοριοποίησης Εικόνων με τη Χρήση του Oracle Multimedia

```
CREATE OR REPLACE PROCEDURE ORAMULTI.classify_image(cimid in integer, weights in
varchar2) IS

tmpVar NUMBER;

/******************************************************************************
****

  NAME:       classify_image

  PURPOSE:


  REVISIONS:

  Ver        Date        Author           Description

  ---------  ----------  ---------------  ------------------------------------

  1.0        9/1/2010   Administrator      1. Created this procedure.


  NOTES:


  Automatically available Auto Replace Keywords:

     Object Name:     classify_image

     Sysdate:         9/1/2010

     Date and Time:   9/1/2010, 2:33:06 PM, and 9/1/2010 2:33:06 PM

     Username:        Administrator (set in TOAD Options, Procedure Editor)

     Table Name:      (set in the "New PL/SQL Object" dialog)


******************************************************************************
***/

cat_class number;

minscore float;

minscoreId integer;

tmpscore float;
```

```
toClassify ORDImageSignature;

classId integer;

temp image_table%rowtype;


cursor templates is

select * from image_table where category_template=1;


BEGIN

  tmpVar := 0;

  minscore := 101.09;

  select image_signature into toClassify from image_table where image_id = cimid;

  OPEN templates;

   LOOP

      FETCH templates INTO temp;


      tmpscore := ORDSYS.ORDImageSignature.evaluateScore(toClassify,
temp.image_signature, weights);


      if tmpscore <= minscore then

         minscore := tmpscore;

         minscoreId := temp.image_id;

      end if;


      EXIT WHEN templates%NOTFOUND;

  END LOOP;


  CLOSE templates;


   update image_table set medcat_id = (select medcat_id from image_table where
image_id = minscoreId) where image_id = cimid;
```

```
  EXCEPTION
    WHEN NO_DATA_FOUND THEN
      NULL;
    WHEN OTHERS THEN
      -- Consider logging the error and then re-raise
      RAISE;
END classify_image;
/
```

Πίνακας 3. Διαδικασία Μαζικής Εισαγωγής Εικόνων στη Βάση Δεδομένων

```
EXECUTE AS SYS

DECLARE

pattern VARCHAR2(1024) := 'C:\TEMP\924pics\flowers\*';
folder VARCHAR2(1024) := 'C:\TEMP\924pics\flowers\';
descr VARCHAR2(1024) := 'flowers';
ns VARCHAR2(1024);

BEGIN
SYS.DBMS_BACKUP_RESTORE.searchFiles(pattern, ns);

FOR each_file IN (SELECT replace(FNAME_KRBMSFT, upper(folder), '') AS name FROM
X$KRBMSFT) LOOP
DBMS_OUTPUT.PUT_LINE(each_file.name);
ORAMULTI.LOADIMAGETODB(folder, each_file.name, descr, null);
END LOOP;
commit;

END;
/
```

Πίνακας 4. Διαδικασία Μαζικής Εισαγωγής και Τροποποίησης Εικόνων μέσω του SQL Loader

```
LOAD DATA
INFILE 'c:/temp/fullexport_newblack.txt'
INSERT
INTO TABLE imagetext_table
TRAILING NULLCOLS
(
image_id      POSITION (1) INTEGER EXTERNAL TERMINATED BY '|',
rawnumber char(1000000) TERMINATED BY '|',
imagetext1 char(4000) TERMINATED BY '|' "substr(:rawnumber, 1, 4000)",
imagetext2 char(4000) TERMINATED BY '|' "substr(:rawnumber, 4001, 4000)",
imagetext3 char(4000) TERMINATED BY '|' "substr(:rawnumber, 8001, 4000)",
imagetext4 char(4000) TERMINATED BY '|' "substr(:rawnumber, 12001, 4000)",
imagetext5 char(4000) TERMINATED BY '|' "substr(:rawnumber, 16001, 4000)",
imagetext6 char(4000) TERMINATED BY '|' "substr(:rawnumber, 20001, 4000)",
imagetext7 char(4000) TERMINATED BY '|' "substr(:rawnumber, 24001, 4000)",
imagetext8 char(4000) TERMINATED BY '|' "substr(:rawnumber, 28001, 4000)",
imagetext9 char(4000) TERMINATED BY '|' "substr(:rawnumber, 32001, 4000)",
imagetext10 char(4000) TERMINATED BY '|' "substr(:rawnumber, 36001, 4000)",
imagetext11 char(4000) TERMINATED BY '|' "substr(:rawnumber, 40001, 4000)",
imagetext12 char(4000) TERMINATED BY '|' "substr(:rawnumber, 44001, 4000)",
imagetext13 char(4000) TERMINATED BY '|' "substr(:rawnumber, 48001, 4000)",
imagetext14 char(4000) TERMINATED BY '|' "substr(:rawnumber, 52001, 4000)",
imagetext15 char(4000) TERMINATED BY '|' "substr(:rawnumber, 56001, 4000)",
imagetext16 char(4000) TERMINATED BY '|' "substr(:rawnumber, 60001, 4000)",
imagetext17 char(4000) TERMINATED BY '|' "substr(:rawnumber, 64001, 4000)",
```

```
imagetext18 char(4000) TERMINATED BY '|' "substr(:rawnumber, 68001, 4000)",
imagetext19 char(4000) TERMINATED BY '|' "substr(:rawnumber, 72001, 4000)",
imagetext20 char(4000) TERMINATED BY '|' "substr(:rawnumber, 76001, 4000)",
imagetext21 char(4000) TERMINATED BY '|' "substr(:rawnumber, 80001, 4000)",
imagetext22 char(4000) TERMINATED BY '|' "substr(:rawnumber, 84001, 4000)",
imagetext23 char(4000) TERMINATED BY '|' "substr(:rawnumber, 88001, 4000)",
imagetext24 char(4000) TERMINATED BY '|' "substr(:rawnumber, 92001, 4000)",
imagetext25 char(4000) TERMINATED BY '|' "substr(:rawnumber, 96001, 4000)",
imagetext26 char(4000) TERMINATED BY '|' "substr(:rawnumber, 100001, 4000)",
imagetext27 char(4000) TERMINATED BY '|' "substr(:rawnumber, 104001, 4000)",
imagetext28 char(4000) TERMINATED BY '|' "substr(:rawnumber, 108001, 4000)",
imagetext29 char(4000) TERMINATED BY '|' "substr(:rawnumber, 112001, 4000)",
imagetext30 char(4000) TERMINATED BY '|' "substr(:rawnumber, 116001, 4000)",
imagetext31 char(4000) TERMINATED BY '|' "substr(:rawnumber, 120001, 4000)",
imagetext32 char(4000) TERMINATED BY '|' "substr(:rawnumber, 124001, 4000)",
imagetext33 char(4000) TERMINATED BY '|' "substr(:rawnumber, 128001, 4000)",
imagetext34 char(4000) TERMINATED BY '|' "substr(:rawnumber, 132001, 4000)",
imagetext35 char(4000) TERMINATED BY '|' "substr(:rawnumber, 136001, 4000)",
imagetext36 char(4000) TERMINATED BY '|' "substr(:rawnumber, 140001, 4000)",
imagetext37 char(4000) TERMINATED BY '|' "substr(:rawnumber, 144001, 4000)",
imagetext38 char(4000) TERMINATED BY '|' "substr(:rawnumber, 148001, 4000)",
imagetext39 char(4000) TERMINATED BY '|' "substr(:rawnumber, 152001, 4000)",
imagetext40 char(4000) TERMINATED BY '|' "substr(:rawnumber, 156001, 4000)",
imagetext41 char(4000) TERMINATED BY '|' "substr(:rawnumber, 160001, 4000)",
imagetext42 char(4000) TERMINATED BY '|' "substr(:rawnumber, 164001, 4000)",
imagetext43 char(4000) TERMINATED BY '|' "substr(:rawnumber, 168001, 4000)",
imagetext44 char(4000) TERMINATED BY '|' "substr(:rawnumber, 172001, 4000)",
```

```
imagetext45 char(4000) TERMINATED BY '|' "substr(:rawnumber, 176001, 4000)",
imagetext46 char(4000) TERMINATED BY '|' "substr(:rawnumber, 180001, 4000)",
imagetext47 char(4000) TERMINATED BY '|' "substr(:rawnumber, 184001, 4000)",
imagetext48 char(4000) TERMINATED BY '|' "substr(:rawnumber, 188001, 4000)",
imagetext49 char(4000) TERMINATED BY '|' "substr(:rawnumber, 192001, 4000)",
imagetext50 char(4000) TERMINATED BY '|' "substr(:rawnumber, 196001, 4000)",
im_category    char(100) TERMINATED BY '|'
)
```

Πίνακας 5. Διαδικασία PL/SQL για το Data Mining

```
CREATE PACKAGE "DATAMININGACTIVITY1" AUTHID DEFINER AS


PROCEDURE "10FIELD_NB_IMAG917515591_BA"(case_table IN VARCHAR2 DEFAULT
'"ORAMULTI"."IMAGETEXT_TABLE"',
                additional_table_1 IN VARCHAR2 DEFAULT NULL,
                model_name IN VARCHAR2 DEFAULT 'IMAGETEXT_DMMODEL_DT',
                confusion_matrix_name IN VARCHAR2 DEFAULT
'"DM4J$T660537187079_M"',
                lift_result_name IN VARCHAR2 DEFAULT '"DM4J$T66053713463_L"',
                roc_result_name IN VARCHAR2 DEFAULT '"DM4J$T660537169832_R"',
                test_metric_name IN VARCHAR2 DEFAULT '"IMAGETEXT_TESTMETRICS_DT"',
                feature_table IN VARCHAR2 DEFAULT NULL,
                mapping_table IN VARCHAR2 DEFAULT NULL,
                drop_output IN BOOLEAN DEFAULT FALSE);


PROCEDURE "IMAGETEXT_SVM219913624_BA"(case_table IN VARCHAR2 DEFAULT
'"ORAMULTI"."IMAGETEXT_TABLE"',
                additional_table_1 IN VARCHAR2 DEFAULT NULL,
                model_name IN VARCHAR2 DEFAULT 'IMAGETEXT_MODEL_SVM',
                confusion_matrix_name IN VARCHAR2 DEFAULT NULL,
                lift_result_name IN VARCHAR2 DEFAULT NULL,
                roc_result_name IN VARCHAR2 DEFAULT NULL,
                test_metric_name IN VARCHAR2 DEFAULT NULL,
                feature_table IN VARCHAR2 DEFAULT NULL,
                mapping_table IN VARCHAR2 DEFAULT NULL,
                drop_output IN BOOLEAN DEFAULT FALSE);
```

```
END;



/

CREATE PACKAGE BODY "DATAMININGACTIVITY1" AS


c_long_sql_statement_length CONSTANT INTEGER := 32767;


SUBTYPE SQL_STATEMENT_TYPE     IS VARCHAR2(32767);

SUBTYPE LONG_SQL_STATEMENT_TYPE IS DBMS_SQL.VARCHAR2A;


TYPE TABLE_ARRAY is TABLE OF VARCHAR2(62);

TYPE LSTMT_REC_TYPE IS RECORD (

   lstmt dbms_sql.VARCHAR2A,

   lb    BINARY_INTEGER DEFAULT 1,

   ub    BINARY_INTEGER DEFAULT 0);

TYPE LSTMT_REC_TYPE_ARRAY is TABLE OF LSTMT_REC_TYPE;

TYPE QUERY_ARRAY is TABLE OF SQL_STATEMENT_TYPE;

TYPE TARGET_VALUES_LIST     IS TABLE OF  VARCHAR2(32);

TYPE VALUE_COUNT_LIST       IS TABLE OF  NUMBER;


PROCEDURE dump_varchar2a(vc2a dbms_sql.VARCHAR2A) IS

 v_str varchar2(32767);

BEGIN

 DBMS_OUTPUT.PUT_LINE('dump_varchar2a:');

 FOR i IN 1..vc2a.COUNT LOOP

   v_str := vc2a(i);
```

```
   DBMS_OUTPUT.PUT_LINE(v_str);
  END LOOP;
END;


PROCEDURE ls_append(
  r_lstmt IN OUT NOCOPY LSTMT_REC_TYPE,
  p_txt   VARCHAR2)
IS
BEGIN
  r_lstmt.ub := r_lstmt.ub + 1;
  r_lstmt.lstmt(r_lstmt.ub) := p_txt;
END ls_append;


PROCEDURE ls_append(
  r_lstmt IN OUT NOCOPY LSTMT_REC_TYPE,
  p_txt   LSTMT_REC_TYPE) IS
BEGIN
  FOR i IN p_txt.lb..p_txt.ub LOOP
    r_lstmt.ub := r_lstmt.ub + 1;
    r_lstmt.lstmt(r_lstmt.ub) := p_txt.lstmt(i);
  END LOOP;
END ls_append;


FUNCTION query_valid(
  p_query VARCHAR2) RETURN BOOLEAN
IS
  v_is_valid  BOOLEAN;
```

```
BEGIN
 BEGIN
  EXECUTE IMMEDIATE p_query;
  v_is_valid := TRUE;
 EXCEPTION WHEN OTHERS THEN
  v_is_valid := FALSE;
 END;
 RETURN v_is_valid;
END query_valid;

FUNCTION table_exist(
 p_table_name  VARCHAR2) RETURN BOOLEAN IS
BEGIN
 RETURN query_valid('SELECT * FROM ' || dbms_assert.simple_sql_name(p_table_name));
END table_exist;

FUNCTION model_exist(
 p_model_name  VARCHAR2) RETURN BOOLEAN
IS
 v_model_cnt NUMBER;
 v_model_exists BOOLEAN := FALSE;
BEGIN
 SELECT COUNT(*) INTO v_model_cnt FROM DM_USER_MODELS WHERE NAME =
UPPER(p_model_name);
 IF v_model_cnt > 0 THEN
  v_model_exists := TRUE;
 END IF;
```

```
 --DBMS_OUTPUT.PUT_LINE('model exist: '||v_model_exists);
 RETURN v_model_exists;
EXCEPTION WHEN OTHERS THEN
 RETURN FALSE;
END model_exist;

PROCEDURE drop_table(
 p_table_name  VARCHAR2)
IS
 v_stmt  SQL_STATEMENT_TYPE;
BEGIN
 v_stmt := 'DROP TABLE '||dbms_assert.simple_sql_name(p_table_name)||' PURGE';
 EXECUTE  IMMEDIATE v_stmt;
EXCEPTION WHEN OTHERS THEN
 NULL;
 --DBMS_OUTPUT.PUT_LINE('Failed drop_table: '||p_table_name);
END drop_table;

PROCEDURE drop_view(
 p_view_name  VARCHAR2)
IS
 v_stmt  SQL_STATEMENT_TYPE;
BEGIN
 v_stmt := 'DROP VIEW '||dbms_assert.simple_sql_name(p_view_name);
 EXECUTE  IMMEDIATE v_stmt;
EXCEPTION WHEN OTHERS THEN
 NULL;
```

```
  --DBMS_OUTPUT.PUT_LINE('Failed drop_view: '||p_view_name);
END drop_view;

PROCEDURE drop_model(
  p_model_name  VARCHAR2)
IS
  v_diagnostics_table VARCHAR2(30);
BEGIN
  DBMS_DATA_MINING.DROP_MODEL(p_model_name);
  SELECT SETTING_VALUE INTO v_diagnostics_table
  FROM TABLE(DBMS_DATA_MINING.GET_MODEL_SETTINGS(p_model_name))
  WHERE SETTING_NAME = 'GLMS_DIAGNOSTICS_TABLE_NAME';
  IF (v_diagnostics_table IS NOT NULL) THEN
    drop_table(v_diagnostics_table);
  END IF;
EXCEPTION WHEN OTHERS THEN
  NULL;
  --DBMS_OUTPUT.PUT_LINE('Failed drop_model: '||p_model_name);
END drop_model;

FUNCTION create_new_temp_table_name(prefix IN VARCHAR2, len IN NUMBER)
RETURN VARCHAR2 IS
  v_table_name     VARCHAR2(30);
  v_seed           NUMBER;
BEGIN
  dbms_random.seed(SYS_GUID());
  v_table_name := 'DM$T' || SUBSTR(prefix, 0, 4) || dbms_random.string(NULL, len-8);
```

```
 --DBMS_OUTPUT.PUT_LINE('create_new_temp_table_name: '||v_table_name);
 RETURN v_table_name;
END create_new_temp_table_name;

FUNCTION create_new_temp_table_name(prefix IN VARCHAR2)
RETURN VARCHAR2 IS
BEGIN
 RETURN create_new_temp_table_name(prefix, 30);
END create_new_temp_table_name;

FUNCTION ADD_TEMP_TABLE(tempTables IN OUT NOCOPY TABLE_ARRAY, temp_table IN
VARCHAR2) RETURN VARCHAR2 IS
BEGIN
 tempTables.EXTEND;
 tempTables(tempTables.COUNT) := temp_table;
 return temp_table;
END;

PROCEDURE DROP_TEMP_TABLES(tempTables IN OUT NOCOPY TABLE_ARRAY) IS
 v_temp  VARCHAR2(30);
BEGIN
 FOR i IN 1..tempTables.COUNT LOOP
  v_temp := tempTables(i);
  drop_table(v_temp);
  drop_view(v_temp);
  tempTables.DELETE(i);
 END LOOP;
```

```
END;


PROCEDURE CHECK_RESULTS(drop_output IN BOOLEAN,
                        result_name IN VARCHAR2) IS
BEGIN
 -- drop all results if drop = true, otherwise make sure all results don't exist already (raise exception)
 IF result_name IS NOT NULL THEN
  IF drop_output THEN
   drop_table(result_name);
   drop_view(result_name);
  ELSIF (table_exist(result_name)) THEN
   RAISE_APPLICATION_ERROR(-20000, 'Result table exists: '||result_name);
  END IF;
 END IF;
END;


PROCEDURE CHECK_MODEL(drop_output IN BOOLEAN,
                      model_name IN VARCHAR2) IS
BEGIN
 -- drop all results if drop = true, otherwise make sure all results don't exist already (raise exception)
 IF model_name IS NOT NULL THEN
  IF drop_output THEN
   drop_model(model_name);
  ELSIF (model_exist(model_name)) THEN
   RAISE_APPLICATION_ERROR(-20001, 'Model exists: '||model_name);
  END IF;
```

```
 END IF;
END;

PROCEDURE create_table_from_query(query IN OUT NOCOPY LSTMT_REC_TYPE)
IS
 v_cursor     NUMBER;
 v_feedback    INTEGER;
BEGIN
 v_cursor := DBMS_SQL.OPEN_CURSOR;

 DBMS_SQL.PARSE(
  c          => v_cursor,
  statement     => query.lstmt,
  lb          => query.lb,
  ub           => query.ub,
  lfflg        => FALSE,
  language_flag => dbms_sql.native);
 v_feedback := DBMS_SQL.EXECUTE(v_cursor);
 DBMS_SQL.CLOSE_CURSOR(v_cursor);

EXCEPTION WHEN OTHERS THEN
 IF DBMS_SQL.IS_OPEN(v_cursor) THEN
  DBMS_SQL.CLOSE_CURSOR(v_cursor);
 END IF;
 RAISE;
END;
```

```
FUNCTION get_row_count(tableName IN VARCHAR2)
RETURN INTEGER IS
  v_stmt  VARCHAR(100);
  qcount INTEGER := 0;
BEGIN
 v_stmt := 'SELECT COUNT(*) FROM '|| tableName;
 EXECUTE  IMMEDIATE v_stmt INTO qcount;
 RETURN qcount;
END get_row_count;

PROCEDURE SET_EQUAL_DISTRIBUTION (
  counts IN OUT VALUE_COUNT_LIST )
IS
 v_minvalue         NUMBER := 0;
BEGIN
 FOR i IN counts.FIRST..counts.LAST
 LOOP
  IF ( i = counts.FIRST )
    THEN
      v_minvalue := counts(i);
    ELSIF ( counts(i) > 0 AND v_minvalue > counts(i) )
    THEN
     v_minvalue := counts(i);
  END IF;
 END LOOP;

 FOR i IN counts.FIRST..counts.LAST
```

```
 LOOP
  counts(i) := v_minvalue;
 END LOOP;
END SET_EQUAL_DISTRIBUTION;

PROCEDURE GET_STRATIFIED_DISTRIBUTION (
  table_name        VARCHAR2,
  attribute_name    VARCHAR2,
  percentage        NUMBER,
  attr_values   IN OUT NOCOPY TARGET_VALUES_LIST,
  counts        IN OUT NOCOPY VALUE_COUNT_LIST,
  counts_sampled IN OUT NOCOPY VALUE_COUNT_LIST )
IS
 v_tmp_stmt         VARCHAR2(4000);
BEGIN
 v_tmp_stmt :=
    'SELECT /*+ noparallel(t)*/ ' || attribute_name ||
    ', count(*), ROUND ( ( count(*) * ' || percentage || ') / 100.0 ) FROM '|| table_name ||
    ' WHERE ' || attribute_name ||' IS NOT NULL GROUP BY ' || attribute_name;
 EXECUTE IMMEDIATE v_tmp_stmt
 BULK COLLECT INTO attr_values, counts, counts_sampled;
END GET_STRATIFIED_DISTRIBUTION;

FUNCTION GENERATE_STRATIFIED_SQL (
                v_2d_temp_view    VARCHAR2,
                src_table_name    VARCHAR2,
                attr_names        TARGET_VALUES_LIST,
```

```
                attribute_name    VARCHAR2,
                percentage        NUMBER,
                op                VARCHAR2,
                equal_distribution IN BOOLEAN DEFAULT FALSE) RETURN LSTMT_REC_TYPE
IS
 v_tmp_lstmt         LSTMT_REC_TYPE;
 attr_values_res     TARGET_VALUES_LIST;
 counts_res          VALUE_COUNT_LIST;
 counts_sampled_res  VALUE_COUNT_LIST;
 tmp_str             VARCHAR2(4000);
 sample_count        PLS_INTEGER;

BEGIN
 GET_STRATIFIED_DISTRIBUTION(src_table_name, attribute_name, percentage,
attr_values_res, counts_res, counts_sampled_res);
 IF ( equal_distribution = TRUE )
 THEN
  SET_EQUAL_DISTRIBUTION(counts_sampled_res);
 END IF;

 v_tmp_lstmt.ub := 0; -- initialize
 ls_append(v_tmp_lstmt, 'CREATE TABLE ');
 ls_append(v_tmp_lstmt, v_2d_temp_view);
 ls_append(v_tmp_lstmt, ' AS ');
 ls_append(v_tmp_lstmt, '( SELECT ');

 FOR i IN attr_names.FIRST..attr_names.LAST
```

```
 LOOP

  IF ( i != attr_names.FIRST )

  THEN

    ls_append(v_tmp_lstmt,',');

  END IF;


  ls_append(v_tmp_lstmt, attr_names(i));

 END LOOP;


 ls_append(v_tmp_lstmt, ' FROM (SELECT /*+ no_merge */ t.*, row_number()
over(partition by '||attribute_name||' order by ora_hash(ROWNUM)) RNUM FROM ' ||
src_table_name || ' t) WHERE RNUM = 1 OR ');


 FOR i IN attr_values_res.FIRST..attr_values_res.LAST

 LOOP

  IF ( i != attr_values_res.FIRST )

  THEN

    tmp_str := ' OR ';

  END IF;

  IF ( counts_res(i) <= 2 ) THEN

   sample_count := counts_res(i);

  ELSE

   sample_count := counts_sampled_res(i);

  END IF;

  tmp_str := tmp_str ||

   '( ' || attribute_name || ' = ''' || REPLACE(attr_values_res(i), '''', '''''') || '''' ||

    ' AND ORA_HASH(RNUM,(' || counts_res(i) || ' -1),12345) ' || op || sample_count || ')
';
```

```
  ls_append(v_tmp_lstmt, tmp_str );
 END LOOP;
 ls_append(v_tmp_lstmt, ') ');
 return v_tmp_lstmt;
END GENERATE_STRATIFIED_SQL;



PROCEDURE "10FIELD_NB_IMAG917515591_BA"(case_table IN VARCHAR2 DEFAULT
'"ORAMULTI"."IMAGETEXT_TABLE"',
                additional_table_1 IN VARCHAR2 DEFAULT NULL,
                model_name IN VARCHAR2 DEFAULT 'IMAGETEXT_DMMODEL_DT',
                confusion_matrix_name IN VARCHAR2 DEFAULT
'"DM4J$T660537187079_M"',
                lift_result_name IN VARCHAR2 DEFAULT '"DM4J$T66053713463_L"',
                roc_result_name IN VARCHAR2 DEFAULT '"DM4J$T660537169832_R"',
                test_metric_name IN VARCHAR2 DEFAULT '"IMAGETEXT_TESTMETRICS_DT"',
                feature_table IN VARCHAR2 DEFAULT NULL,
                mapping_table IN VARCHAR2 DEFAULT NULL,
                drop_output IN BOOLEAN DEFAULT FALSE)
IS
 additional_data  TABLE_ARRAY := TABLE_ARRAY(
  additional_table_1
 );
 v_tempTables           TABLE_ARRAY := TABLE_ARRAY();
 v_2d_view              VARCHAR2(30);
 v_2d_view_build         VARCHAR2(30);
 v_2d_view_test          VARCHAR2(30);
 v_2d_temp_view          VARCHAR2(30);
```

```
v_txn_views             TABLE_ARRAY := TABLE_ARRAY();
v_txn_views_build       TABLE_ARRAY := TABLE_ARRAY();
v_txn_views_test        TABLE_ARRAY := TABLE_ARRAY();
v_txn_temp_views        TABLE_ARRAY := TABLE_ARRAY();
v_case_data             SQL_STATEMENT_TYPE := case_table;
v_case_id               VARCHAR2(30) := 'DMR$CASE_ID';
v_tmp_lstmt             LSTMT_REC_TYPE;
v_target_value          VARCHAR2(4000) := 'sailboats';
v_num_quantiles         NUMBER := 10;
v_build_data            VARCHAR2(30);
v_test_data             VARCHAR2(30);
v_prior                 VARCHAR2(30);
v_build_setting         VARCHAR2(30);
v_apply_result          VARCHAR2(30);
v_build_cm              VARCHAR2(30);
v_test_cm               VARCHAR2(30);
v_diagnostics_table     VARCHAR2(30);
v_accuracy              NUMBER;
v_area_under_curve      NUMBER;
v_avg_accuracy          NUMBER;
v_predictive_confidence NUMBER;
v_confusion_matrix      VARCHAR2(30);
v_gen_caseId            BOOLEAN := FALSE;
v_txt_build             VARCHAR2(30);
v_txt_test              VARCHAR2(30);
v_content_index         VARCHAR2(30);
v_content_index_pref    VARCHAR2(30);
```

```
  v_category_temp_table     VARCHAR2(30);
  v_term_final_table        VARCHAR2(30);
  v_term_final_table_index  VARCHAR2(30);
  v_mapping_table_index     VARCHAR2(30);
  v_term_final_table_test   VARCHAR2(30);
  pragma autonomous_transaction;
BEGIN
  execute immediate 'Alter session set NLS_NUMERIC_CHARACTERS="."';

  CHECK_MODEL(drop_output, model_name);
  CHECK_RESULTS(drop_output, feature_table);
  CHECK_RESULTS(drop_output, mapping_table);
  CHECK_RESULTS(drop_output, test_metric_name);
  CHECK_RESULTS(drop_output, confusion_matrix_name);
  CHECK_RESULTS(drop_output, lift_result_name);
  CHECK_RESULTS(drop_output, roc_result_name);

  IF (v_gen_caseId) THEN
    v_case_data := ADD_TEMP_TABLE(v_tempTables,
create_new_temp_table_name('DM$T'));
    EXECUTE IMMEDIATE 'CREATE TABLE '||v_case_data||' as SELECT rownum as
DMR$CASE_ID, t.* FROM ('||case_table||') t ';
    EXECUTE IMMEDIATE 'ALTER TABLE '||v_case_data||' add constraint
'||create_new_temp_table_name('PK')||' primary key (DMR$CASE_ID)';
  END IF;

  ----- Start: Input Data Preparation -----
  v_2d_temp_view := ADD_TEMP_TABLE(v_tempTables,
```

```
create_new_temp_table_name('DM$T'));

 ls_append(v_tmp_lstmt, 'CREATE VIEW ');

 ls_append(v_tmp_lstmt, v_2d_temp_view);

 ls_append(v_tmp_lstmt, ' AS ');

 ls_append(v_tmp_lstmt, ' ( ');

 ls_append(v_tmp_lstmt, 'SELECT "IMAGETEXT_TABLE"."IMTEXT_ID" as "DMR$CASE_ID",
"IMAGETEXT_TABLE"."IMAGETEXT1" AS "IMAGETEXT1",

 "IMAGETEXT_TABLE"."IMAGETEXT2" AS "IMAGETEXT2",

 "IMAGETEXT_TABLE"."IMAGETEXT3" AS "IMAGETEXT3",

 "IMAGETEXT_TABLE"."IMAGETEXT4" AS "IMAGETEXT4",

 "IMAGETEXT_TABLE"."IMAGETEXT5" AS "IMAGETEXT5",

 "IMAGETEXT_TABLE"."IMAGETEXT6" AS "IMAGETEXT6",

 "IMAGETEXT_TABLE"."IMAGETEXT7" AS "IMAGETEXT7",

 "IMAGETEXT_TABLE"."IMAGETEXT8" AS "IMAGETEXT8",

 "IMAGETEXT_TABLE"."IMAGETEXT9" AS "IMAGETEXT9",

 "IMAGETEXT_TABLE"."IM_CATEGORY" AS "IM_CATEGORY" FROM (' || v_case_data || ')
"IMAGETEXT_TABLE" ');

 ls_append(v_tmp_lstmt, ' ) ');

 create_table_from_query(v_tmp_lstmt);

 v_2d_view := v_2d_temp_view;



 ----- End: Input Data Preparation -----




 ----- Start: Stratified Split Transformation -----

 v_tmp_lstmt.ub := 0; -- initialize

 v_2d_temp_view := ADD_TEMP_TABLE(v_tempTables,
```

```
create_new_temp_table_name('DM$T'));

ls_append(v_tmp_lstmt, GENERATE_STRATIFIED_SQL(v_2d_temp_view, v_2d_view,
TARGET_VALUES_LIST('"DMR$CASE_ID"',

'"IMAGETEXT1"',

'"IMAGETEXT2"',

'"IMAGETEXT3"',

'"IMAGETEXT4"',

'"IMAGETEXT5"',

'"IMAGETEXT6"',

'"IMAGETEXT7"',

'"IMAGETEXT8"',

'"IMAGETEXT9"',

'"IM_CATEGORY"'), '"IM_CATEGORY"', 60, ' < ' ));

 create_table_from_query(v_tmp_lstmt);

 v_2d_view_build := v_2d_temp_view;


 v_tmp_lstmt.ub := 0; -- initialize

 v_2d_temp_view := ADD_TEMP_TABLE(v_tempTables,
create_new_temp_table_name('DM$T'));

ls_append(v_tmp_lstmt, GENERATE_STRATIFIED_SQL(v_2d_temp_view, v_2d_view,
TARGET_VALUES_LIST('"DMR$CASE_ID"',

'"IMAGETEXT1"',

'"IMAGETEXT2"',

'"IMAGETEXT3"',

'"IMAGETEXT4"',

'"IMAGETEXT5"',

'"IMAGETEXT6"',

'"IMAGETEXT7"',
```

```
'"IMAGETEXT8"',

'"IMAGETEXT9"',

'"IM_CATEGORY"'), '"IM_CATEGORY"', 60, ' >= ' ));

 create_table_from_query(v_tmp_lstmt);

 v_2d_view_test := v_2d_temp_view;


 ----- End: Stratified Split Transformation -----




 ----- Start: Mining Data Preparation -----

 v_tmp_lstmt.ub := 0; -- initialize

 v_2d_temp_view := ADD_TEMP_TABLE(v_tempTables,
create_new_temp_table_name('DM$T'));

 ls_append(v_tmp_lstmt, 'CREATE VIEW ');

 ls_append(v_tmp_lstmt, v_2d_temp_view);

 ls_append(v_tmp_lstmt, ' AS ');

 ls_append(v_tmp_lstmt, ' ( ');

 ls_append(v_tmp_lstmt,

 'SELECT caseTable."DMR$CASE_ID"

, caseTable."IMAGETEXT1"

, caseTable."IMAGETEXT2"

, caseTable."IMAGETEXT3"

, caseTable."IMAGETEXT4"

, caseTable."IMAGETEXT5"

, caseTable."IMAGETEXT6"

, caseTable."IMAGETEXT7"
```

```
, caseTable."IMAGETEXT8"

, caseTable."IMAGETEXT9"

, caseTable."IM_CATEGORY"

 FROM ('); ls_append(v_tmp_lstmt, v_2d_view_build); ls_append(v_tmp_lstmt, ') caseTable


'

  );

  ls_append(v_tmp_lstmt, ' ) ');

  create_table_from_query(v_tmp_lstmt);

  v_2d_view_build := v_2d_temp_view;


  v_tmp_lstmt.ub := 0; -- initialize

  v_2d_temp_view := ADD_TEMP_TABLE(v_tempTables,
create_new_temp_table_name('DM$T'));

  ls_append(v_tmp_lstmt, 'CREATE VIEW ');

  ls_append(v_tmp_lstmt, v_2d_temp_view);

  ls_append(v_tmp_lstmt, ' AS ');

  ls_append(v_tmp_lstmt, ' ( ');

  ls_append(v_tmp_lstmt,

  'SELECT caseTable."DMR$CASE_ID"

, caseTable."IMAGETEXT1"

, caseTable."IMAGETEXT2"

, caseTable."IMAGETEXT3"

, caseTable."IMAGETEXT4"

, caseTable."IMAGETEXT5"

, caseTable."IMAGETEXT6"

, caseTable."IMAGETEXT7"
```

```
, caseTable."IMAGETEXT8"

, caseTable."IMAGETEXT9"

, caseTable."IM_CATEGORY"

 FROM ('); ls_append(v_tmp_lstmt, v_2d_view_test); ls_append(v_tmp_lstmt, ') caseTable


'

 );

 ls_append(v_tmp_lstmt, ' ) ');

 create_table_from_query(v_tmp_lstmt);

 v_2d_view_test := v_2d_temp_view;


 v_build_data := v_2d_view_build;

 v_test_data := v_2d_view_test;


 ----- End: Mining Data Preparation -----









 v_build_cm := ADD_TEMP_TABLE(v_tempTables,
create_new_temp_table_name('DM$T'));

 EXECUTE IMMEDIATE 'CREATE TABLE ' || v_build_cm || ' (actual_target_value
VARCHAR2(4000), predicted_target_value VARCHAR2(4000), cost NUMBER)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_cm || ' VALUES (''flowers'', ''flowers'',
0.0)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_cm || ' VALUES (''flowers'', ''cars'',
3.514388489208633)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_cm || ' VALUES (''flowers'', ''medpics'',
```

```
3.514388489208633)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_cm || ' VALUES (''flowers'', ''sailboats'',
3.514388489208633)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_cm || ' VALUES (''cars'', ''flowers'',
3.565693430656934)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_cm || ' VALUES (''cars'', ''cars'', 0.0)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_cm || ' VALUES (''cars'', ''medpics'',
3.565693430656934)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_cm || ' VALUES (''cars'', ''sailboats'',
3.565693430656934)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_cm || ' VALUES (''sailboats'', ''flowers'',
2.684065934065934)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_cm || ' VALUES (''sailboats'', ''cars'',
2.684065934065934)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_cm || ' VALUES (''sailboats'', ''medpics'',
2.684065934065934)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_cm || ' VALUES (''sailboats'', ''sailboats'',
0.0)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_cm || ' VALUES (''medpics'', ''flowers'',
16.016393442622952)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_cm || ' VALUES (''medpics'', ''cars'',
16.016393442622952)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_cm || ' VALUES (''medpics'', ''medpics'',
0.0)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_cm || ' VALUES (''medpics'', ''sailboats'',
16.016393442622952)';

 COMMIT;



 v_build_setting := ADD_TEMP_TABLE(v_tempTables,
create_new_temp_table_name('DM$T'));

 EXECUTE IMMEDIATE 'CREATE TABLE ' || v_build_setting || ' (setting_name
```

```
VARCHAR2(30), setting_value VARCHAR2(128))';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_setting || ' VALUES
(''TREE_TERM_MINREC_SPLIT'', ''20'')';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_setting || ' VALUES
(''CLAS_COST_TABLE_NAME'', :costMatrix)' USING v_build_cm;

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_setting || ' VALUES
(''TREE_TERM_MINREC_NODE'', ''10'')';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_setting || ' VALUES
(''TREE_IMPURITY_METRIC'', ''TREE_IMPURITY_GINI'')';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_setting || ' VALUES
(''TREE_TERM_MAX_DEPTH'', ''7'')';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_setting || ' VALUES (''ALGO_NAME'',
''ALGO_DECISION_TREE'')';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_setting || ' VALUES
(''TREE_TERM_MINPCT_SPLIT'', ''0.1'')';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_setting || ' VALUES
(''TREE_TERM_MINPCT_NODE'', ''0.05'')';

 COMMIT;



 -- BUILD MODEL

 DBMS_DATA_MINING.CREATE_MODEL(

  model_name          => model_name,

  mining_function     => dbms_data_mining.classification,

  data_table_name     => v_build_data,

  case_id_column_name => v_case_id,

  target_column_name  => 'IM_CATEGORY',

  settings_table_name => v_build_setting);
```

```
 -- TEST MODEL

 IF (test_metric_name IS NOT NULL) THEN

   -- CREATE APPLY RESULT FOR TEST

   v_apply_result := ADD_TEMP_TABLE(v_tempTables,
create_new_temp_table_name('DM$T'));


   DBMS_DATA_MINING.APPLY(

     model_name         => model_name,

     data_table_name     => v_test_data,

     case_id_column_name => v_case_id,

     result_table_name   => v_apply_result);


   EXECUTE IMMEDIATE 'CREATE TABLE ' || test_metric_name || ' (METRIC_NAME
VARCHAR2(30), METRIC_VARCHAR_VALUE VARCHAR2(31), METRIC_NUM_VALUE
NUMBER)';

   EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_VARCHAR_VALUE) VALUES (''MODEL_NAME'', :model)' USING model_name;

   EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_VARCHAR_VALUE) VALUES (''TEST_DATA_NAME'', :test_data)' USING v_test_data;

   EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_VARCHAR_VALUE) VALUES (''MINING_FUNCTION'', ''CLASSIFICATION'')';

   EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_VARCHAR_VALUE) VALUES (''TARGET_ATTRIBUTE'', :target)' USING
'IM_CATEGORY';

   EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_VARCHAR_VALUE) VALUES (''POSITIVE_TARGET_VALUE'', :target_value)' USING
v_target_value;

   COMMIT;


   IF confusion_matrix_name IS NULL THEN
```

```
    v_confusion_matrix := ADD_TEMP_TABLE(v_tempTables,
create_new_temp_table_name('DM$T'));
  ELSE
    v_confusion_matrix := confusion_matrix_name;
  END IF;

  DBMS_DATA_MINING.COMPUTE_CONFUSION_MATRIX (
    accuracy                => v_accuracy,
    apply_result_table_name     => v_apply_result,
    target_table_name          => v_test_data,
    case_id_column_name        => v_case_id,
    target_column_name         => 'IM_CATEGORY',
    confusion_matrix_table_name => v_confusion_matrix,
    score_column_name          => 'PREDICTION',
    score_criterion_column_name => 'PROBABILITY',
    cost_matrix_table_name     => v_test_cm,
    apply_result_schema_name    => null,
    target_schema_name         => null,
    cost_matrix_schema_name     => null
    , score_criterion_type => 'COST'
    );
  -- DBMS_OUTPUT.PUT_LINE('**** MODEL ACCURACY ****: ' || ROUND(accuracy, 4));

  IF (confusion_matrix_name IS NOT NULL) THEN
    EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_NUM_VALUE) VALUES (''ACCURACY'', :accuracy)' USING v_accuracy;
    EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_VARCHAR_VALUE) VALUES (''CONFUSION_MATRIX_TABLE'',
```

```
:confusion_matrix_name)' USING confusion_matrix_name;
    COMMIT;


    -- Average Accuracy
    EXECUTE IMMEDIATE '
     WITH
     a as
      (SELECT a.actual_target_value, sum(a.value) recall_total
        FROM ' || confusion_matrix_name || ' a
        group by a.actual_target_value)
      ,
     b as
      (SELECT count(distinct b.actual_target_value) num_recalls
        FROM ' || confusion_matrix_name || ' b)
      ,
     c as
      (SELECT c.actual_target_value, value
        FROM ' || confusion_matrix_name || ' c
        where actual_target_value = predicted_target_value)
      ,
     d as
      (SELECT sum(c.value/a.recall_total) tot_accuracy
        FROM a, c
        where a.actual_target_value = c.actual_target_value)
     SELECT d.tot_accuracy/b.num_recalls * 100 avg_accuracy
     FROM b, d' INTO v_avg_accuracy;
    EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_NUM_VALUE) VALUES (''AVG_ACCURACY'', :avg_accuracy)' USING
```

```
v_avg_accuracy;
    COMMIT;
  END IF;

  -- Predictive Confidence
  EXECUTE IMMEDIATE '
   WITH
   a as
    (SELECT a.actual_target_value, sum(a.value) recall_total
      FROM ' || v_confusion_matrix || ' a
      group by a.actual_target_value)
    ,
   b as
    (SELECT count(distinct b.actual_target_value) num_classes
      FROM ' || v_confusion_matrix || ' b)
    ,
   c as
    (SELECT c.actual_target_value, value
      FROM ' || v_confusion_matrix || ' c
      where actual_target_value = predicted_target_value)
    ,
   d as
   (SELECT sum(c.value/a.recall_total) tot_accuracy
      FROM a, c
      where a.actual_target_value = c.actual_target_value)
   SELECT (1 - (1 - d.tot_accuracy/b.num_classes) / GREATEST(0.0001, ((b.num_classes-
1)/b.num_classes))) * 100
```

```
    FROM b, d' INTO v_predictive_confidence;

  EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_NUM_VALUE) VALUES (''PREDICTIVE_CONFIDENCE'', :predictive_confidence)'
USING v_predictive_confidence;

  COMMIT;


  IF lift_result_name IS NOT NULL AND v_target_value IS NOT NULL THEN

   DBMS_DATA_MINING.COMPUTE_LIFT (

    apply_result_table_name   => v_apply_result,

    target_table_name        => v_test_data,

    case_id_column_name      => v_case_id,

    target_column_name       => 'IM_CATEGORY',

    lift_table_name          => lift_result_name,

    positive_target_value    => v_target_value,

    num_quantiles            => v_num_quantiles,

    cost_matrix_table_name   => v_test_cm,

    apply_result_schema_name  => null,

    target_schema_name       => null,

    cost_matrix_schema_name   => null

    , score_criterion_type => 'COST'

    );

   EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_VARCHAR_VALUE) VALUES (''LIFT_TABLE'', :lift_result_name)' USING
lift_result_name;

   COMMIT;

  END IF;


  IF roc_result_name IS NOT NULL AND v_target_value IS NOT NULL THEN
```

```
    DBMS_DATA_MINING.COMPUTE_ROC (
      roc_area_under_curve       => v_area_under_curve,
      apply_result_table_name    => v_apply_result,
      target_table_name          => v_test_data,
      case_id_column_name        => v_case_id,
      target_column_name         => 'IM_CATEGORY',
      roc_table_name             => roc_result_name,
      positive_target_value      => v_target_value,
      score_column_name          => 'PREDICTION',
      score_criterion_column_name => 'PROBABILITY');
    -- DBMS_OUTPUT.PUT_LINE('**** AREA UNDER ROC CURVE ****: ' ||
area_under_curve );

    EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_VARCHAR_VALUE) VALUES (''ROC_TABLE'', :roc_result_name)' USING
roc_result_name;
    EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_NUM_VALUE) VALUES (''AREA_UNDER_CURVE'', :v_area_under_curve)' USING
v_area_under_curve;
    COMMIT;
   END IF;
  END IF;

  DROP_TEMP_TABLES(v_tempTables);

EXCEPTION WHEN OTHERS THEN
  DROP_TEMP_TABLES(v_tempTables);
  RAISE;
END;
```

```
PROCEDURE "IMAGETEXT_SVM219913624_BA"(case_table IN VARCHAR2 DEFAULT
'"ORAMULTI"."IMAGETEXT_TABLE"',
                additional_table_1 IN VARCHAR2 DEFAULT NULL,
                model_name IN VARCHAR2 DEFAULT 'IMAGETEXT_MODEL_SVM',
                confusion_matrix_name IN VARCHAR2 DEFAULT NULL,
                lift_result_name IN VARCHAR2 DEFAULT NULL,
                roc_result_name IN VARCHAR2 DEFAULT NULL,
                test_metric_name IN VARCHAR2 DEFAULT NULL,
                feature_table IN VARCHAR2 DEFAULT NULL,
                mapping_table IN VARCHAR2 DEFAULT NULL,
                drop_output IN BOOLEAN DEFAULT FALSE)
IS
 additional_data  TABLE_ARRAY := TABLE_ARRAY(
  additional_table_1
 );
 v_tempTables          TABLE_ARRAY := TABLE_ARRAY();
 v_2d_view             VARCHAR2(30);
 v_2d_view_build       VARCHAR2(30);
 v_2d_view_test        VARCHAR2(30);
 v_2d_temp_view        VARCHAR2(30);
 v_txn_views           TABLE_ARRAY := TABLE_ARRAY();
 v_txn_views_build     TABLE_ARRAY := TABLE_ARRAY();
 v_txn_views_test      TABLE_ARRAY := TABLE_ARRAY();
 v_txn_temp_views      TABLE_ARRAY := TABLE_ARRAY();
 v_case_data           SQL_STATEMENT_TYPE := case_table;
 v_case_id             VARCHAR2(30) := 'DMR$CASE_ID';
```

```
v_tmp_lstmt             LSTMT_REC_TYPE;
v_target_value          VARCHAR2(4000) := 'sailboats';
v_num_quantiles         NUMBER := 10;
v_build_data            VARCHAR2(30);
v_test_data             VARCHAR2(30);
v_prior                 VARCHAR2(30);
v_build_setting         VARCHAR2(30);
v_apply_result          VARCHAR2(30);
v_build_cm              VARCHAR2(30);
v_test_cm               VARCHAR2(30);
v_diagnostics_table     VARCHAR2(30);
v_accuracy              NUMBER;
v_area_under_curve      NUMBER;
v_avg_accuracy          NUMBER;
v_predictive_confidence NUMBER;
v_confusion_matrix      VARCHAR2(30);
v_gen_caseId            BOOLEAN := FALSE;
v_txt_build             VARCHAR2(30);
v_txt_test              VARCHAR2(30);
v_content_index         VARCHAR2(30);
v_content_index_pref    VARCHAR2(30);
v_category_temp_table   VARCHAR2(30);
v_term_final_table      VARCHAR2(30);
v_term_final_table_index VARCHAR2(30);
v_mapping_table_index   VARCHAR2(30);
v_term_final_table_test VARCHAR2(30);
pragma autonomous_transaction;
```

```
BEGIN
 execute immediate 'Alter session set NLS_NUMERIC_CHARACTERS="."';

 CHECK_MODEL(drop_output, model_name);
 CHECK_RESULTS(drop_output, feature_table);
 CHECK_RESULTS(drop_output, mapping_table);
 CHECK_RESULTS(drop_output, test_metric_name);
 CHECK_RESULTS(drop_output, confusion_matrix_name);
 CHECK_RESULTS(drop_output, lift_result_name);
 CHECK_RESULTS(drop_output, roc_result_name);

 IF (v_gen_caseId) THEN
  v_case_data := ADD_TEMP_TABLE(v_tempTables,
create_new_temp_table_name('DM$T'));
  EXECUTE IMMEDIATE 'CREATE TABLE '||v_case_data||' as SELECT rownum as
DMR$CASE_ID, t.* FROM ('||case_table||') t ';
  EXECUTE IMMEDIATE 'ALTER TABLE '||v_case_data||' add constraint
'||create_new_temp_table_name('PK')||' primary key (DMR$CASE_ID)';
 END IF;

 ----- Start: Input Data Preparation -----
 v_2d_temp_view := ADD_TEMP_TABLE(v_tempTables,
create_new_temp_table_name('DM$T'));
 ls_append(v_tmp_lstmt, 'CREATE VIEW ');
 ls_append(v_tmp_lstmt, v_2d_temp_view);
 ls_append(v_tmp_lstmt, ' AS ');
 ls_append(v_tmp_lstmt, ' ( ');
 ls_append(v_tmp_lstmt, 'SELECT "IMAGETEXT_TABLE"."IMTEXT_ID" as "DMR$CASE_ID",
"IMAGETEXT_TABLE"."IMAGETEXT1" AS "IMAGETEXT1",
```

```
"IMAGETEXT_TABLE"."IMAGETEXT10" AS "IMAGETEXT10",
"IMAGETEXT_TABLE"."IMAGETEXT11" AS "IMAGETEXT11",
"IMAGETEXT_TABLE"."IMAGETEXT12" AS "IMAGETEXT12",
"IMAGETEXT_TABLE"."IMAGETEXT13" AS "IMAGETEXT13",
"IMAGETEXT_TABLE"."IMAGETEXT14" AS "IMAGETEXT14",
"IMAGETEXT_TABLE"."IMAGETEXT15" AS "IMAGETEXT15",
"IMAGETEXT_TABLE"."IMAGETEXT16" AS "IMAGETEXT16",
"IMAGETEXT_TABLE"."IMAGETEXT17" AS "IMAGETEXT17",
"IMAGETEXT_TABLE"."IMAGETEXT18" AS "IMAGETEXT18",
"IMAGETEXT_TABLE"."IMAGETEXT19" AS "IMAGETEXT19",
"IMAGETEXT_TABLE"."IMAGETEXT2" AS "IMAGETEXT2",
"IMAGETEXT_TABLE"."IMAGETEXT20" AS "IMAGETEXT20",
"IMAGETEXT_TABLE"."IMAGETEXT21" AS "IMAGETEXT21",
"IMAGETEXT_TABLE"."IMAGETEXT22" AS "IMAGETEXT22",
"IMAGETEXT_TABLE"."IMAGETEXT23" AS "IMAGETEXT23",
"IMAGETEXT_TABLE"."IMAGETEXT24" AS "IMAGETEXT24",
"IMAGETEXT_TABLE"."IMAGETEXT25" AS "IMAGETEXT25",
"IMAGETEXT_TABLE"."IMAGETEXT26" AS "IMAGETEXT26",
"IMAGETEXT_TABLE"."IMAGETEXT27" AS "IMAGETEXT27",
"IMAGETEXT_TABLE"."IMAGETEXT28" AS "IMAGETEXT28",
"IMAGETEXT_TABLE"."IMAGETEXT29" AS "IMAGETEXT29",
"IMAGETEXT_TABLE"."IMAGETEXT3" AS "IMAGETEXT3",
"IMAGETEXT_TABLE"."IMAGETEXT30" AS "IMAGETEXT30",
"IMAGETEXT_TABLE"."IMAGETEXT31" AS "IMAGETEXT31",
"IMAGETEXT_TABLE"."IMAGETEXT32" AS "IMAGETEXT32",
"IMAGETEXT_TABLE"."IMAGETEXT33" AS "IMAGETEXT33",
"IMAGETEXT_TABLE"."IMAGETEXT34" AS "IMAGETEXT34",
```

```
 "IMAGETEXT_TABLE"."IMAGETEXT35" AS "IMAGETEXT35",

 "IMAGETEXT_TABLE"."IMAGETEXT36" AS "IMAGETEXT36",

 "IMAGETEXT_TABLE"."IMAGETEXT37" AS "IMAGETEXT37",

 "IMAGETEXT_TABLE"."IMAGETEXT38" AS "IMAGETEXT38",

 "IMAGETEXT_TABLE"."IMAGETEXT39" AS "IMAGETEXT39",

 "IMAGETEXT_TABLE"."IMAGETEXT4" AS "IMAGETEXT4",

 "IMAGETEXT_TABLE"."IMAGETEXT40" AS "IMAGETEXT40",

 "IMAGETEXT_TABLE"."IMAGETEXT41" AS "IMAGETEXT41",

 "IMAGETEXT_TABLE"."IMAGETEXT42" AS "IMAGETEXT42",

 "IMAGETEXT_TABLE"."IMAGETEXT43" AS "IMAGETEXT43",

 "IMAGETEXT_TABLE"."IMAGETEXT44" AS "IMAGETEXT44",

 "IMAGETEXT_TABLE"."IMAGETEXT45" AS "IMAGETEXT45",

 "IMAGETEXT_TABLE"."IMAGETEXT46" AS "IMAGETEXT46",

 "IMAGETEXT_TABLE"."IMAGETEXT47" AS "IMAGETEXT47",

 "IMAGETEXT_TABLE"."IMAGETEXT48" AS "IMAGETEXT48",

 "IMAGETEXT_TABLE"."IMAGETEXT49" AS "IMAGETEXT49",

 "IMAGETEXT_TABLE"."IMAGETEXT5" AS "IMAGETEXT5",

 "IMAGETEXT_TABLE"."IMAGETEXT50" AS "IMAGETEXT50",

 "IMAGETEXT_TABLE"."IMAGETEXT6" AS "IMAGETEXT6",

 "IMAGETEXT_TABLE"."IMAGETEXT7" AS "IMAGETEXT7",

 "IMAGETEXT_TABLE"."IMAGETEXT8" AS "IMAGETEXT8",

 "IMAGETEXT_TABLE"."IMAGETEXT9" AS "IMAGETEXT9",

 "IMAGETEXT_TABLE"."IMAGE_ID" AS "IMAGE_ID",

 "IMAGETEXT_TABLE"."IM_CATEGORY" AS "IM_CATEGORY" FROM (' || v_case_data || ')
"IMAGETEXT_TABLE" ');

 ls_append(v_tmp_lstmt, ' ) ');

 create_table_from_query(v_tmp_lstmt);
```

```
  v_2d_view := v_2d_temp_view;


  ----- End: Input Data Preparation -----




  ----- Start: Mining Data Preparation -----

  v_tmp_lstmt.ub := 0; -- initialize

  v_2d_temp_view := ADD_TEMP_TABLE(v_tempTables,
create_new_temp_table_name('DM$T'));

  ls_append(v_tmp_lstmt, 'CREATE VIEW ');

  ls_append(v_tmp_lstmt, v_2d_temp_view);

  ls_append(v_tmp_lstmt, ' AS ');

  ls_append(v_tmp_lstmt, ' ( ');

  ls_append(v_tmp_lstmt,

  'SELECT caseTable."DMR$CASE_ID"

, caseTable."IMAGETEXT1"

, caseTable."IMAGETEXT10"

, caseTable."IMAGETEXT11"

, caseTable."IMAGETEXT12"

, caseTable."IMAGETEXT13"

, caseTable."IMAGETEXT14"

, caseTable."IMAGETEXT15"

, caseTable."IMAGETEXT16"

, caseTable."IMAGETEXT17"

, caseTable."IMAGETEXT18"

, caseTable."IMAGETEXT19"
```

```
, caseTable."IMAGETEXT2"
, caseTable."IMAGETEXT20"
, caseTable."IMAGETEXT21"
, caseTable."IMAGETEXT22"
, caseTable."IMAGETEXT23"
, caseTable."IMAGETEXT24"
, caseTable."IMAGETEXT25"
, caseTable."IMAGETEXT26"
, caseTable."IMAGETEXT27"
, caseTable."IMAGETEXT28"
, caseTable."IMAGETEXT29"
, caseTable."IMAGETEXT3"
, caseTable."IMAGETEXT30"
, caseTable."IMAGETEXT31"
, caseTable."IMAGETEXT32"
, caseTable."IMAGETEXT33"
, caseTable."IMAGETEXT34"
, caseTable."IMAGETEXT35"
, caseTable."IMAGETEXT36"
, caseTable."IMAGETEXT37"
, caseTable."IMAGETEXT38"
, caseTable."IMAGETEXT39"
, caseTable."IMAGETEXT4"
, caseTable."IMAGETEXT40"
, caseTable."IMAGETEXT41"
, caseTable."IMAGETEXT42"
, caseTable."IMAGETEXT43"
```

```
, caseTable."IMAGETEXT44"

, caseTable."IMAGETEXT45"

, caseTable."IMAGETEXT46"

, caseTable."IMAGETEXT47"

, caseTable."IMAGETEXT48"

, caseTable."IMAGETEXT49"

, caseTable."IMAGETEXT5"

, caseTable."IMAGETEXT50"

, caseTable."IMAGETEXT6"

, caseTable."IMAGETEXT7"

, caseTable."IMAGETEXT8"

, caseTable."IMAGETEXT9"

, caseTable."IM_CATEGORY"

 FROM ('); ls_append(v_tmp_lstmt, v_2d_view); ls_append(v_tmp_lstmt, ') caseTable


'

 );

 ls_append(v_tmp_lstmt, ' ) ');

 create_table_from_query(v_tmp_lstmt);

 v_2d_view := v_2d_temp_view;


 v_build_data := v_2d_view;

 v_test_data := v_2d_view;


 ----- End: Mining Data Preparation -----
```

```
 v_prior := ADD_TEMP_TABLE(v_tempTables, create_new_temp_table_name('DM$T'));

 EXECUTE IMMEDIATE 'CREATE TABLE ' || v_prior || ' (TARGET_VALUE VARCHAR2(4000),
PRIOR_PROBABILITY NUMBER)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_prior || ' VALUES (''flowers'',
0.13631942203162073)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_prior || ' VALUES (''cars'',
0.13830948658682685)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_prior || ' VALUES (''medpics'',
0.6212590053244353)';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_prior || ' VALUES (''sailboats'',
0.10411208605711691)';

 COMMIT;




 v_build_setting := ADD_TEMP_TABLE(v_tempTables,
create_new_temp_table_name('DM$T'));

 EXECUTE IMMEDIATE 'CREATE TABLE ' || v_build_setting || ' (setting_name
VARCHAR2(30), setting_value VARCHAR2(128))';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_setting || ' VALUES
(''SVMS_ACTIVE_LEARNING'', ''SVMS_AL_ENABLE'')';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_setting || ' VALUES (''ALGO_NAME'',
''ALGO_SUPPORT_VECTOR_MACHINES'')';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_setting || ' VALUES
(''SVMS_CONV_TOLERANCE'', ''0.0010'')';

 EXECUTE IMMEDIATE 'INSERT INTO ' || v_build_setting || ' VALUES
(''CLAS_PRIORS_TABLE_NAME'', :priorTable)' USING v_prior;

 COMMIT;
```

```
 -- BUILD MODEL
 DBMS_DATA_MINING.CREATE_MODEL(
  model_name          => model_name,
  mining_function     => dbms_data_mining.classification,
  data_table_name     => v_build_data,
  case_id_column_name => v_case_id,
  target_column_name  => 'IM_CATEGORY',
  settings_table_name => v_build_setting);




 -- TEST MODEL
 IF (test_metric_name IS NOT NULL) THEN
  -- CREATE APPLY RESULT FOR TEST
  v_apply_result := ADD_TEMP_TABLE(v_tempTables,
create_new_temp_table_name('DM$T'));


  DBMS_DATA_MINING.APPLY(
   model_name          => model_name,
   data_table_name     => v_test_data,
   case_id_column_name => v_case_id,
   result_table_name   => v_apply_result);


  EXECUTE IMMEDIATE 'CREATE TABLE ' || test_metric_name || ' (METRIC_NAME
VARCHAR2(30), METRIC_VARCHAR_VALUE VARCHAR2(31), METRIC_NUM_VALUE
NUMBER)';
  EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
```

```
METRIC_VARCHAR_VALUE) VALUES (''MODEL_NAME'', :model)' USING model_name;

   EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_VARCHAR_VALUE) VALUES (''TEST_DATA_NAME'', :test_data)' USING v_test_data;

   EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_VARCHAR_VALUE) VALUES (''MINING_FUNCTION'', ''CLASSIFICATION'')';

   EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_VARCHAR_VALUE) VALUES (''TARGET_ATTRIBUTE'', :target)' USING
'IM_CATEGORY';

   EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_VARCHAR_VALUE) VALUES (''POSITIVE_TARGET_VALUE'', :target_value)' USING
v_target_value;

   COMMIT;



   IF confusion_matrix_name IS NULL THEN

    v_confusion_matrix := ADD_TEMP_TABLE(v_tempTables,
create_new_temp_table_name('DM$T'));

   ELSE

    v_confusion_matrix := confusion_matrix_name;

   END IF;



   DBMS_DATA_MINING.COMPUTE_CONFUSION_MATRIX (

    accuracy                => v_accuracy,

    apply_result_table_name    => v_apply_result,

    target_table_name         => v_test_data,

    case_id_column_name       => v_case_id,

    target_column_name        => 'IM_CATEGORY',

    confusion_matrix_table_name => v_confusion_matrix,

    score_column_name         => 'PREDICTION',

    score_criterion_column_name => 'PROBABILITY',

    cost_matrix_table_name     => v_test_cm,
```

```
    apply_result_schema_name    => null,
    target_schema_name          => null,
    cost_matrix_schema_name     => null
    , score_criterion_type => 'COST'
    );
  -- DBMS_OUTPUT.PUT_LINE('**** MODEL ACCURACY ****: ' || ROUND(accuracy, 4));


  IF (confusion_matrix_name IS NOT NULL) THEN
    EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_NUM_VALUE) VALUES (''ACCURACY'', :accuracy)' USING v_accuracy;
    EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_VARCHAR_VALUE) VALUES (''CONFUSION_MATRIX_TABLE'',
:confusion_matrix_name)' USING confusion_matrix_name;
    COMMIT;


    -- Average Accuracy
    EXECUTE IMMEDIATE '
     WITH
     a as
      (SELECT a.actual_target_value, sum(a.value) recall_total
        FROM ' || confusion_matrix_name || ' a
        group by a.actual_target_value)
      ,
     b as
      (SELECT count(distinct b.actual_target_value) num_recalls
        FROM ' || confusion_matrix_name || ' b)
      ,
     c as
```

```
        (SELECT c.actual_target_value, value
          FROM ' || confusion_matrix_name || ' c
          where actual_target_value = predicted_target_value)
        ,
       d as
        (SELECT sum(c.value/a.recall_total) tot_accuracy
          FROM a, c
          where a.actual_target_value = c.actual_target_value)
       SELECT d.tot_accuracy/b.num_recalls * 100 avg_accuracy
       FROM b, d' INTO v_avg_accuracy;
     EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_NUM_VALUE) VALUES (''AVG_ACCURACY'', :avg_accuracy)' USING
v_avg_accuracy;
     COMMIT;
   END IF;


   -- Predictive Confidence
   EXECUTE IMMEDIATE '
    WITH
    a as
     (SELECT a.actual_target_value, sum(a.value) recall_total
       FROM ' || v_confusion_matrix || ' a
       group by a.actual_target_value)
     ,
    b as
     (SELECT count(distinct b.actual_target_value) num_classes
       FROM ' || v_confusion_matrix || ' b)
     ,
```

```
    c as
     (SELECT c.actual_target_value, value
       FROM ' || v_confusion_matrix || ' c
       where actual_target_value = predicted_target_value)
     ,
    d as
     (SELECT sum(c.value/a.recall_total) tot_accuracy
       FROM a, c
       where a.actual_target_value = c.actual_target_value)
    SELECT (1 - (1 - d.tot_accuracy/b.num_classes) / GREATEST(0.0001, ((b.num_classes-
1)/b.num_classes))) * 100
    FROM b, d' INTO v_predictive_confidence;
  EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_NUM_VALUE) VALUES (''PREDICTIVE_CONFIDENCE'', :predictive_confidence)'
USING v_predictive_confidence;
  COMMIT;


  IF lift_result_name IS NOT NULL AND v_target_value IS NOT NULL THEN
   DBMS_DATA_MINING.COMPUTE_LIFT (
    apply_result_table_name   => v_apply_result,
    target_table_name       => v_test_data,
    case_id_column_name      => v_case_id,
    target_column_name       => 'IM_CATEGORY',
    lift_table_name         => lift_result_name,
    positive_target_value    => v_target_value,
    num_quantiles           => v_num_quantiles,
    cost_matrix_table_name   => v_test_cm,
    apply_result_schema_name  => null,
```

```
      target_schema_name       => null,
      cost_matrix_schema_name   => null
      , score_criterion_type => 'COST'
      );
    EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_VARCHAR_VALUE) VALUES (''LIFT_TABLE'', :lift_result_name)' USING
lift_result_name;
    COMMIT;
  END IF;

  IF roc_result_name IS NOT NULL AND v_target_value IS NOT NULL THEN
    DBMS_DATA_MINING.COMPUTE_ROC (
      roc_area_under_curve       => v_area_under_curve,
      apply_result_table_name    => v_apply_result,
      target_table_name          => v_test_data,
      case_id_column_name        => v_case_id,
      target_column_name         => 'IM_CATEGORY',
      roc_table_name             => roc_result_name,
      positive_target_value      => v_target_value,
      score_column_name          => 'PREDICTION',
      score_criterion_column_name => 'PROBABILITY');
    -- DBMS_OUTPUT.PUT_LINE('**** AREA UNDER ROC CURVE ****: ' ||
area_under_curve );

    EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_VARCHAR_VALUE) VALUES (''ROC_TABLE'', :roc_result_name)' USING
roc_result_name;
    EXECUTE IMMEDIATE 'INSERT INTO ' || test_metric_name || ' (METRIC_NAME,
METRIC_NUM_VALUE) VALUES (''AREA_UNDER_CURVE'', :v_area_under_curve)' USING
v_area_under_curve;
```

```
    COMMIT;

   END IF;

  END IF;


  DROP_TEMP_TABLES(v_tempTables);


EXCEPTION WHEN OTHERS THEN

  DROP_TEMP_TABLES(v_tempTables);

  RAISE;

END;


END;



/
```

Παράρτημα Β – Ακρωνύμια

- IDE – Integrated Development Environment, Ολοκληρωμένο Περιβάλλον Ανάπτυξης
- Διάγραμμα ER – Διάγραμμα Entity Relationship, Διάγραμμα σχέσεων οντοτήτων (βάσεως δεδομένων)
- SVM - Support Vector Machine (αλγόριθμος data mining)
- RDBMS – Relational DataBase Management System, Σχεσιακή Βάση Δεδομένων
- ORDBMS – Oracle RDBMS
- PMML - Predictive Model Markup Language
- XML – eXtensible Markup Language
- API - Application Programming Interfaces
- JSP - Java Server Pages
- EMR - Electronic Medical Record
- DICOM - Digital Imaging and Communications in Medicine
- MRI - Magnetic Resonance Imaging
- DBA - DataBase Administrator
- OLS - Ordinary Least Squares
- DT – Decision Trees
- SQL - Structured Query Language
- PL/SQL - Procedural Language/Structured Query Language
- PCA - Principal Component Analysis
- AOG - Algorithm Output Granularity
- AWSOM - Arbitrary Window Stream mOdeling Method

Βιβλιογραφία

[1] Oracle Corporation: http://www.oracle.com/index.html

[2] Alapati, Sam R. (2008). Expert Oracle Database 11g Administration. The expert's voice in Oracle. Apress. p. 170. ISBN 9781430210153. Retrieved 2010-07-07. "Oracle databases are logically divided into one or more tablespaces. An Oracle tablespace is a logical entity that contains the physical datafiles."

[3] PMML Project Page, http://sourceforge.net/projects/pmml

[4] Alex Guazzelli, Michael Zeller, Wen-Ching Lin, Graham Williams. PMML: An Open Standard for Sharing Models. The R Journal, vol 1/1, May 2009.

[5] Y. Peng, G. Kou, Y. Shi, Z. Chen (2008). "A Descriptive Framework for the Field of Data Mining and Knowledge Discovery". International Journal of Information Technology and Decision Making, Volume 7, Issue 4 7: 639 – 682.

[6] Alex Guazzelli, Wen-Ching Lin, Tridivesh Jena. PMML in Action: Unleashing the Power of Open Standards for Data Mining and Predictive Analytics. CreateSpace, 2010.

[7] The Data Mining Group (DMG). The DMG is an independent, vendor led group which develops data mining standards, such as the Predictive Model Markup Language (PMML).

[8] Proceedings, International Conferences on Knowledge Discovery and Data Mining, ACM, New York. http://www.kdd.org/conferences.php

[9] SIGKDD Explorations, ACM, New York. http://www.kdd.org/explorations/about.php

[10] 5th International Conference on Data Mining (2009) http://www.dmin--2009.com/

[11] Ian H. Witten, Eibe Frank – 2005 – «Data mining: practical machine learning tools and techniques»

[12] Manish Mehta, Rakesh Agrawal and Jorma Rissanen, «SLIQ: A fast scalable classifier for data mining», ADVANCES IN DATABASE TECHNOLOGY — EDBT '96, Lecture Notes in Computer Science, 1996, Volume 1057/1996, 18-32, DOI: 10.1007/BFb0014141

[13] Jiawei Han, Micheline Kamber – 2006 - Data mining: concepts and techniques

[14] Simon Tong, Daphne Koller, «Support vector machine active learning with applications to text classification», The Journal of Machine Learning Research archive Volume 2, 3/1/2002

[15] Lim, T.-S., Loh, W.-Y., & Shih, Y.-S. (1997). An empirical comparison of decision trees and other classification methods. Technical Report 979, Department of Statistics, University of Wisconsin, Madison.

[16] Cristianini, N., & Shawe-Taylor, J. (2000). Introduction to support vector machines and other kernel-based learning methods. Cambridge, UK: Cambridge University Press.

[17] Hosmer, D. W and Lemeshow, S. (1989), Applied Logistic Regression, John Wiley & Sons, Inc.

[18] Firmin, R. (2002). Advanced time series modeling for semiconductor process control: The fab as a time machine. In Mackulak, G. T., Fowler, J. W., & Schomig, A. (eds.). Proceedings of the International Conference on Modeling and Analysis of Semiconductor Manufacturing (MASM 2002).

[19] Iakovidis, D.K.; Pelekis, N.; Kotsifakos, E.E.; Kopanakis, I.; Karanikas, H.; Theodoridis, Y. “A Pattern Similarity Scheme for Medical Image Retrieval”, IEEE Transactions on Information Technology in Biomedicine, Issue Date: July 2009, Volume: 13 Issue:4, p. 442 – 450

[20] D.E. Maroulis, M. Savelonas, D.K. Iakovidis, S.A. Karkanis, N. Dimitropoulos, “Variable Background Active Contour Model for Computer-Aided Delineation of

Nodules in Thyroid Ultrasound Images," IEEE Trans. Inf. Tech. Biomed., vol. 11, no. 5, pp. 537-543, 2007.
[21] C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic, and W. Equitz, "Efficient and Effective Querying by Image Content," J. of Intelligent Inform. Systems, vol. 3, pp. 231-262, 1994.
[22] W. Cai, D. D. Feng, R. Fulton, "Content Based Retrieval of Dynamic PET Functional Images," IEEE Trans. Inf. Tech. Biomed., vol. 4, no. 2, pp. 152-158, 2000.
[23] Berry, M., J., A., & Linoff, G., S., (2000). Mastering data mining. New York: Wiley.
[24] Agrawal, R. & Srikant, R. (1994). Fast algorithms for mining association rules. Proceedings of the 20th VLDB Conference. Santiago, Chile.
[25] J. Gehrke, F. Korn, and D. Srivastava. On computing correlated aggregates over continual data streams. In Proc. Of the 2001 ACM SIGMOD Intl. Conf. on Management of Data, pages 13–24. acmpress, June 2001.
[26] R. Agrawal, T. Imielinski, and A. Swami. Database mining A performance perspective. IEEE Transactions on Knowledge and Data Eng., 5(6):914-925,, December 1993.
[27] A. Dobra, J. Gehrke, M. Garofalakis, and R. Rastogi. Processing complex aggregate queries over data streams. In Proc. of the 2002 ACM SIGMOD Intl. Conf. on Management of Data, June 2002.
[28] J. Catlett. Megainduction: Machine Learning on Very Large Databases. PhD thesis, Departement of Computer Science, University of Sydney, Sydney, Australia, 1991.
[29] Ruoming Jin and Gagan Agrawal. Communication and Memory Efficient Parallel Decision Tree Construction. In Proceedings of Third SIAM Conference on Data Mining, May 2003.
[30] E H Herskovits, R Chen, "Integrating Data-Mining Support into a Brain-Image Database Using Open-Source Components", Journal of Advances in Medical Sciences, Volume 53, Number 2 / December 2008.
[31] C. Aggarwal, J. Han, J. Wang, P. S. Yu, A Framework for Clustering Evolving Data Streams, Proc. 2003 Int. Conf. on Very Large Data Bases, Berlin, Germany, Sept. 2003.
[32] C. Aggarwal, J. Han, J. Wang, and P. S. Yu, A Framework for Projected Clustering of High Dimensional Data Streams, Proc. 2004 Int. Conf. on Very Large Data Bases, Toronto, Canada, 2004.
[33] C. Aggarwal, J. Han, J. Wang, and P. S. Yu, On Demand Classification of Data Streams, Proc. 2004 Int. Conf. on Knowledge Discovery and Data Mining, Seattle, WA, Aug. 2004.
[34] B. Babcock, S. Babu, M. Datar, R. Motwani, and J. Widom. Models and issues in data stream systems. In Proceedings of PODS, 2002.
[35] B. Babcock, M. Datar, and R. Motwani. Load Shedding Techniques for Data Stream Systems (short paper) In Proc. of the 2003 Workshop on Management and Processing of Data Streams, June 2003
[36] G. Cormode, S. Muthukrishnan What' s hot and what' s not: tracking most frequent items dynamically. PODS 2003: 296-306
[37] Q. Ding, Q. Ding, and W. Perrizo, Decision Tree Classification of Spatial Data Streams Using Peano Count Trees, Proceedings of the ACM Symposium on Applied Computing, Madrid, Spain, March 2002.
[38] P. Domingos and G. Hulten. Mining High-Speed Data Streams. In Proceedings of the Association for Computing Machinery Sixth International Conference on Knowledge Discovery and Data Mining, 2000.
[39] P. Domingos and G. Hulten, A General Method for Scaling Up Machine Learning Algorithms and its Application to Clustering, Proceedings of the Eighteenth

International Conference on Machine Learning, 2001, Williamstown, MA, Morgan Kaufmann

[40] G. Dong, J. Han, L.V.S. Lakshmanan, J. Pei, H. Wang and P.S. Yu. Online mining of changes from data streams: Research problems and preliminary results, In Proceedings of the 2003 ACM SIGMOD Workshop on Management and Processing of Data Streams. In cooperation with the 2003 ACM-SIGMOD International Conference on Management of Data, San Diego, CA, June 8, 2003.

[41] V. Ganti, Johannes Gehrke, Raghu Ramakrishnan: Mining Data Streams under Block Evolution. SIGKDD Explorations 3(2), 2002.

[42] Gaber, M, M., Krishnaswamy, S., and Zaslavsky, A., On-board Mining of Data Streams in Sensor Networks, Accepted as a chapter in the forthcoming book Advanced Methods of Knowledge Discovery from Complex Data, (Eds.) Sanghamitra Badhyopadhyay, Ujjwal Maulik, Lawrence Holder and Diane Cook, Springer Verlag.

[43] Gaber, M, M., Zaslavsky, A., and Krishnaswamy, S., A Cost-Efficient Model for Ubiquitous Data Stream Mining, the Tenth International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, Perugia Italy, July 4-9.

[44] Gaber, M, M., Zaslavsky, A., and Krishnaswamy, S., Towards an Adaptive Approach for Mining Data Streams in Resource Constrained Environments, the Proceedings of Sixth International Conference on Data Warehousing and Knowledge Discovery - Industry Track (DaWak 2004), Zaragoza, Spain, 30 August - 3 September, Lecture Notes in Computer Science (LNCS), Springer Verlag.

[45] A. C. Gilbert, Y. Kotidis, S. Muthukrishnan, M. Strauss: One-Pass Wavelet Decompositions of Data Streams. TKDE 15(3), 2003

[46] H. Kargupta, R. Bhargava, K. Liu, M. Powers, P. Blair, S. Bushra, J. Dull, K. Sarkar, M. Klein, M. Vasa, and D. Handy, VEDAS: A Mobile and Distributed Data Stream Mining System for Real-Time Vehicle Monitoring, Proceedings of SIAM International Conference on Data Mining, 2004.

[47] M. Last, Online Classification of Nonstationary Data Streams, Intelligent Data Analysis, Vol. 6, No. 2, pp. 129-147, 2002.

[48] S. Muthukrishnan (2003), Data streams: algorithms and applications. Proceedings of the fourteenth annual ACM-SIAM symposium on discrete algorithms.

[49] S. Papadimitriou, C. Faloutsos, and A. Brockwell, ωAdaptive, Hands-Off Stream Mining, 29th International Conference on Very Large Data Bases VLDB, 2003.

[50] N. Tatbul, U. Cetintemel, S. Zdonik, M. Cherniack, M. Stonebraker. Load Shedding on Data Streams, In Proceedings of the Workshop on Management and Processing of Data Streams, San Diego, CA, USA, June 8, 2003.

[51] H. Wang, W. Fan, P. Yu and J. Han, Mining Concept-Drifting Data Streams using Ensemble Classifiers, in the 9th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD), Aug. 2003, Washington DC, USA.