

Ατομική Διπλωματική Εργασία

**ΤΑΞΙΝΟΜΗΣΗ ΣΥΝΟΨΕΩΝ ΑΝΤΙΚΕΙΜΕΝΩΝ ΜΕ ΤΗ ΧΡΗΣΗ ΤΕΧΝΙΚΩΝ
ΕΞΟΡΥΞΗΣ ΔΕΔΟΜΕΝΩΝ**

Ιουλία Βανέζη

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2010

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

**Ταξινόμηση συνόψεων αντικειμένων με τη χρήση τεχνικών εξόρυξης
δεδομένων**

Ιουλία Βανέζη

Επιβλέπων Καθηγητής
Δρ.Χρίστος Ν. Σχίζας

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων
απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου
Κύπρου

Μάιος 2010

Ευχαριστίες

Θα ήθελα με την ευκαιρία της συμπλήρωσης της Ατομικής Διπλωματικής Εργασίας μου, να ευχαριστήσω θερμά τον Δρ. Γεώργιο Φάκα ο οποίος για ένα ολόκληρο χρόνο με καθοδηγούσε και μου παρείχε συμβουλές οι οποίες συνέβαλαν στην ολοκλήρωση της εργασίας. Ήταν παρών σε όλη την πορεία ανάπτυξής της και πάντα διέθετε χρόνο για να συζητήσουμε οποιαδήποτε θέματα προέκυπταν.

Ακόμη, θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή Δρ. Χρίστο Σχίζα και φυσικά την οικογένειά μου για την πολύτιμη βοήθεια και την αμέριστη συμπαράσταση που μου έχει προσφέρει όλα αυτά τα χρόνια.

Σας ευχαριστώ

Περίληψη

Στόχος της διπλωματικής εργασίας αυτής, είναι η ανάπτυξη μίας νέας μεθοδολογίας η οποία στηρίζεται σε τεχνικές εξόρυξης δεδομένων (IR-Style τεχνικές), για ταξινόμηση των αποτελεσμάτων που λαμβάνονται από την εκτέλεση αναζητήσεων σε σχεσιακές βάσεις δεδομένων.

Το σύστημα που εφαρμόζει τη νέα μεθοδολογία επιτρέπει την αναζήτηση σε μία σχεσιακή βάση δεδομένων με χρήση δύο λέξεων – κλειδιών. Η πρώτη λέξη – κλειδί καθορίζει το αντικείμενο ή αλλιώς το Data Subject (DS) για το οποίο επιθυμεί να πάρει πληροφορίες ο χρήστης, ενώ η δεύτερη λέξη – κλειδί καθορίζει τη θεματική ενότητα που τον ενδιαφέρει περισσότερο. Το αποτέλεσμα κάθε αναζήτησης είναι ένα σύνολο από συνόψεις αντικειμένων, τα λεγόμενα Object Summaries (OSs), τα οποία ταξινομούνται με εφαρμογή IR-Style τεχνικών.

Πιο συγκεκριμένα, για την ταξινόμηση των OSs εφαρμόζεται η μέθοδος stemming και η μέθοδος TF – IDF χρησιμοποιώντας τη δεύτερη λέξη – κλειδί που καθορίζει τη θεματική ενότητα. Ωστόσο, αυτό που έρχεται να παίζει καθοριστικό ρόλο στην ταξινόμηση των αποτελεσμάτων είναι η έννοια του *Affinity*, η οποία συμβάλλει έτσι ώστε η ταξινόμηση των OSs να βασίζεται στο γεγονός ότι οι πληροφορίες γύρω από τη θεματική ενότητα σχετίζονται όσο το δυνατόν περισσότερο με το αντικείμενο για το οποίο ενδιαφέρεται ο χρήστης.

Περιεχόμενα

Κεφάλαιο 1	Εισαγωγή	1
1.1	Βάσεις δεδομένων και η χρήση τους σήμερα	1
1.2	Μηχανές Αναζήτησης	2
1.3	Στόχοι διπλωματικής εργασίας	4
1.4	Οργάνωση τόμου	5
Κεφάλαιο 2	Σχετικές εργασίες	7
2.1	Εισαγωγή	7
2.2	Αναζήτηση χρησιμοποιώντας λέξεις – κλειδιά (Keyword Search)	8
2.3	Ανάκτηση Πληροφορίας (Information Retrieval – IR)	9
2.4	IR – Style τεχνικές	10
2.4.1	Μέθοδος stemming	10
2.4.2	Μέθοδος TF – IDF	11
2.5	Proximity Search	15
2.6	Object Summary (OS)	16
2.6.1	Τι εννοούμε με τον όρο Object Summary;	16
2.6.2	Data Subject Schema Graph (G ^{DS})	17
2.6.3	Σχηματισμός OS δέντρου	19
2.6.4	Affinity	20
2.7	Συμπεράσματα	25
Κεφάλαιο 3	Προτεινόμενη μεθοδολογία ταξινόμησης OSs	26
3.1	Εισαγωγή	26
3.2	Σύντομη περιγραφή της προτεινόμενης μεθοδολογίας	27
3.3	Είσοδος από τον χρήστη	28
3.3.1	Πρώτη λέξη – κλειδί : Καθορισμός DS	28
3.3.2	Δεύτερη λέξη – κλειδί : Καθορισμός θεματικής ενότητας	29
3.4	Μέθοδος ταξινόμησης των αποτελεσμάτων	30
3.4.1	Εφαρμογή μεθόδου stemming	30

3.4.2 Συνδυασμός μεθόδου TF –IDF και Affinity	31
3.5 Αρχιτεκτονική συστήματος προτεινόμενης μεθοδολογίας	38
3.6 Φάση προεπεξεργασίας	40
3.6.1 Αρχείο προεπιλεγμένων λέξεων	40
3.6.2 Εύρεση συνόλου πλειάδων T^{OSs}	42
3.6.3 Δημιουργία OSs και υπολογισμός τιμών $Af - tf$	42
3.6.4 Δημιουργία ευρετηρίου	43
3.7 Φάση επεξεργασίας	44
3.7.1 Εύρεση συνόλου πλειάδων T^{k1}	44
3.7.2 Εύρεση τιμών $Af - tf$	45
3.7.3 Ταξινόμηση τιμών $Af - tf$	47
3.7.4 Δημιουργία και παρουσίαση των OSs	47
3.8 Συμπεράσματα	47
Κεφάλαιο 4 Υλοποίηση συστήματος προτεινόμενης μεθοδολογίας	49
4.1 Εισαγωγή	49
4.2 Επιλογή γλώσσας προγραμματισμού και εργαλείου ανάπτυξης	49
4.3 Βάση δεδομένων Northwind	51
4.4 Επιλογή συστήματος διαχείρισης βάσεων δεδομένων	53
4.5 Λεπτομέρειες υλοποίησης	53
4.5.1 Υλοποίηση φάσης προεπεξεργασίας	53
4.5.2 Υλοποίηση φάσης επεξεργασίας	55
4.6 Πρωτότυπο συστήματος	57
4.7 Συμπεράσματα	61
Κεφάλαιο 5 Αξιολόγηση	63
5.1 Εισαγωγή	63
5.2 Αξιολόγηση αποτελεσματικότητας συστήματος	63
5.2.1 Μεθοδολογία αξιολόγησης	64
5.2.2 Αποτελέσματα αξιολόγησης	65
5.3 Αξιολόγηση απόδοσης συστήματος	71
5.3.1 Μεθοδολογία αξιολόγησης	71
5.3.2 Αποτελέσματα αξιολόγησης	72

5.4 Συμπεράσματα	74
Κεφάλαιο 6 Συμπεράσματα	75
6.1 Σύνοψη	75
6.2 Τελικά συμπεράσματα και μελλοντικές επεκτάσεις	76
 Βιβλιογραφία	 77
 Παράρτημα Α	 A - 1
 Παράρτημα Β	 B - 1
 Παράρτημα Γ	 Γ - 1

Κεφάλαιο 1

Εισαγωγή

1.1 Βάσεις δεδομένων και η χρήση τους σήμερα	1
1.2 Μηχανές αναζήτησης	2
1.3 Στόχοι διπλωματικής εργασίας	4
1.4 Οργάνωση τόμου	5

1.1 Βάσεις δεδομένων και η χρήση τους σήμερα

Ζούμε αναμφίβολα στην «Εποχή της Πληροφορίας», όπου κύριο χαρακτηριστικό είναι η συνεχής και ραγδαία αύξηση των δεδομένων και πληροφοριών. Το γεγονός αυτό, καθιστά επιτακτική την ανάγκη δημιουργίας και διαχείρισης βάσεων δεδομένων, οι οποίες χρησιμοποιούνται πλέον σε αναρίθμητους τομείς της καθημερινής μας ζωής. Τι ακριβώς εννοούμε, όμως, με τον όρο βάση δεδομένων;

Στον κόσμο των υπολογιστών, όταν λέμε βάση δεδομένων αναφερόμαστε σε μία συλλογή πληροφοριών που σχετίζονται μεταξύ τους και που βρίσκονται αποθηκευμένες σε ένα υπολογιστικό σύστημα με οργανωμένο τρόπο. Πέραν όμως της αποθήκευσης, μία βάση δεδομένων προσφέρει ένα πλήρη έλεγχο των δεδομένων επιτρέποντας την ανάκτηση, την ταξινόμηση, την ανάλυση και τη σύνοψή τους καθώς επίσης την αποτελεσματική και ακριβή εισαγωγή νέων δεδομένων σε αυτή.

Οι βάσεις δεδομένων διακρίνονται σε διάφορες κατηγορίες. Σύμφωνα με το περιεχόμενό τους, υπάρχουν οι βιβλιογραφικές βάσεις δεδομένων οι οποίες περιέχουν βιβλιογραφικές αναφορές και περιλήψεις επιστημονικών έργων, οι βάσεις δεδομένων πλήρους κειμένου οι οποίες προσφέρουν τη δυνατότητα πρόσβασης στο πλήρες κείμενο διαφόρων άρθρων, οι βάσεις αριθμητικών δεδομένων, εικόνων, προγραμμάτων κτλ. Σύμφωνα με τη μέθοδο οργάνωσης που ακολουθούν,

διακρίνονται επίσης σε δικτυακές βάσεις δεδομένων και σε σχεσιακές βάσεις δεδομένων. Οι δικτυακές βάσεις δεδομένων είναι ιδιαίτερα πολύπλοκες κι όταν

αναπαρασταθούν γραφικά παρομοιάζονται ως ένα δίκτυο. Οι σχεσιακές βάσεις δεδομένων στις οποίες και θα επικεντρωθούμε, αναπαριστούν τα δεδομένα με χρήση δισδιάστατων πινάκων οι οποίοι αποτελούνται από ένα σύνολο πεδίων (στήλες) και ένα σύνολο πλειάδων – εγγραφών (γραμμές). Η ύπαρξη μίας κοινής τιμής σε πεδία δύο διαφορετικών πινάκων καθορίζει μία σχέση μεταξύ τους. Οι βάσεις αυτές έχουν το πλεονέκτημα ότι είναι λογικά κατανοητές και πολύ ευέλικτες και δεκτικές σε αλλαγές.

Όπως έχουμε προαναφέρει, τη σημερινή εποχή οι βάσεις δεδομένων χρησιμοποιούνται σε πολλούς και διάφορους τομείς. Αν και τις πλείστες φορές δεν μπορούμε να το αντιληφθούμε, πολλές από τις απλές, καθημερινές μας δραστηριότητες σχετίζονται άμεσα με τη χρήση βάσεων δεδομένων. Όταν, για παράδειγμα, αγοράζουμε προϊόντα από μία υπεραγορά, η τιμή του κάθε προϊόντος ανακτάται από μία βάση δεδομένων σύμφωνα με τον γραμμωτό κώδικα (barcode). Ακόμα κι όταν εκτελούμε ένα απλό τηλεφώνημα, όσο περίεργο κι αν ακούγεται, γίνεται και πάλι χρήση μίας βάσης δεδομένων για την εύρεση πληροφοριών δρομολόγησης της κλήσης. Παράλληλα με αυτά, οι βάσεις δεδομένων χρησιμοποιούνται επίσης σε προσωπικούς υπολογιστές για διαχείριση των δεδομένων του χρήστη, σε δίκτυα επιχειρήσεων για αποθήκευση και διαχείριση δεδομένων που αφορούν μία συγκεκριμένη επιχείρηση, σε βιβλιοθήκες ώστε να διατηρούνται οι απαραίτητες πληροφορίες για όλα τα διαθέσιμα βιβλία ενώ είναι ιδιαίτερα δημοφιλείς στο διαδίκτυο. Βρίσκονται κυριολεκτικά παντού.

1.2 Μηχανές αναζήτησης

Ωστόσο, για να είναι πραγματικά χρήσιμες οι βάσεις δεδομένων που δημιουργούνται, είναι απαραίτητο να αναπτυχθούν κατάλληλα προγράμματα τα οποία θα μας δίνουν τη δυνατότητα ανάκτησης των πληροφοριών αποτελεσματικά και αποδοτικά. Τέτοια προγράμματα καλούνται «μηχανές αναζήτησης».

Οι εφαρμογές αυτές επιτρέπουν την αναζήτηση και ανάκτηση πληροφοριών από βάσεις δεδομένων χρησιμοποιώντας ερωτήσεις αναζήτησης λέξεων – κλειδιών (keyword search queries). Πιο συγκεκριμένα, ο χρήστης μίας μηχανής αναζήτησης εισάγει ένα σύνολο από λέξεις – κλειδιά για το αντικείμενο που τον ενδιαφέρει και του

επιστρέφονται από τη βάση δεδομένων πληροφορίες που σχετίζονται άμεσα με τις συγκεκριμένες λέξεις – κλειδιά.

Οι μηχανές αναζήτησης σημείωσαν ιδιαίτερη επιτυχία στο χώρο του διαδικτύου. Το γνωστό σε όλους μας πλέον Google, αποτελεί μία από τις δημοφιλέστερες, εάν όχι τη δημοφιλέστερη μηχανή αναζήτησης που υπάρχει σήμερα. Κάθε αναζήτηση που εκτελείται στο Google, επιστρέφει μία λίστα από συνδέσμους (links) σε ιστοσελίδες οι οποίες προσεγγίζουν όσο το δυνατόν περισσότερο τις λέξεις- κλειδιά που έχουν εισαχθεί. Αξιοσημείωτο είναι το γεγονός ότι οι ιστοσελίδες που επιστρέφονται σε κάθε αναζήτηση παρουσιάζονται στον χρήστη ταξινομημένες, ανάλογα με το πόσο σχετικές είναι με το αντικείμενο που ψάχνει.

Στο σχήμα 1.1 δίνεται ένα μέρος των αποτελεσμάτων στην περίπτωση που έχουν εισαχθεί στη μηχανή αναζήτησης Google οι δύο λέξεις – κλειδιά “Christodoulou algorithms”. Παρουσιάζεται, ουσιαστικά, ένα σύνολο από συνδέσμους σε ιστοσελίδες που περιέχουν τις συγκεκριμένες λέξεις - κλειδιά. Παρατηρούμε ότι τα αποτελέσματα αυτά είναι ταξινομημένα ξεκινώντας από τον πιο σημαντικό, τον πιο δυνατό σύνδεσμο που μπορεί να είναι σε θέση να προσφέρει όλες τις απαραίτητες πληροφορίες στον χρήστη και προχωρώντας σε λιγότερο σημαντικούς συνδέσμους. Επίσης, βλέπουμε πως μαζί με τον κάθε σύνδεσμο δίνεται μία μικρή περίληψη για το περιεχόμενο της ιστοσελίδας. Το γεγονός αυτό κατευθύνει τον χρήστη στο να επιλέξει την πιο κατάλληλη ιστοσελίδα γι’ αυτόν.

Homepage: Giorgos Christodoulou (Max-Planck-Institut für Informatik)

20 Jul 2009 ... Scheduling selfish tasks: about the performance of truthful *algorithms*. George **Christodoulou**, Laurent Gourves, Fanny Pascual. ...
www.mpi-inf.mpg.de/~gchristo/ -

Giorgos Christodoulou

Scheduling selfish tasks: about the performance of truthful *algorithms*. [PDF]
George **Christodoulou**, Laurent Gourves, Fanny Pascual. ...
www.csc.liv.ac.uk/~gchristo/ -

Βιογραφικό Σημείωμα - TEI of Chalkida | Dept. of Electrical ...

[12] S. D. Voliotis and M. A. **Christodoulou**, “Parallel *algorithms* for Path tracking of Two Robot Arms via a Non-inverting Method”, Journal of Intelligent ...
www.ee.teihal.gr/~svoliotis.asp -

Anthony Christodoulou - Football Algorithms

Anthony**Christodoulou**.com: The intertubes home of Anthony **Christodoulou**, a Ph.D. student in Electrical and Computer Engineering at the University of Illinois ...
anthonychristodoulou.com/index.php?... -

Σχήμα 1.1. Αποτελέσμα από τη μηχανή αναζήτησης Google δίνοντας τις λέξεις – κλειδιά “Christodoulou algorithms”

Τα δεδομένα που βρίσκονται αποθηκευμένα σε μία βάση δεδομένων μας είναι εντελώς αχρείαστα εάν δεν μπορούμε να τα εκμεταλλευτούμε σωστά. Οι μηχανές αναζήτησης προσπαθούν να επιλύσουν το πρόβλημα αυτό, μετατρέποντας τα δεδομένα σε χρήσιμες πληροφορίες για τον κάθε χρήστη.

Ωστόσο, η ακατάπαυστη αύξηση των δεδομένων καθιστά τη διαδικασία ανάκτησης ολοένα πιο δύσκολη και πολύπλοκη. Είναι σημαντικό, λοιπόν, οι μηχανές αυτές να αναπτυχθούν με τέτοιο τρόπο ώστε να υποστηρίζουν διάφορες τεχνικές, οι οποίες θα μπορούν να συμβάλουν στη γρήγορη, αποδοτική και αποτελεσματική ανάκτηση πληροφοριών από τις βάσεις δεδομένων.

1.3 Στόχοι διπλωματικής εργασίας

Η διπλωματική εργασία αυτή, στοχεύει στην ανάπτυξη μίας καινούργιας μεθοδολογίας για την ταξινόμηση των αποτελεσμάτων που λαμβάνονται από την εκτέλεση αναζητήσεων σε σχεσιακές βάσεις δεδομένων

Σύμφωνα με τη μέθοδο αυτή, θα υλοποιηθεί ένα σύστημα, μία μηχανή αναζήτησης, η οποία θα δέχεται από τον χρήστη δύο λέξεις – κλειδιά. Η πρώτη λέξη – κλειδί θα προσδιορίζει το αντικείμενο για το οποίο επιθυμεί να πάρει πληροφορίες ο χρήστης, ενώ η δεύτερη λέξη - κλειδί θα καθορίζει τη θεματική ενότητα που τον ενδιαφέρει περισσότερο. Η καινοτομία βρίσκεται στην εφαρμογή διάφορων τεχνικών, που θα αναλύσουμε σε επόμενα κεφάλαια, με σκοπό την ταξινόμηση των αποτελεσμάτων που θα ληφθούν για το συγκεκριμένο αντικείμενο χρησιμοποιώντας τη δεύτερη λέξη – κλειδί.

Ένας ακόμη στόχος που έχει τεθεί είναι η ελαχιστοποίηση του χρόνου απόκρισης του συστήματος. Είναι πολύ σημαντικό το σύστημα που θα υλοποιηθεί να λειτουργεί όσο το δυνατόν πιο γρήγορα και αποδοτικά. Η αναζήτηση που θα εκτελεί κάθε φορά ο χρήστης θα πρέπει να επιστρέφει άμεσα και χωρίς ιδιαίτερες καθυστερήσεις όλες τις απαραίτητες πληροφορίες. Ένα σύστημα το οποίο είναι αργό και χρειάζεται αρκετό

χρονικό διάστημα για να δώσει αποτελέσματα σίγουρα δεν είναι κάτι το οποίο επιθυμούμε.

Η υλοποίηση του συστήματος θα γίνει στα πλαίσια της βάσης δεδομένων Northwind. Η βάση δεδομένων Northwind αφορά εμπορικές συναλλαγές και περιλαμβάνει πολλές σχέσεις και πεδία. Το γεγονός αυτό θα συμβάλει τόσο στη δημιουργία ποιοτικών αποτελεσμάτων που θα έχουν νόημα για τον χρήστη όσο και στη διαδικασία ταξινόμησής τους.

Αφού υλοποιηθεί το σύστημα τότε θα ακολουθήσει η διαδικασία αξιολόγησής του , όπου θα καταγραφεί σε ποιο βαθμό έχει λειτουργήσει αποδοτικά και αποτελεσματικά καθώς επίσης σε ποιά σημεία υστερεί και παρουσιάζεται ανάγκη βελτίωσης.

1.4 Οργάνωση τόμου

Το παρόν κεφάλαιο είναι μια σύντομη εισαγωγή στη διπλωματική.

Το κεφάλαιο 2 παρουσιάζει σχετικές εργασίες οι οποίες θα χρησιμοποιηθούν ως βάση στην ανάπτυξη της προτεινόμενης μεθοδολογίας. Συγκεκριμένα, η ενότητα 2.2 αναφέρεται στη μέθοδο αναζήτησης με χρήση λέξεων - κλειδιών (Keyword Search) που εφαρμόζεται στις πλείστες μηχανές αναζήτησης σήμερα ενώ η ενότητα 2.3 δίνει μία σύντομη περιγραφή για την επιστήμη της Ανάκτησης Πληροφορίας (Information Retrieval). Στην ενότητα 2.4 παρουσιάζονται οι IR-Style τεχνικές που πρόκειται να χρησιμοποιήσουμε για την ανάπτυξη της νέας μεθοδολογίας ενώ στην ενότητα 2.5 γίνεται μία αναφορά στη μέθοδο αναζήτησης Proximity Search. Στην ενότητα 2.6 δίνεται μία ουσιαστική περιγραφή για την έννοια του OS και τα διάφορα επιμέρους στοιχεία του.

Το κεφάλαιο 3 αποτελεί το βασικότερο κεφάλαιο της διπλωματικής εργασίας καθώς περιγράφει αναλυτικά τη μεθοδολογία που θα αναπτύξουμε. Η ενότητα 3.2 δίνει μία σύντομη περιγραφή της προτεινόμενης μεθοδολογίας ενώ η ενότητα 3.3 αναφέρεται στις εισόδους που θα πρέπει να δώσει ο χρήστης στο σύστημα επεξηγώντας το τι καθορίζει η κάθε μία από αυτές. Στην ενότητα 3.4 δίνεται μία εκτενής ανάλυση της μεθόδου ταξινόμησης των αποτελεσμάτων που θα εφαρμόσουμε ενώ στην ενότητα 3.5 παρουσιάζεται η βασική αρχιτεκτονική του συστήματος της προτεινόμενης

μεθοδολογίας. Στις ενότητες 3.6 και 3.7 επεξηγούνται οι διάφορες διεργασίες του συστήματος σύμφωνα με την αρχιτεκτονική που έχει σχεδιαστεί.

Το κεφάλαιο 4 σχετίζεται με τη φάση της υλοποίησης. Ειδικότερα, οι ενότητες 4.2, 4.3 και 4.4 περιγράφουν τα χαρακτηριστικά της συγκεκριμένης υλοποίησης, όπως είναι η γλώσσα προγραμματισμού, το εργαλείο ανάπτυξης του συστήματος, η βάση δεδομένων που θα χρησιμοποιηθεί, όπως επίσης και το σύστημα διαχείρισης βάσεων δεδομένων. Η ενότητα 4.5 παρουσιάζει τις λεπτομέρειες της υλοποίησης, δηλαδή αντιστοιχίζει τις διεργασίες που περιέχονται στην αρχιτεκτονική του συστήματος σε προγράμματα / συναρτήσεις. Η ενότητα 4.6 παρουσιάζει το πρωτότυπο του συστήματος.

Το κεφάλαιο 5 πραγματεύεται την αξιολόγηση του συστήματος. Στην ενότητα 5.2 παρουσιάζεται η αξιολόγηση της αποτελεσματικότητας του συστήματος με τα αντίστοιχα αποτελέσματα που έχουν εξαχθεί ενώ η ενότητα 5.3 παρουσιάζει την αξιολόγηση του συστήματος σε θέμα απόδοσης.

Το κεφάλαιο 6 είναι ο επίλογος της διπλωματικής. Στην ενότητα 6.1 δίνεται μία σύνοψη για τη διπλωματική εργασία ενώ η ενότητα 6.2 αναφέρεται στα τελικά συμπεράσματα και σε μελλοντικές επεκτάσεις.

Ο τόμος αυτός ολοκληρώνεται με την παρουσίαση της βιβλιογραφίας.

Κεφάλαιο 2

Σχετικές εργασίες

2.1 Εισαγωγή	7
2.2 Αναζήτηση χρησιμοποιώντας λέξεις – κλειδιά (Keyword Search)	8
2.3 Ανάκτηση πληροφορίας (Information Retrieval – IR)	9
2.4 IR- Style τεχνικές	10
2.4.1 Μέθοδος stemming	10
2.4.2 Μέθοδος TF-IDF	11
2.5 Proximity Search	15
2.6 Object Summary (OS)	16
2.6.1 Τι εννοούμε με τον όρο Object Summary ;	16
2.6.2 Data Subject Schema Graph (G^{DS})	17
2.6.3 Σχηματισμός OS δέντρου	19
2.6.4 Affinity	20
2.7 Συμπεράσματα	25

2.1 Εισαγωγή

Στο κεφάλαιο αυτό θα αναφερθούμε σε διάφορες σχετικές εργασίες τις οποίες θα χρησιμοποιήσουμε ως βάση στην ανάπτυξη της καινούργιας μεθοδολογίας. Αρχικά θα μιλήσουμε για την αναζήτηση με τη χρήση λέξεων – κλειδίων , το λεγόμενο Keyword Search, δίνοντας ιδιαίτερη έμφαση στην αναζήτηση σε σχεσιακές βάσεις δεδομένων, ενώ στη συνέχεια θα αναφερθούμε στην επιστήμη της Ανάκτησης Πληροφορίας (Information Retrieval – IR) και σε IR – Style τεχνικές που έχουν αναπτυχθεί, οι οποίες πρόκειται να παίξουν καθοριστικό ρόλο στην προτεινόμενη μεθοδολογία. Ακολουθώς, θα δούμε τη μέθοδο αναζήτησης Proximity Search και θα επικεντρωθούμε σε μία καινούργια έννοια, τα Object Summaries, η οποία έχει εισαχθεί από τον Δρ. Γεώργιο Φάκα [1] αναλύοντας κάποια επιμέρους στοιχεία της.

2.2 Αναζήτηση χρησιμοποιώντας λέξεις – κλειδιά (Keyword Search)

Το Keyword Search αποτελεί την πιο διαδεδομένη μέθοδο αναζήτησης που υπάρχει σήμερα τόσο στο διαδίκτυο (Web Keyword Search), όπως στην περίπτωση της μηχανής αναζήτησης Google, όσο και σε σχεσιακές βάσεις δεδομένων (Relational Keyword Search) [5,6,7].

Το γεγονός ότι η συγκεκριμένη μέθοδος έχει γίνει άμεσα αποδεκτή, οφείλεται στο ότι ο χρήστης δεν χρειάζεται να γνωρίζει οποιαδήποτε γλώσσα σχετική με Queries όπως είναι η SQL για να πάρει τις πληροφορίες που επιθυμεί, ούτε είναι απαραίτητο να γνωρίζει τη δομή των δεδομένων που υπάρχουν στη βάση. Το μόνο που χρειάζεται να κάνει είναι να εισαγάγει στη μηχανή αναζήτησης κάποιες λέξεις - κλειδιά για το αντικείμενο που τον ενδιαφέρει και του επιστρέφονται, σε περίπτωση Web Keyword Search (W- KWS) για παράδειγμα, όλες οι ιστοσελίδες που περιέχουν μία ή περισσότερες από τις λέξεις αυτές.

Η εφαρμογή της μεθόδου σε σχεσιακές βάσεις δεδομένων (R-KWS) διακρίνεται σε δύο περιπτώσεις : (1) το Query αποτελείται από μία λέξη - κλειδί και (2) το Query αποτελείται από περισσότερες από μία λέξεις – κλειδιά.

Στην πρώτη περίπτωση, όπου το Query που δίνει ο χρήστης αποτελείται από μία λέξη – κλειδί, το αποτέλεσμα της αναζήτησης θα είναι όλες οι ανεξάρτητες πλειάδες που περιέχουν τη συγκεκριμένη λέξη - κλειδί σε κάποιο από τα πεδία τους. Για παράδειγμα, εάν υποθέσουμε ότι η μηχανή αναζήτησης διαχειρίζεται τη βάση δεδομένων Northwind και έχει εισαχθεί σαν λέξη – κλειδί το όνομα “Maria”, τότε θα επιστραφούν όλες οι πλειάδες που περιέχουν το όνομα “Maria” σε κάποιο από τα πεδία τους, όπως φαίνεται πιο κάτω.

CustomerID	CompanyName	ContactName	ContactTitle	...
ALFKI	Alfreds Futterkiste	Maria Anders	Sales Representative	...
FOLKO	Folk och fä HB	Maria Larsson	Owner	...

Σχήμα 2.1 Αποτέλεσμα αναζήτησης στη σχεσιακή βάση δεδομένων Northwind με βάση τη λέξη – κλειδί “Maria”

Υπάρχουν, ωστόσο, τεχνικές για R-KWS που επιτρέπουν στους χρήστες να κάνουν αναζήτηση συσχετίζοντας λέξεις – κλειδιά μεταξύ τους [5,6,7].

Στη δεύτερη, λοιπόν, περίπτωση R-KwS όπου ο χρήστης δίνει ως είσοδο περισσότερες από μία λέξεις - κλειδιά, το αποτέλεσμα της αναζήτησης θα είναι δέντρα από πλειάδες που συσχετίζουν τις συγκεκριμένες λέξεις - κλειδιά. Για να γίνει πιο κατανοητό, έστω ότι ο χρήστης έχει δώσει, στα πλαίσια και πάλι της βάσης δεδομένων Northwind, τις λέξεις - κλειδιά “Maria” και “Leverling”. Η αναζήτηση αυτή θα επιστρέψει πληροφορίες που αφορούν και τους δύο όρους, όπως για παράδειγμα, Orders του Customer “Maria” τις οποίες έχει αναλάβει ο Employee “Leverling”.

Συνεπώς, μπορούμε να πούμε ότι η μέθοδος R-KwS, σε αντίθεση με τη W-KwS όπου τα αποτελέσματα είναι ιστοσελίδες, δεν είναι ιδιαίτερα αποτελεσματική όταν θέλουμε να πάρουμε μία περίληψη όλων των στοιχείων που αφορούν ένα συγκεκριμένο αντικείμενο. Εάν πάρουμε το πρώτο παράδειγμα όπου έχει εισαχθεί στη μηχανή αναζήτησης μόνο η λέξη - κλειδί “Maria” (Σχήμα 2.1), παρατηρούμε ότι το αποτέλεσμα της αναζήτησης είναι απλά οι πλειάδες που περιέχουν τη συγκεκριμένη λέξη - κλειδί σε κάποιο από τα πεδία τους, χωρίς όμως να μας δίνονται περαιτέρω πληροφορίες όπως για τις παραγγελίες που έχει κάνει ή για τον υπάλληλο που την έχει εξυπηρετήσει κτλ. Παρ’όλα αυτά, η μέθοδος R-KwS φαίνεται να είναι ιδιαίτερα χρήσιμη όταν θέλουμε να συνδυάσουμε λέξεις – κλειδιά μεταξύ τους.

2.3 Ανάκτηση πληροφορίας (Information Retrieval – IR)

Με τον όρο Ανάκτηση Πληροφορίας (Information Retrieval – IR) αναφερόμαστε σε μία επιστήμη η οποία μελετά προβλήματα που σχετίζονται με την αναπαράσταση, αποθήκευση, οργάνωση και προσπέλαση πληροφοριακών αντικειμένων (π.χ έγγραφα, εικόνες, ήχοι κτλ.). Η μέθοδος αυτή εφαρμόστηκε αρχικά πάνω σε μικρές συλλογές εγγράφων και στη συνέχεια επεκτάθηκε σε πολυμεσικά αντικείμενα, στο διαδίκτυο αλλά και σε σχεσιακές βάσεις δεδομένων. [9]

Ο χρήστης ενός συστήματος Ανάκτησης Πληροφορίας θα πρέπει αρχικά να διατυπώσει την πληροφοριακή του ανάγκη υπό τη μορφή ενός Query, όπως συμβαίνει σε όλες τις μηχανές αναζήτησης. Ακολούθως, το σύστημα θα εντοπίσει όλα τα αντικείμενα πληροφορίας που σχετίζονται με το Query που έχει εισαχθεί και τα

αποτελέσματα που θα εξαχθούν θα ταξινομηθούν με βάση τον βαθμό συσχέτισης με το Query. [9]

Το σημαντικό στοιχείο που χαρακτηρίζει την επιστήμη της Ανάκτησης Πληροφορίας, είναι πως δεν περιορίζεται στην εύρεση εγγράφων, για παράδειγμα, τα οποία περιέχουν απλά μερικούς από τους όρους του Query που έχει εισαχθεί, αλλά αναγνωρίζει το γεγονός ότι ο χρήστης του συστήματος επιθυμεί να ανακτήσει πληροφορίες γύρω από ένα συγκεκριμένο θέμα. Για το λόγο αυτό, προσπαθεί να εντοπίσει συσχετίσεις μεταξύ των όρων του Query και των όρων των εγγράφων, έτσι ώστε να μεταβεί από τους όρους σε θεματικές περιοχές. Για να το επιτύχει αυτό, εφαρμόζει διάφορες μεθόδους ανάλυσης τόσο των όρων των εγγράφων όσο και των όρων του Query, όπως απαλοιφή συχνών λέξεων (stopwords), απομάκρυνση επιθημάτων και προθεμάτων από τις λέξεις (stemming) κτλ. [9]

Αξιοσημείωτο είναι το γεγονός, ότι τα συστήματα Ανάκτησης Πληροφορίας σπάνια ψάχνουν απευθείας στα αντικείμενα πληροφορίας, αλλά τις πιο πολλές φορές εφαρμόζουν μία διαδικασία ευρετηρίασης (indexing) η οποία μειώνει σημαντικά τον χρόνο απόκρισης του συστήματος. [9]

2.4 IR-Style τεχνικές

Η επιστήμη της Ανάκτησης Πληροφορίας εφαρμόζει διάφορες τεχνικές για την εξαγωγή και ταξινόμηση των αποτελεσμάτων. Πιο κάτω αναλύονται δύο βασικές IR-Style τεχνικές, η μέθοδος stemming και η μέθοδος TF – IDF.

2.4.1 Μέθοδος stemming

Η μέθοδος stemming είναι πολύ διαδεδομένη σε Query συστήματα γι' αυτό και έχει εφαρμοστεί σε πολλές μηχανές αναζήτησης. Το γεγονός αυτό οφείλεται στο ότι εάν ένας χρήστης ενδιαφέρεται, για παράδειγμα, να πάρει πληροφορίες σχετικά με “computers” και εισάγει στη μηχανή αναζήτησης αυτή τη λέξη – κλειδί , τότε είναι πολύ πιθανόν να ενδιαφέρεται και για έγγραφα τα οποία περιέχουν τη λέξη - κλειδί “computer” χωρίς το γράμμα ‘s’.

Βασικός στόχος είναι, ουσιαστικά, η βελτίωση της Ανάκτησης Πληροφορίας με αυτόματο χειρισμό των καταλήξεων των λέξεων και τη μείωση τους στις ρίζες τους. Η μέθοδος stemming, θα πάρει τη λέξη – κλειδί που έχει εισαγάγει ο χρήστης στη μηχανή αναζήτησης και θα τη μειώσει στη ρίζα της αφαιρώντας οποιοδήποτε επίθημα ή πρόθεμα από αυτή. Ακολούθως, η ρίζα της λέξης - κλειδί θα συγκριθεί με τις λέξεις που υπάρχουν σε κάθε έγγραφο του συνόλου στο οποίο εκτελείται η αναζήτηση και θα επιστραφούν έγγραφα που περιέχουν λέξεις με τη ρίζα αυτή [12,15].

Για παράδειγμα, έστω ότι ο χρήστης έχει εισαγάγει τη λέξη – κλειδί “viewer” στη μηχανή αναζήτησης. Τότε, η λέξη αυτή θα μειωθεί στη ρίζα της, δηλ. το “view”, με αφαίρεση του επιθέματος –er και θα επιστραφούν έγγραφα που περιέχουν λέξεις όπως view, viewer, preview, review κτλ. Θα μπορούσαμε να πούμε ότι η μέθοδος stemming καθορίζει στην ουσία μία ομάδα συνωνύμων για μία συγκεκριμένη λέξη.

Μέχρι σήμερα έχουν αναπτυχθεί πολλοί και διάφοροι stemming αλγόριθμοι ή αλλιώς stemmers. Ο πρώτος αλγόριθμος που εντοπίζουμε είναι ο αλγόριθμος Lovins [11] ο οποίος αναπτύχθηκε το 1968. Υπάρχουν, όμως, κι άλλοι αλγόριθμοι όπως ο αλγόριθμος Nice [12], ο οποίος έχει υλοποιηθεί στα πλαίσια του συστήματος IRIS για ανάκτηση πληροφορίας στο πανεπιστήμιο του North Carolina και αποτελεί ένα συνδυασμό τεσσάρων διαφορετικών stemmers. Ωστόσο, ο γνωστότερος αλγόριθμος ο οποίος εφαρμόζεται πιο συχνά, είναι ο αλγόριθμος του Porter [10]. Πολλές εφαρμογές του αλγορίθμου αυτού εκδόθηκαν και διανεμήθηκαν ελεύθερα αλλά περιείχαν αρκετά λάθη. Για το λόγο αυτό, ο Martin Porter εξέδωσε επίσημα μία εφαρμογή ελεύθερου λογισμικού του αλγορίθμου το 2000 [10], ενώ αργότερα επέκτεινε την εργασία αυτή με τη δημιουργία μίας γλώσσας για stemming αλγορίθμους, το Snowball, και εφάρμοσε ένα βελτιωμένο stemmer για την αγγλική γλώσσα.

2.4.2 Μέθοδος TF-IDF

Ο όρος TF-IDF (Term Frequency – Inverse Document Frequency) αναφέρεται σε μία IR-style τεχνική η οποία καθορίζει πόσο σημαντική είναι μία λέξη – κλειδί σε ένα έγγραφο δίνοντας σε αυτή ένα συγκεκριμένο βάρος [8,15]. Η τεχνική αυτή έχει

εφαρμοστεί σε διάφορες μηχανές αναζήτησης για την εύρεση του βαθμού συσχέτισης ενός εγγράφου με τις λέξεις - κλειδιά που έχουν εισαχθεί.

Για κάθε λέξη – κλειδί του Query που εισάγει ο χρήστης στη μηχανή αναζήτησης, υπολογίζεται το βάρος *tf-idf* που της αναλογεί για το κάθε έγγραφο του συνόλου ξεχωριστά. Ακολουθώς, συναθροίζονται τα βάρη όλων των λέξεων – κλειδιών του Query για το κάθε έγγραφο και τα αποτελέσματα χρησιμοποιούνται για την ταξινόμηση του συνόλου. Πρώτο στην ταξινόμηση θα βρίσκεται το έγγραφο για το οποίο οι λέξεις – κλειδιά έχουν το μεγαλύτερο άθροισμα *tf-idf*. [15]

Η σημασία της λέξης - κλειδί αυξάνει ανάλογα με το πόσες φορές εμφανίζεται στο έγγραφο. Σίγουρα, όμως, υπάρχουν κάποιες λέξεις οι οποίες είναι πολύ συνηθισμένες και πιθανόν να εμφανίζονται σε πολλά έγγραφα, όπως είναι οι προθέσεις και τα άρθρα. Αυτό έχει ως αποτέλεσμα να δίνεται στο τέλος έμφαση σε έγγραφα που δεν είναι και τόσο σχετικά με αυτό που πραγματικά αναζητά ο χρήστης. [15]

Για παράδειγμα, έστω ότι ο χρήστης έχει εισαγάγει στη μηχανή αναζήτησης τις λέξεις – κλειδιά “the Porter algorithm”. Τότε, λόγω του ότι η λέξη – κλειδί “the” είναι πολύ συνηθισμένη και εμφανίζεται σίγουρα σε μεγάλο αριθμό εγγράφων, θα δοθεί λανθασμένα μεγαλύτερη έμφαση σε έγγραφα τα οποία περιέχουν πολλές φορές τη λέξη – κλειδί “the” και μικρότερη έμφαση σε έγγραφα που περιέχουν τις λέξεις – κλειδιά “Porter” και “algorithm”, οι οποίες αν και εμφανίζονται λιγότερο συχνά εμπερικλείουν το πραγματικό νόημα της αναζήτησης.

Για την επίλυση του προβλήματος αυτού, χρησιμοποιείται ο παράγοντας Inverse Document Frequency (IDF) . Ο παράγοντας αυτός δίνει μικρό βάρος σε λέξεις - κλειδιά που εμφανίζονται συχνά και μεγάλο βάρος σε λέξεις – κλειδιά που εμφανίζονται σπάνια ώστε να γίνει σωστά η ταξινόμηση των αποτελεσμάτων. [15]

Το term frequency (*tf*) αντιστοιχεί στον αριθμό των εμφανίσεων μίας λέξης – κλειδί σε ένα έγγραφο και η τιμή του κανονικοποιείται με τη χρήση της πιο κάτω εξίσωσης :

$$(1) \quad tf = \frac{n_i}{\sum_k n_k}$$

, όπου n_i ο αριθμός των εμφανίσεων της λέξης - κλειδί i στο συγκεκριμένο έγγραφο και ο παρονομαστής το άθροισμα του αριθμού των εμφανίσεων κάθε λέξης k (n_k) που περιέχεται στο έγγραφο. Ουσιαστικά, ο παρονομαστής αντιστοιχεί στο συνολικό αριθμό των λέξεων του εγγράφου. Η κανονικοποίηση διαιρώντας με το άθροισμα $\sum_k n_k$ κρίνεται απαραίτητη εάν τα έγγραφα έχουν μεγάλη διαφορά ως προς το μέγεθός τους. Με τον τρόπο αυτό, θα αποφεύγεται η ανάθεση μεγαλύτερου βάρους σε έγγραφα τα οποία είναι απλά μεγάλα σε μέγεθος και πιθανόν να περιέχουν περισσότερες φορές τη λέξη - κλειδί, ανεξάρτητα από τη σημασία που έχει πραγματικά η λέξη - κλειδί στο συγκεκριμένο έγγραφο. [15]

Ο παράγοντας idf υπολογίζεται ως εξής :

$$(2) \quad idf = \log (|D| / f_{w,D})$$

,όπου $f_{w,D}$ είναι ο αριθμός των εγγράφων στα οποία εμφανίζεται η λέξη - κλειδί w και $|D|$ το σύνολο των εγγράφων[8]. Στον παρονομαστή $f_{w,D}$ προστίθεται πολλές φορές μία επιπλέον μονάδα , δηλ. $f_{w,D} + 1$, ώστε να αποφευχθεί η διαίρεση με το 0 σε περίπτωση που η λέξη – κλειδί δεν εμφανίζεται σε κανένα έγγραφο. [15]

Το βάρος που θα ανατεθεί τελικά στη λέξη - κλειδί θα ισούται με το πιο κάτω γινόμενο :

$$(3) \quad tf-idf = tf * idf$$

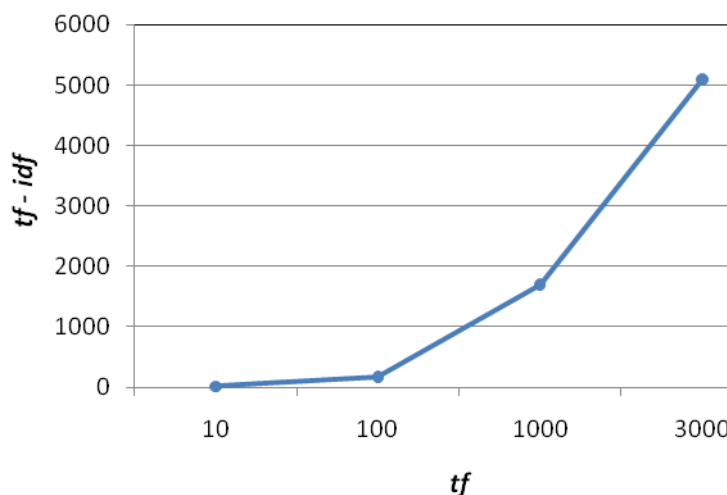
Με λίγα λόγια, λέξεις – κλειδιά του Query που εμφανίζονται σε ένα μικρό σύνολο εγγράφων έχουν μεγαλύτερο βάρος ($tf-idf$) από λέξεις – κλειδιά που είναι πολύ συνηθισμένες και εμφανίζονται σε πολλά έγγραφα. Γενικά, στην επιστήμη Ανάκτησης Πληροφορίας ένας όρος με υψηλή συχνότητα όρου αλλά χαμηλή συχνότητα εγγράφων εκτιμάται ως καλός όρος ευρετηρίασης. [15]

Στη γραφική παράσταση του σχήματος 2.2, παρατηρούμε πως μεταβάλλεται η τιμή του $tf-idf$, δηλ. το βάρος που ανατίθεται στη λέξη - κλειδί, καθώς αυξάνεται η συχνότητα εμφάνισής της (tf) στο ίδιο σύνολο εγγράφων. Βλέπουμε πως όσο

αυξάνεται η συχνότητα εμφάνισης της λέξης στα έγγραφα τόσο αυξάνεται και το βάρος που της αναλογεί.

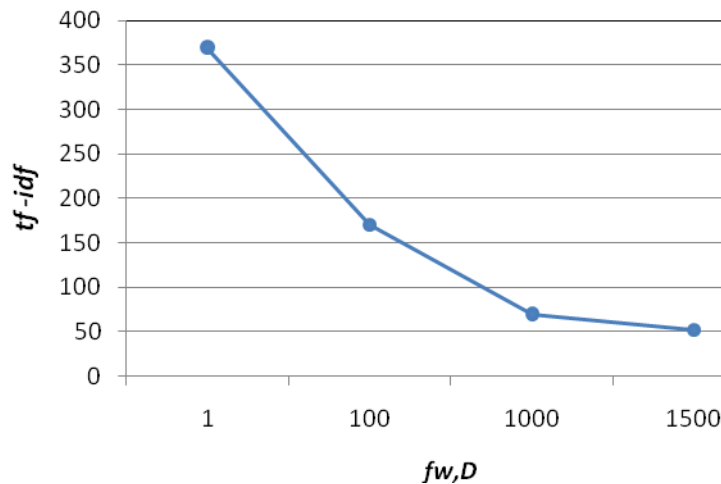
Στη γραφική παράσταση του σχήματος 2.3, παρατηρούμε πως μεταβάλλεται η τιμή του $tf-idf$ καθώς αυξάνεται ο αριθμός των εγγράφων στα οποία εμφανίζεται η λέξη - κλειδί. Βλέπουμε πως όσο μεγαλώνει το σύνολο των εγγράφων στα οποία εμφανίζεται τόσο μειώνεται το βάρος που της αναλογεί.

tf	D	fw,D	idf	$tf-idf$
10	5000	100	1,69	16,9
100	5000	100	1,69	169,9
1000	5000	100	1,69	1698,97
3000	5000	100	1,69	5096,91



Σχήμα 2.2 Γραφική παράσταση του $tf-idf$ συναρτήσεως του παράγοντα tf

tf	D	fw,D	idf	$tf-idf$
100	5000	1	3,69	369,9
100	5000	100	1,69	169,9
100	5000	1000	0,69	69,9
100	5000	1500	0,522	52,28



Σχήμα 2.3 Γραφική παράσταση του $tf-idf$ συναρτήσεως του fw,D

2.5 Proximity Search

Το Proximity Search αποτελεί μία μέθοδο αναζήτησης η οποία έχει εφαρμοστεί επιτυχώς σε διάφορα συστήματα ανάκτησης πληροφορίας. Αυτό που κάνει η συγκεκριμένη μέθοδος είναι να ταξινομεί έγγραφα σύμφωνα με το πόσο «κοντά» βρίσκονται οι λέξεις – κλειδιά του Query σε κάθε ένα από αυτά. Η απόσταση μεταξύ των λέξεων – κλειδίων στο έγγραφο καθορίζεται από τον αριθμό των λέξεων που βρίσκονται ανάμεσά τους. [16,17]

Η μέθοδος αυτή έχει επίσης εφαρμοστεί για εντοπισμό αντικειμένων της βάσης δεδομένων που βρίσκονται «κοντά» σε άλλα σχετικά αντικείμενα. Στην προκειμένη περίπτωση η βάση δεδομένων αναπαριστάται υπό τη μορφή ενός γράφου, όπου οι κόμβοι αντιπροσωπεύουν τα αντικείμενα ενώ οι ακμές τις σχέσεις μεταξύ των αντικειμένων. Η απόσταση μεταξύ των αντικειμένων καθορίζεται από το συντομότερο μονοπάτι που υπάρχει μεταξύ τους στο γράφο αυτό.[17]

Για την εκτέλεση της αναζήτησης, δίνεται ως είσοδος ένα ζεύγος από Queries το οποίο αποτελείται από το *Find query* και το *Near query*. Το *Find query* καθορίζει το σύνολο των αντικειμένων που μας ενδιαφέρουν. Ο σκοπός είναι να ταξινομηθούν τα αντικείμενα που περιέχονται στο *Find query* με βάση την απόστασή τους από τα αντικείμενα που καθορίζονται από το *Near query*. Για παράδειγμα, έστω ότι ένας χρήστης ενδιαφέρεται να πάρει πληροφορίες, στα πλαίσια της βάσης δεδομένων

Northwind, για Products που έχουν παραγγείλει οι Customers Maria και Leverling. Αυτό μπορεί να εκφραστεί ως “Find Products Near Maria Leverling”. Με τον τρόπο αυτό θα εντοπιστούν αντικείμενα τύπου Products που συσχετίζονται με τα αντικείμενα Maria AND/OR Leverling. [17]

2.6 Object Summary (OS)

Η αναζήτηση σε μία σχεσιακή βάση δεδομένων με τη μέθοδο Keyword Search (R-KwS) εισάγοντας μόνο μία λέξη – κλειδί μας επιστρέφει , όπως έχουμε δει, μόνο τις πλειάδες που περιέχουν τη συγκεκριμένη λέξη - κλειδί σε κάποιο από τα πεδία τους. Αδυνατεί , όμως, στο να μας προσφέρει οποιεσδήποτε επιπλέον πληροφορίες που πιθανόν να σχετίζονται με το συγκεκριμένο αντικείμενο. Για την επίλυση του προβλήματος αυτού έχει εισαχθεί η έννοια του Object Summary [1].

2.6.1 Τι εννοούμε με τον όρο Object Summary;

Ένα Object Summary (OS) μας δίνει μία περίληψη όλων των στοιχείων που αφορούν ένα συγκεκριμένο Data Subject (DS) της σχεσιακής βάσης δεδομένων. Με τον όρο DS αναφερόμαστε σε ένα αντικείμενο για το οποίο έχουμε κάποια δεδομένα. Για παράδειγμα, στη βάση δεδομένων Northwind ένα DS μπορεί να είναι ένας Customer ή ένας Employee ή ένα Product κτλ. [1]

Έστω, ότι θέλουμε να πάρουμε πληροφορίες για τον Employee με το όνομα “Leverling”. Τότε, θα δημιουργηθεί ένα OS το οποίο συνοψίζει όλες τις πληροφορίες από τη βάση δεδομένων που αφορούν τον Employee “Leverling”. Το OS αυτό θα έχει στη ρίζα του την πλειάδα από την σχέση Employees που περιέχει το όνομα “Leverling” σε κάποιο από τα πεδία της και κόμβους – παιδιά που περιέχουν επιπλέον πληροφορίες για τον συγκεκριμένο υπάλληλο, όπως για τις παραγγελίες που έχει αναλάβει, τους πελάτες που έχει εξυπηρετήσει κτλ. Το OS είναι, ουσιαστικά, ένα δέντρο αποτελούμενο από κάποιες πλειάδες [1].

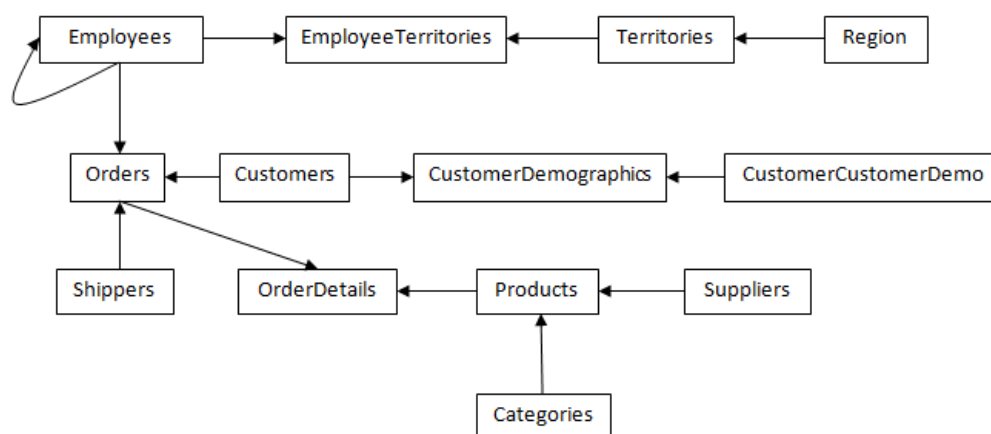
2.6.2 Data Subject Schema Graph (G^{DS})

Η δημιουργία ενός OS προϋποθέτει τη δημιουργία ενός Data Subject Schema Graph (G^{DS}). Το G^{DS} είναι ένα Directed Labeled Tree το οποίο αναπαριστά ένα υποσύνολο του σχεσιακού μοντέλου της βάσης δεδομένων που χρησιμοποιείται [1].

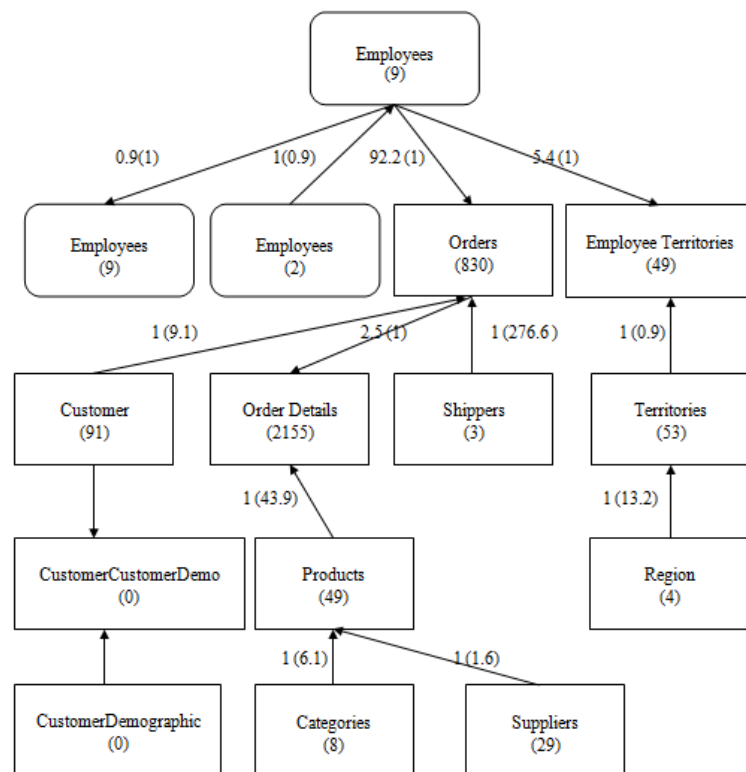
Η ρίζα του δέντρου G^{DS} αντιπροσωπεύει τη σχέση R^{DS} στην οποία ανήκει το DS για το οποίο ενδιαφέρεται ο χρήστης. Για παράδειγμα, εάν ο χρήστης επιθυμεί να πάρει πληροφορίες, στα πλαίσια της βάσης δεδομένων Northwind, για τον υπάλληλο “Leverling” τότε η σχέση R^{DS} θα είναι η σχέση Employees στην οποία ανήκει το DS “Leverling”. Τα παιδιά της σχέσης R^{DS} θα είναι οι γειτονικές σχέσεις σύμφωνα το σχεσιακό μοντέλο της βάσης δεδομένων (σχήμα 2.4) , καθώς επίσης και οποιεσδήποτε αναδρομικές σχέσεις είναι πιθανόν να υπάρχουν.

Σε κάθε κόμβο του G^{DS} δέντρου ανατίθεται ο Cardinality βαθμός δηλ. ο συνολικός αριθμός εγγραφών που περιέχει η συγκεκριμένη σχέση, αλλά και η τιμή του Affinity την οποία θα εξετάσουμε στη συνέχεια. Επίσης, σε κάθε ακμή ανατίθεται το Relative Cardinality και το Reverse Relative Cardinality μεταξύ των δύο σχέσεων που συνδέονται [1].

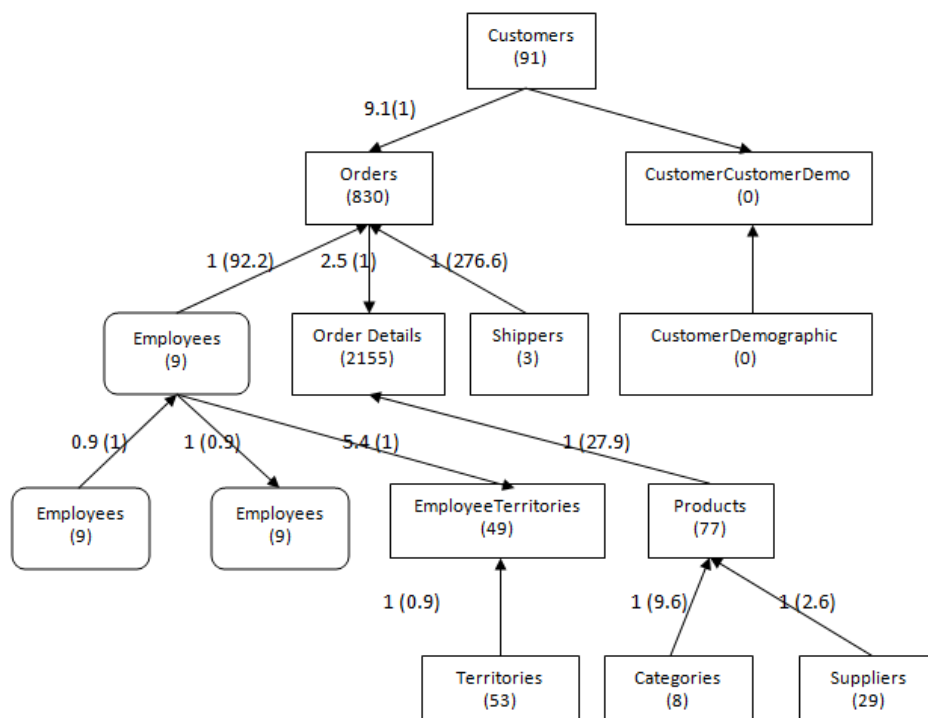
Στο σχήμα 2.5 βλέπουμε , στα πλαίσια και πάλι της βάσης δεδομένων Northwind, το G^{DS} δέντρο που έχει δημιουργηθεί για τη σχέση Employees ενώ στο σχήμα 2.6 βλέπουμε το G^{DS} δέντρο για τη σχέση Customers. Σε κάθε ακμή του δέντρου ανατίθεται, όπως έχουμε προαναφέρει, το Relative Cardinality και το Reverse Relative Cardinality μεταξύ των δύο σχέσεων, ενώ σε κάθε κόμβο αναγράφεται και ο Cardinality βαθμός.



Σχήμα 2.4 Σχεσιακό μοντέλο της βάσης δεδομένων Northwind



Σχήμα 2.5 G^{DS} δέντρο για τη σχέση Employees στα πλαίσια της βάσης δεδομένων Northwind



Σχήμα 2.6 G^{DS} δέντρο για τη σχέση Customers στα πλαίσια της βάσης δεδομένων Northwind

2.6.3 Σχηματισμός OS δέντρου

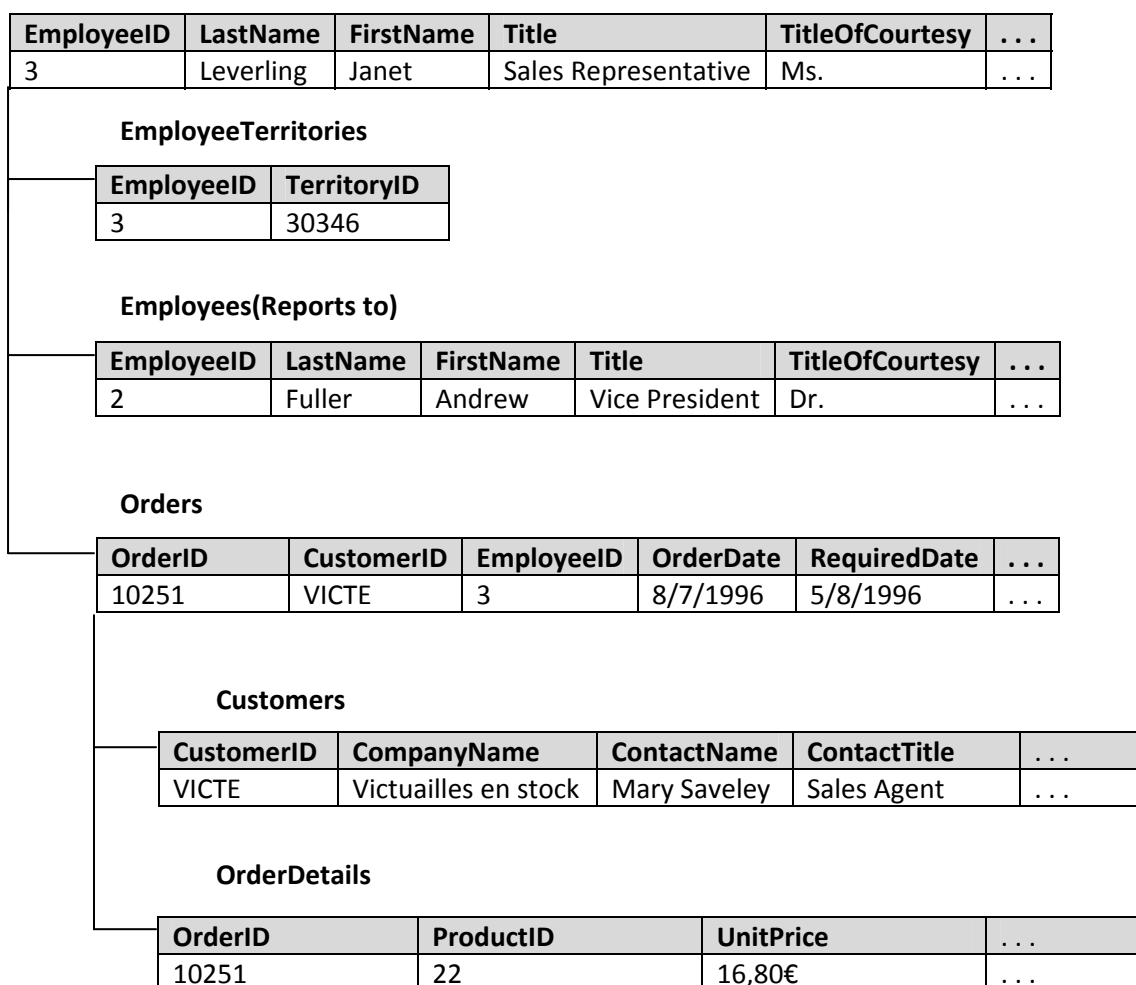
Όταν ο χρήστης ζητήσει πληροφορίες για ένα συγκεκριμένο DS, τότε αρχικά θα εντοπιστούν όλες οι πλειάδες από τη βάση δεδομένων που περιέχουν το DS σε κάποιο από τα πεδία τους [1]. Στη συνέχεια, για κάθε μία από τις πλειάδες αυτές θα σχηματιστεί ένα OS ως ακολούθως.

Επιλέγουμε μία από τις πλειάδες και την ορίζουμε ως τη ρίζα του OS δέντρου. Σύμφωνα με τη σχέση στην οποία ανήκει η πλειάδα αυτή (π.χ Employees, Customers), επιλέγουμε το αντίστοιχο G^{DS} δέντρο και το διαπερνούμε με κατά πλάτος αναζήτηση εφαρμόζοντας SQL Queries της μορφής $\sigma_{R_i.FK = t_j.PK}(R_i)$ για σχέσεις $R_j \rightarrow R_i$ και $\sigma_{R_i.PK = t_j.FK}(R_i)$ για σχέσεις $R_j \leftarrow R_i$, όπου R_i η τρέχον σχέση στο G^{DS} δέντρο και t_j η πλειάδα η οποία ανήκει στο γονέα R_j . Τα αποτελέσματα των πράξεων αυτών, δηλ. οι πλειάδες που ανακτώνται κάθε φορά, προστίθενται στο δέντρο του OS σαν κόμβοι - παιδιά της πλειάδας t_j [1].

Η διαδικασία αυτή επαναλαμβάνεται, όπως έχουμε πει, για κάθε μία από τις πλειάδες που περιέχουν το DS σε κάποιο από τα πεδία τους, οπότε στο τέλος θα έχουμε στη διάθεσή μας ένα σύνολο από OSs.

Στο σχήμα 2.7 βλέπουμε πως έχει δημιουργηθεί ένα OS, στα πλαίσια της βάσης δεδομένων Northwind, στην περίπτωση που ο χρήστης έχει δώσει σαν λέξη – κλειδί το όνομα “Leverling”. Παρατηρούμε πως στη ρίζα του δέντρου βρίσκεται η πλειάδα στην οποία ανήκει το DS “Leverling” και ακολουθούν γειτονικές πλειάδες οι οποίες συσχετίζονται μέσω των πρωτεύοντων κλειδίων τους.

Employees



Σχήμα 2.7 OS για το DS “Leverling” στα πλαίσια της βάσης δεδομένων Northwind

2.6.4 Affinity

Ένα OS, όπως έχει προαναφερθεί, περιέχει πληροφορίες που αφορούν ένα συγκεκριμένο DS. Υπάρχει μεγάλη πιθανότητα όμως, να περιλαμβάνει πλειάδες οι οποίες δεν σχετίζονται ιδιαίτερα με το DS για το οποίο ενδιαφέρεται ο χρήστης και συνεπώς να μην υπήρχε λόγος να συμπεριληφθούν. Για να αποφευχθεί το γεγονός αυτό εφαρμόζεται η τεχνική του Affinity [1].

Στο δέντρο G^D έχουμε στη ρίζα τη σχέση R^{DS} στην οποία ανήκει το DS που έχει καθορίσει ο χρήστης. Σχέσεις που συνδέονται γύρω από τη ρίζα στο δέντρο αυτό, μπορούν να μας δώσουν επιπλέον πληροφορίες για το συγκεκριμένο DS. Η χρήση του Affinity θα μας βοηθήσει να καθορίσουμε ποιές από αυτές τις σχέσεις

σχετίζονται άμεσα με το DS και θα πρέπει συνεπώς να διαπεραστούν κατά τη δημιουργία ενός OS και ποιές όχι.

Για κάθε σχέση R_i που υπάρχει στο G^{DS} δέντρο υπολογίζεται το Affinity που έχει ως προς τη σχέση R^{DS} , δηλ. σε ποίο βαθμό συνδέεται με αυτή. Η τιμή που θα έχει το Affinity εξαρτάται από τις πιο κάτω μετρικές [1]:

1. Relations' Distance

Η απόσταση μεταξύ των σχέσεων R_i και R^{DS} καθορίζεται από το μήκος του μονοπατιού που τις συνδέει στο δέντρο G^{DS} . Όσο πιο μικρή είναι η απόσταση αυτή τόσο πιο μεγάλο είναι το Affinity που υπάρχει μεταξύ τους. Αν πάρουμε για παράδειγμα το Σχήμα 2.5, παρατηρούμε ότι η σχέση Orders βρίσκεται πιο κοντά στη σχέση R^{DS} (Employees) σε σύγκριση με τη σχέση Products η οποία βρίσκεται σε μεγαλύτερη απόσταση από αυτή. Συνεπώς, η σχέση Orders θα έχει μεγαλύτερο Affinity ως προς τη σχέση Employees σε σύγκριση με τη σχέση Products [1].

Ωστόσο, οποιεσδήποτε φυσικές σχέσεις υπάρχουν στο μονοπάτι από τη σχέση R_i στη σχέση R^{DS} , δηλ. σχέσεις $M:N$ οι οποίες περιλαμβάνουν μόνο τα πρωτεύοντα κλειδιά των σχέσεων που συμμετέχουν, εξαιρούνται κατά τον υπολογισμό της απόστασης. Ορίζουμε ως ld_i τη λογική απόσταση μεταξύ των σχέσεων R_i και R^{DS} και ως fd_i τη φυσική απόσταση. Έστω ότι, $|M:N|$ είναι ο αριθμός των φυσικών σχέσεων που υπάρχουν στο μονοπάτι. Τότε, $ld_i = fd_i - |M:N|$. Οπότε, ουσιαστικά αυτό που λαμβάνεται υπόψη στη μετρική αυτή είναι η απόσταση ld_i [1].

2. Connectivity

Μία άλλη μετρική που καθορίζει πόσο κοντά βρίσκεται μία σχέση R_i στη σχέση R^{DS} είναι η συνδετικότητα (Connectivity) της σχέσης R_i η οποία συνδετικότητα αφορά το Schema Connectivity (Co_i) και το Relative Cardinality ($RC_{i->j}$) [1].

Το Schema Connectivity (Co_i) είναι στην ουσία ο αριθμός των σχέσεων στις οποίες συμμετέχει η σχέση R_i . Αν πάρουμε, για παράδειγμα, το σχήμα 2.5 η σχέση Orders συμμετέχει σε 4 σχέσεις συνεπώς και έχει $Co_i = 4$ ενώ η σχέση Region συμμετέχει σε μία μόνο σχέση γι' αυτό και έχει $Co_i = 1$.

Το Relative Cardinality μίας σχέσης R_i ως προς τη σχέση R_j ($RC_{i->j}$) αντιπροσωπεύει τον αριθμό των πλειάδων της R_i που συνδέονται με κάθε πλειάδα της σχέσης R_j . Για

τον υπολογισμό του Affinity, αυτό που μας ενδιαφέρει είναι μόνο το Relative Cardinality της σχέσης R_i σε σχέση με το γονέα της στο δέντρο G^{DS} [1].

Ο υπολογισμός του Relative Cardinality γίνεται σχετικά εύκολα. Για 1: M σχέσεις το $RC_{i \rightarrow j}$ υπολογίζεται σύμφωνα με την πράξη $RC_{i \rightarrow j} = |R_i| / |R_j|$, όπου $|R_i|$ και $|R_j|$ είναι ο αριθμός των πλειάδων της σχέσης R_i και R_j αντίστοιχα. Για M:1 σχέσεις όσο και για 1:1 σχέσεις το $RC_{i \rightarrow j}$ ισούται με 1.

Σύμφωνα, λοιπόν, με το Σχήμα 2.5 το $RC_{Orders \rightarrow Employees}$ υπολογίζεται ως εξής :

$$RC_{Orders \rightarrow Employees} = |R_{Orders}| / |R_{Employees}| = 830 / 9 = 92.2$$

Με τον ίδιο τρόπο :

$$RC_{Orders \rightarrow OrderDetails} = |R_{Orders}| / |R_{OrderDetails}| = 830 / 2155 = 0.385$$

Όσο πιο μικρή είναι η τιμή του $RC_{i \rightarrow j}$ τόσο πιο κοντά στο γονέα της βρίσκεται η σχέση R_i [1].

Το Reverse Relative Cardinality ορίζεται ως $\overline{RC}_{i \rightarrow j} = RC_{j \rightarrow i}$ και συνεπώς μπορεί να υπολογιστεί με τον ίδιο τρόπο όπως και το Relative Cardinality [1].

3. Penalization of Lateral Data

Σχέσεις που λειτουργούν σαν γέφυρες (hubs) μεταξύ της σχέσης R^{DS} και μίας σχέσης R_i , έστω $R^{DS} \dots \leftarrow R_{hub} \rightarrow R_i$, μας δίνουν κάποια πλευρικά στοιχεία που δεν είναι απαραίτητο να συμπεριληφθούν στο OS. Για παράδειγμα, εάν πάρουμε τη σχέση $R_{Supplier}$ η οποία βρίσκεται στο μονοπάτι $R_{Customers} \leftarrow R_{Nation} \rightarrow R_{Supplier}$, στα πλαίσια της βάσης δεδομένων TPC – H αυτή τη φορά, το αποτέλεσμα θα είναι μία λίστα με όλους τους Suppliers που ανήκουν στο ίδιο Nation με τον συγκεκριμένο Customer. Προφανώς, όμως, αυτό δεν έχει κάποια άμεση σχέση με τον Customer. Συνεπώς, τέτοια πλευρικά στοιχεία θα πρέπει οπωσδήποτε να εξαιρεθούν από ένα OS. Για να επιτευχθεί αυτό, αυξάνεται η επίδραση της απόστασης ld_i κατά ένα παράγοντα h , δηλ. θα έχουμε $ld_i = ld_i * h$, αλλά και του Relative Cardinality [1].

4. Affinity Descriptor of Ri (DAf(Ri))

Ο όρος αυτός αναφέρεται σε μία λίστα από σταθμισμένες μετρικές $m_1, m_2, \dots, m_n$ με τα αντίστοιχά τους βάρη $w_1, w_2, \dots, w_n$. Για τον υπολογισμό του Affinity χρησιμοποιούνται οι εξής σταθμισμένες μετρικές [1] :

- $m_1 = f_1(ld_i)$ για την απόσταση
- $m_2 = f_1(\log(10 \cdot RC_i))$ για το Relative Cardinality
- $m_3 = f_1(\log(10 \cdot \overline{RC}_i))$ για το Reverse Relative Cardinality
- $m_4 = f_1(Co_i)$ για το Schema Connectivity

Σε περίπτωση που υπάρχουν σχέσεις που λειτουργούν σαν γέφυρες, όπως έχουμε δει προηγουμένως, η μετρική για την απόσταση θα μεταβληθεί σε $m_1 = f_1(ld_i \cdot h)$ και του Relative Cardinality σε $m_2 = f_1(RC_i)$ [1].

Η συνάρτηση f_1 χρησιμοποιείται ουσιαστικά, για την κανονικοποίηση των σταθμισμένων μετρικών στο 1, ώστε η τιμή του Affinity να μην παρουσιάζει απότομες μεταβολές. Για τον ίδιο λόγο, χρησιμοποιούνται οι λογάριθμοι στο Relative Cardinality (m_2) και το Reverse Relative Cardinality (m_3) [1].

Η συνάρτηση f_1 ορίζεται ως εξής:

$$f_1(\alpha) = \begin{cases} (11 - \alpha)/10 & 1 \leq \alpha \leq 10 \\ 1 & 0 < \alpha < 1 \\ 0 & \alpha > 10 \end{cases}$$

Ο τελικός υπολογισμός του Affinity της σχέσης R_i ως προς τη σχέση R^{DS} γίνεται με βάση την πιο κάτω εξίσωση :

$$(4) \quad Af_{R_i \rightarrow R^{DS}} = \sum_j m_j w_j \cdot Af_{R_{parent} \rightarrow R^{DS}}$$

, όπου $Af_{R_{parent} \rightarrow R^{DS}}$ είναι η τιμή του Affinity του γονέα της σχέσης R_i ως προς τη σχέση R^{DS} και $m_j \cdot w_j$ το γινόμενο της σταθμισμένης μετρικής j με το αντίστοιχό της βάρος [1].

Το αποτέλεσμα της πράξης αυτής θα είναι μία τιμή από το πεδίο $[0, 1]$. Η σχέση R^{DS} παίρνει τη μέγιστη τιμή όσον αφορά το Affinity δηλ. 1.

Έστω ότι χρησιμοποιούμε τα βάρη $w_1=0.5$, $w_2=0.4$, $w_3=0.05$ και $w_4=0.05$ για τις μετρικές m_1, m_2, m_3 και m_4 , έτσι ώστε να υπολογίσουμε την τιμή του Affinity για κάθε σχέση του G^{DS} δέντρου της σχέσης Employees (σχήμα 2.5) και για κάθε σχέση του G^{DS} δέντρου της σχέσης Customer (σχήμα 2.6). Τότε, σύμφωνα με την εξίσωση (4) προκύπτουν οι ακόλουθες τιμές:

R_i \ R^{DS}	Employees		
	$Id_i, RC_i, \overline{RC}_i, Co_i$	$m_1, \dots, m_4$	Af_{R_i}
Employees	R^{DS}	R^{DS}	1.00
Employees (ReportsTo)	1,1,0.9,4	1,1,1,0.7	0.98
Employees (ReportedBy)	1,0.9,1,4	1,1,1,0.7	0.98
Territories	1,5.4,1,2	1,0.9,1,0.9	0.96
Region	2,1,13.2,1	0.9,1,0.88,1	0.91
Order	1,92.2,1,4	1,0.8,1,0.7	0.90
Customer	2,1,9.1,2	0.9,1,0.9,0.9	0.85
Shipper	2,1,276.6,1	0.9,1,0.75,1	0.85
OrderDetails	2,2.5,1,2	0.9,0.96,1,0.9	0.84
Products	3,1,43.9,3	0.8,1,0.83,0.8	0.74
Supplier	4,1,1.6,1	0.7,1,0.9,1	0.63
Categories	4,1,6.1,1	0.7,1,0.92,1	0.62
CustomerDemographics	3, null, null, 1	0.8,null,null,1	Null

Πίνακας 2.1 Τιμές Affinity για κάθε σχέση του G^{DS} δέντρου των Employees

R_i \ R^{DS}	Customers		
	$Id_i, RC_i, \overline{RC}_i, Co_i$	$m_1, \dots, m_4$	Af_{R_i}
Customers	R^{DS}	R^{DS}	1.00
Order	1,9.1,1,4	1,0.9,1,0.7	0.94
Employees	2,1,92.2,4	0.9,1,0.8,0.7	0.88
Employees (ReportsTo)	3,1,0.9,4	0.8,1,1,0.7	0.78
Employees (ReportedBy)	3,0.9,1,4	0.8,1,1,0.7	0.70
Territories	3,5.4,1,2	0.8,0.9,1,0.9	0.55
Region	4,1,13.2,1	0.7,1,0.88,1	0.46
Shipper	2,1,276.6,1	0.9,1,0.75,1	0.88
OrderDetails	2,2.5,1,2	0.9,0.96,1,0.9	0.88
Products	3,1,27.9,3	0.8,1,0.8,0.8	0.77
Supplier	4,1,2.6,1	0.7,1,0.96,1	0.65
Categories	4,1,9.6,1	0.7,1,0.9,1	0.65
CustomerDemographics	1, null, null, 1	1,null,null,1	Null

Πίνακας 2.2 Τιμές Affinity για κάθε σχέση του G^{DS} δέντρου των Customers

Ο σωστός και ακριβής υπολογισμός του Affinity είναι ένας πολύ σημαντικός παράγοντας για τη δημιουργία ενός καλού OS.

2.7 Συμπεράσματα

Στο κεφάλαιο αυτό, λοιπόν, αναλύθηκαν κάποιες πολύ σημαντικές έννοιες πάνω στις οποίες θα βασιστεί η υλοποίηση του συστήματος της προτεινόμενης μεθοδολογίας. Αρχικά, αναφερθήκαμε στη μέθοδο αναζήτησης Keyword Search η οποία χρησιμοποιείται τόσο στο χώρο του διαδικτύου όσο και σε σχεσιακές βάσεις δεδομένων ενώ στη συνέχεια μιλήσαμε για την επιστήμη της Ανάκτησης Πληροφορίας και αναλύσαμε δύο πολύ βασικές IR-Style τεχνικές, τη μέθοδο stemming και τη μέθοδο TF-IDF. Ακολούθως, αναφερθήκαμε στη μέθοδο αναζήτησης Proximity Search ενώ επίσης μιλήσαμε για την έννοια του Object Summary και για δύο βασικές του παραμέτρους, το Data Subject Schema Graph (G^{DS}) και το Affinity στο οποίο δόθηκε ιδιαίτερη λεπτομέρεια.

Κεφάλαιο 3

Προτεινόμενη μεθοδολογία ταξινόμησης OSs

3.1 Εισαγωγή	26
3.2 Σύντομη περιγραφή της προτεινόμενης μεθοδολογίας	27
3.3 Είσοδος από τον χρήστη	28
3.3.1 Πρώτη λέξη – κλειδί : Καθορισμός DS	28
3.3.2 Δεύτερη λέξη – κλειδί : Καθορισμός θεματικής ενότητας	29
3.4 Μέθοδος ταξινόμησης των αποτελεσμάτων	30
3.4.1 Εφαρμογή μεθόδου stemming	30
3.4.2 Συνδυασμός μεθόδου TF-IDF και Affinity	31
3.5 Αρχιτεκτονική συστήματος προτεινόμενης μεθοδολογίας	38
3.6 Φάση προεπεξεργασίας	40
3.6.1 Αρχείο προεπιλεγμένων λέξεων	40
3.6.2 Εύρεση συνόλου πλειάδων T^{OSs}	42
3.6.3 Δημιουργία OSs και υπολογισμός τιμών $Af-tf$	42
3.6.4 Δημιουργία ευρετηρίου	43
3.7 Φάση επεξεργασίας	44
3.7.1 Εύρεση συνόλου πλειάδων T^{k1}	44
3.7.2 Εύρεση τιμών $Af-tf$	45
3.7.3 Ταξινόμηση τιμών $Af-tf$	47
3.7.4 Δημιουργία και παρουσίαση των OSs	47
3.8 Συμπεράσματα	47

3.1 Εισαγωγή

Αντικείμενο μελέτης της παρούσας διπλωματικής εργασίας, όπως έχει αναφερθεί και στο πρώτο κεφάλαιο, είναι η ανάπτυξη μίας καινούργιας μεθοδολογίας για την ταξινόμηση των αποτελεσμάτων που λαμβάνονται από την εκτέλεση αναζητήσεων σε σχεσιακές βάσεις δεδομένων. Συγκεκριμένα, η μεθοδολογία που θα αναπτύξουμε θα

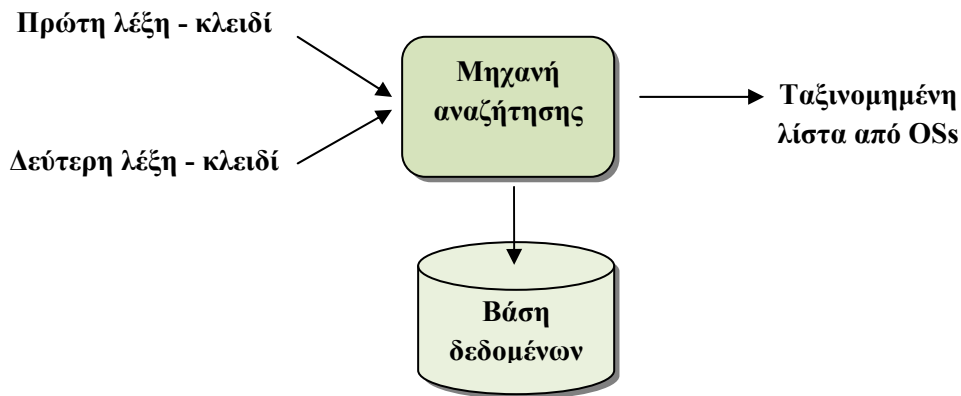
υιοθετήσει την έννοια του OS, η οποία έχει επεξηγηθεί εκτενώς στο προηγούμενο κεφάλαιο, ενώ πρόκειται να συνδυάσει IR-Style τεχνικές με την έννοια του Affinity για την τελική ταξινόμηση των αποτελεσμάτων. Όπως έχουμε πει, η υλοποίηση του συστήματος της μεθόδου αυτής θα γίνει στα πλαίσια της βάσης δεδομένων Northwind. Στο κεφάλαιο αυτό, λοιπόν, θα αναλύσουμε την καινούργια μεθοδολογία και θα δούμε μέσα από διάφορα παραδείγματα από τη βάση δεδομένων Northwind, τον τρόπο με τον οποίο μπορούν να συνδυαστούν οι πιο πάνω έννοιες για τη δημιουργία μίας όσο το δυνατόν πιο αποδοτικής μηχανής αναζήτησης.

3.2 Σύντομη περιγραφή της προτεινόμενης μεθοδολογίας

Σύμφωνα με την προτεινόμενη μεθοδολογία, πρόκειται να υλοποιηθεί ένα σύστημα, μία μηχανή αναζήτησης, η οποία θα προσφέρει τη δυνατότητα στον χρήστη να αναζητά πληροφορίες για ένα συγκεκριμένο αντικείμενο από μία σχεσιακή βάση δεδομένων. Ο χρήστης θα πρέπει να εισαγάγει στη μηχανή αναζήτησης (1) μία λέξη – κλειδί η οποία θα καθορίζει το αντικείμενο για το οποίο επιθυμεί να πάρει πληροφορίες και (2) μία δεύτερη λέξη – κλειδί η οποία θα καθορίζει τη θεματική ενότητα που τον ενδιαφέρει περισσότερο βάσει της οποίας θα γίνεται η ταξινόμηση των αποτελεσμάτων. Στο σημείο αυτό, θα πρέπει να αναφέρουμε ότι ο καθορισμός του αντικειμένου αλλά και της θεματικής ενότητας θα μπορούσε να γίνει με περισσότερες από μία λέξεις – κλειδιά. Στην προκειμένη περίπτωση εφαρμόζονται τελεστές του τύπου AND/OR.

Τα αποτελέσματα που θα λαμβάνει ο χρήστης για το αντικείμενο του ενδιαφέροντός του, θα είναι ένα σύνολο από OSs. Τα OSs αυτά, θα ταξινομηθούν με την εφαρμογή IR-Style τεχνικών σε συνδυασμό με την έννοια του Affinity χρησιμοποιώντας τη δεύτερη λέξη – κλειδί. Με τον τρόπο αυτό, ευελπιστούμε ότι ο χρήστης θα παίρνει κάθε φορά πληροφορίες οι οποίες βρίσκονται όσο το δυνατόν πιο κοντά σε αυτό που επιθυμεί.

Στο σχήμα 3.1 παρουσιάζεται ένα απλοποιημένο διάγραμμα της προτεινόμενης μεθοδολογίας, όπου δίνονται ως είσοδος στη μηχανή αναζήτησης οι δύο λέξεις – κλειδιά και δημιουργείται ως έξοδος μία ταξινομημένη λίστα από OSs.



Σχήμα 3.1 Απλοποιημένο διάγραμμα προτεινόμενης μεθοδολογίας

3.3 Είσοδος από τον χρήστη

Κάθε μηχανή αναζήτησης απαιτεί από τον χρήστη να εισαγάγει ένα Query, ώστε να εκτελέσει την αναζήτησή του και να πάρει τις πληροφορίες που επιθυμεί. Έτσι κι εδώ, το σύστημα που θα υλοποιήσουμε θα προσφέρει τη δυνατότητα εισαγωγής ενός Query το οποίο θα αποτελείται, όπως έχει προαναφερθεί, από δύο λέξεις – κλειδιά.

Έστω ότι χρησιμοποιούμε τον πιο κάτω συμβολισμό για να αναπαραστήσουμε το κάθε Query:

$$Q = \{k_1, k_2\}$$

, όπου k_1 και k_2 , η πρώτη και δεύτερη λέξη – κλειδί αντίστοιχα.

3.3.1 Πρώτη λέξη – κλειδί : Καθορισμός DS

Στην προτεινόμενη μεθοδολογία τα αποτελέσματα που θα λαμβάνει ο χρήστης με την εκτέλεση κάθε αναζήτησης θα είναι, όπως έχουμε πει, ένα σύνολο από OSs. Έτσι, για να καταστεί δυνατή η δημιουργία του συγκεκριμένου συνόλου θα πρέπει πρώτα να καθοριστεί το DS στο οποίο θα απευθύνονται. Ο καθορισμός αυτός θα γίνεται σύμφωνα με την λέξη – κλειδί k_1 του Query.

Δεδομένου ότι θα χρησιμοποιηθεί η βάση δεδομένων Northwind, η λέξη – κλειδί k_1 που θα δώσει ο χρήστης στο συγκεκριμένο σύστημα θα μπορεί να αφορά είτε

κάποιον υπάλληλο (σχέση Employees) είτε κάποιον πελάτη (σχέση Customers). Ο χρήστης, λοιπόν, θα μπορεί να εισαγάγει σαν λέξη – κλειδί k_1 το όνομα ή το επίθετο κάποιου υπαλλήλου ή το όνομα, ακόμα και την ταυτότητα κάποιου πελάτη.

Για παράδειγμα, έστω ότι έχει εισαχθεί σαν λέξη – κλειδί k_1 το όνομα “Leverling”. Τότε, η μηχανή αναζήτησης θα δημιουργήσει ένα σύνολο από OSs για όλους τους “Leverling”, είτε Customers είτε Employees, που υπάρχουν στη βάση δεδομένων Northwind.

3.3.2 Δεύτερη λέξη – κλειδί : Καθορισμός θεματικής ενότητας

Το βασικό στοιχείο που χαρακτηρίζει τη νέα αυτή μεθοδολογία είναι η χρήση μίας δεύτερης λέξης – κλειδί με σκοπό την ταξινόμηση του συνόλου των αποτελεσμάτων, δηλ. του συνόλου των OSs. Στο εν λόγω σύστημα, αυτή η δεύτερη λέξη – κλειδί θα καθορίζει τη θεματική ενότητα για την οποία ενδιαφέρεται περισσότερο ο χρήστης και η οποία θα αφορά φυσικά το αντικείμενο που έχει καθοριστεί. Έτσι, λοιπόν, αφού γίνει ο καθορισμός του DS για το οποίο θα δημιουργηθεί το σύνολο των OSs σύμφωνα με τη λέξη – κλειδί k_1 του Query, θα πρέπει επίσης να καθοριστεί πριν από την εκτέλεση της αναζήτησης και η θεματική ενότητα που ενδιαφέρει τον χρήστη σύμφωνα με τη λέξη – κλειδί k_2 του Query.

Η λέξη – κλειδί k_2 θα χρησιμοποιηθεί αποκλειστικά για την ταξινόμηση του συνόλου των OSs με την εφαρμογή IR-Style τεχνικών, όπως θα δούμε στη συνέχεια.

Έστω ότι ο χρήστης έχει δώσει ως είσοδο το πιο κάτω Query :

$$Q = \{\text{“Leverling”}, \text{“Sales”}\}$$

Η αναζήτηση αυτή θα επιστρέψει ένα σύνολο από OSs για όλους τους “Leverling” , ταξινομημένο με IR-Style τεχνικές σύμφωνα με τη λέξη - κλειδί “Sales”. Πρώτα στην ταξινόμηση θα βρίσκονται, για παράδειγμα, τα OSs των “Leverling” που έχουν γνώσεις ή κάποια εμπειρία όσον αφορά τον τομέα των πωλήσεων.

3.4 Μέθοδος ταξινόμησης των αποτελεσμάτων

Στην παρούσα ενότητα θα επεξηγήσουμε λεπτομερώς τη μέθοδο ταξινόμησης των αποτελεσμάτων, δηλαδή των OSs, που θα λαμβάνει ο χρήστης με την εκτέλεση κάθε αναζήτησης. Η ταξινόμηση αποτελεί κεντρικό θέμα στην εν λόγω διπλωματική εργασία γι' αυτό και χρήζει ιδιαίτερης προσοχής. Όπως θα δούμε στη συνέχεια, η συγκεκριμένη μέθοδος θα συνδυάσει τις δύο IR –Style τεχνικές που μελετήσαμε στο προηγούμενο κεφάλαιο, δηλαδή τη μέθοδο stemming και τη μέθοδο TF-IDF, ενώ κύριος πρωταγωνιστής θα είναι η έννοια του Affinity.

3.4.1 Εφαρμογή μεθόδου stemming

Η μέθοδος stemming η οποία, όπως έχουμε δει, μειώνει τη λέξη – κλειδί που δίνεται ως είσοδος στη ρίζα της και τη συγκρίνει με τις λέξεις που περιέχονται σε κάθε έγγραφο, θα αποτελέσει ένα απλό βήμα προεπεξεργασίας στη μέθοδο αυτή.

Κάθε OS μπορεί να θεωρηθεί ως ένα απλό κείμενο με δομημένη ωστόσο μορφή. Οπότε, αυτό που θα κάνουμε είναι να μειώσουμε τη λέξη – κλειδί k_2 που θα δώσει ο χρήστης στη ρίζα της, εφαρμόζοντας τη συγκεκριμένη μέθοδο, και ακολούθως να εξετάσουμε πόσες λέξεις σε κάθε OS ξεχωριστά περιέχουν τη ρίζα αυτή. Τα αποτελέσματα θα χρησιμοποιηθούν, όπως θα δούμε στη συνέχεια, για την εφαρμογή της μεθόδου TF-IDF σε συνδυασμό με την έννοια του Affinity.

Για παράδειγμα, έστω ότι ο χρήστης έχει δώσει ως είσοδο το ακόλουθο Query :

$$Q = \{ \text{“Leverling”}, \text{“Sales”} \}$$

Η λέξη – κλειδί k_2 αντιστοιχεί στη λέξη “Sales”. Συνεπώς, η λέξη αυτή θα μειωθεί στη ρίζα της, δηλ. στο “Sale”, με αφαίρεση του χαρακτήρα ‘s’ και ακολούθως θα υπολογιστεί σε κάθε OS των DSs με το όνομα “Leverling”, πόσες λέξεις περιέχουν τη ρίζα αυτή (π.χ. sale, sales, salesman κτλ.).

Ωστόσο, η υλοποίηση της συγκεκριμένης μεθόδου δεν εμπίπτει στους στόχους της διπλωματικής εργασίας. Για το λόγο αυτό, θα χρησιμοποιηθεί ένας ήδη γνωστός αλγόριθμος για το stemming των λέξεων και συγκεκριμένα ο αλγόριθμος Porter στον οποίο έγινε μία αναφορά στο προηγούμενο κεφάλαιο. Ο αλγόριθμος Porter,

ακολουθεί ένα συγκεκριμένο αριθμό βημάτων για τη μείωση μίας λέξης στη ρίζα της εφαρμόζοντας κάποιους κανόνες, οι οποίοι στηρίζονται στη μορφή της κάθε λέξης.

Στον πίνακα 3.1 βλέπουμε πως έχουν μεταβληθεί κάποιες λέξεις μετά την εφαρμογή του αλγόριθμου Porter.

Αρχική λέξη	Λέξη μετά το stemming
Sales	Sale
Marketing	Market
Commercial	Commerci
Management	Manag
Representative	Repres
International	Intern
Economics	Econom
Selling	Sell
Psychology	Psycholog
Office	Offic
Business	Busi
Administrator	Administr
Assistant	Assist
Accounting	Account
Express	Express
Shipping	Ship

Πίνακας 3.1 Λέξεις μετά την εφαρμογή του αλγορίθμου Porter

3.4.2 Συνδυασμός TF-IDF και Affinity

Προβλέποντας σε ένα καλύτερο τρόπο ταξινόμησης των αποτελεσμάτων, όπου οι πληροφορίες γύρω από τη θεματική ενότητα θα σχετίζονται όσο το δυνατόν περισσότερο με το DS που θα καθοριστεί, θα συνδυάσουμε τις μεθόδους TF – IDF και Affinity οι οποίες έχουν αναλυθεί με ιδιαίτερη λεπτομέρεια στο προηγούμενο κεφάλαιο. Πώς όμως μπορεί να γίνει ένας τέτοιος συνδυασμός δύο εντελώς ανεξάρτητων εννοιών ;

Όπως ήδη γνωρίζουμε, η μέθοδος TF-IDF μπορεί να καθορίσει πόσο σημαντική είναι μία λέξη – κλειδί σε ένα έγγραφο δίνοντας σε αυτή ένα συγκεκριμένο βάρος. Στην προτεινόμενη μεθοδολογία θα εφαρμόσουμε τη μέθοδο TF-IDF σε συνδυασμό με το Affinity, ώστε να δίνει ένα βάρος στη λέξη – κλειδί k_2 του Query για το κάθε OS που θα δημιουργηθεί. Με βάση το βάρος αυτό θα γίνεται και η τελική ταξινόμηση των OSs.

Πιο κάτω, παρουσιάζονται οι εξισώσεις που θα χρησιμοποιηθούν για την εφαρμογή της μεθόδου και τον υπολογισμό του βάρους της λέξης – κλειδί k_2 . Οι εξισώσεις αυτές αντιστοιχούν στις εξισώσεις που έχουν παρουσιαστεί στο προηγούμενο κεφάλαιο για τον υπολογισμό του βάρους $tf-idf$, με διαφορετική ωστόσο ερμηνεία των όρων. Υπενθυμίζουμε, ότι αρχικά εφαρμόζεται η μέθοδος stemming, όπως ακριβώς έχει επεξηγηθεί στην προηγούμενη παράγραφο, και ακολουθεί ο υπολογισμός των πιο κάτω εξισώσεων.

Το term – frequency (tf) της λέξης – κλειδί k_2 σε κάποιο OS, υπολογίζεται ως εξής :

$$(1) \quad tf = \frac{n_{k2}}{\sum_k n_k}$$

, όπου n_{k2} ο αριθμός των λέξεων στο συγκεκριμένο OS που περιέχουν τη ρίζα της λέξης – κλειδί k_2 και ο παρονομαστής το άθροισμα του αριθμού των εμφανίσεων κάθε λέξης k (n_k) που υπάρχει στο OS. Η εξίσωση αυτή αντιστοιχεί στην εξίσωση (1) του προηγούμενου κεφαλαίου. Διαιρώντας με το άθροισμα $\sum_k n_k$, αποφεύγονται περιπτώσεις όπου δίνεται μεγαλύτερο βάρος σε OSs τα οποία είναι απλά μεγάλα σε μέγεθος και πιθανόν να περιέχουν περισσότερες λέξεις με τη ρίζα της λέξης – κλειδί k_2 , ανεξάρτητα από τη σημασία που έχουν πραγματικά οι λέξεις αυτές στο συγκεκριμένο OS.

Κατά τον υπολογισμό του παράγοντα tf της λέξης – κλειδί k_2 , έρχεται να πρωταγωνιστήσει και η έννοια του Affinity. Το Affinity, όπως έχουμε δει, αντιστοιχεί σε μία τιμή από το πεδίο $[0,1]$, η οποία ανατίθεται σε κάθε σχέση του G^{DS} δέντρου που χρησιμοποιείται για τη δημιουργία του OS, ούτως ώστε να καθορίσει σε ποιο βαθμό η συγκεκριμένη σχέση σχετίζεται με τη σχέση που περιέχει το DS,

δηλαδή με τη σχέση που βρίσκεται στη ρίζα του δέντρου G^{DS} και συνεπώς στη ρίζα του OS.

Έστω, λοιπόν, ότι σε κάθε σχέση των G^{DS} δέντρων που θα χρησιμοποιηθούν για τη δημιουργία των OSs, αντιστοιχεί μία προκαθορισμένη τιμή όσον αφορά το Affinity η οποία έχει υπολογιστεί με βάση την εξίσωση (4) του προηγούμενου κεφαλαίου. Τότε, ανάλογα σε ποιά σχέση του OS έχουν παρουσιαστεί λέξεις οι οποίες περιέχουν τη ρίζα της λέξης – κλειδί k_2 , θα πολλαπλασιάσουμε τον αριθμό των λέξεων αυτών στη συγκεκριμένη σχέση με την αντίστοιχη τιμή του Affinity.

Μετονομάζουμε τον όρο tf σε $Af\text{-}tf$ λόγω της προσθήκης του Affinity και η εξίσωση (1) τροποποιείται ως εξής:

$$(2) \quad Af\text{-}tf = \frac{\sum_i (n_{k2,i} * Af_i)}{\sum_k n_k}$$

, όπου ο αριθμητής αντιστοιχεί τώρα στο άθροισμα των γινομένων του αριθμού των λέξεων που περιέχουν τη ρίζα της λέξης – κλειδί k_2 σε κάθε σχέση i ($n_{k2,i}$) του OS με την αντίστοιχη τιμή του Affinity (Af_i).

Λόγω της αλλαγής αυτής στον αριθμητή της εξίσωσης θα μπορούσαμε να τροποποιήσουμε ανάλογα και τον παρονομαστή, ώστε να δίνεται και πάλι περισσότερη έμφαση σε λέξεις που βρίσκονται πιο κοντά στη ρίζα του OS. Έτσι, για κάθε σχέση που λαμβάνει μέρος στο OS, μπορούμε να υπολογίσουμε πόσες συνολικά λέξεις περιέχονται σε αυτή και να πολλαπλασιάσουμε τον αριθμό που θα βρούμε με την αντίστοιχη τιμή του Affinity.

Η εξίσωση που προκύπτει είναι η εξής :

$$(3) \quad Af\text{-}tf = \frac{\sum_i (n_{k2,i} * Af_i)}{\sum_i (n_i * Af_i)}$$

, όπου ο αριθμητής, όπως έχουμε αναφέρει και προηγουμένως, αντιστοιχεί στο άθροισμα των γινομένων του αριθμού των λέξεων που περιέχουν τη ρίζα της λέξης –

κλειδί k_2 σε κάθε σχέση i ($n_{k_2,i}$) του OS με την αντίστοιχη τιμή του Affinity (Af_i), ενώ ο παρονομαστής αντιστοιχεί στο άθροισμα των γινομένων του πλήθους των λέξεων σε κάθε σχέση i (n_i) με την αντίστοιχη τιμή του Affinity (Af_i).

Για λόγους ευκολίας συμβολισμού ορίζουμε ως $Af\text{-}tf$ το αποτέλεσμα της εξίσωσης (3) στην οποία χρησιμοποιείται το Affinity και στον παρονομαστή, και ως $\overline{Af\text{-}tf}$ το αποτέλεσμα της εξίσωσης (2) όπου δεν χρησιμοποιείται το Affinity στον παρονομαστή.

Για να γίνουν πιο κατανοητά όλα αυτά που έχουμε αναφέρει μέχρι εδώ, έστω ότι ο χρήστης έχει δώσει και πάλι ως είσοδο το Query $Q=\{\text{"Leverling"}, \text{"Sales"}\}$ στα πλαίσια της βάσης δεδομένων Northwind, κι έχουν δημιουργηθεί τα OSs του σχήματος 3.2 και 3.3. Όσον αφορά, λοιπόν, το OS του σχήματος 3.2, παρατηρούμε ότι η λέξη – κλειδί “Sales” εμφανίζεται δύο φορές στη σχέση Employees στην οποία ανήκει το DS “Leverling” και μία φορά στη σχέση Customers. Είναι λογικό πως η εμφάνιση της λέξης - κλειδί k_2 στη σχέση στην οποία ανήκει το DS, θα πρέπει να συνεισφέρει περισσότερο στο βάρος που θα της ανατεθεί από ότι εάν εμφανιζόταν σε μία σχέση η οποία δεν σχετίζεται άμεσα με αυτό, όπως στη σχέση Customers. Ακριβώς γι’ αυτό τον λόγο χρησιμοποιείται το Affinity, όπου για το συγκεκριμένο OS θα έχει τη μέγιστη τιμή για τη σχέση Employees, δηλ. 1, ενώ μικρότερη τιμή για τη σχέση Customers. Το ίδιο παρατηρείται και στο OS του σχήματος 3.3. Η εμφάνιση της λέξης - κλειδί “Sales” στη σχέση Employees(ReportsTo) θα πρέπει να συνεισφέρει περισσότερο στο βάρος που θα της ανατεθεί από την εμφάνισή της στη σχέση Customers. Η σχέση Employees(ReportsTo) βρίσκεται πλησιέστερα στη ρίζα του OS και συνεπώς οι πληροφορίες που περιέχει σχετίζονται περισσότερο με το DS που έχει καθοριστεί σε σύγκριση με τις πληροφορίες που προσφέρει η σχέση Customers.

Στους πίνακες 3.2 και 3.3 αναγράφονται όλα τα στοιχεία για τα OSs του σχήματος 3.2 και 3.3 αντίστοιχα, μαζί με τις τιμές $Af\text{-}tf$ και $\overline{Af\text{-}tf}$ που έχουν υπολογιστεί με βάση το Query που έχει εισαχθεί. Όσον αφορά το Affinity, γίνεται χρήση των τιμών που έχουν υπολογιστεί στο προηγούμενο κεφάλαιο για το G^{DS} δέντρο της σχέσης Employees (πίνακας 2.1). Παρατηρούμε πως τα δύο αυτά OSs είναι ισομεγεθή ενώ έχουν επίσης τον ίδιο αριθμό εμφανίσεων της λέξης – κλειδί “Sales”. Ωστόσο, σύμφωνα με τις τιμές $Af\text{-}tf$ και $\overline{Af\text{-}tf}$, φαίνεται να δίνεται μεγαλύτερη έμφαση στο OS

του σχήματος 3.2, το οποίο περιέχει δύο φορές τη λέξη – κλειδί “Sales” στη ρίζα του OS και μία φορά στη σχέση Customers, σε σύγκριση με το OS του σχήματος 3.3 το οποίο περιέχει δύο φορές τη λέξη – κλειδί “Sales” στη σχέση Employee(ReportsTo), η οποία βρίσκεται πιο μακριά από τη ρίζα του OS, και μία φορά στη σχέση Customers.

Employees

EmployeeID	LastName	FirstName	Title	...	Notes
5	Leverling	Maria	Sales Representative	...	Maria graduated from St. Andrews University, Scotland, with a BSC degree in 1976. Upon joining the company as a sales representative in 1992 ...

EmployeeID	TerritoryID
3	30346

EmployeeID	LastName	FirstName	Title	...	Notes
2	Fuller	Robert	Vice President	...	Robert received his BTS commercial in 1974 and a Ph.D. in international marketing from the University of Dallas in 1981.

OrderID	CustomerID	EmployeeID	OrderDate	RequiredDate	...
10251	VICTE	3	8/7/1996	5/8/1996	...

CustomerID	CompanyName	ContactName	ContactTitle	...
VICTE	Victuailles en stock	Mary Saveley	Sales Agent	...

OrderID	ProductID	UnitPrice	...
10251	22	16,80€	...

Σχήμα 3.2 OS που έχει δημιουργηθεί σύμφωνα με το Query Q = {“Leverling”, “Sales”}

Σχέση (i)	Af_i	$n_{k2,i}$	n_i	$n_{k2,i} * Af_i$	$n_i * Af_i$	$Af-tf$	$\overline{Af-tf}$
Employees	1	2	27	2	27	0.0419	0.0401
Employee(ReportsTo)	0.98	0	25	0	24.5		
EmployeeTerritories	0.96	0	2	0	1.92		
Orders	0.9	0	5	0	4.5		
Customers	0.85	1	8	0.85	6.8		
OrderDetails	0.84	0	4	0	3.36		
Σύνολο		3	71	2.85	68.08		

Πίνακας 3.2 Στοιχεία που έχουν ληφθεί από το OS του σχήματος 3.3 για υπολογισμό των τιμών $Af\text{-}tf$ και $\overline{Af\text{-}tf}$ της λέξης – κλειδί “Sales”

Employees

EmployeeID	LastName	FirstName	Title	...	Notes
9	Leverling	Anne	Coordinator	...	Anne has a BA degree in English from St. Lawrence College. She is fluent in French and German.

EmployeeTerritories

EmployeeID	TerritoryID
9	03049

Employee (ReportsTo)

EmployeeID	LastName	FirstName	Title	...	Notes
5	Leverling	Maria	Sales Representative	...	Maria graduated from St. Andrews University, Scotland, with a BSC degree in 1976. Upon joining the company as a sales representative in 1992, she spent 6 months in ...

Orders

OrderID	CustomerID	EmployeeID	OrderDate	RequiredDate	...
10255	RICSU	9	12/7/1996	9/8/1996	...

Customers

CustomerID	CompanyName	ContactName	ContactTitle	...
RICSU	Richter Supermarkt	Michael Holz	Sales Manager	...

OrderDetails

OrderID	ProductID	UnitPrice	...
10255	2	19,00 €	...

Σχήμα 3.3 OS που έχει δημιουργηθεί σύμφωνα με το Query Q = {“Leverling”, “Sales”}

Σχέση (i)	Af_i	$n_{k2,i}$	n_i	$n_{k2,i} * Af_i$	$n_i * Af_i$	$Af\text{-}tf$	$\overline{Af\text{-}tf}$
Employees	1	0	21	0	21	0.0413	0.0395
Employee(ReportsTo)	0.98	2	32	1.96	31.36		
EmployeeTerritories	0.96	0	2	0	1.92		
Orders	0.9	0	5	0	4.5		
Customers	0.85	1	7	0.85	5.95		
OrderDetails	0.84	0	4	0	3.36		
Σύνολο		3	71	2.81	68.09		

Πίνακας 3.3 Στοιχεία που έχουν ληφθεί από το OS του σχήματος 3.4 για υπολογισμό των τιμών $Af\text{-}tf$ και $\overline{Af\text{-}tf}$ της λέξης – κλειδί “Sales”

Συνεχίζουμε με τον παράγοντα idf , ο οποίος υπολογίζεται σύμφωνα με την πιο κάτω εξίσωση:

$$(4) \quad idf = \log (|OSs| / (f_{k2,oss} + 1))$$

, όπου $|OSs|$ ο αριθμός όλων των πιθανών OSs, δηλ. ο συνολικός αριθμός των DSs που υπάρχουν στη βάση δεδομένων, και $f_{k2,oss}$ ο αριθμός των OSs στα οποία εμφανίζονται λέξεις που περιέχουν τη ρίζα της λέξης - κλειδί k_2 . Στον παρονομαστή $f_{k2,oss}$ θα πρέπει να προστεθεί μία επιπλέον μονάδα ώστε να αποφευχθεί η διαίρεση με το 0 σε περίπτωση που τέτοιες λέξεις δεν εμφανίζονται σε κανένα OS. Η εξίσωση αυτή αντιστοιχεί στην εξίσωση (2) του προηγούμενου κεφαλαίου.

Όπως έχει επεξηγηθεί στο προηγούμενο κεφάλαιο, ο παράγοντας idf χρησιμοποιείται ώστε να δίνει μικρό βάρος σε λέξεις - κλειδιά του Query που εμφανίζονται πολύ συχνά και μεγάλο βάρος σε λέξεις - κλειδιά του Query που εμφανίζονται σπάνια για να γίνεται σωστά η ταξινόμηση των αποτελεσμάτων. Ωστόσο, λόγω του ότι στην προτεινόμενη μεθοδολογία χρησιμοποιείται μόνο η λέξη - κλειδί k_2 του Query για την ταξινόμηση των OSs, ο παράγοντας idf μπορεί να αγνοηθεί καθώς δεν πρόκειται να συνεισφέρει κάτι στην ταξινόμηση των αποτελεσμάτων. Σε περίπτωση που χρησιμοποιούσαμε περισσότερες από μία λέξεις - κλειδιά για την ταξινόμηση των OSs, ο παράγοντας idf θα ήταν απαραίτητος ώστε να δοθεί η ανάλογη βαρύτητα στην κάθε λέξη - κλειδί. Τότε, το τελικό βάρος για το OS θα ήταν ίσο με το πιο κάτω άθροισμα :

$$(5) \quad \sum_k (Af \cdot tf_k * idf_k)$$

, όπου k κάθε λέξη που χρησιμοποιείται για την ταξινόμηση των αποτελεσμάτων.

Για παράδειγμα, έστω ότι ο χρήστης έχει δώσει ως είσοδο το ακόλουθο Query στα πλαίσια της βάσης δεδομένων Northwind :

$$Q = \{ \text{“Leverling”}, \{ \text{“Sales Marketing”} \} \}$$

, όπου η λέξη - κλειδί “Leverling” καθορίζει, όπως και προηγουμένως, το DS για το οποίο θα δημιουργηθούν τα OSs, ενώ οι λέξεις - κλειδιά “Sales” και “Marketing” καθορίζουν τη θεματική ενότητα.

Υποθέτουμε ότι λέξεις που περιέχουν τη ρίζα της λέξης – κλειδί “Sales” παρουσιάστηκαν σε 50 από τα OSs που δημιουργήθηκαν για το DS “Leverling” , δηλ. $f_{\text{Sales,oss}} = 50$, ενώ λέξεις που περιέχουν τη ρίζα της λέξης – κλειδί “Marketing” παρουσιάστηκαν μόνο σε 5 OSs, δηλ. $f_{\text{Marketing,oss}}=5$. Ο αριθμός των πιθανών OSs ανέρχεται στα 100, καθώς υπάρχουν συνολικά 9 εγγραφές στη σχέση Employees και 91 εγγραφές στη σχέση Customers.

Επιπρόσθετα, έστω ότι στο σύνολο των OSs που έχουν δημιουργηθεί για το DS “Leverling” περιλαμβάνεται και το OS του σχήματος 3.2. Η τιμή $Af\text{-}tf$ για τη λέξη – κλειδί “Sales” στο OS αυτό ισούται με 0.0419, όπως την έχουμε υπολογίσει προηγουμένως ενώ η τιμή του παράγοντα idf θα είναι ίση με $\log(100/50) = 0.3$. Η λέξη – κλειδί “Marketing” , εμφανίζεται μόνο μία φορά στη σχέση Employees(ReportsTo), οπότεν $Af\text{-}tf = (0.98 * 1) / 68.08 = 0.014$, ενώ η τιμή του παράγοντα idf για τη λέξη αυτή θα είναι ίση με $\log(100/5) = 1.3$. Συνεπώς, το τελικό βάρος που θα ανατεθεί στο OS του σχήματος 3.2 στην περίπτωση αυτή θα ισούται με:

$$\sum_k (Af\text{-}tf_k * idf_k) = (0.0419 * 0.3) + (0.014 * 1.3) = 0.195$$

Παρατηρούμε ότι ο παράγοντας idf δίνει μεγαλύτερη έμφαση στη λέξη – κλειδί “Marketing” ($idf_{\text{Marketing}} = 1.3$) η οποία εμφανίζεται μόνο σε 5 OSs, όπως το έχουμε καθορίσει, και μικρότερη έμφαση στη λέξη – κλειδί “Sales” ($idf_{\text{Sales}} = 0.3$) η οποία εμφανίζεται πιο συχνά, σε 50 OSs.

Έτσι, λοιπόν, αφού δεν τίθεται λόγος να χρησιμοποιηθεί ο παράγοντας idf στην προτεινόμενη μεθοδολογία, η ταξινόμηση του συνόλου των OSs θα βασιστεί ουσιαστικά μόνο στις τιμές $Af\text{-}tf$. Πρώτο στην ταξινόμηση θα βρίσκεται το OS για το οποίο το βάρος της λέξης – κλειδί k_2 , δηλ. το $Af\text{-}tf$, έχει τη μέγιστη τιμή.

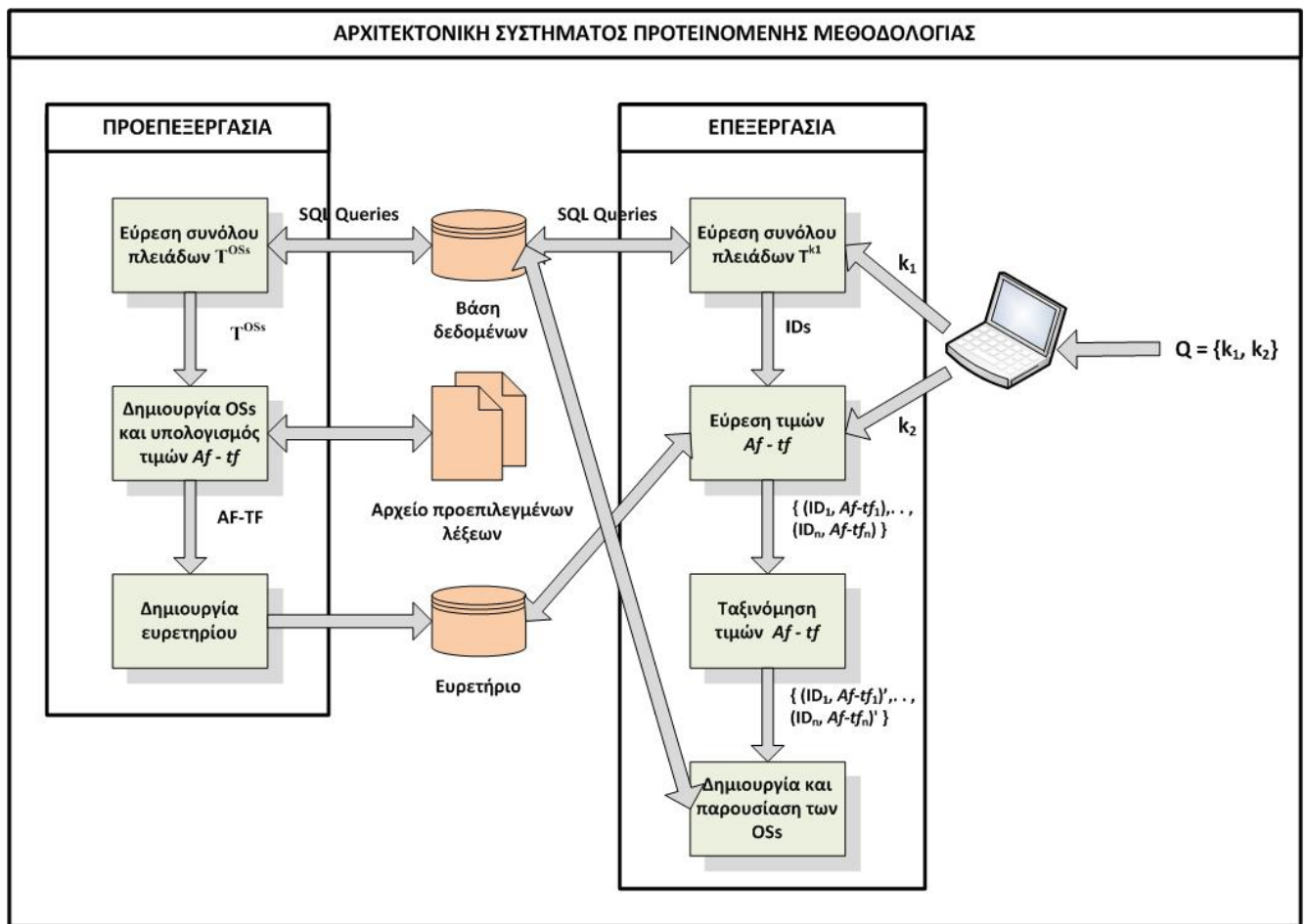
3.5 Αρχιτεκτονική συστήματος προτεινόμενης μεθοδολογίας

Έχοντας κατά νου τους στόχους που έχουμε θέσει, μπορούμε να προχωρήσουμε στην προσεκτική σχεδίαση της αρχιτεκτονικής του συστήματος της προτεινόμενης μεθοδολογίας. Το βήμα αυτό είναι αρκετά σημαντικό και καθοριστικό για τη συνέχεια της εργασίας, καθώς θα κατευθύνει στην ουσία τον τρόπο υλοποίησης του συστήματος. Έτσι, αφενός απαιτείται να ενσωματώσουμε όλη τη λειτουργικότητα και

τα απαιτούμενα χαρακτηριστικά του συστήματος, και αφετέρου υποχρεούμαστε να μην υπερβάλλουμε σε πολυπλοκότητα, αλλά να διατηρούμε εφικτή την υλοποίησή του. Η αρχιτεκτονική που έχει σχεδιαστεί παρουσιάζεται στο σχήμα 3.4.

Όπως βλέπουμε, το σύστημά μας διακρίνεται σε δύο φάσεις, τη φάση προεπεξεργασίας και τη φάση επεξεργασίας. Η φάση προεπεξεργασίας αποσκοπεί στη δημιουργία ενός ευρετηρίου το οποίο θα εξυπηρετήσει το στόχο που έχουμε θέσει για μείωση του χρόνου απόκρισης του συστήματος. Το ευρετήριο αυτό θα περιέχει όλες τις πιθανές τιμές *Af-tf* για όλα τα πιθανά OSs, έτσι ώστε με την εκτέλεση κάθε αναζήτησης οι τιμές *Af-tf* για τα OSs που θα δημιουργηθούν και θα παρουσιαστούν στο χρήστη να μην χρειάζεται να υπολογιστούν την ίδια ώρα από το σύστημα. Για τη δημιουργία του ευρετηρίου, όπως φαίνεται στο σχήμα, εντοπίζονται αρχικά όλες οι πλειάδες (T^{OSs}) από τη βάση δεδομένων που θα χρησιμοποιηθούν για τη δημιουργία όλων των πιθανών OSs ενώ ακολουθεί η δημιουργία των OSs και ο υπολογισμός των τιμών *Af-tf* με χρήση λέξεων από το αρχείο προεπιλεγμένων λέξεων. Οι τιμές αυτές (AF-TF) θα χρησιμοποιηθούν εν συνεχεία για τη δημιουργία του ευρετηρίου. Από την άλλη, η φάση επεξεργασίας περιλαμβάνει τις διάφορες διεργασίες που λαμβάνουν χώρα κάθε φορά που ο χρήστης εκτελεί μία αναζήτηση. Όπως βλέπουμε στο σχήμα 3.4, αφού εισαχθούν οι δύο λέξεις – κλειδιά, τότε γίνεται πρόσβαση στη βάση δεδομένων με χρήση της λέξης – κλειδί k_1 και ανακτώνται όλες οι πλειάδες (T^{k_1}) που περιέχουν τη λέξη – κλειδί k_1 σε κάποιο από τα πεδία τους. Από τις πλειάδες αυτές λαμβάνονται τα πρωτεύοντα κλειδιά (IDs), τα οποία χρησιμοποιούνται μαζί με τη λέξη – κλειδί k_2 του Query για πρόσβαση στο ευρετήριο και την ανάκτηση των τιμών *Af-tf*. Ακολούθως, οι τιμές αυτές ταξινομούνται και με βάση τη σειρά ταξινόμησης δημιουργούνται και παρουσιάζονται τα OSs στον χρήστη.

Μία πιο ουσιαστική περιγραφή και ανάλυση για το κάθε βήμα που εκτελείται σε κάθε μία από τις φάσεις αυτές, δίνεται στις ενότητες που ακολουθούν.



Σχήμα 3.4 Αρχιτεκτονική συστήματος προτεινόμενης μεθοδολογίας

3.6 Φάση προεπεξεργασίας

Στην παρούσα ενότητα περιγράφονται όλα τα βήματα που εκτελούνται κατά τη φάση προεπεξεργασίας του συστήματος, η οποία όπως έχει προαναφερθεί είναι υπεύθυνη για τη δημιουργία ενός ευρετηρίου.

3.6.1 Αρχείο προεπιλεγμένων λέξεων

Καταρχήν, αυτό που πρέπει να εξηγήσουμε είναι το τι ακριβώς περιέχεται στο αρχείο προεπιλεγμένων λέξεων που χρησιμοποιείται για τον υπολογισμό των τιμών $Af - tf$. Το αρχείο αυτό, λοιπόν, περιέχει ορισμένες λέξεις που έχουν επιλεγεί από τη βάση δεδομένων, οι οποίες εμφανίζονται αρκετά συχνά σε αυτή και οι οποίες θα

μπορούσαν να αποτελέσουν τη λέξη – κλειδί k_2 του Query. Δηλαδή, λέξεις οι οποίες απευθύνονται σε μία θεματική ενότητα, βάση της οποίας θα μπορούσε να γίνει η ταξινόμηση των OSs.

Για κάθε μία από τις λέξεις αυτές, όπως θα δούμε στη συνέχεια, θα υπολογιστεί η τιμή $Af-tf$ για κάθε πιθανό OS. Εάν, για παράδειγμα, χρησιμοποιείται η βάση δεδομένων Northwind για την οποία γνωρίζουμε πως υπάρχουν 100 πιθανά OSs και το συγκεκριμένο αρχείο περιέχει 50 λέξεις, τότε θα υπολογιστούν συνολικά 5000 τιμές $Af-tf$, 100 για κάθε λέξη.

Στον πιο κάτω πίνακα, παρουσιάζονται ορισμένες λέξεις που έχουν επιλεγεί από τη βάση δεδομένων Northwind και οι οποίες θα μπορούσαν να περιέχονται στο αρχείο αυτό.

Λέξεις
Sales
Marketing
Commercial
Management
Food
Economics
University
Psychology
Accounting
Shipping
Beverages
Liquids
Delights
Seasoning

Πίνακας 3.4 Λέξεις που έχουν επιλεγεί από τη βάση δεδομένων Northwind

3.6.2 Εύρεση συνόλου πλειάδων T^{OSs}

Η πρώτη διεργασία που εκτελείται κατά τη φάση προεπεξεργασίας, είναι η ανάκτηση από τη βάση δεδομένων των πλειάδων που θα χρησιμοποιηθούν για τη δημιουργία όλων των πιθανών OSs. Ορίζουμε ως T^{OSs} το σύνολο των πλειάδων αυτών.

Το σύνολο T^{OSs} περιέχει, ουσιαστικά, όλες τις πλειάδες των σχέσεων της βάσης δεδομένων που καθορίζουν ένα DS. Στα πλαίσια της βάσης δεδομένων Northwind, για παράδειγμα, θα περιείχε όλες τις πλειάδες των σχέσεων Employees και Customers. Οι πλειάδες του συνόλου αυτού, θα χρησιμοποιηθούν στη συνέχεια για τη δημιουργία των OSs και υπολογισμό των τιμών $Af-tf$.

3.6.3 Δημιουργία OSs και υπολογισμός τιμών $Af-tf$

Η διεργασία δημιουργίας των OSs και υπολογισμού των τιμών $Af-tf$ έχει ως ακολούθως.

Αρχικά, επιλέγεται μία λέξη από το αρχείο προεπιλεγμένων λέξεων στην οποία εφαρμόζεται η μέθοδος stemming, όπως έχει επεξηγηθεί στην παράγραφο 3.4.1. Στη συνέχεια, για κάθε μία από τις πλειάδες του συνόλου T^{OSs} σχηματίζεται το δέντρο του OS όπως ακριβώς έχει επεξηγηθεί στην παράγραφο 2.5.3 του προηγούμενου κεφαλαίου, διαπερνώντας το κατάλληλο G^{DS} δέντρο με κατά πλάτος αναζήτηση. Κατά τον σχηματισμό του κάθε OS υπολογίζεται σε κάθε του σχέση ο συνολικός αριθμός των λέξεων όπως επίσης και ο αριθμός των λέξεων που περιέχουν τη ρίζα της λέξης που έχει επιλεγεί. Ακολούθως, χρησιμοποιώντας τις κατάλληλες τιμές για το Affinity, υπολογίζεται η τιμή $Af-tf$ για το συγκεκριμένο OS και τη συγκεκριμένη λέξη. Η διαδικασία αυτή επαναλαμβάνεται για όλες τις λέξεις που περιέχονται στο αρχείο προεπιλεγμένων λέξεων.

Μετά το πέρας της όλης διαδικασίας θα έχουμε στη διάθεσή μας όλες τις πιθανές τιμές $Af-tf$ για όλα τα πιθανά OSs, έστω το σύνολο AF-TF, όπως ακριβώς το επιθυμούμε.

3.6.4 Δημιουργία ευρετηρίου

Τελευταία διεργασία που απομένει να εκτελεστεί είναι η δημιουργία του ευρετηρίου.

Κάθε τιμή του συνόλου AF-TF αποθηκεύεται στο ευρετήριο, το οποίο μπορεί να είναι οποιασδήποτε μορφής (π.χ βάση δεδομένων, αρχείο κτλ), μαζί με το πρωτεύον κλειδί του DS για το OS του οποίου υπολογίστηκε και την αντίστοιχη λέξη που χρησιμοποιήθηκε από το αρχείο των προεπιλεγμένων λέξεων.

Ένα ευρετήριο που έχει δημιουργηθεί στα πλαίσια της βάσης δεδομένων Northwind, παρουσιάζεται στο σχήμα 3.5. Για κάθε πιθανό OS, δηλ. για κάθε Customer και Employee, αποθηκεύεται το πρωτεύον κλειδί (ID) , η λέξη που έχει χρησιμοποιηθεί από το αρχείο και η αντίστοιχη τιμή *Af-tf*.

CustomerID	Λέξη	<i>Af-tf</i>
ALFKI	Sales	0.0262
ALFKI	Marketing	0.0065
ALFKI	Commercial	0.0028
ALFKI	Management	0.0122
.....		
ANATR	Sales	0.0241
ANATR	Marketing	0.0547
ANATR	Commercial	0.0022
ANATR	Management	0.0098
.....		
BERGS	Sales	0.025
BERGS	Marketing	0.0065
BERGS	Commercial	0.0025
BERGS	Management	0.0121
.....		
EmployeeID	Λέξη	<i>Af-tf</i>
1	Sales	0.096
1	Marketing	0.007
1	Commercial	0.00006
1	Management	0.0088
.....		
2	Sales	0.009

2	Marketing	0.0072
2	Commercial	0.00008
2	Management	0.0089
.....		
3	Sale	0.0099
3	Marketing	0.0062
3	Commercial	0.00006
3	Management	0.008
.....		

Σχήμα 3.5 Ευρετήριο που έχει δημιουργηθεί στα πλαίσια της βάσης δεδομένων Northwind

3.7 Φάση επεξεργασίας

Στην ενότητα αυτή περιγράφονται όλες οι διεργασίες της φάσης επεξεργασίας του συστήματος, οι οποίες λαμβάνουν χώρα κάθε φορά που ο χρήστης εκτελεί μία αναζήτηση.

3.7.1 Εύρεση συνόλου πλειάδων T^{k_1}

Η πρώτη διεργασία που θα εκτελέσει το σύστημα μετά την εισαγωγή του Query από τον χρήστη, είναι η εύρεση των πλειάδων από τη βάση δεδομένων που περιέχουν το DS, δηλαδή τη λέξη – κλειδί k_1 , σε κάποιο από τα πεδία τους. Έστω T^{k_1} το σύνολο των πλειάδων αυτών. Η αναζήτηση μπορεί να περιοριστεί σε συγκεκριμένες σχέσεις και πεδία της βάσης δεδομένων που χρησιμοποιείται, όπως πρωτεύοντα κλειδιά και ονόματα, ώστε να αποφευχθούν περιπτώσεις όπου η λέξη - κλειδί k_1 εμφανίζεται σε πεδία που περιέχουν ασήμαντα δεδομένα, όπως σχόλια, περιγραφές κτλ. Σε μία τέτοια περίπτωση, θα δημιουργούνταν OSs που δεν θα είχαν κάποιο ιδιαίτερο νόημα για τον χρήστη.

Όσον αφορά τη βάση δεδομένων Northwind, η αναζήτηση περιορίζεται στις εξής σχέσεις και πεδία:

- Σχέση Employees:
 - ✓ LastName
 - ✓ FirstName
- Σχέση Customers:

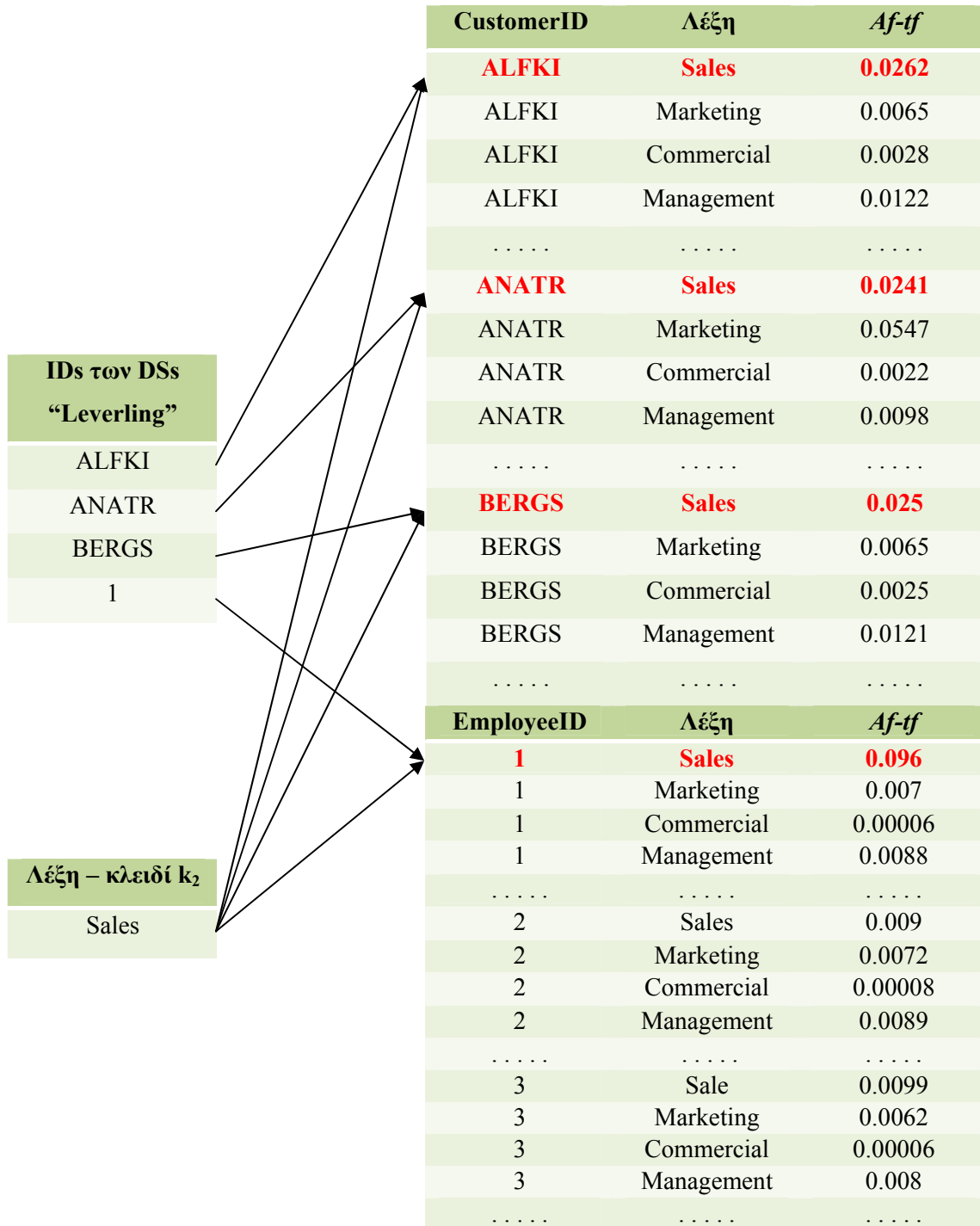
- ✓ CustomerID
- ✓ ContactName

Από τις πλειάδες του συνόλου T^{k_1} ανακτώνται τα πρωτεύοντα κλειδιά (IDs), τα οποία ταυτοποιούν ουσιαστικά το κάθε OS που πρόκειται να δημιουργηθεί και να παρουσιαστεί στον χρήστη.

3.7.2 Εύρεση τιμών $Af-tf$

Μετά την εύρεση των πλειάδων και την ανάκτηση των πρωτεύοντων κλειδιών, ακολουθεί η εύρεση των τιμών $Af-tf$ στις οποίες θα βασιστεί η ταξινόμηση των OSs. Οι τιμές αυτές, όπως ήδη έχουμε αναφέρει, δεν θα υπολογίζονται από το σύστημα με την εκτέλεση κάθε αναζήτησης αλλά θα ανακτώνται από το ευρετήριο που θα δημιουργηθεί κατά τη φάση προεπεξεργασίας.

Η πρόσβαση στο ευρετήριο με σκοπό την εύρεση των τιμών $Af - tf$ θα γίνεται με χρήση των πρωτεύοντων κλειδιών IDs και τη λέξη – κλειδί k_2 που έχει εισαγάγει ο χρήστης. Η διαδικασία αυτή παρουσιάζεται στο σχήμα 3.6, στην περίπτωση που έχει εισαχθεί το Query $Q = \{\text{“Leverling”}, \text{“Sales”}\}$ στα πλαίσια και πάλι της βάσης δεδομένων Northwind. Αφού ανακτηθούν τα IDs των DSs “Leverling”, που μπορεί να είναι είτε Customers είτε Employees, τότε με βάση το κάθε ID και τη λέξη – κλειδί “Sales”, πραγματοποιείται πρόσβαση στο ευρετήριο και λαμβάνονται οι αντίστοιχες τιμές $Af-tf$.



Σχήμα 3.6 Διαδικασία ευρετηρίασης στα πλαίσια της βάσης δεδομένων Northwind στην περίπτωση που έχει εισαχθεί το Query $Q = \{\text{"Leverling"}, \text{"Sales"}\}$

3.7.3 Ταξινόμηση τιμών $Af-tf$

Μετά τον εντοπισμό όλων των τιμών $Af-tf$ για τα OSs που θα δημιουργηθούν, ακολουθεί η διαδικασία ταξινόμησής τους.

Συγκεκριμένα, οι τιμές αυτές θα ταξινομηθούν κατά φθίνουσα σειρά έτσι ώστε το πρώτο OS που θα παρουσιαστεί στον χρήστη να έχει τη μέγιστη τιμή $Af-tf$, όπως ακριβώς το επιθυμούμε.

3.7.4 Δημιουργία και παρουσίαση των OSs

Τελευταίο βήμα που απομένει να εκτελεστεί είναι η δημιουργία και η παρουσίαση των OSs στον χρήστη.

Σύμφωνα, λοιπόν, με το ID με το οποίο συσχετίζεται η κάθε τιμή $Af-tf$ που ανακτήθηκε από το ευρετήριο, επιλέγεται το κατάλληλο G^{DS} δέντρο, το οποίο διαπερνάται με κατά πλάτος αναζήτηση όπως ακριβώς περιγράφεται στην παράγραφο 2.5.3 του προηγούμενου κεφαλαίου, για τη δημιουργία του OS. Εάν για παράδειγμα το ID απευθύνεται σε κάποιο Employee στα πλαίσια της βάσης δεδομένων Northwind, τότε το OS θα δημιουργηθεί με βάση το G^{DS} δέντρο της σχέσης Employees ενώ εάν απευθύνεται σε κάποιο Customer, τότε το OS θα δημιουργηθεί με βάση το G^{DS} δέντρο της σχέσης Customers. Στη ρίζα του OS θα βρίσκεται η πλειάδα με πρωτεύον κλειδί το ID.

Αφού δημιουργηθούν όλα τα OSs, τότε θα παρουσιαστούν στον χρήστη σε μία όσο το δυνατόν πιο κατανοητή μορφή.

3.8 Συμπεράσματα

Στο κεφάλαιο αυτό, λοιπόν, δόθηκε μία ουσιαστική περιγραφή της προτεινόμενης μεθοδολογίας. Επεξηγήθηκε εκτενώς η μέθοδος ταξινόμησης των αποτελεσμάτων που θα λαμβάνονται με την εκτέλεση κάθε αναζήτησης, η οποία μέθοδος συνδυάζει IR-Style τεχνικές μαζί με την έννοια του Affinity. Ακολούθως, παρουσιάστηκε η αρχιτεκτονική του συστήματος της προτεινόμενης μεθοδολογίας, η οποία διακρίνεται σε δύο φάσεις, τη φάση προεπεξεργασίας και τη φάση επεξεργασίας. Για κάθε μία

από τις φάσεις αυτές δόθηκε εν συνεχεία μία αναλυτική περιγραφή των διάφορων διεργασιών που εκτελούν έτσι ώστε να γίνει κατανοητός ο τρόπος λειτουργίας του συστήματος.

Κεφάλαιο 4

Υλοποίηση συστήματος προτεινόμενης μεθοδολογίας

4.1 Εισαγωγή	49
4.2 Επιλογή γλώσσας προγραμματισμού και εργαλείου ανάπτυξης	49
4.3 Βάση δεδομένων Northwind	51
4.4 Επιλογή συστήματος διαχείρισης βάσεων δεδομένων	53
4.5 Λεπτομέρειες υλοποίησης	53
4.5.1 Υλοποίηση φάσης προεπεξεργασίας	53
4.5.2 Υλοποίηση φάσης επεξεργασίας	55
4.6 Πρωτότυπο συστήματος	57
4.7 Συμπεράσματα	61

4.1 Εισαγωγή

Στο κεφάλαιο αυτό περιγράφονται τα βασικά χαρακτηριστικά της υλοποίησης του συστήματός μας, όπως η γλώσσα προγραμματισμού και το εργαλείο ανάπτυξης που επιλέξαμε, η βάση δεδομένων Northwind στα πλαίσια της οποίας υλοποιήθηκε το σύστημα καθώς επίσης και το σύστημα διαχείρισης βάσεων δεδομένων που χρησιμοποιήσαμε. Στη συνέχεια, παρουσιάζονται αρκετές λεπτομέρειες όσον αφορά τη διαδικασία υλοποίησης μαζί με screenshots που λήφθηκαν από το πρωτότυπο του συστήματος.

4.2 Επιλογή γλώσσας προγραμματισμού και εργαλείου ανάπτυξης

Μία από τις πρώτες αποφάσεις που έπρεπε να ληφθούν ήταν η επιλογή της γλώσσας προγραμματισμού και του εργαλείου ανάπτυξης που θα χρησιμοποιούσαμε για την υλοποίηση του συστήματός μας. Επιλέξαμε τη γλώσσα προγραμματισμού Java και

το εργαλείο ανάπτυξης NetBeans IDE, λαμβάνοντας υπόψη τα διάφορα πλεονεκτήματα που μπορούν να μας προσφέρουν.

Η γλώσσα Java συνοδεύεται από ένα πλήθος έτοιμων κλάσεων και μεθόδων που μειώνουν σημαντικά την προγραμματιστική προσπάθεια. Το σημαντικότερο είναι ότι η πρόσβαση σε βάσεις δεδομένων με τη χρήση Java προγραμμάτων μπορεί να γίνει με πολύ απλό και ευέλικτο τρόπο, χρησιμοποιώντας το JDBC (Java DataBase Connectivity API) driver. Η διευκόλυνση αυτή κρίνεται απαραίτητη ειδικά στην περίπτωση μας, όπου το σύστημα σχετίζεται άμεσα με τη χρήση βάσεων δεδομένων.

Επιπρόσθετα, ένα από τα βασικότερα πλεονεκτήματα της Java έναντι των υπόλοιπων γλωσσών προγραμματισμού είναι το χαρακτηριστικό γνώρισμα "write once, run everywhere" που συναντάται μόνο σε αυτή τη γλώσσα. Τα προγράμματα Java γράφονται μία φορά και μπορούν να τρέξουν σε οποιοδήποτε λειτουργικό σύστημα χωρίς να χρειαστεί να ξαναγίνει μεταγλώττιση ή να αλλάξει ο πηγαίος κώδικας για κάθε διαφορετικό λειτουργικό σύστημα. Το εκάστοτε JVM (Java Virtual Machine) αναλαμβάνει να μετατρέψει τα Java bytecodes, δηλαδή τον μεταφρασμένο (compiled) κώδικα, σε εκτελέσιμο.

Από την άλλη, το εργαλείο NetBeans IDE παρέχει ένα πλήρες περιβάλλον ανάπτυξης εφαρμογών. Είναι γραμμένο σε Java αλλά μπορεί να υποστηρίξει όλες τις γλώσσες προγραμματισμού. Το περιβάλλον αυτό προσφέρει μία πληθώρα βοηθημάτων στον προγραμματιστή τα οποία κάνουν ευκολότερη και αποδοτικότερη την ανάπτυξη των εφαρμογών. Τα βασικότερα βοηθήματα αφορούν κυρίως τον Source Code Editor (επισήμανση συντακτικού στον κώδικα, live επισήμανση λαθών και live parsing, pop up παράθυρα βοήθειας κτλ.). Ωστόσο εξίσου σημαντικά είναι και τα βοηθήματα που παρέχει για την κατασκευή ενός GUI, τα οποία θα μας βοηθήσουν να δημιουργήσουμε ένα πρωτότυπο του συστήματός μας, αλλά και για τη διαχείριση βάσεων δεδομένων. Θα πρέπει επίσης να αναφέρουμε ότι το εργαλείο NetBeans IDE έχει εγκατεστημένο τον JDBC - ODBC Bridge driver που χρησιμοποιείται για την επικοινωνία με βάσεις δεδομένων.

Όλα αυτά συνηγορούν στην επιλογή της Java ως γλώσσας προγραμματισμού και του NetBeans IDE ως εργαλείο ανάπτυξης του συστήματος.

4.3 Βάση δεδομένων Northwind

Όπως έχουμε ήδη αναφέρει σε προηγούμενα κεφάλαια, για την υλοποίηση και συνεπώς την αξιολόγηση του συστήματος, έχουμε επιλέξει να χρησιμοποιήσουμε τη βάση δεδομένων Northwind, της οποίας το μέγεθος κυμαίνεται μέσα σε λογικά πλαίσια (3.7MB). Αν και δεν είναι ιδιαίτερα μεγάλη σε μέγεθος σε σχέση με άλλες βάσεις δεδομένων που υπάρχουν όπως η TPC – H (1 GB), περιέχει αρκετά δεδομένα τα οποία θα μπορούν να μας δώσουν ικανοποιητικά αποτελέσματα.

Η βάση δεδομένων Northwind αφορά εμπορικές συναλλαγές και οι σχέσεις από τις οποίες αποτελείται συνοψίζονται στον πιο κάτω πίνακα.

Σχέση	Σύντομη περιγραφή
❖ <i>Categories</i>	➤ κατηγορίες των Northwind προϊόντων
❖ <i>Customers</i>	➤ ονόματα πελατών, διευθύνσεις, τηλέφωνα κτλ.
❖ <i>Employees</i>	➤ ονόματα υπαλλήλων, τίτλοι και προσωπικές πληροφορίες
❖ <i>Order Details</i>	➤ λεπτομέρειες για τα προϊόντα, ποσότητες και τιμές για κάθε παραγγελία κτλ.
❖ <i>Orders</i>	➤ όνομα πελάτη, ημερομηνία παραγγελίας, κόστος κτλ.
❖ <i>Products</i>	➤ ονόματα προϊόντων, προμηθευτές, τιμές, μονάδες αποθέματος κτλ.
❖ <i>Shippers</i>	➤ ονόματα εταιρειών αποστολής των προϊόντων και αριθμοί τηλεφώνων
❖ <i>Suppliers</i>	➤ ονόματα προμηθευτών, διευθύνσεις, αριθμοί τηλεφώνων, υπερσυνδέσμοι σε ιστοσελίδες κτλ.
❖ <i>Territories</i>	➤ ονόματα και ταυτότητες περιοχών
❖ <i>Employee Territories</i>	➤ ταυτότητα υπαλλήλου και περιοχή που έχει αναλάβει

DSs. Είναι λογικό πως εάν δεν υπάρχουν DSs στη βάση δεδομένων με το ίδιο όνομα, τότε σίγουρα δεν θα είναι δυνατόν να δημιουργηθεί ένα σύνολο από OSs και πόσο μάλλον να ταξινομηθεί. Οπότε, αλλάξαμε ονόματα και επίθετα μερικών Customers και Employees έτσι ώστε να υπάρχουν ορισμένες συνωνυμίες και να καταστεί δυνατός ο έλεγχος των αποτελεσμάτων του συστήματος.

4.4 Επιλογή συστήματος διαχείρισης βάσεων δεδομένων

Όσον αφορά το σύστημα διαχείρισης βάσεων δεδομένων που θα χρησιμοποιούσαμε υπήρχαν στην ουσία δύο επιλογές, ο Microsoft SQL Server και η Microsoft Office Access. Λόγω του ότι η βάση δεδομένων Northwind, όπως αναφέραμε και προηγουμένως, δεν είναι ιδιαίτερα μεγάλη σε μέγεθος, η τελική μας επιλογή ήταν η Microsoft Office Access που βρισκόταν ήδη εγκατεστημένη στον υπολογιστή. Η Microsoft Office Access φαίνεται να είναι ιδιαίτερα εύχρηστη και αποδοτική για επεξεργασία μερικών mega byte δεδομένων. Από την άλλη, ο Microsoft SQL Server προτιμάται κυρίως σε περιπτώσεις όπου οι βάσεις δεδομένων είναι μεγάλου μεγέθους και υπάρχει πρόσβαση σε αυτές από πολλούς χρήστες ταυτόχρονα.

4.5 Λεπτομέρειες υλοποίησης

Στην παρούσα ενότητα επεξηγούμε την διαδικασία που ακολουθήθηκε για την υλοποίηση του συστήματος της προτεινόμενης μεθοδολογίας.

4.5.1 Υλοποίηση φάσης προεπεξεργασίας

Πρώτο μας μέλημα ήταν η υλοποίηση της φάσης προεπεξεργασίας του συστήματος για τη δημιουργία του ευρετηρίου. Έτσι, προχωρήσαμε στην υλοποίηση ενός προγράμματος σε γλώσσα προγραμματισμού Java και με χρήση του εργαλείου NetBeans IDE, για τον υπολογισμό όλων των τιμών $Af-tf$ που θα χρειαζόμασταν.

Αρχικά, έπρεπε να επιλέξουμε από τη βάση δεδομένων Northwind ένα σύνολο λέξεων οι οποίες μπορούσαν να καθορίσουν μία θεματική ενότητα. Συνολικά επιλέξαμε 72 λέξεις τις οποίες παρουσιάζουμε στο παράρτημα Β. Οι λέξεις αυτές

τοποθετήθηκαν εν συνεχεία σε ένα αρχείο, το οποίο αντιστοιχούσε στο αρχείο προεπιλεγμένων λέξεων που περιγράψαμε στην ενότητα 3.6.1 του προηγούμενου κεφαλαίου, έτσι ώστε να είναι εύκολα και άμεσα προσβάσιμες από το πρόγραμμα.

Το πρώτο ζήτημα που προέκυψε κατά την υλοποίηση του προγράμματος, ήταν ο τρόπος εγκαθίδρυσης επικοινωνίας με τη βάση δεδομένων Northwind. Λόγω της ύπαρξης του JDBC – ODBC Bridge driver στο εργαλείο NetBeans IDE, η διαδικασία αυτή μπορούσε να επιτευχθεί με ένα πολύ μικρό αριθμό βημάτων. Συγκεκριμένα, γινόταν αρχικά φόρτωση του JDBC – ODBC Bridge driver με κλήση της *Class.forName* και ακολουθούσε ο καθορισμός του URL της βάσης δεδομένων Northwind. Το URL αυτό, το περνούσαμε ως παράμετρο κατά την κλήση της μεθόδου *getConnection* της κλάσης *DriverManager* και η επικοινωνία με τη βάση δεδομένων Northwind ήταν πλέον γεγονός.

Επομένως, είχαμε τη δυνατότητα ανάκτησης πλειάδων με εκτέλεση των κατάλληλων SQL Queries. Όλα τα SQL queries μπορούσαμε να τα δηλώσουμε υπό τη μορφή ενός *String* και εκτελούνταν με κλήση της κατάλληλης μεθόδου της κλάσης *Statement*. Τα αποτελέσματα κάθε εκτέλεσης αποθηκεύονταν σε αντικείμενα τύπου *ResultSet*, που μας παρείχαν διάφορες μεθόδους διαχείρισης των εν λόγω αποτελεσμάτων.

Για να υπολογίσουμε τις τιμές *Af-tf* ακολουθήσαμε βήμα προς βήμα τη διαδικασία που περιγράφεται στην ενότητα 3.6.3 του προηγούμενου κεφαλαίου. Οι λέξεις που τοποθετήσαμε στο αρχείο διαβάστηκαν και αποθηκεύτηκαν σε ένα *ArrayList*, γεγονός που επέτρεψε την εύκολη διαχείρισή τους. Για την εφαρμογή της μεθόδου stemming δημιουργήσαμε την κλάση *Stemmer*, η οποία περιείχε μεθόδους που υλοποιούσαν τα διάφορα βήματα του αλγόριθμου Porter. Θα πρέπει να αναφέρουμε πως λόγω του ότι ο αλγόριθμος Porter λειτουργούσε μόνο για λέξεις που άρχιζαν με μικρό γράμμα έγιναν όλες οι απαραίτητες τροποποιήσεις έτσι ώστε να μπορεί να εφαρμοστεί και σε λέξεις που ξεκινούσαν με κεφαλαίο γράμμα. Για τη δημιουργία των OSs και τον υπολογισμό των τιμών *Af-tf* όλων των Employees, ήταν υπεύθυνη η μέθοδος *AfTf_Calculation* της κλάσης *Emp_OS*, ενώ για τη δημιουργία των OSs και τον υπολογισμό των τιμών *Af-tf* όλων των Customers ήταν υπεύθυνη η αντίστοιχη μέθοδος της κλάσης *Cus_OS*. Όσον αφορά το Affinity, χρησιμοποιήθηκαν οι τιμές που υπολογίστηκαν στο κεφάλαιο 2 (πίνακας 2.1 και πίνακας 2.2), τις οποίες δηλώσαμε ως σταθερές.

Η εκάστοτε τιμή $Af\text{-}tf$ που υπολογιζόταν στις μεθόδους $AfTf_Calculation$, αποθηκευόταν σε ένα *HashMap*. Στο σημείο αυτό θα πρέπει να αναφέρουμε, πως κάθε τιμή που προστίθεται σε ένα *HashMap* συσχετίζεται πάντα με ένα κλειδί. Εάν επιθυμείς να ανακτήσεις τη συγκεκριμένη τιμή τότε θα την εντοπίσεις με βάση το κλειδί με το οποίο την συσχέτισες. Στην προκειμένη περίπτωση το ρόλο του κλειδιού για κάθε τιμή $Af\text{-}tf$ τον διαδραμάτιζε το ID, ανάλογα είτε του Customer είτε του Employee για το OS του οποίου υπολογίστηκε η συγκεκριμένη τιμή, μαζί με τη λέξη που χρησιμοποιήθηκε από το *ArrayList*.

Μετά το πέρας της όλης διαδικασίας, τα αποτελέσματα για όλους τους Employees και για όλους τους Customers, δηλαδή τα περιεχόμενα των δύο *HashMaps* που χρησιμοποιήθηκαν στις δύο μεθόδους $AfTf_Calculation$, καταγράφηκαν όλα σε ένα αρχείο με το όνομα *Index*. Το αρχείο αυτό αντιστοιχούσε ουσιαστικά στο ευρετήριο μας, το οποίο μπορούσε πλέον να μεταφερθεί και χρησιμοποιηθεί άμεσα από το σύστημα που θα υλοποιούσαμε.

Επιπρόσθετα, θα πρέπει να αναφέρουμε ότι υλοποιήθηκαν ουσιαστικά δύο παραλλαγές του συγκεκριμένου προγράμματος, όπου η μία υπολόγιζε τις τιμές $Af\text{-}tf$ με χρήση του Affinity και στον παρονομαστή, δηλαδή εφαρμόζε την εξίσωση (3) του προηγούμενου κεφαλαίου, ενώ η δεύτερη υπολόγιζε τις τιμές $\overline{Af\text{-}tf}$ χωρίς χρήση του Affinity στον παρονομαστή, δηλαδή εφαρμόζε την εξίσωση (2) του προηγούμενου κεφαλαίου. Ένα μέρος των δύο αρχείων που λήφθηκαν παρουσιάζονται στο παράρτημα Β.

4.5.2 Υλοποίηση φάσης επεξεργασίας

Μετά την υλοποίηση της φάσης προεπεξεργασίας για τη δημιουργία ευρετηρίου, μπορούσαμε να προχωρήσουμε στην υλοποίηση της φάσης επεξεργασίας.

Θα πρέπει να αναφέρουμε, ότι το σύστημά μας υλοποιήθηκε με τέτοιο τρόπο έτσι ώστε η ταξινόμηση των αποτελεσμάτων να μπορεί να γίνει είτε σύμφωνα με τις τιμές $Af\text{-}tf$, είτε με βάση τις τιμές $\overline{Af\text{-}tf}$. Ο χρήστης καλείται να επιλέξει πριν την εκτέλεση της αναζήτησής του τη μέθοδο ταξινόμησης που επιθυμεί. Η διαφορά μεταξύ των δύο μεθόδων, όσον αφορά το θέμα της υλοποίησης, έγκειται απλά στο ευρετήριο που χρησιμοποιείται. Δηλαδή, υπάρχει ανάλογα πρόσβαση είτε στο αρχείο/ευρετήριο που

λήφθηκε από την παραλλαγή του προγράμματος της φάσης προεπεξεργασίας το οποίο λάμβανε υπόψη το Affinity και στον παρονομαστή της εξίσωσης, είτε στο αρχείο/ευρετήριο που λήφθηκε από την παραλλαγή εκείνη του εν λόγω προγράμματος που δεν εφάρμοσε την έννοια του Affinity στον παρονομαστή.

Ακολούθως, το σύστημά μας δίνει τη δυνατότητα στον χρήστη να εισαγάγει δύο λέξεις – κλειδιά για να εκτελέσει την αναζήτησή του, όπως ορίζει η προτεινόμενη μεθοδολογία. Όσον αφορά την περάτωση της πρώτης διεργασίας, δηλ. την εύρεση των πλειάδων από τη βάση δεδομένων Northwind που περιέχουν τη πρώτη λέξη – κλειδί σε κάποιο από τα πεδία τους, εκτελούνται τα κατάλληλα SQL queries και τα αποτελέσματα, δηλαδή οι πλειάδες που ανακτώνται, αποθηκεύονται σε αντικείμενα τύπου *ResultSet*. Για την εκτέλεση των SQL queries και την ανάκτηση των πλειάδων γίνεται φυσικά πρώτα εγκαθίδρυση επικοινωνίας με τη βάση δεδομένων Northwind, με τον τρόπο που έχει επεξηγηθεί στην προηγούμενη παράγραφο. Από τις πλειάδες που ανακτώνται κατά τη συγκεκριμένη διεργασία, λαμβάνονται τα πρωτεύοντα κλειδιά (IDs), τα οποία τοποθετούνται σε ένα *ArrayList*.

Τα επόμενα βήματα που εκτελεί το σύστημά μας είναι η εύρεση των τιμών *Af-tf* από το κατάλληλο ευρετήριο για τα OSs που πρόκειται να δημιουργηθούν, και η ταξινόμησή τους. Στο σημείο αυτό, θα πρέπει να αναφέρουμε πως το περιεχόμενο του αρχείου/ευρετηρίου που χρησιμοποιείται σύμφωνα με την επιλογή του χρήστη για τη μέθοδο ταξινόμησης, διαβάζεται από το σύστημα και αποθηκεύεται κατάλληλα σε ένα *HashMap* για ευκολία πρόσβασης. Για την υλοποίηση των διεργασιών εύρεσης και ταξινόμησης των τιμών *Af-tf* είναι υπεύθυνες οι δύο μέθοδοι της κλάσης *Ranker* οι οποίες περιγράφονται πιο κάτω:

```
void find_AfTf(ArrayList l1, HashMap index , String keyword2):
```

Η μέθοδος αυτή εκτελεί τη διεργασία για εύρεση των τιμών *Af-tf* από το ευρετήριο. Δέχεται ως παραμέτρους εισόδου το *ArrayList* με τα IDs, το *HashMap* με τα περιεχόμενα του αρχείου/ευρετηρίου, και τη δεύτερη λέξη – κλειδί που έχει δώσει ως είσοδο ο χρήστης. Η συγκεκριμένη μέθοδος εντοπίζει όλες τις τιμές *Af-tf* για τα OSs που θα παρουσιαστούν στο χρήστη από το συγκεκριμένο *HashMap*, χρησιμοποιώντας κάθε φορά ως κλειδί πρόσβασης ένα από τα IDs του *ArrayList* μαζί με τη δεύτερη λέξη – κλειδί. Η τιμή *Af-tf* που λαμβάνεται σε κάθε περίπτωση αποθηκεύεται σε ένα δεύτερο *HashMap* με κλειδί το ID που χρησιμοποιήθηκε από το *ArrayList*.

`LinkedHashMap sortHashMapByValuesD(HashMap passedMap)` : Η μέθοδος αυτή είναι υπεύθυνη για την εκτέλεση της διεργασίας ταξινόμησης των τιμών *Af-tf* των OSs που θα παρουσιαστούν στο χρήστη. Δέχεται ως παράμετρο εισόδου το *HashMap* στο οποίο αποθηκεύτηκαν οι συγκεκριμένες τιμές και επιστρέφει μία συνδεδεμένη λίστα *LinkedHashMap* που τις περιέχει ταξινομημένες με τα αντίστοιχα κλειδιά τους κατά φθίνουσα σειρά. Όπως έχουμε αναφέρει και στο προηγούμενο κεφάλαιο, η φθίνουσα σειρά ταξινόμησης επιλέγεται έτσι ώστε το πρώτο OS που θα παρουσιαστεί στον χρήστη να έχει τη μέγιστη τιμή *Af-tf*.

Τελευταία διεργασία που εκτελεί το σύστημά μας είναι η παρουσίαση των αποτελεσμάτων, δηλαδή των OSs, στον χρήστη. Υπεύθυνες για τη διεργασία αυτή είναι οι μέθοδοι *createEmployeeOS* και *createCustomerOS* της κλάσης *OS*. Για κάθε τιμή *Af-tf* που βρίσκεται αποθηκευμένη στη συνδεδεμένη λίστα (*LinkedHashMap*), λαμβάνεται το κλειδί με το οποίο συσχετίστηκε δηλ. το ID είτε ενός *Employee* είτε ενός *Customer*, και καλείται ανάλογα η κατάλληλη μέθοδος. Οι δύο αυτές μέθοδοι, των οποίων η διαφορά έγκειται στον τρόπο σχηματισμού και παρουσίασης του OS, έχουν την εξής μορφή:

`void createEmployeeOS(String id, AttributedTextArea area)`: Η μέθοδος αυτή δέχεται ως παραμέτρους εισόδου το πρωτεύον κλειδί (*id*) ενός *Employee* για τον οποίο θα σχηματιστεί το OS και την περιοχή στην οποία θα παρουσιαστεί το συγκεκριμένο OS. Ο σχηματισμός και η παρουσίαση του OS θα γίνει σύμφωνα με το G^{DS} δέντρο της σχέσης *Employees*.

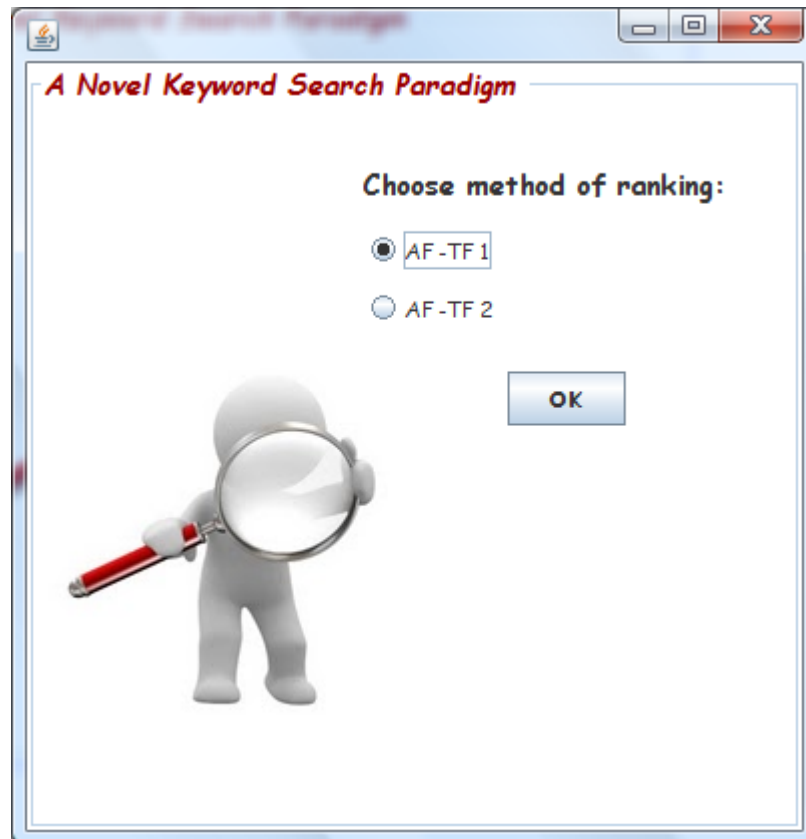
`void createCustomerOS(String id, AttributedTextArea area)`: Η ομοιότητα της μεθόδου αυτής με τη μέθοδο *createEmployeeOS* είναι προφανής. Δέχεται ως παραμέτρους εισόδου το πρωτεύον κλειδί (*id*) ενός *Customer* για τον οποίο θα σχηματιστεί το OS και την περιοχή στην οποία θα παρουσιαστεί το συγκεκριμένο OS. Ο σχηματισμός και η παρουσίαση του OS θα γίνει σύμφωνα με το G^{DS} δέντρο της σχέσης *Customers*.

4.6 Πρωτότυπο συστήματος

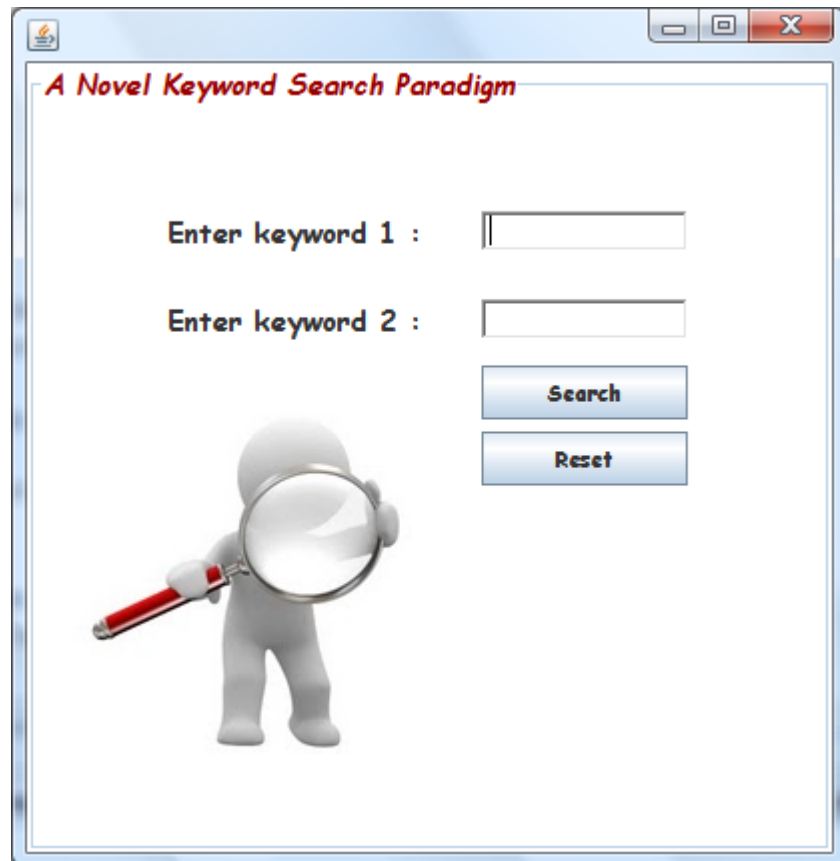
Ένας από τους λόγους που επιλέξαμε να χρησιμοποιήσουμε το εργαλείο NetBeans IDE ως εργαλείο ανάπτυξης του συστήματός μας, ήταν τα διάφορα βοηθήματα που

μπορούσε να μας προσφέρει όσον αφορά τη δημιουργία ενός GUI. Καταφέραμε έτσι να φτιάξουμε ένα απλό πρωτότυπο του συστήματος στο οποίο διαφαίνεται ο τρόπος αλληλεπίδρασης με τον χρήστη αλλά και η μορφή παρουσίασης των αποτελεσμάτων.

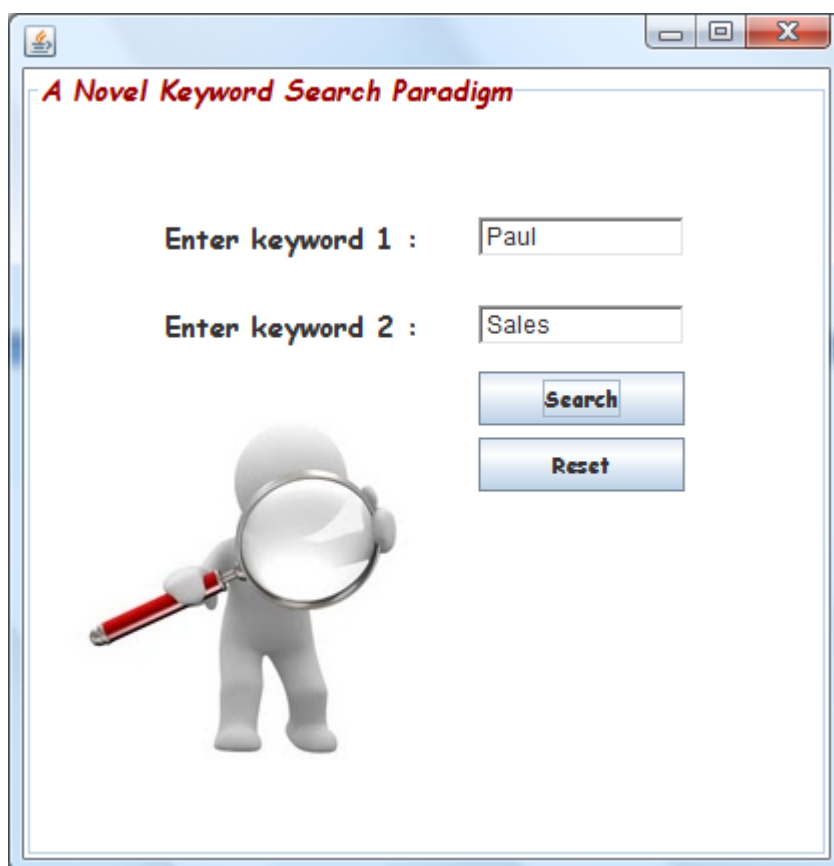
Πιο κάτω δίνουμε μερικά screenshots που λήφθηκαν από το πρωτότυπο αυτό, τα οποία παρουσιάζουν όλα τα βήματα που εκτελούνται μέχρι την τελική παρουσίαση των αποτελεσμάτων στον χρήστη.



Η πιο πάνω εικόνα αντιστοιχεί στο αρχικό παράθυρο που εμφανίζεται στον χρήστη. Αρχικά, του ζητείται να επιλέξει τη μέθοδο ταξινόμησης των αποτελεσμάτων που επιθυμεί, δηλ. είτε με χρήση των τιμών $Af-tf$ (AF –TF 1), είτε με χρήση των τιμών $\overline{Af-tf}$ (AF – TF 2). Πατώντας το κουμπί OK καταχωρεί την επιλογή του και μπορεί να προχωρήσει στην εκτέλεση της αναζήτησής του.



Αμέσως μετά την επιλογή της μεθόδου ταξινόμησης, παρουσιάζεται ένα νέο παράθυρο στον χρήστη, όπως φαίνεται πιο πάνω, το οποίο του δίνει τη δυνατότητα εισαγωγής δύο λέξεων – κλειδιών. Η πρώτη λέξη – κλειδί που θα δώσει θα αντιστοιχεί, όπως ήδη γνωρίζουμε, στο DS για το οποίο ενδιαφέρεται να πάρει πληροφορίες και η δεύτερη λέξη – κλειδί στη θεματική ενότητα που τον ενδιαφέρει.



Η πιο πάνω εικόνα δηλώνει πως ο χρήστης επιθυμεί να πάρει πληροφορίες για το DS με το όνομα “Paul” γύρω από τη θεματική ενότητα “Sales”. Για την εκτέλεση της αναζήτησης αρκεί να πατηθεί το κουμπί *Search*, ενώ το κουμπί *Reset* επαναφέρει τα δύο πεδία κειμένου στην αρχική άδεια μορφή τους.



Σε περίπτωση που το DS που καθόρισε ο χρήστης δεν εντοπιστεί στη βάση δεδομένων, τότε παρουσιάζεται το ανάλογο μήνυμα λάθους. Επίσης, εάν η λέξη που καθορίζει τη θεματική ενότητα δεν βρίσκεται στο ευρετήριο τότε τα αποτελέσματα για το συγκεκριμένο DS θα παρουσιαστούν μη ταξινομημένα.

Επίσης, αναφερθήκαμε με αρκετή λεπτομέρεια στον τρόπο υλοποίησης τόσο της φάσης προεπεξεργασίας όσο και της φάσης επεξεργασίας του συστήματός μας και τέλος παρουσιάσαμε μερικά screenshots που λήφθηκαν από το πρωτότυπο του συστήματος.

Κεφάλαιο 5

Αξιολόγηση

5.1 Εισαγωγή	63
5.2 Αξιολόγηση αποτελεσματικότητας συστήματος	63
5.2.1 Μεθοδολογία αξιολόγησης	64
5.2.2 Αποτελέσματα αξιολόγησης	65
5.3 Αξιολόγηση απόδοσης συστήματος	71
5.3.1 Μεθοδολογία αξιολόγησης	71
5.3.2 Αποτελέσματα αξιολόγησης	72
5.4 Συμπεράσματα	74

5.1 Εισαγωγή

Το παρόν κεφάλαιο πραγματεύεται τη διαδικασία αξιολόγησης του συστήματός μας, η οποία έχει γίνει στα πλαίσια της βάσης δεδομένων Northwind. Γνωρίζοντας ότι η ποιότητα του συστήματος είναι θέμα πρωταρχικής σημασίας για την αποδοχή του εξετάστηκαν δύο πολύ βασικά στοιχεία που την καθορίζουν, η αποτελεσματικότητα και η απόδοσή του. Έτσι, λοιπόν, η διαδικασία αξιολόγησης διακρίνεται σε δύο μέρη, όπου το πρώτο μέρος αφορά την αξιολόγηση της αποτελεσματικότητας του συστήματος ενώ το δεύτερο μέρος αφορά την αξιολόγησή του σε θέμα απόδοσης.

5.2 Αξιολόγηση αποτελεσματικότητας συστήματος

Στην ενότητα αυτή, διερευνούμε την αποτελεσματικότητα του συστήματός μας. Ως αποτελεσματικότητα ορίζουμε το κατά πόσο η μεθοδολογία που ακολουθεί είναι όντως χρήσιμη και μπορεί να συμβάλει στην εξαγωγή αποτελεσμάτων που βρίσκονται πολύ κοντά στις προσδοκίες των χρηστών.

5.2.1 Μεθοδολογία αξιολόγησης

Γι' αυτό το μέρος της αξιολόγησης, χρειάστηκε η συμμετοχή πέντε φοιτητών του Τμήματος Πληροφορικής. Οι φοιτητές αυτοί, δεν είχαν οποιαδήποτε προηγούμενη επαφή με την προτεινόμενη μεθοδολογία αλλά έρχονταν για πρώτη φορά αντιμέτωποι με έννοιες όπως το OS.

Ως πρώτο βήμα, λοιπόν, δημιουργήσαμε πέντε αρχεία τα οποία περιείχαν ένα υποσύνολο των OSs που είχαν ληφθεί από το σύστημα για πέντε διαφορετικές εισόδους, δηλαδή για πέντε διαφορετικά Queries. Κάθε αρχείο περιείχε πέντε OSs. Τα αρχεία αυτά δόθηκαν εν συνεχεία σε κάθε ένα από τους πέντε αξιολογητές, και μετά από όλες τις απαραίτητες επεξηγήσεις τους ζητήθηκε να δώσουν τη δική τους ταξινόμηση για τα OSs του κάθε αρχείου. Ωστόσο, λαμβάνοντας υπόψη το γεγονός ότι οι αξιολογητές δεν ήταν ιδιαίτερα εξοικειωμένοι με την έννοια του OS και τον τρόπο αναζήτησης που εφαρμόζει το σύστημα, κάθε αρχείο συνοδευόταν από ένα πίνακα με όλα τα απαραίτητα στοιχεία για κάθε OS καθώς επίσης και δύο υποσύνολα των G^{DS} δέντρων των Employees και Customers. Ένα μέρος των αρχείων αυτών παρουσιάζεται στο παράρτημα Γ.

Η ταξινόμηση που έδινε το σύστημά μας για τα OSs του κάθε αρχείου έπρεπε να προσιδιάζει όσο το δυνατόν περισσότερο με τα αποτελέσματα των αξιολογητών, έτσι ώστε να μπορεί να τεκμηριωθεί η αποτελεσματικότητά του. Θα πρέπει ωστόσο να αναφέρουμε πως θέλοντας να εξετάσουμε και την περίπτωση όπου η ταξινόμηση των OSs γινόταν με συνδυασμό λέξεων – κλειδιών, όπως συμβαίνει στις πλείστες μηχανές αναζήτησης, δημιουργήσαμε μία παραλλαγή του συστήματός μας όπου η ταξινόμηση των OSs βασιζόταν σε δύο λέξεις – κλειδιά κι όχι σε μία. Η παραλλαγή αυτή εφαρμόζε την εξίσωση (5) του κεφαλαίου 3. Ένα από τα πέντε αρχεία που δόθηκαν στους αξιολογητές αφορά την συγκεκριμένη περίπτωση.

Τα Queries που χρησιμοποιήσαμε για την αξιολόγηση του συστήματος είναι τα εξής:

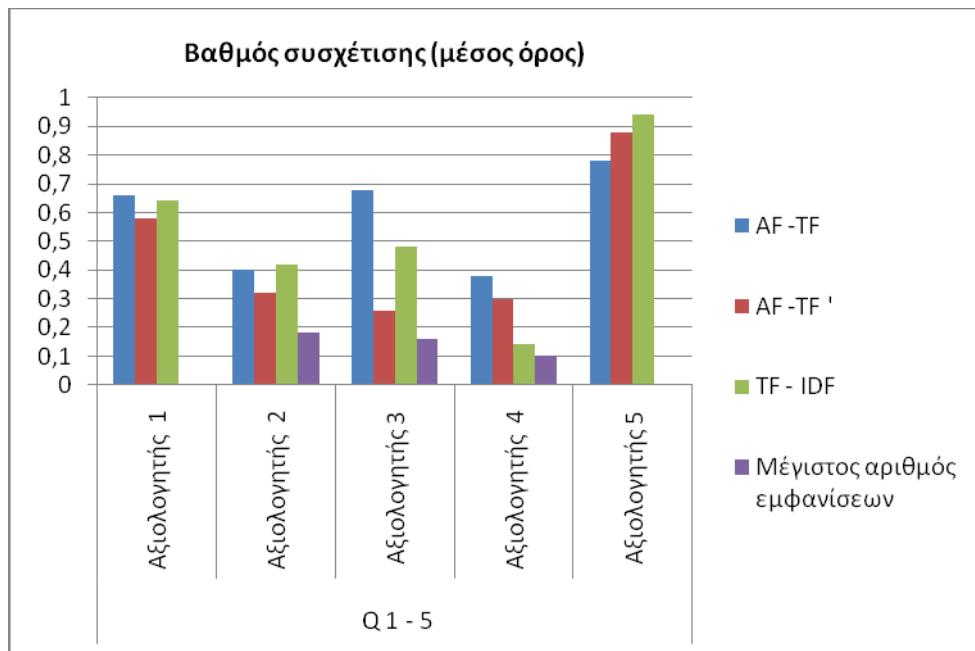
- $Q1 = \{\text{"Maria"}, \text{"Sales"}\}$
- $Q2 = \{\text{"Robert"}, \text{"Marketing"}\}$
- $Q3 = \{\text{"Michael"}, \text{"Sales"}\}$
- $Q4 = \{\text{"Anne"}, \text{"Management"}\}$
- $Q5 = \{\text{"Leverling"}, \{\text{"Sales Marketing"}\}\}$

Για να εξετάσουμε, λοιπόν, σε ποιο βαθμό συσχετίζονταν τα αποτελέσματα του συστήματός μας με τα αποτελέσματα των αξιολογητών για τα πέντε διαφορετικά Queries, χρησιμοποιήσαμε τη συνάρτηση CORREL που παρέχει η Microsoft Excel. Συγκεκριμένα, η συνάρτηση αυτή δίνει ένα βαθμό από το πεδίο $[-1,1]$ για να ορίσει τη συσχέτιση που υπάρχει μεταξύ δύο συνόλων τιμών. Εάν αυτός ο βαθμός συσχέτισης είναι 1 τότε τα δύο σύνολα έχουν μία τέλεια γραμμική σχέση (όσο αυξάνονται οι τιμές του ενός συνόλου αυξάνονται οι τιμές και του άλλου συνόλου) ενώ εάν ο βαθμός είναι -1 τότε έχουν μία τέλεια αρνητική γραμμική σχέση (όσο αυξάνονται οι τιμές του ενός συνόλου τόσο μειώνονται οι τιμές του άλλου συνόλου). Εάν ο βαθμός συσχέτισης είναι 0 τότε τα δύο σύνολα δεν έχουν καμία σχέση μεταξύ τους.

Επιπρόσθετα, θέλοντας να συγκρίνουμε τη μέθοδο $Af-tf$ που υποστηρίζει το σύστημά μας για την ταξινόμηση των αποτελεσμάτων, όπως επίσης και τη μέθοδο $\overline{Af-tf}$, με άλλες μεθόδους που θα μπορούσαν να εφαρμοστούν για την ταξινόμηση των OSs, βρήκαμε τα αποτελέσματα που δίνουν δύο ακόμη μέθοδοι ταξινόμησης για τα πέντε Queries. Με τον τρόπο αυτό θα μπορούσαμε να εξετάσουμε κατά πόσο η μέθοδος μας υπερτερεί ή όχι έναντι άλλων μεθόδων.

5.2.2 Αποτελέσματα αξιολόγησης

Στο σημείο αυτό, παρουσιάζουμε τα αποτελέσματα έχουν εξαχθεί κατά τη διαδικασία αξιολόγησης της αποτελεσματικότητας του συστήματός μας.



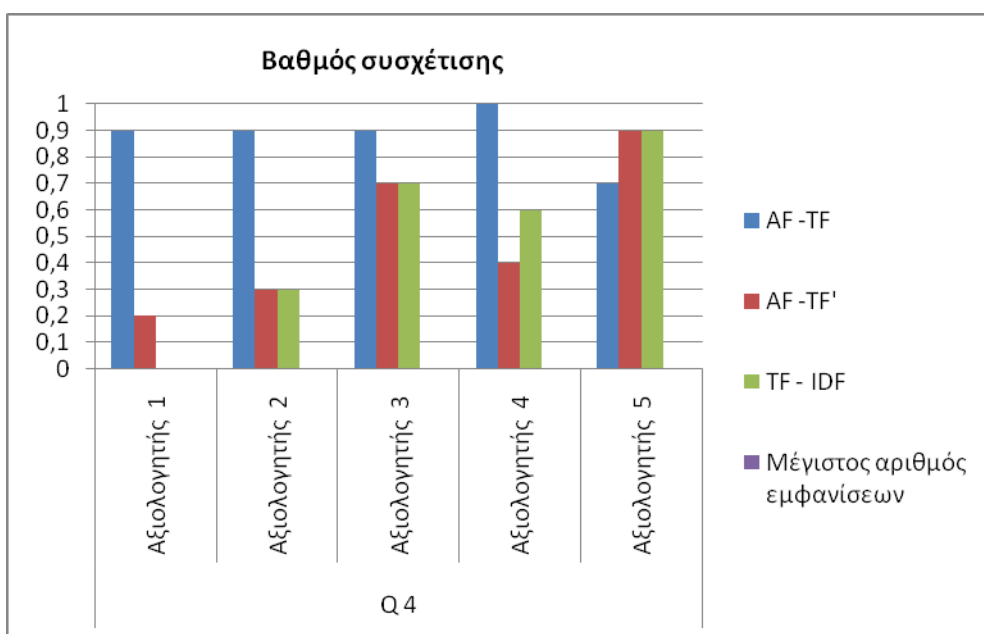
Σχήμα 5.2 Γραφική παράσταση βαθμού συσχέτισης των αποτελεσμάτων των αξιολογητών με τα αποτελέσματα των μεθόδων ταξινόμησης κατά μέσο όρο και για τα πέντε Queries

Η πιο πάνω γραφική παράσταση παρουσιάζει κατά μέσο όρο το βαθμό συσχέτισης των αποτελεσμάτων που έχει δώσει ο κάθε αξιολογητής για τα πέντε Queries με τα αποτελέσματα που δίνουν τέσσερις μέθοδοι ταξινόμησης για τα αντίστοιχα Queries. Οι τέσσερις μέθοδοι ταξινόμησης που εξετάζονται είναι οι εξής:

- AF-TF: απευθύνεται στη μέθοδο που εφαρμόζει το σύστημά μας για ταξινόμηση των OSs με βάση τις τιμές $Af-tf$
- AF- TF': απευθύνεται στη μέθοδο που επίσης υποστηρίζει το σύστημά μας για ταξινόμηση των OSs με βάση τις τιμές $\overline{Af-tf}$
- TF – IDF: η μέθοδος αυτή ταξινομεί τα OSs με βάση το βάρος $tf-idf$ των λέξεων – κλειδιών που χρησιμοποιούνται για την ταξινόμηση.
- Μέγιστος αριθμός εμφανίσεων : η ταξινόμηση των OSs γίνεται με βάση τον αριθμό των εμφανίσεων των λέξεων – κλειδιών που χρησιμοποιούνται για την ταξινόμηση. Πρώτο στην ταξινόμηση θα βρίσκεται το OS με τον μέγιστο αριθμό εμφανίσεων των λέξεων – κλειδιών.

Από τη γραφική παράσταση, λοιπόν, παρατηρούμε ότι τα αποτελέσματα που έχουν δώσει οι αξιολογητές 1, 3 και 4 για τα πέντε Queries, έχουν κατά μέσο όρο μεγαλύτερο βαθμό συσχέτισης με τα αποτελέσματα που έχει δώσει η μέθοδος AF–

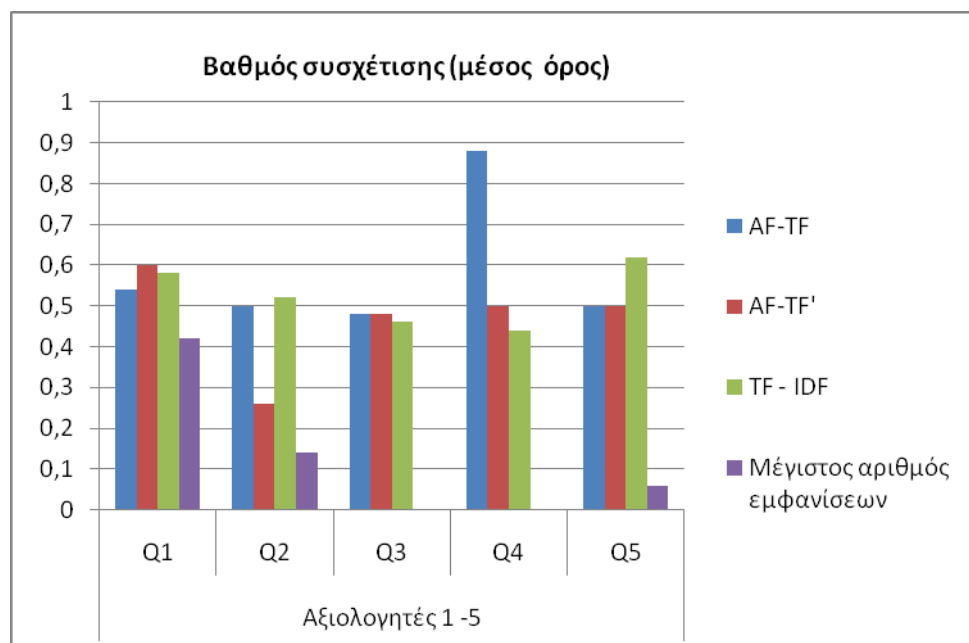
TF, σε σύγκριση με τα αποτελέσματα των υπόλοιπων μεθόδων ταξινόμησης. Όσον αφορά τα αποτελέσματα των αξιολογητών 2 και 5 φαίνεται να συσχετίζονται περισσότερο με τα αποτελέσματα της μεθόδου TF-IDF, με ελάχιστη ωστόσο διαφορά από τις μεθόδους AF-TF και AF-TF'. Επιπρόσθετα τα αποτελέσματα της μεθόδου μέγιστου αριθμού εμφανίσεων σχετίζονται ελάχιστα έως καθόλου με τα αποτελέσματα των πέντε αξιολογητών.



Σχήμα 5.3 Γραφική παράσταση βαθμού συσχέτισης των αποτελεσμάτων των αξιολογητών για ένα συγκεκριμένο Query με τα αποτελέσματα των μεθόδων ταξινόμησης

Η πιο πάνω γραφική παράσταση παρουσιάζει τον βαθμό συσχέτισης των αποτελεσμάτων των φοιτητών για το Q4 με τα αποτελέσματα που δίνουν για το συγκεκριμένο Query οι μέθοδοι που εξετάζουμε. Παρατηρούμε ότι τα αποτελέσματα των αξιολογητών 1, 2, 3 και 4 έχουν πολύ μεγαλύτερο βαθμό συσχέτισης με τα αποτελέσματα που δίνει η μέθοδος AF-TF σε σχέση με τα αποτελέσματα των υπόλοιπων μεθόδων ταξινόμησης. Συγκεκριμένα, ο βαθμός συσχέτισης των αξιολογητών 1, 2 και 3 κυμαίνεται γύρω στο 0.9 που δηλώνει ότι τα αποτελεσματά τους έχουν μία πολύ καλή γραμμική συσχέτιση με τα αποτελέσματα της μεθόδου AF

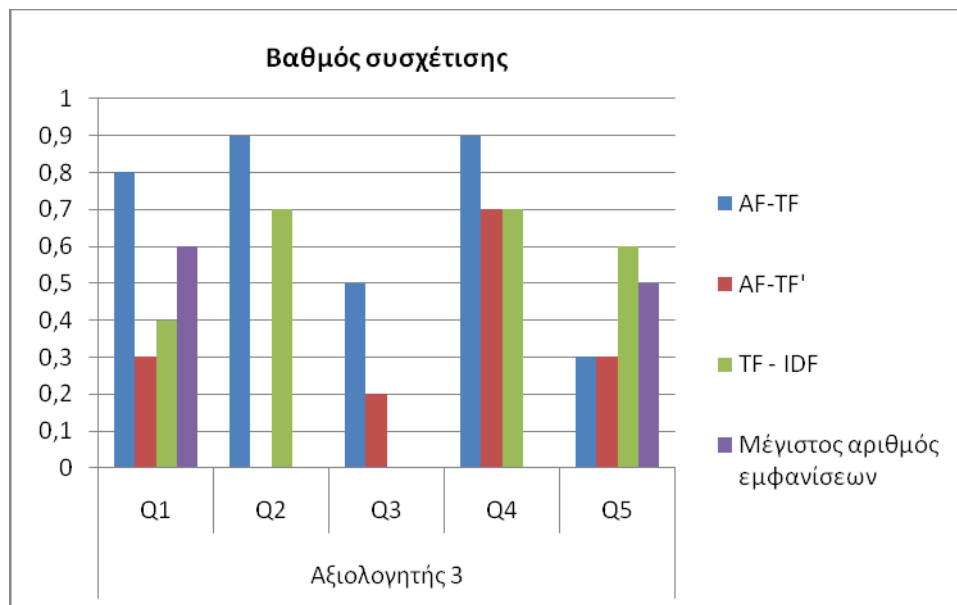
-TF αλλά η συσχέτιση δεν είναι τέλεια, ενώ του αξιολογητή 4 ο βαθμός συσχέτισης παίρνει τη μέγιστη τιμή δηλαδή 1, που δηλώνει μία τέλεια γραμμική σχέση με τα αποτελέσματα της μεθόδου AF-TF. Επίσης, τα αποτελέσματα του αξιολογητή 5 φαίνεται να έχουν τον ίδιο βαθμό συσχέτισης τόσο με τα αποτελέσματα της μεθόδου AF-TF' όσο και με τα αποτελέσματα της μεθόδου TF - IDF. Όσον αφορά τα αποτελέσματα της μεθόδου μέγιστου αριθμού εμφανίσεων των λέξεων – κλειδιών, είναι εμφανές ότι δεν σχετίζονται καθόλου με τα αποτελέσματα των αξιολογητών για το συγκεκριμένο Query.



Σχήμα 5.4 Βαθμός συσχέτισης κατά μέσο όρο των αποτελεσμάτων των αξιολογητών για κάθε Query ξεχωριστά με τα αποτελέσματα των μεθόδων ταξινόμησης

Στην πιο πάνω γραφική παράσταση μπορούμε να δούμε κατά μέσο όρο για κάθε ένα από τα πέντε Queries τον βαθμό συσχέτισης των αποτελεσμάτων των αξιολογητών με τα αποτελέσματα της κάθε μεθόδου ταξινόμησης. Όσον αφορά το Q4 τα αποτελέσματα των αξιολογητών συσχετίζονται σε πολύ μεγαλύτερο βαθμό με τα αποτελέσματα της μεθόδου AF-TF σε σχέση με τις υπόλοιπες μεθόδους ταξινόμησης. Συγκεκριμένα, ο βαθμός συσχέτισης των αποτελεσμάτων των αξιολογητών για το Q4 με τα αποτελέσματα της μεθόδου AF-TF φτάνει μέχρι και το 0.88 δηλώνοντας μία

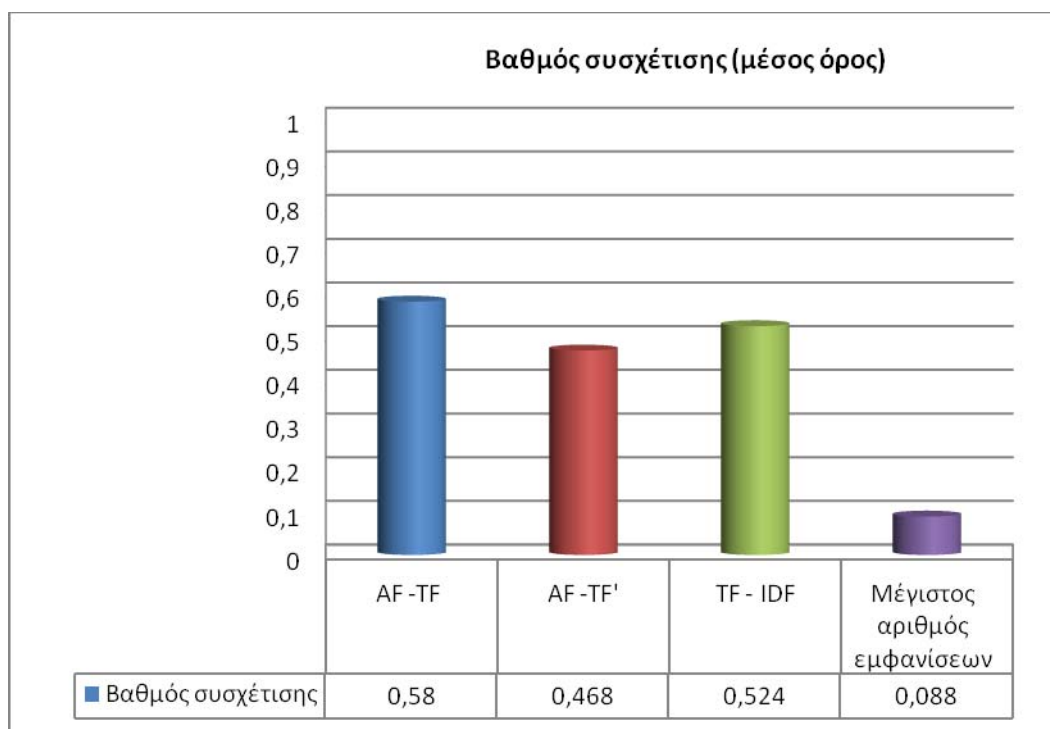
πολύ καλή γραμμική συσχέτιση ενώ ακολουθεί η μέθοδος AF-TF' με βαθμό συσχέτισης 0.5. Στην περίπτωση του Q1 μεγαλύτερη συσχέτιση υπάρχει με τα αποτελέσματα της μεθόδου AF – TF' ενώ στην περίπτωση του Q3 υπάρχει μία ισοδυναμία ως προς το βαθμό συσχέτισης με τις μεθόδους AF – TF και AF –TF'. Τα αποτελέσματα των αξιολογητών για το Q2 φαίνεται να σχετίζονται περισσότερο με τα αποτελέσματα της μεθόδου TF – IDF. Το ίδιο παρατηρείται και στην περίπτωση του Q5 , στο οποίο γίνεται χρήση δύο λέξεων – κλειδίων για την ταξινόμηση των OSs, με αρκετή μάλιστα διαφορά από τις υπόλοιπες μεθόδους ταξινόμησης. Το γεγονός αυτό πιθανόν να οφείλεται στο ότι δεν έχει γίνει κατανοητό από τους αξιολογητές ο τρόπος ταξινόμησης των αποτελεσμάτων με χρήση περισσότερων της μίας λέξεων – κλειδίων. Όσον αφορά τα αποτελέσματα της μεθόδου μέγιστου αριθμού εμφανίσεων, σχετίζονται και πάλι ελάχιστα έως καθόλου με τα αποτελέσματα των αξιολογητών για το κάθε Query ξεχωριστά.



Σχήμα 5.5 Βαθμός συσχέτισης αποτελεσμάτων ενός συγκεκριμένου αξιολογητή για κάθε Query ξεχωριστά με τα αποτελέσματα των μεθόδων ταξινόμησης

Η πιο πάνω γραφική παράσταση παρουσιάζει για κάθε ένα από τα πέντε Queries τον βαθμό συσχέτισης των αποτελεσμάτων του αξιολογητή 3 με τα αποτελέσματα των μεθόδων ταξινόμησης. Παρατηρούμε, πως για τα τέσσερα πρώτα Queries ο βαθμός

συσχέτισης των αποτελεσμάτων του αξιολογητή με τα αποτελέσματα της μεθόδου AF-TF είναι πολύ μεγαλύτερος σε σύγκριση με τα αποτελέσματα των υπόλοιπων μεθόδων ταξινόμησης, παίρνοντας μάλιστα μέγιστη τιμή 0.9 για τα Queries Q2 και Q4. Όσον αφορά το Q5, τα αποτελέσματα του αξιολογητή φαίνεται να συσχετίζονται περισσότερο με τα αποτελέσματα της μεθόδου TF – IDF.



Σχήμα 5.6 Βαθμός συσχέτισης κατά μέσο όρο των αποτελεσμάτων όλων των αξιολογητών για τα πέντε Queries με τα αποτελέσματα των μεθόδων ταξινόμησης

Τα αποτελέσματα που παρουσιάζονται στις προηγούμενες γραφικές παραστάσεις συνοψίζονται στην πιο πάνω γραφική παράσταση. Η γραφική παράσταση αυτή παρουσιάζει κατά μέσο όρο το βαθμό συσχέτισης των αποτελεσμάτων όλων των αξιολογητών για τα πέντε Queries με τα αποτελέσματα της κάθε μεθόδου ταξινόμησης. Αυτό που παρατηρούμε είναι ότι τα αποτελέσματα των αξιολογητών συσχετίζονται σε μεγαλύτερο βαθμό με τα αποτελέσματα που δίνει η μέθοδος AF – TF σε σχέση με τα αποτελέσματα των υπόλοιπων μεθόδων ταξινόμησης. Ο βαθμός συσχέτισης κυμαίνεται στο 0.58. Ακολουθούν, οι μέθοδοι TF – IDF και AF –TF’

ενώ τελευταία στην κατάταξη με πολύ μικρό βαθμό συσχέτισης βρίσκεται η μέθοδος μέγιστου αριθμού εμφανίσεων των λέξεων - κλειδιών.

Μπορούμε, λοιπόν, να πούμε ότι η μέθοδος AF-TF πλεονεκτεί έναντι των υπόλοιπων μεθόδων ταξινόμησης καθώς δίνει αποτελέσματα που βρίσκονται πλησιέστερα στις προσδοκίες των χρηστών.

5.3 Αξιολόγηση απόδοσης συστήματος

Στην παρούσα ενότητα, παρουσιάζουμε την αξιολόγηση του συστήματός μας σε θέμα απόδοσης. Ως απόδοση ορίζουμε τον χρόνο που απαιτείται για την εκτέλεση μίας αναζήτησης και την ανάκτηση των αποτελεσμάτων. Πιο συγκεκριμένα παραθέτουμε μία σύγκριση της περίπτωση που το σύστημα δεν χρησιμοποιεί ευρετήριο αλλά υπολογίζει τις τιμές *Af-tf* με την εκτέλεση κάθε αναζήτησης, με την περίπτωση χρήσης ευρετηρίου όπως γίνεται τώρα.

5.3.1 Μεθοδολογία αξιολόγησης

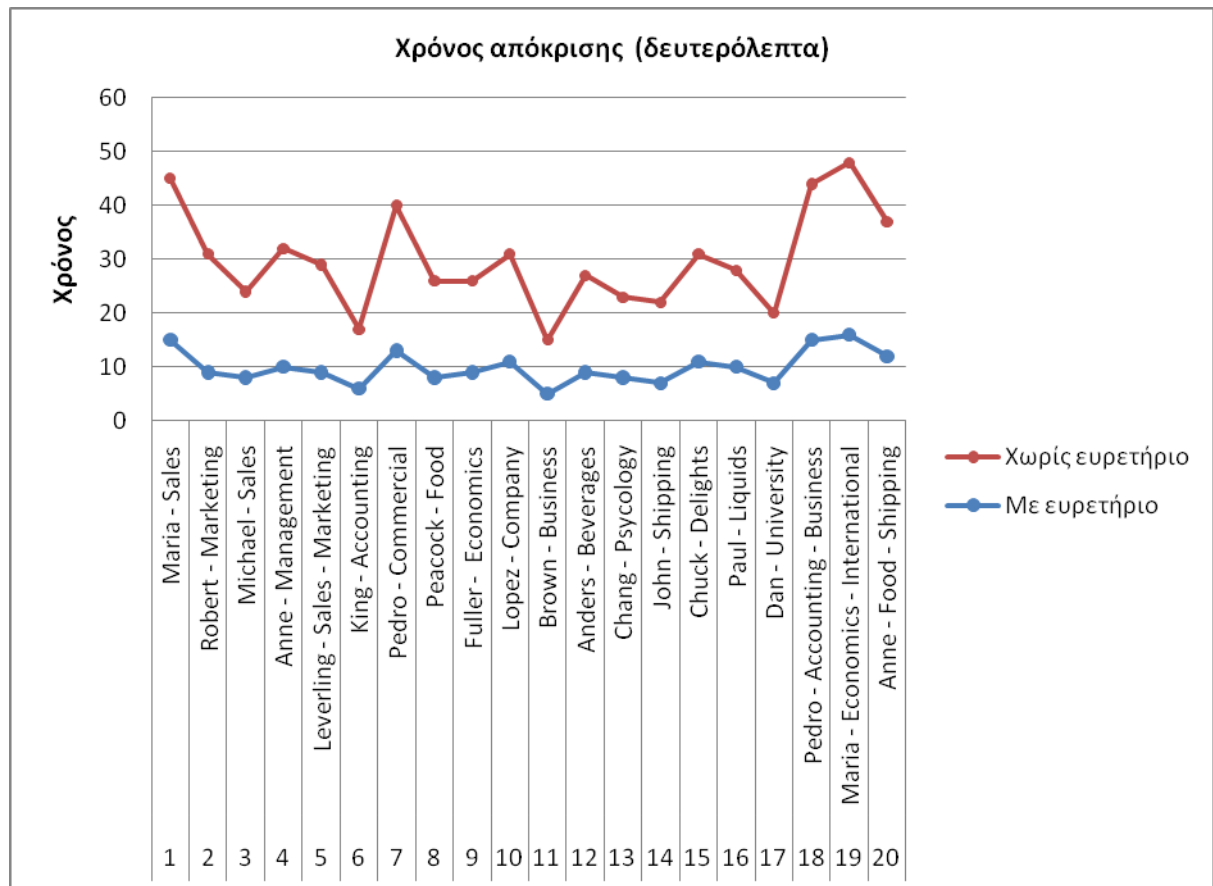
Για την αξιολόγηση αυτή, χρειάστηκε να υλοποιήσουμε και πάλι μία παραλλαγή του συστήματός μας κατά την οποία δεν γινόταν χρήση του ευρετηρίου αλλά οι τιμές *Af-tf* υπολογίζονταν με την εκτέλεση κάθε αναζήτησης. Ακολούθως, επιλέξαμε είκοσι διαφορετικά Queries για τα οποία υπολογίσαμε τον χρόνο απόκρισης του συστήματος στην περίπτωση που γινόταν χρήση του ευρετηρίου και στην περίπτωση που δεν χρησιμοποιείτο το ευρετήριο. Τα Queries αυτά παρουσιάζονται στον πιο κάτω πίνακα μαζί με τον αριθμό των OSs που αντιστοιχούν για το καθένα και τον συνολικό αριθμό των πλειάδων. Επίσης, θα πρέπει να αναφέρουμε πως τα πέντε πρώτα OSs αντιστοιχούν στα OSs που χρησιμοποιήσαμε για την αξιολόγηση της αποτελεσματικότητας του συστήματός μας ενώ τα τρία τελευταία αφορούν την περίπτωση συνδυασμού δύο λέξεων – κλειδιών για την ταξινόμηση των OSs.

Query		Αριθμός OSs	Συνολικός αριθμός πλειάδων
1	Maria - Sales	19	4395
2	Robert - Marketing	16	7189
3	Michael - Sales	15	2504
4	Anne - Management	20	3356
5	Leverling - Sales - Marketing	13	2766
6	King - Accounting	9	1440
7	Pedro - Commercial	15	4079
8	Peacock - Food	8	2421
9	Fuller - Economics	10	2277
10	Lopez - Company	22	3289
11	Brown - Business	10	1608
12	Anders - Beverages	10	2697
13	Chang - Psychology	18	2268
14	John - Shipping	17	2202
15	Chuck - Delights	22	3253
16	Paul - Liquids	18	2895
17	Dan - University	14	1988
18	Pedro - Accounting - Business	15	4079
19	Maria - Economics - International	19	4395
20	Anne - Food - Shipping	20	3356

Πίνακας 5.1 Queries που χρησιμοποιήθηκαν για την αξιολόγηση της απόδοσης του συστήματος

5.3.2 Αποτελέσματα αξιολόγησης

Στο σημείο αυτό, παρουσιάζουμε τα αποτελέσματα που έχουν εξαχθεί κατά τη διαδικασία αξιολόγησης της απόδοσης του συστήματός μας.



Σχήμα 5.7 Χρόνος απόκρισης συστήματος για είκοσι Queries με και χωρίς τη χρήση ευρετηρίου

Η πιο πάνω γραφική παράσταση παρουσιάζει το χρόνο απόκρισης του συστήματος για τα είκοσι διαφορετικά Queries με και χωρίς τη χρήση ευρετηρίου. Παρατηρούμε, ότι χωρίς χρήση ευρετηρίου ο χρόνος που χρειάζεται το σύστημα για να δώσει αποτελέσματα για κάθε ένα από τα είκοσι Queries κυμαίνεται σε αρκετά υψηλά επίπεδα φτάνοντας μέχρι και τα 32 δευτερόλεπτα. Ωστόσο, με χρήση ευρετηρίου ο χρόνος αυτός μειώνεται περίπου στο μισό για κάθε ένα από τα είκοσι Queries με τιμές να κυμαίνονται από 5 έως 16 δευτερόλεπτα.

Γνωρίζοντας ότι ο χρόνος απόκρισης αποτελεί ένα από τους μεγαλύτερους εχθρούς της ποιότητας μίας μηχανής αναζήτησης, η βελτίωση αυτή θεωρείται πολύ σημαντική για την απόδοση του συστήματός μας. Το κόστος που είχαμε σε μνήμη όσον αφορά τη χρήση των ευρετηρίων με τις τιμές $Af\text{-}tf$ και $Af\text{-}\overline{tf}$ δεν ήταν ιδιαίτερα μεγάλο καθώς κυμαινόταν συνολικά στα 394 KB. Οπότε, μπορούμε να πούμε πως η χρήση των ευρετηρίων επέδρασε μόνο θετικά στην απόδοση του συστήματός μας.

5.4 Συμπεράσματα

Στο κεφάλαιο αυτό, λοιπόν, έγινε η αξιολόγηση του συστήματός μας τόσο σε θέμα αποτελεσματικότητας όσο και σε θέμα απόδοσης. Από τα αποτελέσματα που έχουμε πάρει διαφάνηκε ότι η μεθοδολογία που εφαρμόζει το σύστημά μας για ταξινόμηση των OSs με βάση τις τιμές $Af-tf$, είναι όντως χρήσιμη και αποτελεσματική. Συγκριτικά με άλλες μεθόδους ταξινόμησης που εφαρμόζονται σε μηχανές αναζήτησης έδωσε αποτελέσματα τα οποία συσχετίζονταν σε μεγαλύτερο βαθμό με τα αποτελέσματα των πέντε αξιολογητών του συστήματος. Παράλληλα με αυτά, η διαδικασία ευρετηρίασης που εφαρμόζει το σύστημά μας είχε ως αποτέλεσμα τη σημαντική βελτίωση της απόδοσής του όσον αφορά το χρόνο που απαιτείται από τη στιγμή της εκτέλεσης μίας αναζήτησης μέχρι την τελική παρουσίαση των αποτελεσμάτων.

Κεφάλαιο 6

Συμπεράσματα

6.1 Σύνοψη	75
6.2 Τελικά συμπεράσματα και μελλοντικές επεκτάσεις	76

6.1 Σύνοψη

Στόχος της διπλωματικής εργασίας ήταν να διερευνήσει μία νέα μεθοδολογία για ταξινόμηση OSs που λαμβάνονται από την εκτέλεση αναζητήσεων σε σχεσιακές βάσεις δεδομένων. Η νέα αυτή μεθοδολογία στηρίχθηκε σε προϋπάρχουσες μεθόδους ταξινόμησης που εφαρμόζονται από πολλές μηχανές αναζήτησης σήμερα και συγκεκριμένα στις μεθόδους stemming και TF – IDF, συνδυάζοντας παράλληλα την έννοια του *Affinity*. Το *Affinity* έχει παίξει ένα πολύ σημαντικό και καθοριστικό ρόλο στην ανάπτυξη της εν λόγω μεθοδολογίας.

Η ιδέα ήταν να παρέχεται η δυνατότητα στο χρήστη της μηχανής αναζήτησης να καθορίζει το αντικείμενο για το οποίο θέλει να πάρει πληροφορίες αλλά και τη θεματική ενότητα που τον ενδιαφέρει, ούτως ώστε τα αποτελέσματα που θα λάβει να βρίσκονται όσο το δυνατόν πιο κοντά σε αυτό που επιθυμεί. Για το λόγο αυτό το σύστημα που αναπτύχθηκε προς μελέτη της νέας μεθοδολογίας δεχόταν από το χρήστη δύο λέξεις – κλειδιά, μία για τον καθορισμό του αντικειμένου και μία για τον καθορισμό της θεματικής ενότητας.

Σύμφωνα με το αντικείμενο που καθόριζε ο χρήστης δημιουργείτο ένα σύνολο από OSs τα οποία ταξινομούνταν στη συνέχεια με βάση τη λέξη – κλειδί που αφορούσε τη θεματική ενότητα. Η λέξη αυτή χρησιμοποιήθηκε για την εφαρμογή των μεθόδων ταξινόμησης που προαναφέραμε και της έννοιας του *Affinity*. Από τον συνδυασμό των μεθόδων αυτών προέκυψε η νέα μεθοδολογία *Af-tf*.

Ειδικότερα, η μεθοδολογία αυτή εφαρμόζε τη μέθοδο stemming ως ένα απλό βήμα προεπεξεργασίας και στη συνέχεια συνδύαζε τις εξισώσεις της μεθόδου TF-IDF μαζί με την έννοια του *Affinity* για τον υπολογισμό των τιμών που θα χρησιμοποιούνταν με σκοπό την ταξινόμηση των OSs. Η προσθήκη του *Affinity* στις εξισώσεις της μεθόδου TF-IDF συνέβαλε στην καλύτερη ταξινόμηση των OSs, όπου οι πληροφορίες γύρω από τη θεματική ενότητα σχετίζονταν όσο το δυνατόν περισσότερο με το αντικείμενο που καθόριζε ο χρήστης.

6.2 Τελικά συμπεράσματα και μελλοντικές επεκτάσεις

Έχοντας ολοκληρώσει την εργασία και αξιολογήσει το σύστημα που αναπτύχθηκε στα πλαίσια της νέας μεθοδολογίας, η γενική παρατήρηση που απορρέει είναι πως η εν λόγω μεθοδολογία είναι ιδιαίτερα αποτελεσματική και αποδοτική όσον αφορά την ταξινόμηση OSs.

Συγκρίνοντάς την με άλλες μεθόδους ταξινόμησης όπως τη μέθοδο TF-IDF, παρατηρήθηκε πως υπερτερεί έναντι αυτών καθώς τα αποτελέσματά που έδινε για συγκεκριμένες εισόδους προσιδίαζαν περισσότερο με τα αποτελέσματα των αξιολογητών του συστήματός μας. Παρόλο που ο αριθμός των αξιολογητών αλλά και των δειγμάτων που χρησιμοποιήθηκαν ήταν σχετικά μικρός, διαφάνηκε ακόμα και σε αυτή την περίπτωση πως η μέθοδος *Af-tf* λειτουργεί αποδοτικά και αποτελεσματικά έναντι άλλων μεθόδων ταξινόμησης.

Το σύστημα που υλοποιήθηκε θα μπορούσε να επεκταθεί έτσι ώστε να δέχεται από τον χρήστη περισσότερες από μία λέξεις – κλειδιά τόσο για τον καθορισμό του αντικειμένου όσο και για τον καθορισμό της θεματικής ενότητας. Επιπρόσθετα, η προτεινόμενη μεθοδολογία θα μπορούσε να συμμετάσχει σε υβριδικές μεθόδους ταξινόμησης στα πλαίσια πάντα σχεσιακών βάσεων δεδομένων.

Βιβλιογραφία

- [1] G.Fakas, “Automated Generation of Object Summaries from Relational Databases: A Novel Keyword Searching Paradigm”, DBRank Workshop 2008, ICDE 2008.
- [2] G.Fakas, Z.Cai, “Ranking of Object Summaries”, DBRank Workshop 2009, ICDE 2009.
- [3] V.Hristidis, L.Gravano and Y.Papakonstantinou, “Efficient IR-Style Keyword Search over Relational Databases”, Proc. Very Large Data Bases Conf., 2003.
- [4] A.Balmin, V.Hristidis and Y.Papakonstantinou, “Objectrank: Authority – based Keyword Search in Databases”, Proc. Very Large Data Bases Conf., 2004.
- [5] F.Liu, C.Yu, W.Meng, A.Chowdhury, “Effective Keyword Search in Relational Databases”, SIGMOD 2006, June 27-29, 2006, Chicago, Illinois, USA.
- [6] B.Aditya, G.Bhalotia, S.Chakrabarti, A.Hulgeri, C.Nakhe, P.S.Sudarshan, “BANKS: Browsing and Keyword Searching in Relational Databases”, Proc Very Large Data Bases Conf., 2002.
- [7] V.Hristidis, Y.Papakonstantinou, “DISCOVER: Keyword Search in Relational Databases”, Proc Very Large Data Bases Conf., 2002.
- [8] J.Ramos, “Using TF-IDF to Determine Word Relevance in Document Queries”, Instructional Conference on Machine Learning Web Site and Instructions, December 2003, Piscataway NJ USA.
- [9] B.Yates, R.Neto, “Modern Information Retrieval”, Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA. 1999.
- [10] Porter stemming algorithm, <http://www.tartarus.org/~martin/PorterStemmer/>
- [11] Lovins stemming algorithm, <http://snowball.tartarus.org/algorithms/lovins/stemmer.html>
- [12] Nice Stemmer, <http://ils.unc.edu/iris/irisnstem.htm>
- [13] Snowball, <http://snowball.tartarus.org/>
- [14] M.Kantrowitz, B.Mohit, V.Mittal, “Stemming and its effects on TF-IDF Ranking”, Annual ACM Conference on Research and Development in Information Retrieval, 2000.
- [15] TF – IDF, <http://en.wikipedia.org/wiki/Tf-idf>

- [16] Proximity Search (text), http://en.wikipedia.org/wiki/Proximity_search_text
- [17] R.Goldman, N.Shivakumar, S.Venkatasubramanian, H.Garcia-Molina, "Proximity Search in Databases", 24th VLDB Conference New York, USA, 1998.

Παράρτημα Α

Στο παράρτημα αυτό δίνεται ένα μέρος των προγραμμάτων που έχουν υλοποιηθεί για την ανάπτυξη του συστήματος της προτεινόμενης μεθοδολογίας. Η λειτουργία που επιτελεί το καθένα περιγράφεται στην αρχή του κάθε προγράμματος ενώ υπάρχουν επίσης κάποια σχόλια για καλύτερη κατανόηση του κώδικα. Ωστόσο, λόγω μεγάλου μεγέθους των προγραμμάτων παραλείπεται ένα μεγάλο μέρος από αυτά. Τα προγράμματα έχουν υλοποιηθεί σε περιβάλλον Java με χρήση της πλατφόρμας ανάπτυξης NetBeans IDE.

```
/******
```

Γλώσσα: Java

Περιγραφή : Το πρόγραμμα αυτό αφορά την υλοποίηση της φάσης προεπεξεργασίας του συστήματος της προτεινόμενης μεθοδολογίας. Σκοπός του είναι η δημιουργία ενός ευρετηρίου με τις τιμές Af-tf βάση των οποίων θα γίνεται η ταξινόμηση των OSS

```
*****/
```

```
package indexing_keyword1;
import java.sql.*;
import java.util.*;
import java.io.*;
```

```
public class Preprocessing {
```

```
    public static void main(String[] args) {
```

```
        //δήλωση ArrayList για αποθήκευση των προεπιλεγμένων λέξεων
```

```
        ArrayList l1=new ArrayList();
```

```
        //διασύνδεση με τη βάση δεδομένων Northwind
```

```
        try {
```

```
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
```

```
            String url = "jdbc:odbc:Driver={Microsoft Access Driver " +  
                " (*.mdb, *.accdb)};DBQ=C:\\Program Files
```

```
\\Northwind_Original.mdb\\Northwind_Original.mdb";
```

```
            Connection con = DriverManager.getConnection(url);
```

```
        //δήλωση Query για ανάκτηση των πλειάδων της σχέσης Employees
```

```
        Statement stm_sql1=null;
```

```
        String emp = "SELECT * FROM Employees";
```

```
        ResultSet rs_emp = null;
```

```
        stm_sql1 = con.createStatement
```

```
(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);
```

```
        rs_emp = stm_sql1.executeQuery(emp);
```

```
        //δήλωση Query για ανάκτηση των πλειάδων της σχέσης Customers
```

```
        Statement stm_sql2=null;
```

```
        String cus = "SELECT * FROM Customers";
```

```
        ResultSet rs_cus = null;
```

```
        stm_sql2 = con.createStatement
```

```
(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);
```

```
        rs_cus=stm_sql2.executeQuery(cus);
```

```
        //διάβασμα αρχείου προεπιλεγμένων λέξεων
```

```
    try{
```

```

FileInputStream fstream = new FileInputStream("Words.txt")
DataInputStream in = new DataInputStream(fstream);
BufferedReader br = new BufferedReader(new InputStreamReader
(in));

    String strLine;

    while ((strLine = br.readLine()) != null)    {

        l1.add(strLine);

    }

    in.close();
} catch (Exception e){
    System.err.println("Error: " + e.getMessage());
}

    Emp_OS e = new Emp_OS(); //δήλωση αντικειμένου κλάσης Emp_OS

//κλήση συνάρτησης για υπολογισμό των τιμών Af-tf των Employees
e.AfTf_Calculation(rs_emp,l1);

//δήλωση αντικειμένου κλάσης Cus_OS
Cus_OS c = new Cus_OS();

//κλήση συνάρτησης για υπολογισμό των τιμών Af-tf των Customers
c.AfTf_Calculation(rs_cus,l1);
rs_emp.beforeFirst();
rs_cus.beforeFirst();

int k=l1.size();
int i;

try{
//δημιουργία αρχείου/ευρετηρίου για καταγραφή των τιμών Af-tf
    FileWriter fstream = new FileWriter("Index1.txt");
    BufferedWriter out = new BufferedWriter(fstream);

    //καταγραφή αποτελεσμάτων για τους Employees
    while(rs_emp.next()){

        String t=rs_emp.getString(1);

        for(i=0;i<k;i++){

            String keyword2=(String) l1.get(i);

            out.write(t+"_"+keyword2+" "+e.Af_tf.get(t+"_"+keyword2));
            out.newLine();

        }

        //καταγραφή αποτελεσμάτων για τους Customers
        while(rs_cus.next()){

            String t1=rs_cus.getString(1);

```

```

        for(i=0;i<k;i++){

            String keyword2=(String) l1.get(i);

            out.write(t1+"_"+keyword2+" "+c.Af_tf.get(t1+"_"+keyword2));
            out.newLine();

        }

    }

    out.close();
} catch (Exception e1){
    System.err.println("Error: " + e1.getMessage());
}

    con.close();
    } catch (SQLException e) {
        System.out.println("SQL Exception: "+ e.toString());
    } catch (ClassNotFoundException cE) {
        System.out.println("Class Not Found Exception: "+
            cE.toString());
    }
}
}
}

```

```

/*****
Η κλάση Cus_OS αφορά τον υπολογισμό των τιμών Af-tf για όλους τους
Customers. Παραθέτουμε μόνο ένα μέρος της κλάσης αυτής ενώ
παραλείπεται η κλάση Emp_OS η οποία υλοποιείται με παρόμοιο τρόπο.
*****/

```

```

package indexing_keyword1;
import java.sql.*;
import java.util.*;
import java.io.*;

```

```

public class Cus_OS {

```

```

    HashMap Af_tf=new HashMap(); //αποθήκευση τιμών Af-tf

```

```

    /*δήλωση τιμών Affinity για κάθε σχέση του GDS δέντρου των
    *Customers
    */

```

```

    final float EmployeesA = 0.98f;
    final float ReportsToA = 0.78f;
    final float ReportedByA = 0.70f;
    final float TerritoriesA = 0.55f;
    final float RegionA = 0.46f;
    final float OrderA = 0.94f;

```

```

final float CustomerA = 1.0f;
final float ShipperA = 0.88f;
final float OrderDetailsA = 0.88f;
final float ProductA = 0.77f;
final float SupplierA = 0.65f;
final float CategoriesA = 0.65f;

//συνάρτηση για υπολογισμό των τιμών Af-tf των Customers
public void AfTf_Calculation(ResultSet rs_cus,ArrayList l1){

    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        String url = "jdbc:odbc:Driver={Microsoft Access Driver " +
            " (*.mdb, *.accdb)};DBQ=C:\\Program Files
            \\Northwind_Original.mdb\\Northwind_Original.mdb";
        Connection con = DriverManager.getConnection(url);

        /*δήλωση αντικειμένων τύπου Statement και ResultSet για την
        *εκτέλεση των Queries
        */

        Statement orders_w= null;
        ResultSet orders =null;
        orders_w = con.createStatement
        (ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);

        Statement emp_w= null;
        ResultSet emp =null;
        emp_w = con.createStatement
        (ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);

        Statement reportsTo_w= null;
        ResultSet reportsTo =null;
        reportsTo_w = con.createStatement
        (ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);

        Statement reportedBy_w= null;
        ResultSet reportedBy =null;
        reportedBy_w = con.createStatement
        (ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);

        Statement emp_ter_w= null;
        ResultSet emp_ter=null;
        emp_ter_w = con.createStatement
        (ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);

        Statement ter_w= null;
        ResultSet ter=null;
        ter_w = con.createStatement
        (ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);

        Statement ordd_w= null;

```

```

    ResultSet ordd =null;
    ordd_w = con.createStatement
(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);

    Statement prod_w= null;
    ResultSet prod =null;
    prod_w = con.createStatement
(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);

    Statement cat_w= null;
    ResultSet cat =null;
    cat_w = con.createStatement
(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);

    Statement sup_w= null;
    ResultSet sup =null;
    sup_w = con.createStatement
(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);

    Statement shippers_w= null;
    ResultSet shippers =null;
    shippers_w = con.createStatement
(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);

//δήλωση strings για τα Queries

String s2;
String s4;
String s6;
String s8;
String s10;
String s11;
String s13;
String s14;
String s16;
String s18;
String s20;

/*μεταβλητές οι οποίες κρατούν τον συνολικό αριθμό λέξεων σε
*κάθε σχέση
*/

double count_total;
double count_cus;
double count_ord;
double count_emp;
double count_rt;
double count_rb;
double count_ter;
double count_prod;
double count_cat;
double count_sup;
double count_sh;
double count_ordd;

```

```

/*μεταβλητές οι οποίες κρατούν τον συνολικό αριθμό των
*εμφανίσεων της λέξης-κλειδί k2
*/

double count_total_k2;
double count_cus_k2;
double count_ord_k2;
double count_emp_k2;
double count_rt_k2;
double count_rb_k2;
double count_ter_k2;
double count_prod_k2;
double count_cat_k2;
double count_sup_k2;
double count_sh_k2;

//μέγεθος ArrayList με τις προεπιλεγμένες λέξεις
int k=l1.size();

int i;

double af_tf; //μεταβλητή που κρατά την τιμή Af-tf

//δημιουργία αντικειμένου της κλάσης Stemmer
Stemmer m=new Stemmer();

/*υπολογισμός τιμών Af-tf για όλα τα OSs των Customers για
*την κάθε λέξη του ArrayList
*/

for(i=0;i<k;i++){

    //από UpperCase σε LowerCase

    String k2=(String) l1.get(i);
    String keyword2=k2;
    String k1;
    String a=k2.substring(0,1);
    StringTokenizer str = new StringTokenizer(k2,a);
    String char1=k2.substring(0,1).toLowerCase();
    String string2;
    string2=str.nextToken();
    k1 = char1 + string2;

    rs_cus.beforeFirst();

    //δημιουργία OSs
    while(rs_cus.next()){

        //αρχικοποίηση
        count_cus=0;
        count_ord=0;
        count_emp=0;
        count_rt=0;
        count_rb=0;
        count_ter=0;
        count_prod=0;
        count_cat=0;
        count_sup=0;
        count_sh=0;
    }
}

```

```

count_ordd=0;

count_total_k2=0;
count_cus_k2=0;
count_ord_k2=0;
count_emp_k2=0;
count_rt_k2=0;
count_rb_k2=0;
count_ter_k2=0;
count_prod_k2=0;
count_cat_k2=0;
count_sup_k2=0;
count_sh_k2=0;

/*εφαρμογή μεθόδου stemming στη λέξη που έχει
*επιλεχθεί από το αρχείο
*/

String str_2=k2;
char array[]=str_2.toCharArray();
int size=array.length;
m.add(array,size);
m.stem();

k2=m.toString();

String str_1=k1;
char array2[]=str_1.toCharArray();
int size2=array2.length;
m.add(array2,size2);
m.stem();

k1=m.toString();

String id= rs_cus.getString(1);

count_cus=count_cus+2;

String t1= rs_cus.getString(2);
String t2= rs_cus.getString(4);
String t3 = rs_cus.getString(5);
String t4 = rs_cus.getString(6);
String t6 = rs_cus.getString(3);
String t7 = rs_cus.getString(7);
String t8 = rs_cus.getString(8);
String t9= rs_cus.getString(10);
String t10= rs_cus.getString(11);
String t11= rs_cus.getString(9);

StringTokenizer token = new StringTokenizer(t1);

while (token.hasMoreTokens()){

    t1 = token.nextToken();

    if(t1.contains(k1)||t1.contains(k2)){
        count_cus_k2++;
    }
}

```

```

    }

    count_cus++;

}

StringTokenizer token1 = new StringTokenizer(t2);

while (token1.hasMoreTokens()) {
    t2 = token1.nextToken();
    if (t2.contains(k1) || t2.contains(k2)) {
        count_cus_k2++;
    }

    count_cus++;

}

StringTokenizer token2 = new StringTokenizer(t3);

while (token2.hasMoreTokens()) {
    t3 = token2.nextToken();
    if (t3.contains(k1) || t3.contains(k2)) {
        count_cus_k2++;
    }

    count_cus++;

}

StringTokenizer token3 = new StringTokenizer(t4);

while (token3.hasMoreTokens()) {
    t4 = token3.nextToken();
    if (t4.contains(k1) || t4.contains(k2)) {
        count_cus_k2++;
    }

    count_cus++;

}

StringTokenizer token5 = new StringTokenizer(t6);

while (token5.hasMoreTokens()) {

    t6 = token5.nextToken();
    if (t6.contains(k1) || t6.contains(k2)) {
        count_cus_k2++;
    }

    count_cus++;

}

```

```

if(t7!=null){
    StringTokenizer token6 = new StringTokenizer(t7);

    while (token6.hasMoreTokens()){

        t7 = token6.nextToken();
        if(t7.contains(k1)||t7.contains(k2)){
            count_cus_k2++;
        }

        count_cus++;
    }
}
if(t8!=null){

    StringTokenizer token7 = new StringTokenizer(t8);

    while (token7.hasMoreTokens()){

        t8 = token7.nextToken();
        if(t8.contains(k1)||t8.contains(k2)){
            count_cus_k2++;
        }

        count_cus++;

    }

}

StringTokenizer token8 = new StringTokenizer(t9);

while (token8.hasMoreTokens()){

    t9 = token8.nextToken();
    if(t9.contains(k1)||t9.contains(k2)){
        count_cus_k2++;
    }

    count_cus++;

}

if(t10!=null){
StringTokenizer token9 = new StringTokenizer(t10);

    while (token9.hasMoreTokens()){

        t10 = token9.nextToken();
        if(t10.contains(k1)||t10.contains(k2)){
            count_cus_k2++;
        }

        count_cus++;
    }
}

```

```

        if(t11.contains(k1)||t11.contains(k2)){
            count_cus_k2++;
        }

s20= "SELECT * FROM Shippers WHERE ShipperID="+ship_id;
shippers=shippers_w.executeQuery(s20);

while(shippers.next()){
    count_sh=count_sh+3;
    String st1=shippers.getString(2);

    StringTokenizer tok_1 = new StringTokenizer(st1);

        while (tok_1.hasMoreTokens()){

            st1 = tok_1.nextToken();

            if(st1.contains(k1)||st1.contains(k2)){
                count_sh_k2++;
            }

            count_sh++;

        }

    }

}

/*πολλαπλασιασμός των αριθμών των εμφανίσεων της λέξης-
*κλειδί k2 σε κάθε σχέση του OS με την αντίστοιχη τιμή του
*Affinity
*/

count_emp_k2 = count_emp_k2 * EmployeesA;
count_rt_k2 = count_rt_k2 * ReportsToA;
count_rb_k2 = count_rb_k2 * ReportedByA;
count_ord_k2 = count_ord_k2 * OrderA;
count_cus_k2 = count_cus_k2 * CustomerA;
count_prod_k2 = count_prod_k2 * ProductA;
count_cat_k2 = count_cat_k2 * CategoriesA;
count_sup_k2 = count_sup_k2 * SupplierA;
count_ter_k2 = count_ter_k2 * TerritoriesA;
count_sh_k2 = count_sh_k2 * ShipperA;

/*πολλαπλασιασμός του συνολικού αριθμού των λέξεων
*σε κάθε σχέση του OS με την αντίστοιχη τιμή του Affinity
*/

count_emp = count_emp * EmployeesA;
count_rt = count_rt * ReportsToA;
count_rb = count_rb * ReportedByA;
count_ord = count_ord * OrderA;
count_cus = count_cus * CustomerA;

```

```

        count_prod = count_prod * ProductA;
        count_cat = count_cat * CategoriesA;
        count_sup = count_sup * SupplierA;
        count_ter = count_ter * TerritoriesA;
        count_sh = count_sh * ShipperA;

        //υπολογισμός παρονομαστή της εξίσωσης
        count_total =count_ordd+ count_cus + count_ord + count_emp
        + count_rt + count_rb + count_ter + count_prod + count_cat
        + count_sup + count_sh;

        //υπολογισμός αριθμητή της εξίσωσης
        count_total_k2= count_cus_k2 + count_ord_k2 + count_emp_k2
        + count_rt_k2 + count_rb_k2 + count_ter_k2 + count_prod_k2
        + count_cat_k2 + count_sup_k2 + count_sh_k2;

        /*υπολογισμός Af=tf για το συγκεκριμένο OS και τη
        *συγκεκριμένη λέξη
        */

        af_tf=(Double)count_total_k2/(Double)count_total;

        //αποθήκευση της τιμής σε HashMap
        Af_tf.put(id+"_"+keyword2, af_tf);

    }

}

con.close();
    } catch (SQLException e) {
        System.out.println("SQL Exception: "+ e.toString());
    } catch (ClassNotFoundException cE) {
        System.out.println("Class Not Found Exception: "+
            cE.toString());
    }
}

}

}

```

```

/*****
    Γλώσσα:      Java
    Περιγραφή :   Το πρόγραμμα αυτό αφορά την υλοποίηση της φάσης
                  επεξεργασία του συστήματος της προτεινόμενης
                  μεθοδολογίας. Παραθέτουμε ενδεικτικά ένα μέρος από
                  αυτό.
*****/

//διάβασμα λέξεων - κλειδιών
String keyword1=textField1.getText().toString();
String keyword2=textField2.getText().toString();

JFrame f=new JFrame();

//δήλωση περιοχής στην οποία θα παρουσιαστούν τα OSs
AttributedTextArea area=new AttributedTextArea();

int error=0;

long startTime=System.currentTimeMillis();

//διασύνδεση με τη βάση δεδομένων Northwind
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    String url = "jdbc:odbc:Driver={Microsoft Access Driver " +
        " (*.mdb, *.accdb)};DBQ=C:\\Program
Files\\Northwind_Original.mdb\\Northwind_Original.mdb";
    Connection con = DriverManager.getConnection(url);

    System.out.println();

    /*query για την εύρεση πλειάδων στη σχέση Employees που περιέχουν την
    *πρώτη λέξη - κλειδί σε συγκεκριμένα πεδία
    */

    Statement stm_sql1=null;
    String emp = "SELECT * FROM Employees WHERE LastName = '" + keyword1
    + "'" + "OR FirstName = '" + keyword1+"'";
    ResultSet rs_emp = null;
    stm_sql1 =
    con.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCU
    R_UPDATABLE);
    rs_emp = stm_sql1.executeQuery(emp);

    /*query για την εύρεση πλειάδων στη σχέση Customers που περιέχουν την
    *πρώτη λέξη - κλειδί σε συγκεκριμένα πεδία
    */

    Statement stm_sql2=null;
    String cus = "SELECT * FROM Customers WHERE CustomerID = '" +
    keyword1 + "'" + "OR ContactName LIKE '%" +keyword1+"'" + "OR
    ContactName LIKE '" +keyword1+" '%" + "OR ContactName LIKE '%"
    +keyword1+" %'";//+ "OR CompanyName LIKE '%" +keyword1+"'" + "OR
    CompanyName LIKE '" +keyword1+" %'"+ "OR CompanyName LIKE '%"
    +keyword1+" %'";

    ResultSet rs_cus = null;

```

```

stm_sql2 =
con.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);
rs_cus=stm_sql2.executeQuery(cus);

//ανάκτηση όλων των πλειάδων που περιέχονται στη σχέση Employees
Statement stm_sql3=null;
String emp2 = "SELECT * FROM Employees";
ResultSet rs_emp2 = null;
stm_sql3 =
con.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);
rs_emp2 = stm_sql3.executeQuery(emp2);

//ανάκτηση όλων των πλειάδων που περιέχονται στη σχέση Customers
Statement stm_sql4=null;
String cus2 = "SELECT * FROM Customers";
ResultSet rs_cus2 = null;
stm_sql4 =
con.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);
rs_cus2=stm_sql4.executeQuery(cus2);

//δήλωση ArrayList για την αποθήκευση των IDs όλων των Employees
ArrayList All_Employees=new ArrayList();

//δήλωση ArrayList για την αποθήκευση των IDs όλων των Customers
ArrayList All_Customers=new ArrayList();

//δήλωση HashMap για αποθήκευση του περιεχομένου του αρχείου
HashMap Index=new HashMap();

//διάβασμα κατάλληλου αρχείου σύμφωνα με την επιλογή του χρήστη
if(j1==1){

    try{
        FileInputStream fstream = new FileInputStream("Index1.txt");
        DataInputStream in = new DataInputStream(fstream);
        BufferedReader br = new BufferedReader(new
        InputStreamReader(in));

        String strLine;

        while ((strLine = br.readLine()) != null)    {

            StringTokenizer token = new StringTokenizer(strLine);

            while (token.hasMoreTokens()){
                String str1 = token.nextToken();
                String str2= token.nextToken();
                Double s=Double.parseDouble(str2);
                Index.put(str1, s);
            }

        }

        in.close();
    }catch (Exception e){
        System.err.println("Error: " + e.getMessage());
    }
}

```

```

}else if(j1==0){

    try{
        FileInputStream fstream = new FileInputStream("Index.txt");

        DataInputStream in = new DataInputStream(fstream);

        BufferedReader br = new BufferedReader(new InputStreamReader(in));

        String strLine;

        while ((strLine = br.readLine()) != null)    {

            StringTokenizer token = new StringTokenizer(strLine);

            while (token.hasMoreTokens()){
                String str1 = token.nextToken();
                String str2= token.nextToken();
                Double s=Double.parseDouble(str2);
                Index.put(str1, s);
            }

        }

        in.close();
    }catch (Exception e){
        System.err.println("Error: " + e.getMessage());
    }
}

OS o = new OS();

ArrayList IDs=new ArrayList();
rs_emp.beforeFirst();

while(rs_emp.next()){

    IDs.add(rs_emp.getString(1));
}

while(rs_cus.next()){
    IDs.add(rs_cus.getString(1));
}

while(rs_emp2.next()){

    All_Employees.add(rs_emp2.getString(1));
}

while(rs_cus2.next()){

    All_Customers.add(rs_cus2.getString(1));
}

```

```

if(IDs.size()>0){

    //δημιουργία αντικειμένου τύπου Ranker
    Ranker r=new Ranker();

    //εύρεση τιμών Af-tf από το αρχείο
    r.findAftf(IDs, Index, keyword2);

    //ταξινόμηση τιμών Af-tf
    LinkedHashMap r1=r.sortHashMapByValuesD(r.af_tf);

    Iterator iterator = r1.keySet().iterator();

    int count=1;
    area.setFont(new Font("",Font.BOLD,13));
    area.setForeground(Color.red);

    while( iterator.hasNext() ){
        area.setFont(new Font("",Font.BOLD,13));
        area.setForeground(Color.getHSBColor(0.7f, 1.0f, 0.5f));
        area.append("\n");
        area.append("\n      ~~~~~~\n");
        area.append("      Object Summary "+ count);
        area.append("\n");
        area.append("      ~~~~~~\n");

        count++;

        area.append("\n");
        String id=(String) iterator.next();

        NumberFormat nf = NumberFormat.getNumberInstance() ;
        nf.setGroupingUsed(false) ;
        nf.setMaximumFractionDigits(4) ;
        nf.setMinimumFractionDigits(2) ;

        //δημιουργία OS ενός Employee
        if(All_Employees.contains(id)){

            area.append("      [ Ranking value = "+
            nf.format(r1.get(id))+" ]\n");
            o.createEmployeeOS(id,area);
            area.append("\n\n
            _____
            _____
            _____");

        //δημιουργία OS ενός Customer
        }else if(All_Customers.contains(id)){

            area.append("      [ Ranking value = "+
            nf.format(r1.get(id))+" ]\n");
            o.createCustomerOS(id,area);
            area.append("\n\n
            _____
            _____
            _____");

```

```

        }
    }
} else {
    new Error().setVisible(true);
    error=1;
}

if(error==0){
    JScrollPane s = new JScrollPane(area);

    s.setVerticalScrollBarPolicy(JScrollPane.VERTICAL_SCROLLB
AR_ALWAYS);

    s.setHorizontalScrollBarPolicy(JScrollPane.HORIZONTAL_SCR
OLLBAR_ALWAYS);

    f.getContentPane().add(s);
    f.setBounds( 300, 300, 600, 600);
    f.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
    f.setVisible(true);

    }

con.close();
    } catch (SQLException e) {
        System.out.println("SQL Exception: "+ e.toString());
    } catch (ClassNotFoundException cE) {
        System.out.println("Class Not Found Exception: "+
            cE.toString());
    }

long endTime=System.currentTimeMillis();
    System.out.println((endTime-startTime)/1000);

}

```

Παράρτημα Β

Στο παράρτημα αυτό παρεθέτονται δεδομένα που έχουν χρησιμοποιηθεί αλλά και εξαχθεί κατά την υλοποίηση της φάσης προεπεξεργασίας του συστήματος. Συγκεκριμένα δίνεται το περιεχόμενο του αρχείου προεπιλεγμένων λέξεων και μέρος των δύο ευρετηρίων με τις τιμές $Af-tf$ και $Af-\overline{tf}$.

- Περιεχόμενο αρχείου προεπιλεγμένων λέξεων που χρησιμοποιήθηκε κατά τη φάση προεπεξεργασίας

Προεπιλεγμένες λέξεις			
1	Sales	37	Marketing
2	Commercial	38	Management
3	Company	39	Food
4	Representative	40	University
5	International	41	Economics
6	Selling	42	Psychology
7	Office	43	Business
8	Owner	44	Agent
9	Administrator	45	Order
10	Assistant	46	Accounting
11	USA	47	UK
12	Denmark	48	France
13	Belgium	49	Germany
14	Mexico	50	Brazil
15	Finland	51	Poland
16	Switzerland	52	Venezuela
17	Austria	53	Sweden
18	Italy	54	Spain
19	Portugal	55	Canada
20	Ireland	56	Norway
21	Argentina	57	Stock
22	Vins	58	Chinese
23	Beverages	59	Grocery
24	Bistro	60	Store
25	Import	61	Biscuits
26	Scones	62	Chocolate
27	Mozzarella	63	Lager
28	Sauce	64	Ravioli
29	Tofu	65	Apples
30	Meat	66	Pears
31	Seasoning	67	Syrup
32	Eastern	68	Western
33	Northern	69	Southern
34	Express	70	Package
35	Shipping	71	Liquids
36	Delights	72	Pasta

➤ Μέρος του περιεχομένου του ευρετηρίου των τιμών *Af-tf*

<i>ID_Αέξη</i>	<i>Af-tf</i>
1_Sales	0.00938
1_Marketing	0.00691
1_Commercial	0.00006
1_Management	0.00859
1_Company	0.00006
1_Food	0.00288
1_Representative	0.00572
1_University	0.00000
1_International	0.00117
1_Economics	0.00000
1_Selling	0.00001
1_Psychology	0.00000
1_Office	0.00000
1_Business	0.00000
1_Owner	0.00158
1_Agent	0.00090
1_Administrator	0.00117
1_Order	0.00140
1_Assistant	0.00060
.....	
2_Sales	0.00902
2_Marketing	0.00716
2_Commercial	0.00008
2_Management	0.00883
2_Company	0.00010
2_Food	0.00237
2_Representative	0.00468
2_University	0.00000
2_International	0.00124
2_Economics	0.00000
2_Selling	0.00015
2_Psychology	0.00000
2_Office	0.00016
2_Business	0.00008
2_Owner	0.00222
2_Agent	0.00090
2_Administrator	0.00112
2_Order	0.00123
2_Assistant	0.00085
.....	
3_Sales	0.00974
3_Marketing	0.00605

3_Commercial	0.00006
3_Management	0.00823
3_Company	0.00006
3_Food	0.00330
3_Representative	0.00507
3_University	0.00000
3_International	0.00070
3_Economics	0.00000
3_Selling	0.00012
3_Psychology	0.00000
3_Office	0.00000
3_Business	0.00000
3_Owner	0.00174
3_Agent	0.00133
3_Administrator	0.00162
3_Order	0.00211
3_Assistant	0.00080
3_Accounting	0.00279
.....	
ALFKI_Sales	0.02591
ALFKI_Marketing	0.00646
ALFKI_Commercial	0.00283
ALFKI_Management	0.01209
ALFKI_Company	0.00339
ALFKI_Food	0.00165
ALFKI_Representative	0.01004
ALFKI_University	0.00000
ALFKI_International	0.00623
ALFKI_Economics	0.00070
ALFKI_Selling	0.00070
ALFKI_Psychology	0.00000
ALFKI_Office	0.00198
ALFKI_Business	0.00000
ALFKI_Owner	0.00000
ALFKI_Agent	0.00000
ALFKI_Administrator	0.00000
ALFKI_Order	0.00000
ALFKI_Assistant	0.00000
.....	
BERGS_Sales	0.02480
BERGS_Marketing	0.00643
BERGS_Commercial	0.00245
BERGS_Management	0.01203
BERGS_Company	0.00353
BERGS_Food	0.00303
BERGS_Representative	0.00960
BERGS_University	0.00000
BERGS_International	0.00491
BERGS_Economics	0.00020
BERGS_Selling	0.00079

BERGS_Psychology	0.00000
BERGS_Office	0.00141
BERGS_Business	0.00050
BERGS_Owner	0.00025
BERGS_Agent	0.00050
BERGS_Administrator	0.00031
BERGS_Order	0.00082
.....	

➤ **Μέρος του περιεχομένου του ευρετηρίου με των τιμών $\overline{Af-tf}$**

<i>ID_Λέξη</i>	$\overline{Af-tf}$
1_Sales	0.00705
1_Marketing	0.00520
1_Commercial	0.00004
1_Management	0.00646
1_Company	0.00004
1_Food	0.00024
1_Representative	0.00430
1_University	0.00088
1_International	0.00000
1_Economics	0.00014
1_Selling	0.00000
1_Psychology	0.00000
1_Office	0.00000
1_Business	0.00000
1_Owner	0.00119
1_Agent	0.00071
1_Administrator	0.00088
1_Order	0.00105
1_Assistant	0.00047
.....	
2_Sales	0.00684
2_Marketing	0.00543
2_Commercial	0.00006
2_Management	0.00671
2_Company	0.00012
2_Food	0.00179
2_Representative	0.00355
2_University	0.00000
2_International	0.00094
2_Economics	0.00000
2_Selling	0.00012
2_Psychology	0.00000
2_Office	0.00012

2_Business	0.00006
2_Owner	0.00169
2_Agent	0.00069
2_Administrator	0.00085
2_Order	0.00093
2_Assistant	0.00064
.....	
3_Sales	0.00735
3_Marketing	0.00457
3_Commercial	0.00047
3_Management	0.00622
3_Company	0.00004
3_Food	0.00249
3_Representative	0.00382
3_University	0.00000
3_International	0.00056
3_Economics	0.00000
3_Selling	0.00009
3_Psychology	0.00000
3_Office	0.00000
3_Business	0.00000
3_Owner	0.00131
3_Agent	0.00100
3_Administrator	0.00123
3_Order	0.00159
3_Assistant	0.00061
.....	
ALFKI_Sales	0.02087
ALFKI_Marketing	0.00521
ALFKI_Commercial	0.00228
ALFKI_Management	0.00973
ALFKI_Company	0.00273
ALFKI_Food	0.00133
ALFKI_Representative	0.00809
ALFKI_University	0.00000
ALFKI_International	0.00502
ALFKI_Economics	0.00057
ALFKI_Selling	0.00057
ALFKI_Psychology	0.00000
ALFKI_Office	0.00160
ALFKI_Business	0.00000
ALFKI_Owner	0.00000
ALFKI_Agent	0.00000
ALFKI_Administrator	0.00000
ALFKI_Order	0.00000
ALFKI_Assistant	0.00000
.....	
BERGS_Sales	0.01940
BERGS_Marketing	0.00503
BERGS_Commercial	0.00191

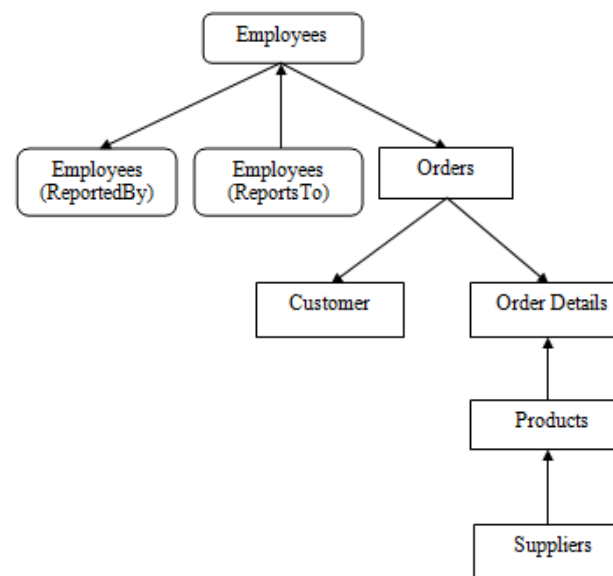
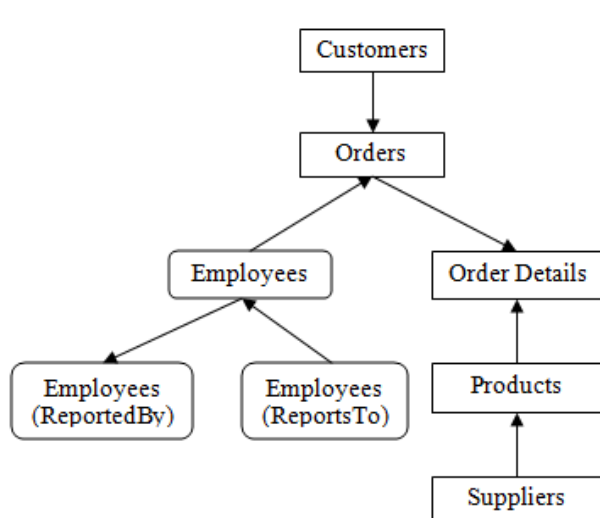
BERGS_Management	0.00942
BERGS_Company	0.00276
BERGS_Food	0.00237
BERGS_Representative	0.00751
BERGS_University	0.00000
BERGS_International	0.00384
BERGS_Economics	0.00021
BERGS_Selling	0.00062
BERGS_Psychology	0.00000
BERGS_Office	0.00110
BERGS_Business	0.00040
BERGS_Owner	0.00019
BERGS_Agent	0.00039
BERGS_Administrator	0.00024
BERGS_Order	0.00064
BERGS_Assistant	0.00000
ALFKI_Assistant	0.00000
.....	

Παράρτημα Γ

Στο παράρτημα αυτό παραθέτονται τα αρχεία που έχουν δοθεί στους αξιολογητές με σκοπό την αξιολόγηση της αποτελεσματικότητας του συστήματος. Ωστόσο, θα πρέπει να αναφέρουμε πως λόγω του μεγάλου μεγέθους των αρχείων δίνεται μόνο ένα μέρος από αυτά.

➤ $Q_1 = \{\text{"Maria"}, \text{"Sales"}\}$

	Στατιστικά στοιχεία για 5 OSs							
	Αριθμός εμφανίσεων λέξης - κλειδί "Sales"					Συνολικός αριθμός εμφανίσεων λέξης – κλειδί "Sales"	Μέγεθος OS σε αριθμό πλειάδων (tuples)	Μέγεθος OS σε αριθμό λέξεων
	Σχέση Customers	Σχέση Employees	Σχέση Employees (ReportsTo)	Σχέση Employees (ReportedBy)	Σχέση Suppliers			
OS ₁	1	25	45	28	13	112	166	4356
OS ₂	1	9	29	0	5	44	79	1710
OS ₃	0	9	10	10	1	30	40	1143
OS ₄	18	4	5	4	57	88	297	7649
OS ₅	0	9	14	10	3	36	49	1399



OS₁

>Customer : RICAR | Ricardo Adocicados | Maria Limeira | Assistant Sales Agent | Av. Copacabana, 267 | Rio de Janeiro | RJ | 02389-890 | Brazil | (21) 555-3412 |

>Order : 10287 | RICAR | 8 | 1996-08-22 00:00:00 | 1996-09-19 00:00:00 | 1996-08-28 00:00:00 | 3 | 12.7600 | Ricardo Adocicados | Av. Copacabana, 267 | Rio de Janeiro | RJ | 02389-890 | Brazil

>Employee : 8 | Fuller | Anne | Inside Sales Coordinator | Ms. | 1958-01-09 00:00:00 | 1994-03-05 00:00:00 | 4726 - 11th Ave. N.E. | Seattle | WA | 98105 | USA | (206) 555-1189 | 2344 | Anne received a BA in psychology from the University of Washington. She has also completed a course in business French. She reads and writes French. | 2 | <http://accweb/emmployees/davolio.bmp>

>ReportsTo : 2 | Fuller | Robert | Vice President, Sales | Dr. | 1952-02-19 00:00:00 | 1992-08-14 00:00:00 | 908 W. Capital Way | Tacoma | WA | 98401 | USA | (206) 555-9482 | 3457 | Robert received his BTS commercial in 1974 and a Ph.D. in international marketing from the University of Dallas in 1981. He is fluent in French and Italian and reads German. He joined the company as a sales representative, was promoted to sales manager in January 1992 and to vice president of sales in March 1993. Andrew is a member of the Sales Management Roundtable, the Seattle Chamber of Commerce, and the Pacific Rim Importers Association. | null | <http://accweb/emmployees/fuller.bmp>

>EmployeeTerritories : 8 | 19428 | Philadelphia

8 | 44122 | Beachwood

8 | 45839 | Findlay

8 | 53404 | Racine

>Order Details : 10287 | 16 | 13.9000 | 40 | 0.15

>Product : 16 | Pavlova | 3 | 7 | 32 - 500 g boxes | 17.4500 | 29 | 0 | 10 | 0

>Category : 3 | Confections | Desserts, candies, and sweet breads

>Supplier : 7 | Pavlova, Ltd. | Ian Devling | Marketing Manager | 74 Rose St. Moonie Ponds | Melbourne | Victoria | 3058 | Australia | (03) 444-2343 | (03) 444-6588

>Order Details : 10287 | 34 | 11.2000 | 20 | 0.0

>Product : 34 | Sasquatch Ale | 1 | 16 | 24 - 12 oz bottles | 14.0000 | 111 | 0 | 15 | 0

>Supplier : 8 | Specialty Biscuits, Ltd. | Peter Wilson | Sales Representative | 29 King's Way | Manchester | null | M14 GSD | UK | (161) 555-4448 |

>Order Details : 10299 | 70 | 12.0000 | 20 | 0.0

>Product : 70 | Outback Lager | 1 | 7 | 24 - 355 ml bottles | 15.0000 | 15 | 10 | 30 | 0

>Category : 1 | Beverages | Soft drinks, coffees, teas, beers, and ales

>Supplier : 7 | Pavlova, Ltd. | Ian Devling | Marketing Manager | 74 Rose St. Moonie Ponds | Melbourne | Victoria | 3058 | Australia | (03) 444-2343 | (03) 444-6588

>Shipper : 2 | United Package | (503) 555-3199

>Order : 10447 | RICAR | 4 | 1997-02-14 00:00:00 | 1997-03-14 00:00:00 | 1997-03-07 00:00:00 | 2 | 68.6600 | Ricardo Adocicados | Av. Copacabana, 267 | Rio de Janeiro | RJ | 02389-890 | Brazil

>Employee : 4 | Peacock | Pedro | Sales Representative | Mr. | 1937-09-19 00:00:00 | 1993-05-03 00:00:00 | 4110 Old Redmond Rd. | Redmond | WA | 98052 | USA | (206) 555-8122 | 5176 | Margaret holds a BA in English literature from Concordia College (1958) and an MA from the American Institute of Culinary Arts (1966). She was assigned to the London office temporarily from July through November 1992. | 2 | <http://accweb/emmployees/peacock.bmp>

>ReportsTo : 2 | Fuller | Robert | Vice President, Sales | Dr. | 1952-02-19 00:00:00 | 1992-08-14 00:00:00 | 908 W. Capital Way | Tacoma | WA | 98401 | USA | (206) 555-9482 | 3457 | Robert received his BTS commercial in 1974 and a Ph.D. in international marketing from the University of Dallas in 1981. He is fluent in French and Italian and reads German. He joined the company as a sales representative, was promoted to sales manager in January 1992 and to vice president of sales in March 1993. Andrew is a member of the Sales Management Roundtable, the Seattle Chamber of Commerce, and the Pacific Rim Importers Association. | null | <http://accweb/emmployees/fuller.bmp>

>EmployeeTerritories : 4 | 20852 | Rockville

4 | 27403 | Greensboro

4 | 27511 | Cary

>Order Details : 10447 | 19 | 7.3000 | 40 | 0.0

>Product : 19 | Teatime Chocolate Biscuits | 3 | 8 | 10 boxes x 12 pieces | 9.2000 | 25 | 0 | 5 | 0

>Category : 3 | Confections | Desserts, candies, and sweet breads

>Supplier : 8 | Specialty Biscuits, Ltd. | Peter Wilson | Sales Representative | 29 King's Way | Manchester | null | M14 GSD | UK | (161) 555-4448 |

>Order Details : 10447 | 65 | 16.8000 | 35 | 0.0

>Product : 65 | Louisiana Fiery Hot Pepper Sauce | 2 | 2 | 32 - 8 oz bottles | 21.0500 | 76 | 0 | 0 | 0

>Category : 2 | Condiments | Sweet and savory sauces, relishes, spreads, and seasonings

>Supplier : 2 | New Orleans Cajun Delights | Shelley Burke | Order Administrator | P.O. Box 78934 | New Orleans | LA | 70117 | USA | (100) 555-4822 |

>Order Details : 10447 | 71 | 17.2000 | 2 | 0.0

>Product : 71 | Flotemysost | 4 | 15 | 10 - 500 g pkgs. | 21.5000 | 26 | 0 | 0 | 0

>Category : 4 | Dairy Products | Cheeses

>Supplier : 15 | Norske Meierier | Beate Vileid | Marketing Manager | Hatlevegen 5 | Sandvika | null | 1320 | Norway | (0)2-953010 |

>Shipper : 2 | United Package | (503) 555-3199

OS₂

>Customer : ALFKI | Alfreds Futterkiste | Maria Anders | Sales Representative | Obere Str. 57 | Berlin | null | 12209 | Germany | 030-0074321 | 030-0076545

>Order : 10643 | ALFKI | 6 | 1997-08-25 00:00:00 | 1997-09-22 00:00:00 | 1997-09-02 00:00:00 | 1 | 29.4600 | Alfreds Futterkiste | Obere Str. 57 | Berlin | null | 12209 | Germany

>Employee : 6 | Peacock | Michael | Sales Representative | Mr. | 1963-07-02 00:00:00 | 1993-10-17 00:00:00 | Coventry House Miner Rd. | London | null | EC2 7JR | UK | (71) 555-7773 | 428 | Michael is a graduate of Sussex University (MA, economics, 1983) and the University of California at Los Angeles (MBA, marketing, 1986). He has also taken the courses "Multi-Cultural Selling" and "Time Management for the Sales Professional." He is fluent in Japanese and can read and write French, Portuguese, and Spanish. | 5 | <http://accweb/emmployees/davolio.bmp>

>ReportsTo : 5 | Leverling | Maria | Sales Manager | Ms. | 1955-03-04 00:00:00 | 1993-10-17 00:00:00 | 14 Garrett Hill | London | null | SW1 8JR | UK | (71) 555-4848 | 3453 | Maria graduated from St. Andrews University, Scotland, with a BSC degree in 1976. Upon joining the company as a sales representative in 1992, she spent 6 months in an orientation program at the Seattle office and then returned to her permanent post in London. She was promoted to sales manager in March 1993. Ms. Leverling has completed the courses "Successful Telemarketing" and "International Sales Management." She is fluent in French. | 2 | <http://accweb/emmployees/buchanan.bmp>

>EmployeeTerritories : 6 | 85014 | Phoenix

6 | 85251 | Scottsdale

6 | 98004 | Bellevue

6 | 98052 | Redmond

6 | 98104 | Seattle

>Order Details : 10643 | 28 | 45.6000 | 15 | 0.25

>Product : 28 | Rossle Sauerkraut | 7 | 12 | 25 - 825 g cans | 45.6000 | 26 | 0 | 0 | 1

>Category : 7 | Produce | Dried fruit and bean curd

>Supplier : 12 | Plutzer Lebensmittelgroßmarkte AG | Martin Bein | International Marketing Mgr. | Bogenallee 51 | Frankfurt | null | 60439 | Germany | (069) 992755 |

>Order Details : 10643 | 39 | 18.0000 | 21 | 0.25

>Product : 39 | Chartreuse verte | 1 | 18 | 750 cc per bottle | 18.0000 | 69 | 0 | 5 | 0

>Category : 1 | Beverages | Soft drinks, coffees, teas, beers, and ales

>Supplier : 18 | Aux joyeux ecclésiastiques | Guylene Nodier | Sales Manager | 203, Rue des Francs-Bourgeois | Paris | null | 75004 | France | (1) 03.83.00.68 | (1) 03.83.00.62

>Order Details : 10643 | 46 | 12.0000 | 2 | 0.25

>Product : 46 | Spegesild | 8 | 21 | 4 - 450 g glasses | 12.0000 | 95 | 0 | 0 | 0

>Category : 8 | Seafood | Seaweed and fish

>Supplier : 21 | Lyngbysild | Niels Petersen | Sales Manager | Lyngbysild Fiskebakken 10 | Lyngby | null | 2800 | Denmark | 43844108 | 43844115

>Shipper : 1 | Speedy Express | (503) 555-9831

>Order : 10692 | ALFKI | 4 | 1997-10-03 00:00:00 | 1997-10-31 00:00:00 | 1997-10-13 00:00:00 | 2 | 61.0200 | Alfred's Futterkiste | Obere Str. 57 | Berlin | null | 12209 | Germany

>Employee : 4 | Peacock | Pedro | Sales Representative | Mr. | 1937-09-19 00:00:00 | 1993-05-03 00:00:00 | 4110 Old Redmond Rd. | Redmond | WA | 98052 | USA | (206) 555-8122 | 5176 | Margaret holds a BA in English literature from Concordia College (1958) and an MA from the American Institute of Culinary Arts (1966). She was assigned to the London office temporarily from July through November 1992. | 2 | <http://accweb/emmployees/peacock.bmp>

>ReportsTo : 2 | Fuller | Robert | Vice President, Sales | Dr. | 1952-02-19 00:00:00 | 1992-08-14 00:00:00 | 908 W. Capital Way | Tacoma | WA | 98401 | USA | (206) 555-9482 | 3457 | Robert received his BTS commercial in 1974 and a Ph.D. in international marketing from the University of Dallas in 1981. He is fluent in French and Italian and reads German. He joined the company as a sales representative, was promoted to sales manager in January 1992 and to vice president of sales in March 1993. Andrew is a member of the Sales Management Roundtable, the Seattle Chamber of Commerce, and the Pacific Rim Importers Association. | null | <http://accweb/emmployees/fuller.bmp>

>EmployeeTerritories : 4 | 20852 | Rockville
4 | 27403 | Greensboro

4 | 27511 | Cary

>Order Details : 10692 | 63 | 43.9000 | 20 | 0.0

>Product : 63 | Vegie-spread | 2 | 7 | 15 - 625 g jars | 43.9000 | 24 | 0 | 5 | 0

>Category : 2 | Condiments | Sweet and savory sauces, relishes, spreads, and seasonings

>Supplier : 7 | Pavlova, Ltd. | Ian Devling | Marketing Manager | 74 Rose St. Moonie Ponds | Melbourne | Victoria | 3058 | Australia | (03) 444-2343 | (03) 444-6588

>Shipper : 2 | United Package | (503) 555-3199

>Order : 10702 | ALFKI | 4 | 1997-10-13 00:00:00 | 1997-11-24 00:00:00 | 1997-10-21 00:00:00 | 1 | 23.9400 | Alfred's Futterkiste | Obere Str. 57 | Berlin | null | 12209 | Germany

>Employee : 4 | Peacock | Pedro | Sales Representative | Mr. | 1937-09-19 00:00:00 | 1993-05-03 00:00:00 | 4110 Old Redmond Rd. | Redmond | WA | 98052 | USA | (206) 555-8122 | 5176 | Margaret holds a BA in English literature from Concordia College (1958) and an MA from the American Institute of Culinary Arts (1966). She was assigned to the London office temporarily from July through November 1992. | 2 | <http://accweb/emmployees/peacock.bmp>

>ReportsTo : 2 | Fuller | Robert | Vice President, Sales | Dr. | 1952-02-19 00:00:00 | 1992-08-14 00:00:00 | 908 W. Capital Way | Tacoma | WA | 98401 | USA | (206) 555-9482 | 3457 | Robert received his BTS commercial in 1974 and a Ph.D. in international marketing from the University of Dallas in 1981. He is fluent in French and Italian and reads German. He joined the company as a sales representative, was promoted to sales manager in January 1992 and to vice president of sales in March 1993. Andrew is a member of the Sales Management Roundtable, the Seattle Chamber of Commerce, and the Pacific Rim Importers Association. | null | <http://accweb/emmployees/fuller.bmp>

>EmployeeTerritories : 4 | 20852 | Rockville

4 | 27403 | Greensboro

4 | 27511 | Cary

>Order Details : 10702 | 3 | 10.0000 | 6 | 0.0

>Product : 3 | Aniseed Syrup | 2 | 1 | 12 - 550 ml bottles | 10.0000 | 13 | 70 | 25 | 0

>Category : 2 | Condiments | Sweet and savory sauces, relishes, spreads, and seasonings

>Supplier : 1 | Exotic Liquids | Charlotte Cooper | Purchasing Manager | 49 Gilbert St. | London | null | EC1 4SD | UK | (171) 555-2222 |

>Order Details : 10702 | 76 | 18.0000 | 15 | 0.0

>Product : 76 | Lakkalikoori | 1 | 23 | 500 ml | 18.0000 | 57 | 0 | 20 | 0
>Category : 1 | Beverages | Soft drinks, coffees, teas, beers, and ales
>Supplier : 23 | Karkki Oy | Anne Heikkonen | Product Manager | Valtakatu 12 | Lappeenranta | null | 53120 | Finland | (953) 10956 |
>Shipper : 1 | Speedy Express | (503) 555-9831

OS₃

>Customer : FRANR | France restauration | Maria Schmitt | Marketing Manager | 54, rue Royale | Nantes | null | 44000 | France | 40.32.21.21 | 40.32.21.20

>Order : 10671 | FRANR | 1 | 1997-09-17 00:00:00 | 1997-10-15 00:00:00 | 1997-09-24 00:00:00 | 1 | 30.3400 | France restauration | 54, rue Royale | Nantes | null | 44000 | France

>Employee : 1 | Fuller | Maria | Sales Representative | Ms. | 1948-12-08 00:00:00 | 1992-05-01 00:00:00 | 507 - 20th Ave.E. | Seattle | WA | 98122 | USA | (206) 555-9857 | 5467 | Education includes a BA in psychology from Colorado State University in 1970. She also completed "The Art of the Cold Call." Nancy is a member of Toastmasters International. | 2 | <http://accweb/emmployees/davolio.bmp>

>ReportsTo : 2 | Fuller | Robert | Vice President, Sales | Dr. | 1952-02-19 00:00:00 | 1992-08-14 00:00:00 | 908 W. Capital Way | Tacoma | WA | 98401 | USA | (206) 555-9482 | 3457 | Robert received his BTS commercial in 1974 and a Ph.D. in international marketing from the University of Dallas in 1981. He is fluent in French and Italian and reads German. He joined the company as a sales representative, was promoted to sales manager in January 1992 and to vice president of sales in March 1993. Andrew is a member of the Sales Management Roundtable, the Seattle Chamber of Commerce, and the Pacific Rim Importers Association. | null | <http://accweb/emmployees/fuller.bmp>

>EmployeeTerritories : 1 | 06897 | Wilton

1 | 19713 | Neward

>Order Details : 10671 | 16 | 17.4500 | 10 | 0.0

>Product : 16 | Pavlova | 3 | 7 | 32 - 500 g boxes | 17.4500 | 29 | 0 | 10 | 0

>Category : 3 | Confections | Desserts, candies, and sweet breads

>Supplier : 7 | Pavlova, Ltd. | Ian Devling | Marketing Manager | 74 Rose St. Moonie Ponds | Melbourne | Victoria | 3058 | Australia | (03) 444-2343 | (03) 444-6588

>Order Details : 10671 | 62 | 49.3000 | 10 | 0.0

>Product : 62 | Tarte au sucre | 3 | 29 | 48 pies | 49.3000 | 17 | 0 | 0 | 0

>Category : 3 | Confections | Desserts, candies, and sweet breads

>Supplier : 29 | Forets d'erable | Chantal Goulet | Accounting Manager | 148 rue Chasseur | Ste-Hyacinthe | Quebec | J2S 7S8 | Canada | (514) 555-2955 | (514) 555-2921

>Order Details : 10671 | 65 | 21.0500 | 12 | 0.0

>Product : 65 | Louisiana Fiery Hot Pepper Sauce | 2 | 2 | 32 - 8 oz bottles | 21.0500 | 76 | 0 | 0 | 0

>Category : 2 | Condiments | Sweet and savory sauces, relishes, spreads, and seasonings

>Supplier : 2 | New Orleans Cajun Delights | Shelley Burke | Order Administrator | P.O. Box 78934 | New Orleans | LA | 70117 | USA | (100) 555-4822 |

>Shipper : 1 | Speedy Express | (503) 555-9831

>Order : 10860 | FRANR | 3 | 1998-01-29 00:00:00 | 1998-02-26 00:00:00 | 1998-02-04 00:00:00 | 3 | 19.2600 | France restauration | 54, rue Royale | Nantes | null | 44000 | France

>Employee : 3 | Leverling | Maria | Sales Representative | Ms. | 1963-08-30 00:00:00 | 1992-04-01 00:00:00 | 722 Moss Bay Blvd. | Kirkland | WA | 98033 | USA | (206) 555-3412 | 3355 | Maria has a BS degree in chemistry from Boston College (1984). She has also completed a certificate program in food retailing management. Maria was hired as a sales associate in 1991 and promoted to sales representative in February 1992. | 2 | <http://accweb/emmployees/leverling.bmp>

>ReportsTo : 2 | Fuller | Robert | Vice President, Sales | Dr. | 1952-02-19 00:00:00 | 1992-08-14 00:00:00 | 908 W. Capital Way | Tacoma | WA | 98401 | USA | (206) 555-9482 | 3457 | Robert received his BTS commercial in 1974 and a Ph.D. in international marketing from the University of Dallas in 1981. He is fluent in French and Italian and reads German. He joined the company as a sales representative, was promoted to sales manager in January 1992 and to vice president of sales in March 1993. Andrew is a member of the Sales Management Roundtable, the Seattle Chamber of Commerce, and the Pacific Rim Importers Association. | null | <http://accweb/emmployees/fuller.bmp>

>EmployeeTerritories : 3 | 30346 | Atlanta

3 | 31406 | Savannah

3 | 32859 | Orlando

3 | 33607 | Tampa

>Order Details : 10860 | 51 | 53.0000 | 3 | 0.0

>Product : 51 | Manjimup Dried Apples | 7 | 24 | 50 - 300 g pkgs. | 53.0000 | 20 | 0 | 10 | 0

>Category : 7 | Produce | Dried fruit and bean curd

>Supplier : 24 | G'day, Mate | Wendy Mackenzie | Sales Representative | 170 Prince Edward Parade Hunter's Hill | Sydney | NSW | 2042 | Australia | (02) 555-5914 | (02) 555-4873

>Order Details : 10860 | 76 | 18.0000 | 20 | 0.0

>Product : 76 | Lakkalikoori | 1 | 23 | 500 ml | 18.0000 | 57 | 0 | 20 | 0

>Category : 1 | Beverages | Soft drinks, coffees, teas, beers, and ales

>Supplier : 23 | Karkki Oy | Anne Heikkinen | Product Manager | Valtakatu 12 | Lappeenranta | null | 53120 | Finland | (953) 10956 |

>Shipper : 3 | Federal Shipping | (503) 555-9931

>Order : 10971 | FRANR | 2 | 1998-03-24 00:00:00 | 1998-04-21 00:00:00 | 1998-04-02 00:00:00 | 2 | 121.8200 | France restauration | 54, rue Royale | Nantes | null | 44000 | France

>Employee : 2 | Fuller | Robert | Vice President, Sales | Dr. | 1952-02-19 00:00:00 | 1992-08-14 00:00:00 | 908 W. Capital Way | Tacoma | WA | 98401 | USA | (206) 555-9482 | 3457 | Robert received his BTS commercial in 1974 and a Ph.D. in international marketing from the University of Dallas in 1981. He is fluent in French and Italian and reads German. He joined the company as a sales representative, was promoted to sales manager in January 1992 and to vice president of sales in March 1993. Andrew is a member of the Sales Management Roundtable, the Seattle Chamber of Commerce, and the Pacific Rim Importers Association. | null | <http://accweb/emmployees/fuller.bmp>

>ReportedBy : 1 | Fuller | Maria | Sales Representative | Ms. | 1948-12-08 00:00:00 | 1992-05-01 00:00:00 | 507 - 20th Ave.E. | Seattle | WA | 98122 | USA | (206) 555-9857 | 5467 | Education includes a BA in psychology from Colorado State University in 1970. She also completed "The Art of the Cold Call." Nancy is a member of Toastmasters International. | 2 | <http://accweb/emmployees/davolio.bmp>

3 | Leverling | Maria | Sales Representative | Ms. | 1963-08-30 00:00:00 | 1992-04-01 00:00:00 | 722 Moss Bay Blvd. | Kirkland | WA | 98033 | USA | (206) 555-3412 | 3355 | Maria has a BS degree in chemistry from Boston College (1984). She has also completed a certificate program in food retailing management. Maria was hired as a sales associate in 1991 and promoted to sales representative in February 1992. | 2 | <http://accweb/emmployees/leverling.bmp>

4 | Peacock | Pedro | Sales Representative | Mr. | 1937-09-19 00:00:00 | 1993-05-03 00:00:00 | 4110 Old Redmond Rd. | Redmond | WA | 98052 | USA | (206) 555-8122 | 5176 | Margaret holds a BA in English literature from Concordia College (1958) and an MA from the American Institute of Culinary Arts (1966). She was assigned to the London office temporarily from July through November 1992. | 2 | <http://accweb/emmployees/peacock.bmp>

5 | Leverling | Maria | Sales Manager | Ms. | 1955-03-04 00:00:00 | 1993-10-17 00:00:00 | 14 Garrett Hill | London | null | SW1 8JR | UK | (71) 555-4848 | 3453 | Maria graduated from St. Andrews University, Scotland, with a BSC degree in 1976. Upon joining the company as a sales representative in 1992, she spent 6 months in an orientation program at the Seattle office and then returned to her permanent post in London. She was promoted to sales manager in March 1993. Ms. Leverling has completed the courses "Successful Telemarketing" and "International Sales Management." She is fluent in French. | 2 | <http://accweb/emmployees/buchanan.bmp>

8 | Fuller | Anne | Inside Sales Coordinator | Ms. | 1958-01-09 00:00:00 | 1994-03-05 00:00:00 | 4726 - 11th Ave. N.E. | Seattle | WA | 98105 | USA | (206) 555-1189 | 2344 | Anne received a BA in psychology from the University of Washington. She has also completed a course in business French. She reads and writes French. | 2 | <http://accweb/emmployees/davolio.bmp>

>EmployeeTerritories : 2 | 01581 | Westboro

2 | 01730 | Bedford

2 | 01833 | Georgetow

2 | 02116 | Boston

2 | 02139 | Cambridge

2 | 02184 | Braintree

2 | 40222 | Louisville

>Order Details : 10971 | 29 | 123.7900 | 14 | 0.0

>Product : 29 | Thuringer Rostbratwurst | 6 | 12 | 50 bags x 30 sausgs. | 123.7900 | 0 | 0 | 0 | 1

>Category : 6 | Meat/ Poultry | Prepared meats

>Supplier : 12 | Plutzer Lebensmittelgro?markte AG | Martin Bein | International Marketing Mgr. | Bogenallee 51 | Frankfurt | null | 60439 | Germany | (069) 992755 |

>Shipper : 2 | United Package | (503) 555-3199

OS₄

>Employee: 5 | Leverling | Maria | Sales Manager | Ms. | 1955-03-04 00:00:00 | 1993-10-17 00:00:00 | 14 Garrett Hill | London | null | SW1 8JR | UK | (71) 555-4848 | 3453 | Maria graduated from St. Andrews University, Scotland, with a BSC degree in 1976. Upon joining the company as a sales representative in 1992, she spent 6 months in an orientation program at the Seattle office and then returned to her permanent post in London. She was promoted to sales manager in March 1993. Ms. Leverling has completed the courses "Successful Telemarketing" and "International Sales Management." She is fluent in French. | 2 | <http://accweb/emmployees/buchanan.bmp>

>ReportsTo : 2 | Fuller | Robert | Vice President, Sales | Dr. | 1952-02-19 00:00:00 | 1992-08-14 00:00:00 | 908 W. Capital Way | Tacoma | WA | 98401 | USA | (206) 555-9482 | 3457 | Robert received his BTS commercial in 1974 and a Ph.D. in international marketing from the University of Dallas in 1981. He is fluent in French and Italian and reads German. He joined the company as a sales representative, was promoted to sales manager in January 1992 and to vice president of sales in March 1993. Andrew is a member of the Sales Management Roundtable, the Seattle Chamber of Commerce, and the Pacific Rim Importers Association. | null | <http://accweb/emmployees/fuller.bmp>

>ReportedBy : 6 | Peacock | Michael | Sales Representative | Mr. | 1963-07-02 00:00:00 | 1993-10-17 00:00:00 | Coventry House Miner Rd. | London | null | EC2 7JR | UK | (71) 555-7773 | 428 | Michael is a graduate of Sussex University (MA, economics, 1983) and the University of California at Los Angeles (MBA, marketing, 1986). He has also taken the courses "Multi-Cultural Selling" and "Time Management for the Sales Professional." He is fluent in Japanese and can read and write French, Portuguese, and Spanish. | 5 | <http://accweb/emmployees/davolio.bmp>

7 | King | Robert | Sales Representative | Mr. | 1960-05-29 00:00:00 | 1994-01-02 00:00:00 | Edgeham Hollow Winchester Way | London | null | RG1 9SP | UK | (71) 555-5598 | 465 | Robert King served in the Peace Corps and traveled extensively before completing his degree in English at the University of Michigan in 1992, the year he joined the company. After completing a course entitled "Selling in Europe," he was transferred to the London office in March 1993. | 5 | <http://accweb/emmployees/davolio.bmp>

9 | Dodsworth | Anne | Sales Representative | Ms. | 1966-01-27 00:00:00 | 1994-11-15 00:00:00 | 7 Houndstooth Rd. | London | null | WG2 7LT | UK | (71) 555-4444 | 452 | Anne has a BA degree in English from St. Lawrence College. She is fluent in French and German. | 5 | <http://accweb/emmployees/davolio.bmp>

>EmployeeTerritories : 5 | 02903 | Providence | Eastern

5 | 07960 | Morristown | Eastern
5 | 08837 | Edison | Eastern
5 | 10019 | New York | Eastern
5 | 10038 | New York | Eastern
5 | 11747 | Mellville | Eastern
5 | 14450 | Fairport | Eastern

>Order : 10248 | VINET | 5 | 1996-07-04 00:00:00 | 1996-08-01 00:00:00 | 1996-07-16 00:00:00 | 3 | 32.3800 | Vins et alcools Chevalier | 59 rue de l'Abbaye | Reims | null | 51100 | France

>Customer : VINET | Vins et alcools Chevalier | Pedro Henriot | Accounting Manager | 59 rue de l'Abbaye | Reims | null | 51100 | France | 26.47.15.10 | 26.47.15.11

>OrderDetails : 10248 | 11 | 14.0000 | 12 | 0.0

>Product : 11 | Queso Cabrales | 4 | 5 | 1 kg pkg. | 21.0000 | 22 | 30 | 30 | 0

>Category : 4 | Dairy Products | Cheeses

>Supplier : 5 | Cooperativa de Quesos 'Las Cabras' | Antonio del Valle Saavedra | Export Administrator | Calle del Rosal 4 | Oviedo | Asturias | 33007 | Spain | (98) 598 76 54 |

>Shipper : 3 | Federal Shipping | (503) 555-9931

>Order : 10254 | CHOPS | 5 | 1996-07-11 00:00:00 | 1996-08-08 00:00:00 | 1996-07-23 00:00:00 | 2 | 22.9800 | Chop-suey Chinese | Hauptstr. 31 | Bern | null | 3012 | Switzerland

>Customer : CHOPS | Chop-suey Chinese | Michael Leverling | Owner | Hauptstr. 29 | Bern | null | 3012 | Switzerland | 0452-076545 |

>OrderDetails : 10254 | 24 | 3.6000 | 15 | 0.15

>Product : 24 | Guarana Fantastica | 1 | 10 | 12 - 355 ml cans | 4.5000 | 20 | 0 | 0 | 1

>Category : 1 | Beverages | Soft drinks, coffees, teas, beers, and ales

>Supplier : 10 | Refrescos Americanas LTDA | Carlos Diaz | Marketing Manager | Av. das Americanas 12.890 | Sao Paulo | null | 5442 | Brazil | (11) 555 4640 |

>Shipper : 2 | United Package | (503) 555-3199

>Order : 10269 | WHITC | 5 | 1996-07-31 00:00:00 | 1996-08-14 00:00:00 | 1996-08-09 00:00:00 | 1 | 4.5600 | White Clover Markets | 1029 - 12th Ave. S. | Seattle | WA | 98124 | USA

>Customer : WHITC | White Clover Markets | Pedro Peacock | Owner | 305 - 14th Ave. S. Suite 3B | Seattle | WA | 98128 | USA | (206) 555-4112 | (206) 555-4115

>OrderDetails : 10269 | 33 | 2.0000 | 60 | 0.05

>Product : 33 | Geitost | 4 | 15 | 500 g | 2.5000 | 112 | 0 | 20 | 0

>Category : 4 | Dairy Products | Cheeses

>Supplier : 15 | Norske Meierier | Beate Vileid | Marketing Manager | Hatlevegen 5 | Sandvika | null | 1320 | Norway | (0)2-953010 |

>Shipper : 1 | Speedy Express | (503) 555-9831

>Order : 10297 | BLONP | 5 | 1996-09-04 00:00:00 | 1996-10-16 00:00:00 | 1996-09-10 00:00:00 | 2 | 5.7400 | Blondel pere et fils | 24, place Kleber | Strasbourg | null | 67000 | France

>Customer : BLONP | Blondesddsl pere et fils | Robert Citeaux | Marketing Manager | 24, place Kleber | Strasbourg | null | 67000 | France | 88.60.15.31 | 88.60.15.32

>OrderDetails : 10297 | 39 | 14.4000 | 60 | 0.0

>Product : 39 | Chartreuse verte | 1 | 18 | 750 cc per bottle | 18.0000 | 69 | 0 | 5 | 0

>Category : 1 | Beverages | Soft drinks, coffees, teas, beers, and ales

>Supplier : 18 | Aux joyeux ecclesiastiques | Guylene Nodier | Sales Manager | 203, Rue des Francs-Bourgeois | Paris | null | 75004 | France | (1) 03.83.00.68 | (1) 03.83.00.62

>Shipper : 2 | United Package | (503) 555-3199

>Order : 10320 | WARTH | 5 | 1996-10-03 00:00:00 | 1996-10-17 00:00:00 | 1996-10-18 00:00:00 | 3 | 34.5700 | Wartian Herkku | Torikatu 38 | Oulu | null | 90110 | Finland

>Customer : WARTH | Wartian Herkku | Pedro Koskitalo | Accounting Manager | Torikatu 38 | Oulu | null | 90110 | Finland | 981-443655 | 981-443655

>OrderDetails : 10320 | 71 | 17.2000 | 30 | 0.0

>Product : 71 | Flotemysost | 4 | 15 | 10 - 500 g pkgs. | 21.5000 | 26 | 0 | 0 | 0

>Category : 4 | Dairy Products | Cheeses

>Supplier : 15 | Norske Meierier | Beate Vileid | Marketing Manager | Hatlevegen 5 | Sandvika | null | 1320 | Norway | (0)2-953010 |
 >Shipper : 3 | Federal Shipping | (503) 555-9931
 >Order : 10333 | WARTH | 5 | 1996-10-18 00:00:00 | 1996-11-15 00:00:00 | 1996-10-25 00:00:00 | 3 | .5900 | Wartian Herkku | Torikatu 38 |
 Oulu | null | 90110 | Finland
 >Customer : WARTH | Wartian Herkku | Pedro Koskitalo | Accounting Manager | Torikatu 38 | Oulu | null | 90110 | Finland | 981-443655 |
 981-443655
 >OrderDetails : 10333 | 14 | 18.6000 | 10 | 0.0
 >Product : 14 | Tofu | 7 | 6 | 40 - 100 g pkgs. | 23.2500 | 35 | 0 | 0 | 0
 >Category : 7 | Produce | Dried fruit and bean curd
 >Supplier : 6 | Mayumi's | Mayumi Ohno | Marketing Representative | 92 Setsuko Chuo-ku | Osaka | null | 545 | Japan | (06) 431-7877 |
 >Shipper : 3 | Federal Shipping | (503) 555-9931
 >Order : 10358 | LAMAI | 5 | 1996-11-20 00:00:00 | 1996-12-18 00:00:00 | 1996-11-27 00:00:00 | 1 | 19.6400 | La maison d'Asie | 1 rue
 Alsace-Lorraine | Toulouse | null | 31000 | France
 >Customer : LAMAI | La maison d'Asie | Maria Roulet | Sales Manager | 1 rue Alsace-Lorraine | Toulouse | null | 31000 | France |
 61.77.61.10 | 61.77.61.11
 >OrderDetails : 10358 | 24 | 3.6000 | 10 | 0.05
 >Product : 24 | Guarana Fantastica | 1 | 10 | 12 - 355 ml cans | 4.5000 | 20 | 0 | 0 | 1
 >Category : 1 | Beverages | Soft drinks, coffees, teas, beers, and ales
 >Supplier : 10 | Refrescos Americanas LTDA | Carlos Diaz | Marketing Manager | Av. das Americanas 12.890 | Sao Paulo | null | 5442 |
 Brazil | (11) 555 4640 |
 >Shipper : 1 | Speedy Express | (503) 555-9831

OS₅

>Customer : THEBI | The Big Cheese | Maria Nixon | Marketing Manager | 89 Jefferson Way Suite 2 | Portland | OR | 97201 | USA | (503) 555-3612 |

>Order : 10310 | THEBI | 8 | 1996-09-20 00:00:00 | 1996-10-18 00:00:00 | 1996-09-27 00:00:00 | 2 | 17.5200 | The Big Cheese | 89 Jefferson Way Suite 2 | Portland | OR | 97201 | USA

>Employee : 8 | Fuller | Anne | Inside Sales Coordinator | Ms. | 1958-01-09 00:00:00 | 1994-03-05 00:00:00 | 4726 - 11th Ave. N.E. | Seattle | WA | 98105 | USA | (206) 555-1189 | 2344 | Anne received a BA in psychology from the University of Washington. She has also completed a course in business French. She reads and writes French. | 2 | <http://accweb/emmployees/davolio.bmp>

>ReportsTo : 2 | Fuller | Robert | Vice President, Sales | Dr. | 1952-02-19 00:00:00 | 1992-08-14 00:00:00 | 908 W. Capital Way | Tacoma | WA | 98401 | USA | (206) 555-9482 | 3457 | Robert received his BTS commercial in 1974 and a Ph.D. in international marketing from the University of Dallas in 1981. He is fluent in French and Italian and reads German. He joined the company as a sales representative, was promoted to sales manager in January 1992 and to vice president of sales in March 1993. Andrew is a member of the Sales Management Roundtable, the Seattle Chamber of Commerce, and the Pacific Rim Importers Association. | null | <http://accweb/emmployees/fuller.bmp>

>EmployeeTerritories : 8 | 19428 | Philadelphia

8 | 44122 | Beachwood

8 | 45839 | Findlay

8 | 53404 | Racine

>Order Details : 10310 | 16 | 13.9000 | 10 | 0.0

>Product : 16 | Pavlova | 3 | 7 | 32 - 500 g boxes | 17.4500 | 29 | 0 | 10 | 0

>Category : 3 | Confections | Desserts, candies, and sweet breads

>Supplier : 7 | Pavlova, Ltd. | Ian Devling | Marketing Manager | 74 Rose St. Moonie Ponds | Melbourne | Victoria | 3058 | Australia | (03) 444-2343 | (03) 444-6588

>Order Details : 10310 | 62 | 39.4000 | 5 | 0.0

>Product : 62 | Tarte au sucre | 3 | 29 | 48 pies | 49.3000 | 17 | 0 | 0 | 0
 >Category : 3 | Confections | Desserts, candies, and sweet breads
 >Supplier : 29 | Forets d'erables | Chantal Goulet | Accounting Manager | 148 rue Chasseur | Ste-Hyacinthe | Quebec | J2S 7S8 | Canada |
 (514) 555-2955 | (514) 555-2921
 >Shipper : 2 | United Package | (503) 555-3199
 >Order : 10708 | THEBI | 6 | 1997-10-17 00:00:00 | 1997-11-28 00:00:00 | 1997-11-05 00:00:00 | 2 | 2.9600 | The Big Cheese | 89 Jefferson
 Way Suite 2 | Portland | OR | 97201 | USA
 >Employee : 6 | Peacock | Michael | Sales Representative | Mr. | 1963-07-02 00:00:00 | 1993-10-17 00:00:00 | Coventry House Miner Rd. |
 London | null | EC2 7JR | UK | (71) 555-7773 | 428 | Michael is a graduate of Sussex University (MA, economics, 1983) and the University of
 California at Los Angeles (MBA, marketing, 1986). He has also taken the courses "Multi-Cultural Selling" and "Time Management for the
 Sales Professional." He is fluent in Japanese and can read and write French, Portuguese, and Spanish. | 5 |
<http://accweb/emmployees/davolio.bmp>
 >ReportsTo : 5 | Leverling | Maria | Sales Manager | Ms. | 1955-03-04 00:00:00 | 1993-10-17 00:00:00 | 14 Garrett Hill | London | null |
 SW1 8JR | UK | (71) 555-4848 | 3453 | Maria graduated from St. Andrews University, Scotland, with a BSC degree in 1976. Upon joining the
 company as a sales representative in 1992, she spent 6 months in an orientation program at the Seattle office and then returned to her permanent
 post in London. She was promoted to sales manager in March 1993. Ms. Leverling has completed the courses "Successful Telemarketing" and
 "International Sales Management." She is fluent in French. | 2 | <http://accweb/emmployees/buchanan.bmp>
 >EmployeeTerritories : 6 | 85014 | Phoenix
 6 | 85251 | Scottsdale
 6 | 98004 | Bellevue
 6 | 98052 | Redmond
 6 | 98104 | Seattle
 >Order Details : 10708 | 5 | 21.3500 | 4 | 0.0
 >Product : 5 | Chef Anton's Gumbo Mix | 2 | 2 | 36 boxes | 21.3500 | 0 | 0 | 0 | 1
 >Category : 2 | Condiments | Sweet and savory sauces, relishes, spreads, and seasonings

>Supplier : 2 | New Orleans Cajun Delights | Shelley Burke | Order Administrator | P.O. Box 78934 | New Orleans | LA | 70117 | USA | (100) 555-4822 |

>Order Details : 10708 | 36 | 19.0000 | 5 | 0.0

>Product : 36 | Inlagd Sill | 8 | 17 | 24 - 250 g jars | 19.0000 | 112 | 0 | 20 | 0

>Category : 8 | Seafood | Seaweed and fish

>Supplier : 17 | Svensk Sjöföda AB | Michael Björn | Sales Representative | Brovallavägen 231 | Stockholm | null | S-123 45 | Sweden | 08-123 45 67 |

>Shipper : 2 | United Package | (503) 555-3199

>Order : 10805 | THEBI | 2 | 1997-12-30 00:00:00 | 1998-01-27 00:00:00 | 1998-01-09 00:00:00 | 3 | 237.3400 | The Big Cheese | 89 Jefferson Way Suite 2 | Portland | OR | 97201 | USA

>Employee : 2 | Fuller | Robert | Vice President, Sales | Dr. | 1952-02-19 00:00:00 | 1992-08-14 00:00:00 | 908 W. Capital Way | Tacoma | WA | 98401 | USA | (206) 555-9482 | 3457 | Robert received his BTS commercial in 1974 and a Ph.D. in international marketing from the University of Dallas in 1981. He is fluent in French and Italian and reads German. He joined the company as a sales representative, was promoted to sales manager in January 1992 and to vice president of sales in March 1993. Andrew is a member of the Sales Management Roundtable, the Seattle Chamber of Commerce, and the Pacific Rim Importers Association. | null | <http://accweb/emmployees/fuller.bmp>

>ReportedBy : 1 | Fuller | Maria | Sales Representative | Ms. | 1948-12-08 00:00:00 | 1992-05-01 00:00:00 | 507 - 20th Ave.E. | Seattle | WA | 98122 | USA | (206) 555-9857 | 5467 | Education includes a BA in psychology from Colorado State University in 1970. She also completed "The Art of the Cold Call." Nancy is a member of Toastmasters International. | 2 | <http://accweb/emmployees/davolio.bmp>

3 | Leverling | Maria | Sales Representative | Ms. | 1963-08-30 00:00:00 | 1992-04-01 00:00:00 | 722 Moss Bay Blvd. | Kirkland | WA | 98033 | USA | (206) 555-3412 | 3355 | Maria has a BS degree in chemistry from Boston College (1984). She has also completed a certificate program in food retailing management. Maria was hired as a sales associate in 1991 and promoted to sales representative in February 1992. | 2 | <http://accweb/emmployees/leverling.bmp>

4 | Peacock | Pedro | Sales Representative | Mr. | 1937-09-19 00:00:00 | 1993-05-03 00:00:00 | 4110 Old Redmond Rd. | Redmond | WA | 98052 | USA | (206) 555-8122 | 5176 | Margaret holds a BA in English literature from Concordia College (1958) and an MA from the American Institute of Culinary Arts (1966). She was assigned to the London office temporarily from July through November 1992. | 2 | <http://accweb/emmployees/peacock.bmp>

5 | Leverling | Maria | Sales Manager | Ms. | 1955-03-04 00:00:00 | 1993-10-17 00:00:00 | 14 Garrett Hill | London | null | SW1 8JR | UK | (71) 555-4848 | 3453 | Maria graduated from St. Andrews University, Scotland, with a BSC degree in 1976. Upon joining the company as a sales representative in 1992, she spent 6 months in an orientation program at the Seattle office and then returned to her permanent post in London. She was promoted to sales manager in March 1993. Ms. Leverling has completed the courses "Successful Telemarketing" and "International Sales Management." She is fluent in French. | 2 | <http://accweb/emmployees/buchanan.bmp>

8 | Fuller | Anne | Inside Sales Coordinator | Ms. | 1958-01-09 00:00:00 | 1994-03-05 00:00:00 | 4726 - 11th Ave. N.E. | Seattle | WA | 98105 | USA | (206) 555-1189 | 2344 | Anne received a BA in psychology from the University of Washington. She has also completed a course in business French. She reads and writes French. | 2 | <http://accweb/emmployees/davolio.bmp>

>EmployeeTerritories : 2 | 01581 | Westboro

2 | 01730 | Bedford

2 | 01833 | Georgetow

2 | 02116 | Boston

2 | 02139 | Cambridge

2 | 02184 | Braintree

2 | 40222 | Louisville

>Order Details : 10805 | 34 | 14.0000 | 10 | 0.0

>Product : 34 | Sasquatch Ale | 1 | 16 | 24 - 12 oz bottles | 14.0000 | 111 | 0 | 15 | 0

>Category : 1 | Beverages | Soft drinks, coffees, teas, beers, and ales

>Supplier : 16 | Bigfoot Breweries | Cheryl Saylor | Regional Account Rep. | 3400 - 8th Avenue Suite 210 | Bend | OR | 97101 | USA | (503) 555-9931 |

>Order Details : 10805 | 38 | 263.5000 | 10 | 0.0

>Product : 38 | Cote de Blaye | 1 | 18 | 12 - 75 cl bottles | 263.5000 | 17 | 0 | 15 | 0

>Category : 1 | Beverages | Soft drinks, coffees, teas, beers, and ales

>Supplier : 18 | Aux joyeux ecclésiastiques | Guylene Nodier | Sales Manager | 203, Rue des Francs-Bourgeois | Paris | null | 75004 | France | (1) 03.83.00.68 | (1) 03.83.00.62

>Shipper : 3 | Federal Shipping | (503) 555-9931

➤ $Q_2 = \{\text{"Robert"}, \text{"Marketing"}\}$

	Στατιστικά στοιχεία για 5 OSs								
	Αριθμός εμφανίσεων λέξης - κλειδί "Marketing"						Συνολικός αριθμός εμφανίσεων λέξης - κλειδί "Marketing"	Μέγεθος OS σε αριθμό πλειάδων (tuples)	Μέγεθος OS σε αριθμό λέξεων
	Σχέση Customers	Σχέση Employees	Σχέση Employees (ReportsTo)	Σχέση Employees (ReportedBy)	Σχέση Suppliers	Σχέση Orders			
OS ₁	0	2	3	1	4	0	10	65	1598
OS ₂	1	3	3	2	4	0	13	46	1587
OS ₃	25	1	0	1	89	7	123	675	15745
OS ₄	14	0	1	0	52	6	73	507	11619
OS ₅	0	4	8	3	7	0	22	152	3763

➤ $Q_3 = \{\text{"Michael"}, \text{"Sales"}\}$

	Στατιστικά στοιχεία για 5 OSs							
	Αριθμός εμφανίσεων λέξης - κλειδί "Sales"					Συνολικός αριθμός εμφανίσεων λέξης – κλειδί "Sales"	Μέγεθος OS σε αριθμό πλειάδων (tuples)	Μέγεθος OS σε αριθμό λέξεων
	Σχέση Customers	Σχέση Employees	Σχέση Employees (ReportsTo)	Σχέση Employees (ReportedBy)	Σχέση Suppliers			
OS_1	0	23	33	30	11	97	134	3800
OS_2	0	19	53	0	10	82	144	3209
OS_3	0	7	25	0	2	34	58	1355
OS_4	35	2	4	0	65	106	427	10880
OS_5	0	8	29	0	5	42	71	1643

➤ $Q_4 = \{\text{"Anne"}, \text{"Management"}\}$

	Στατιστικά στοιχεία για 5 OSs							
	Αριθμός εμφανίσεων λέξης - κλειδί "Management"					Συνολικός αριθμός εμφανίσεων λέξης – κλειδί "Management"	Μέγεθος OS σε αριθμό πλειάδων (tuples)	Μέγεθος OS σε αριθμό λέξεων
	Σχέση Customers	Σχέση Employees	Σχέση Employees (ReportsTo)	Σχέση Employees (ReportedBy)	Σχέση Suppliers			
OS_1	0	4	15	1	10	30	101	2320
OS_2	1	3	21	0	10	35	122	2740
OS_3	14	0	3	0	46	63	304	6922
OS_4	0	2	18	0	6	26	92	2019
OS_5	1	10	28	8	16	63	211	4967

➤ $Q_5 = \{ \text{“Leverling”}, \{ \text{“Sales Marketing”} \} \}$

	Στατιστικά στοιχεία για 5 OSs						
	Αριθμός εμφανίσεων λέξης - κλειδί “Sales”				Συνολικός αριθμός εμφανίσεων λέξης – κλειδί “Sales”	Μέγεθος OS σε αριθμό πλειάδων (tuples)	Μέγεθος OS σε αριθμό λέξεων
	Σχέση Customers	Σχέση Employees	Σχέση Employees (ReportsTo)	Σχέση Suppliers			
OS ₁	0	13	33	5	51	104	2252
OS ₂	61	3	5	147	216	892	20704
OS ₃	1	14	43	110	69	126	2744
OS ₄	1	8	24	7	40	90	1748
OS ₅	1	10	31	7	49	92	2019

	Στατιστικά στοιχεία για 5 OSs							
	Αριθμός εμφανίσεων λέξης - κλειδί “Marketing”					Συνολικός αριθμός εμφανίσεων λέξης – κλειδί “Marketing”	Μέγεθος OS σε αριθμό πλειάδων (tuples)	Μέγεθος OS σε αριθμό λέξεων
	Σχέση Customers	Σχέση Employees	Σχέση Employees (ReportsTo)	Σχέση Suppliers	Σχέση Orders			
OS ₁	0	0	7	4	0	11	104	2252
OS ₂	33	0	1	87	12	133	892	20704
OS ₃	0	1	9	8	0	18	126	2744
OS ₄	0	1	5	6	0	12	90	1748
OS ₅	0	1	7	3	0	11	92	2019