



ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

«Αυτόματη Κατηγοριοποίηση επιστημονικών
δημοσιεύσεων»

ΠΙΤΣΙΚΑΛΗΣ ΧΡΗΣΤΟΣ
A.M. : 20732

Επιβλέπων:
Ηρακλής Βαρλάμης, Λέκτορας
Μέλη: Νικολαΐδη Μ. Καθηγήτρια, Βασιλοπούλου Κ. Λέκτορας

Αθήνα, Σεπτέμβριος 2013

ΕΥΧΑΡΙΣΤΙΕΣ

Αρχικά θα ήθελα να ευχαριστήσω όλους όσους συνέβαλαν στην ολοκλήρωση της πτυχιακής εργασίας μου και γενικότερα των σπουδών μου στο Τμήμα Πληροφορικής και Τηλεματικής του Χαροκόπειου Πανεπιστημίου.

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή της πτυχιακής μου εργασίας κ. Ηρακλή Βαρλάμη, για την καθοδήγησή του και την πολύτιμη συμβολή του σε κάθε φάση της δημιουργίας της.

Τέλος, θα ήθελα να ευχαριστήσω θερμά την οικογένειά μου για την ηθική και οικονομική συμπαράσταση όχι μόνο κατά τη διάρκεια της εκπόνησης της πτυχιακής μου εργασίας αλλά και καθ' όλη τη διάρκεια των σπουδών μου.

ΠΕΡΙΛΗΨΗ

Σκοπός της εργασίας αυτής είναι η δημιουργία ενός εργαλείου το οποίο θα επιτρέπει τη διαχείριση ενός καταλόγου επιστημονικών δημοσιεύσεων και πιο συγκεκριμένα θα δίνει τη δυνατότητα αυτόματης κατηγοριοποίησης/αξιολόγησης επιστημονικών δημοσιεύσεων με βάση διεθνείς κατηγοριοποιήσεις περιοδικών και συνεδρίων. Το εργαλείο είναι διαθέσιμο στο web, χρησιμοποιεί ως βάση για τον κατάλογο δημοσιεύσεων το πρότυπο bibtex και είναι διαθέσιμο προς όλους τους χρήστες, ερευνητές, βιβλιοθήκες, ακαδημαϊκούς κλπ.

Η διαδικασία της αξιολόγησης των επιστημονικών δημοσιεύσεων ξεκίνησε ως ένας τρόπος να βαθμολογούνται τα πανεπιστήμια και τα ερευνητικά ιδρύματα της κάθε χώρας. Για τον σκοπό αυτό χρησιμοποιούνται βιβλιομετρικοί δείκτες και αξιολόγηση από ομότιμους. Υπάρχουν διάφοροι οργανισμοί που παράγουν αξιολογήσεις, ένας από αυτούς είναι ο ERA(Excellence in Research for Australia) του οποίου τα αποτελέσματα αξιολόγησης χρησιμοποιήθηκαν στα πλαίσια της εργασίας αυτής.

Για την υλοποίηση της εργασίας χρησιμοποιήθηκε JSP καθώς και διάφορες βιβλιοθήκες όπως Lucene, jBibtex και Apache poi που παρείχαν τα απαραίτητα εργαλεία. Επίσης, για τις δοκιμές χρησιμοποιήθηκαν δυο πίνακες αξιολόγησης από τον οργανισμό ERA για το έτος 2010, ένας για περιοδικά κι ένας για συνέδρια.

Αποτέλεσμα είναι ένας διαδικτυακός τόπος όπου ο χρήστης μπορεί να ανεβάζει το αρχείο με τις δημοσιεύσεις του σε bibtex format, να βλέπει αυτόματα την αξιολόγηση των δημοσιεύσεων του και παράλληλα να μπορεί να επέμβει σε περίπτωση που ο τίτλος που επιλέχθηκε από το πρόγραμμα για το περιοδικό ή το συνέδριο είναι λάθος. Τέλος, έχοντας κάνει τις αλλαγές που επιθυμεί μπορεί να κατεβάσει το ανανεωμένο αρχείο bib αλλά και να δει ένα διάγραμμα αξιολόγησης των δημοσιεύσεών του.

Table of Contents

1 ΕΙΣΑΓΩΓΗ.....	5
1.1 Αξιολόγηση περιοδικών και συνεδρίων.....	5
1.2 Σκοπός της εργασίας.....	7
1.3 Χρησιμότητα της εργασίας.....	8
1.4 Οφέλη από την εργασία.....	8
2 ΥΠΟΒΑΘΡΟ.....	9
2.1 BibTeX, Bib και η βιβλιοθήκη jbibtex.....	9
2.1.1 Γενικά.....	9
2.1.2 Ιστορικά.....	9
2.1.3 Αρχείο βιβλιογραφικών πληροφοριών.....	9
2.1.4 Η βιβλιοθήκη jbibtex.....	11
2.2 Excellence in Research for Australia (ERA).....	11
2.3 Apache Lucene.....	13
2.3.1 Γενικά.....	13
2.3.2 Κοινή χρήση.....	13
2.3.3 Χαρακτηριστικά.....	14
2.4 Apache Poi.....	14
2.5 Σχετικά project.....	16
3 ΣΧΕΔΙΑΣΗ ΣΥΣΤΗΜΑΤΟΣ.....	17
3.1 Δημιουργία ευρετηρίων.....	17
3.2 Κυρίως πρόγραμμα.....	18
3.3 Το αρχείο πριν και μετά.....	20
4 ΥΛΟΠΟΙΗΣΗ.....	22
4.1 Από τα xls στα Lucene indexes.....	22
4.1.1 Διάβασμα xls.....	22
4.1.2 Δημιουργία ευρετηρίων.....	24
4.2 Ιστότοπος.....	25
4.2.1 Γενικά.....	25
4.2.2 Index.jsp.....	26
4.2.3 Readbib.jsp.....	27
4.2.4 Check.jsp.....	27
4.2.5 Create.jsp.....	28
4.3 Κλάσεις.....	28
4.4 Η κλάση methods.....	29
4.4.1 Μέθοδοι read και parseBibTeX.....	29
4.4.2 Μέθοδοι add_Field, parseBibTeX και formatBibTeX.....	30
4.4.3 Μέθοδοι suggest και search.....	31
5 ΑΞΙΟΛΟΓΗΣΗ – ΣΥΜΠΕΡΑΣΜΑΤΑ.....	34
5.1 Ενδεικτική χρήση.....	34
5.2 Συμπεράσματα.....	39
5.3 Μελλοντικές επεκτάσεις.....	39
BIBΛΙΟΓΡΑΦΙΑ.....	40
ΧΡΗΣΙΜΟΙ ΣΥΝΔΕΣΜΟΙ.....	40

1 ΕΙΣΑΓΩΓΗ

1.1 Αξιολόγηση περιοδικών και συνεδρίων

Οι δημοσιεύσεις σε επιστημονικά περιοδικά αποτελούν το κυριότερο μέσο για τη διάχυση των ερευνητικών αποτελεσμάτων στην επιστημονική κοινότητα και ταυτόχρονα μια σημαντική πηγή δεδομένων για την καταγραφή και την αποτίμηση του ερευνητικού έργου με το οποίο συνδέονται.

Η αξιολόγηση των περιοδικών γίνεται επιλέγοντας ένα ή περισσότερα από τα παρακάτω κριτήρια:

1. Ο μέσος αριθμός αναφορών σε άρθρα που έχουν δημοσιευθεί σε επιστημονικά περιοδικά.
2. Η εκτίμηση της συνολικής σπουδαιότητας του επιστημονικού περιοδικού σύμφωνα με τον αριθμό των αναφορών από άλλα περιοδικά(με ειδικό βάρος ανάλογα την αξιολόγηση του περιοδικού).
3. Ένα μέτρο της επιστημονικής επιρροής των επιστημονικών περιοδικών που αντιπροσωπεύει τόσο τον αριθμό των αναφορών που λαμβάνονται από ένα περιοδικό όσο και την σημασία ή το κύρος των περιοδικών από τα οποία προέρχονται οι αναφορές.
4. Ένα σκορ που αντικατοπτρίζει την συνολική ποιότητα και/ή συμβολή ενός περιοδικού με βάση τα αποτελέσματα της έρευνας ενεργών ερευνητών του τομέα, επαγγελματιών και μαθητών, οι οποίοι κατατάσσουν κάθε περιοδικό με βάση συγκεκριμένα κριτήρια.
5. Η θέση κατάταξης του κάθε περιοδικού με βάση την πραγματική συμπεριφορά της δημοσίευσης από κορυφαίους ακαδημαϊκούς συνεργάτες για παρατεταμένο χρονικό διάστημα. Ως εκ τούτου, η θέση κατάταξης του περιοδικού.

αντικατοπτρίζει την συχνότητα με την οποία οι μελετητές δημοσίευσαν τα άρθρα τους στο περιοδικό αυτό

Αντίστοιχα, η αξιολόγηση των συνεδρίων γίνεται με βάση τα παρακάτω:

1. Ανταπόκριση από βιομηχανίες: Ένα καλό συνέδριο μπορεί να προσελκύσει περισσότερους εταίρους του κλάδου
2. Η ευκολία των θέσεων του συνεδρίου: Οι εύκολες θέσεις των συνεδρίων διευκολύνουν τους ερευνητές και έτσι προσελκύουν περισσότερους
3. Η ποιότητα των μελών της επιτροπής: Είναι ένας σημαντικός παράγοντας αφού επηρεάζει την οργάνωση του συνεδρίου και την διατήρηση της ποιότητας των αποδεκτών εργασιών
4. Η αναλογία αποδοχής εργασιών: Πρόκειται για έναν παράγοντα που διασφαλίζει την ποιότητα των εργασιών του συνεδρίου

Σύμφωνα λοιπόν με τα παραπάνω, τα περιοδικά και τα συνέδρια βαθμολογούνται και παίρνουν έτσι μια θέση στην κατάταξη η οποία μπορεί να τα κάνει περισσότερο ή λιγότερο ελκυστικά σε αναγνώστες, ερευνητές και κυρίως επίδοξους επενδυτές. Μαζί βέβαια με τα περιοδικά και τα συνέδρια αξιολογούνται και τα αντίστοιχα άρθρα και έρευνες που βρίσκονται σε αυτά.

Αυτό το σύστημα, όπως είναι λογικό είναι αμφιλεγόμενο και υπάρχουν πολλά παράπονα επειδή επηρεάζει χρηματοδοτήσεις ερευνών και την σταδιοδρομία των ακαδημαϊκών. Το σύστημα αυτό ήταν μια πρωτοβουλία της κυβέρνησης της Αυστραλίας, ονομαζόμενο ERA(Excellence in Research for Australia)¹, που την βοηθούσε να αποφασίσει πόσα χρήματα θα δώσει σε κάθε ομάδα έρευνας στα πανεπιστήμια(Rowbotham 2010) .

1 <http://www.arc.gov.au/era/>

1.2 Σκοπός της εργασίας

Σκοπός αυτής της εργασίας είναι να δημιουργήσουμε ένα εργαλείο το οποίο θα ενημερώνει αυτόματα αρχεία δημοσιεύσεων με δεδομένα που αφορούν την κατηγοριοποίηση συνεδρίων και περιοδικών. Ο χρήστης θα ανεβάζει απλά στον ιστότοπο το αρχείο με τις δημοσιεύσεις που θέλει και στην συνέχεια αυτό θα βρίσκει τους αντίστοιχους τίτλους συνεδρίων ή περιοδικών και με βάση αυτά θα αξιολογεί τις δημοσιεύσεις. Επίσης, θα μπορεί να αλλάξει την αυτόματη επιλογή του συστήματος και τέλος θα κατεβάζει το ανανεωμένο πλέον αρχείο(με επιπλέον πεδία την αξιολόγηση και τον τίτλο του περιοδικού ή συνεδρίου) και θα βλέπει ένα διάγραμμα αξιολογήσεων καθώς και έναν μέσο βαθμό αξιολόγησης.

Για να πραγματοποιηθούν τα παραπάνω, συνδυάζονται πολλά εργαλεία. Αρχικά, πρέπει να υπάρχουν λίστες κατηγοριοποίησης περιοδικών/συνεδρίων με τίτλους ανά κατηγορία. Στην παρούσα εργασία χρησιμοποιήθηκαν οι λίστες του οργανισμού ERA, αλλά θα μπορούσε να χρησιμοποιηθεί οποιαδήποτε άλλη κατηγοριοποίηση αρκεί να δοθεί σε ένα xls format. Στην συνέχεια, χρειάζεται ένα πρότυπο με το οποίο μπορεί κανείς να δίνει την βιβλιογραφία, όπως είναι τα bib, html, xml κ.ά. Το σύστημα αυτό χρησιμοποιεί την μορφοποίηση bibtex. Επίσης, θα πρέπει να υπάρχει και ένα εργαλείο που να επιτρέπει την ανάγνωση και εγγραφή του αρχείου βιβλιογραφίας, όπως είναι η βιβλιοθήκη jbibtex που χρησιμοποιείται από το πρόγραμμα αυτό. Τέλος, χρειάζεται και ένας μηχανισμός που να ταιριάζει αυτόματα τον τίτλο(περιοδικού ή συνεδρίου) που βρίσκει στο αρχείο με τους ήδη κατηγοριοποιημένους τίτλους. Εδώ χρησιμοποιήθηκε το Lucene, ένα εργαλείο που είναι η βάση για τις περισσότερες μηχανές αναζήτησης.

1.3 Χρησιμότητα της εργασίας

Η εργασία αυτή είναι χρήσιμη τόσο σε μεμονωμένους ερευνητές όσο και σε οργανισμούς που ασχολούνται με την αξιολόγηση της ποιότητας του επιστημονικού έργου σε εθνικό και διεθνές επίπεδο (π.χ. Αρχή διασφάλισης ποιότητας-ΑΔΙΠ, Εθνικό κέντρο τεκμηρίωσης -ΕΚΤ/ΕΙΕ). Πολλοί ερευνητές από τη μία έχουν μια λίστα δημοσιεύσεων και θέλουν είτε να δουν σε τι κατηγορίες ποιότητας ανήκουν αυτές, είτε να γράψουν τον τίτλο του forum(συνέδριο ή περιοδικό) στο οποίο δημοσίευσαν με σωστό τρόπο. Το πρόγραμμα που αναπτύχθηκε επιτρέπει και τα 2. Επίσης, το αποτέλεσμα αυτό μπορεί να συγχωνευτεί με αντίστοιχα άλλων ερευνητών του ίδιου ιδρύματος και να αξιοποιηθεί για την αξιολόγηση του ερευνητή ή της ομάδας ερευνητών. Από την άλλη, υπάρχουν και πολλοί φορείς ελέγχου ποιότητας (κρατικοί φορείς, χορηγοί ερευνητικών προγραμμάτων κλπ) που θέλουν ένα εργαλείο που να τους δείχνει ποιες δημοσιεύσεις έχουν γίνει σε περισσότερο ή λιγότερο σημαντικά forum. Ας μην ξεχνάμε ότι αυτός ήταν και ο λόγος που ξεκίνησαν οι αξιολογήσεις).

1.4 Οφέλη από την εργασία

Πέρα από την χρησιμότητα που αναφέραμε πιο πάνω, η εργασία αυτή ήταν πολύ ωφέλιμη και για εμένα που την έγραψα για τον λόγο ότι με έκανε να ενασχοληθώ με αρκετά πράγματα και μάλιστα όχι επιφανειακά. Κάποια από αυτά είναι:

- JavaServer Pages(JsP), δημιουργία html σελίδων με χρήση Java στο background
- Διαχείριση αρχείων μορφοποίησης bibtex μέσω εφαρμογών Java
- Διαχείριση αρχείων xls μέσω εφαρμογών Java
- Δημιουργία ευρετηρίων και αναζήτηση με το Lucene, ένα εργαλείο που είναι η βάση για πολλές εμπορικές και ερευνητικές μηχανές αναζήτησης

2 ΥΠΟΒΑΘΡΟ

2.1 *BibTeX*, *Bib* και η βιβλιοθήκη *jbibtex*

2.1.1 Γενικά

Το BibTeX είναι ένα λογισμικό διαχείρισης αναφορών για την μορφοποίηση καταλόγων αναφορών. Το εργαλείο BibTeX χρησιμοποιείται συνήθως μαζί με το σύστημα περοετοιμασίας εγγράφων LaTeX. Το BibTeX καθιστά εύκολη την παράθεση πηγών με τρόπο συνεπή, διαχωρίζοντας βιβλιογραφικές πληροφορίες από την παρουσίαση του περιεχομένου, παρόμοια με το διαχωρισμό του περιεχομένου και του τρόπου παρουσίασης που υποστηρίζεται από το LaTeX.

2.1.2 Ιστορικά

Το BibTeX δημιουργήθηκε από τους Oren Patashnik και Leslie Lamport το 1985 και είναι γραμμένο σε WEB/Pascal(<http://ftp.math.utah.edu/pub/tex/bibtex/bibtex.web>). Τον Μάρτιο του 1985 κυκλοφόρησε η έκδοση 0.98f. Τρία χρόνια αργότερα, τον Φεβρουάριο του 1988, με την έκδοση 0.99c που κυκλοφόρησε, επιτεύχθηκε μια στατική κατάσταση για 22 χρόνια έως την έκδοση 0.99d που κυκλοφόρησε τον Μάρτιο του 2010. Ωστόσο, είχαν ανακοινωθεί κι άλλες εκδόσεις².

2.1.3 Αρχείο βιβλιογραφικών πληροφοριών

Το BibTeX χρησιμοποιεί μια ανεξάρτητου στυλ βασισμένη σε κείμενο μορφοποίηση για τις λίστες βιβλιογραφικών στοιχείων, όπως είναι τα άρθρα, τα βιβλία και οι διατριβές. Τα βιβλιογραφικά αρχεία που παράγονται από το BibTeX συνήθως

² <http://www.tex.ac.uk/tex-archive/bibliography/bibtex/base/bibtex.web>

έχουν την κατάληξη .bib. Η κάθε βιβλιογραφική καταχώρηση περιέχει ένα υποσύνολο των τυποποιημένων καταχωρήσεων δεδομένων, όπως είναι ο τίτλος του άρθρου, το περιοδικό στο οποίο δημοσιεύθηκε, το όνομα του συγγραφέα, το έτος δημοσίευσης και άλλα. Αναλόγως τον τύπο της κάθε καταχώρησης, υπάρχουν και τα αντίστοιχα στοιχεία.

Παρακάτω φαίνονται δυο παραδείγματα εγγραφών, ένα για συνέδρια κι ένα για περιοδικά. Με γκρι φόντο ξεχωρίζουν τα κομμάτια που αφορούν το παρόν σύστημα,

```
@inproceedings{domingos00unified,           //Συνέδριο
author = {Pedro Domingos},
title = {A Unified Bias-Variance Decomposition for Zero-One and Squared Loss},
booktitle = {Proc. National Conference on Artificial Intelligence and Proc. Conference Innovative Applications
of Artificial Intelligence},
year = {2000},
isbn = {0-262-51112-6},
pages = {564--569},
publisher = {AAAI Press / The MIT Press},
}
```

```
@article{smeulders00contentbased,           //Περιοδικό
author = {Arnold W. M. Smeulders and Marcel Worring and Simone Santini and Amarnath Gupta and Ramesh
Jain},
title = {Content-Based Image Retrieval at the End of the Early Years},
journal = {IEEE Trans. Pattern Anal. Mach. Intell.},
volume = {22},
number = {12},
year = {2000},
issn = {0162-8828},
pages = {1349--1380},
doi = {dx.doi.org/10.1109/34.895972},
publisher = {IEEE Computer Society},
address = {Washington, DC, USA},
}
```

2.1.4 Η βιβλιοθήκη jbibtex

Η βιβλιοθήκη jbibtex είναι μια πολύ χρήσιμη βιβλιοθήκη για προγραμματιστές Java που θέλουν να επεξεργαστούν αρχεία βιβλιογραφίας. Αυτή η βιβλιοθήκη, που σήμερα βρίσκεται στην έκδοση 1.0.8, έχει τα παρακάτω χαρακτηριστικά³:

- Αναλύει, χειρίζεται και μορφοποιεί αρχεία βιβλιογραφίας BibTeX
- Αναλύει και τυπώνει πεδία τιμών LaTeX
- Είναι ένα μικρό αρχείο JAR, χωρίς επιπλέον εξαρτήσεις
- Είναι επαρκώς γρήγορο και ισχυρό εργαλείο. Αναλύει τα βιβλιογραφικά αρχεία BibTeX Unix και Java του Nelson H. F. Beebe's σε λιγότερο από 1 δευτερόλεπτο.

2.2 *Excellence in Research for Australia (ERA)*

Το συμβούλιο ερευνών της Αυστραλίας(ARC – Australian Research Council) είναι υπεύθυνο για την διαχείριση του ERA, το οποίο αντικατέστησε το Research Quality Framework(Kim Carr, 2008) και έχει ως σκοπό τον εντοπισμό και την προώθηση αριστείας σε όλο το φάσμα της ερευνητικής δραστηριότητας των ιδρυμάτων τριτοβάθμιας εκπαίδευσης της Αυστραλίας(Kim Carr, 2009).

Ο ERA αξιολογεί την ποιότητα της έρευνας που αναλαμβάνουν τα αυστραλιανά πανεπιστήμια έναντι εθνικών και διεθνών σημείων αναφοράς. Οι αξιολογήσεις προσδιορίζονται και εποπτεύονται από επιτροπές διακεκριμένων ερευνητών, που προέρχονται από την Αυστραλία και το εξωτερικό. Η μονάδα αξιολόγησης σε γενικές γραμμές ορίζεται ως πεδίο έρευνας(FoR – Field of Research) εντός ενός θεσμικού οργάνου βασισμένο στην τυποποιημένη ταξινόμηση Αυστραλίας και Νέας Ζηλανδίας(ANZSRC – Australia and New Zealand Standard Classification).

³ <https://code.google.com/p/java-bibtex/>

Οι δείκτες που χρησιμοποιούνται στο ERA περιλαμβάνουν μια σειρά από μετρήσεις, όπως παραπομπές σε άλλους ερευνητές που είναι συνηθισμένο σε κλάδους των φυσικών επιστημών, και αξιολόγηση από ομότιμους ερευνητές ενός δείγματος των αποτελεσμάτων της έρευνας που είναι πιο συχνό σε ανθρωπιστικές και κοινωνικές επιστήμες. Το ERA είναι μια ολοκληρωμένη συλλογή. Τα δεδομένα που υποβάλλονται από πανεπιστήμια καλύπτουν όλους τους αποδεκτούς ερευνητές και τα αποτελέσματα των ερευνών τους. Το συγκεκριμένο σετ δεικτών που χρησιμοποιείται έχει αναπτυχθεί σε στενή διαβούλευση με την ερευνητική κοινότητα. Η προσέγγιση αυτή διασφαλίζει ότι οι δείκτες που χρησιμοποιούνται είναι κατάλληλοι και αναγκαίοι, γεγονός που ελαχιστοποιεί την διάθεση πόρων του ERA για την κυβέρνηση και τα πανεπιστήμια και διασφαλίζει ότι τα αποτελέσματα του ERA είναι και ισχυρά και ευρέως αποδεκτά.

Η πρώτη ολοκληρωμένη έκδοση του ERA εμφανίστηκε το 2010 και τα αποτελέσματά της δημοσιεύθηκαν στις αρχές του 2011. Οι αξιολογήσεις μπορούν να πάρουν τις τιμές A*(top 5%), A(next 15%), B(next 30%) και C(next 50%) (http://www.arc.gov.au/era/tiers_ranking.htm). Η δεύτερη έκδοση ολοκληρώθηκε με την δημοσίευση του ERA 2012 National Report τον Δεκέμβριο του 2012 όπου έχουν αλλάξει και οι κατηγορίες αξιολόγησης (Tim Mazzarol, 2011). Οι εκδόσεις του 2010 και 2012 χρησιμοποιούν το σύστημα Scopus⁴ για βιβλιομετρικούς δείκτες και αναφορές (Margaret Sheil, 2009). Η επόμενη έχει προγραμματιστεί για το 2015 και οι προετοιμασίες βρίσκονται σε εξέλιξη. Στα πλαίσια της εργασίας αυτής χρησιμοποιήθηκε η λίστα του 2010 για περιοδικά και συνέδρια.

Η λίστα αυτή είναι σε μορφή excel και εναλλακτικά θα μπορούσε να φορτωθεί οποιαδήποτε άλλη λίστα κατηγοριοποίησης. Αυτό κάνει την προτεινόμενη υλοποίηση πολύ πιο ευέλικτη.

4 <http://www.elsevier.com/online-tools/scopus/content-overview>

2.3 *Apache Lucene*

2.3.1 Γενικά

Το Apache Lucene⁵ είναι μια βιβλιοθήκη μηχανής αναζήτησης υψηλών επιδόσεων και πλήρως εξοπλισμένη, γραμμένη εξ' ολοκλήρου σε Java που δημιουργήθηκε από τον Doug Cutting το 1999 (Grant Ingersoll, 2007). Εντάχθηκε στην οικογένεια προϊόντων Java ανοιχτού κώδικα της Apache Software Foundation της Τζακάρτα τον Σεπτέμβριο του 2001 και έγινε από μόνο του ανωτάτου επιπέδου έργο του Apache τον Φεβρουάριο του 2005.

2.3.2 Κοινή χρήση

Όντας κατάλληλο για κάθε εφαρμογή που απαιτεί την πλήρη ευρετηρίαση κειμένου και δυνατότητα αναζήτησης, το Lucene έχει αναγνωριστεί ευρέως (McCandless, Michael; Hatcher, Erik; Gospodnetic, Otis, 2010) για τη χρησιμότητά του στην υλοποίηση μηχανών αναζήτησης στο Διαδίκτυο και τοπικά σε ιστότοπους. Στον πυρήνα της λογικής αρχιτεκτονικής του Lucene είναι η ιδέα ενός εγγράφου που περιέχει πεδία κειμένου. Αυτή του η ευελιξία επιτρέπει στο Lucene API να είναι ανεξάρτητο από την μορφή του αρχείου. Κείμενα από αρχεία PDF, HTML, Microsoft Word, OpenDocument έγγραφα καθώς και πολλοί άλλοι τύποι (εκτός από εικόνες), όλα μπορούν να αναπροσαρμοστούν εφόσον είναι δυνατόν να εξαχθούν από αυτά πληροφορίες υπό μορφή κειμένου (Petra Pernert, 2007). Όπως γίνεται κατανοητό, το Lucene αποτελεί μια μηχανή αναζήτησης που μπορεί να βρει χρησιμότητα παντού.

5 <http://lucene.apache.org/>

2.3.3 Χαρακτηριστικά

Οι αλγόριθμοι του Lucene είναι ισχυροί, ακριβείς και αποτελεσματικοί. Κάποια χαρακτηριστικά τους είναι τα παρακάτω⁶:

- Ταξινομημένη αναζήτηση – τα καλύτερα αποτελέσματα επιστρέφονται πρώτα
- Πολλοί ισχυροί τύποι ερωτημάτων όπως query parser(χρησιμοποιείται στην εργασία), phrase queries, wildcard queries, proximity queries, range queries κ.ά.
- Αναζήτηση πεδίων, π.χ. Title, rank, key
- Ταξινόμηση ανα πεδίο
- Αναζήτηση πολλαπλών ευρετηρίων με συγχωνευμένα αποτελέσματα
- Επιτρέπει την ταυτόχρονη αναζήτηση και ενημέρωση
- Ενώνει και ομαδοποιεί αποτελέσματα
- Γρήγορο, αποδοτική χρήση μνήμας και ανεκτικό σε τυπογραφικά λάθη
- Δυνατότητα σύνδεσης με μοντέλα κατάταξης, συμπεριλαμβανομένων των Vector Space Model και Okapi BM25
- Διαμορφώσιμος μηχανισμός αναζήτησης

2.4 Apache Poi

Το Apache Poi⁷, ένα πρόγραμμα που υλοποιήθηκε από την Apache Software Foundation, και προηγουμένως ένα υποέργο του προγράμματος Τζακάρτα, παρέχει βιβλιοθήκες Java για την ανάγνωση και εγγραφή αρχείων σε μορφοποίηση Microsoft Office, όπως είναι το Word, το PowerPoint και το Excel.

⁶ <http://lucene.apache.org/core/>

⁷ <http://poi.apache.org/>

Το όνομά του ήταν αρχικά ένα ακρονύμιο για “ανεπαρκής ασαφής υλοποίηση”(Elango Sundaram, 2004), το οποίο παραπέμπει χιουμοριστικά στο γεγονός ότι οι τύποι των αρχείων φαίνεται να είναι σκοπίμως ασαφής, αλλά κακώς, δεδομένου ότι ήταν επιτυχώς αντίστοιχη μηχανική. Αυτή η εξήγηση – και εκείνες των παρόμοιων ονομάτων για διάφορα υποέργα – απομακρύνθηκαν από τις επίσημες ιστοσελίδες με σκοπό την καλύτερη πώληση των εργαλείων στις επιχειρήσεις που δεν θα θεωρήσουν το εν λόγω χιούμορ σκόπιμο.

Το Apache Poι περιλαμβάνει τα ακόλουθα επιμέρους στοιχεία:

- POIFS (Poor Obfuscation Implementation File System)
- HSSF (Horrible SpreadSheet Format)
- XSSF (XML SpreadSheet Format)
- HPSE (Horrible Property Set Format)
- HWPF (Horrible Word Processor Format)
- HSLF (Horrible Slide Layout Format)
- HDGF (Horrible DiaGram Format)
- HPBF (Horrible PuBlisher Format)
- HSMF (Horrible Stupid Mail Format)
- DDF (Dreadful Drawing Format)

Το στοιχείο HSSF είναι η πιο προηγμένη και πιο χρήσιμη δυνατότητα της βιβλιοθήκης, αφού χάρη σε αυτήν μπορεί η Java να διαβάσει αρχεία Microsoft Excel(αυτό το στοιχείο έχει χρησιμοποιηθεί και στην εργασία). Επίσης τα HPSE, HWPF και HSLF είναι χρήσιμα, αλλά με λιγότερες δυνατότητες.

2.5 Σχετικά project

Γενικά υπάρχουν πολλά και διάφορα project που ανοίγουν και διαβάζουν αρχεία βιβλιογραφίας σε bibtex format. Κανένα όμως από αυτά δεν κάνει αξιολόγηση των εγγραφών γεγονός που κάνει την εργασία αυτή πιο χρήσιμη.

Κάποια από τα projects αυτά είναι τα παρακάτω:

- **bibtexbrowser**⁸: ανοίγει το αρχείο με rhr και εμφανίζει τις εγγραφές με διάφορες κατηγοριοποιήσεις(ανά συγγραφέα, έτος, περιοδικό κ.ά.). Επίσης εμφανίζει δεδομένα από το Google Scholar καθώς και μια διεύθυνση, όπου βρίσκονται αναφορές ανά έγγραφο από το Google Scholar, εάν η εγγραφή περιέχει το πεδίο `gsid`(Google Scholar id). Υποστηρίζει τα προγράμματα Zotero⁹ και Mendeley¹⁰.
- **bibtex2html**¹¹: όπως λέει και ο τίτλος του, το πρόγραμμα αυτό μετατρέπει ένα αρχείο bibtex σε html. Εμφανίζει την βιβλιογραφία και δίπλα έναν σύνδεσμο για το πώς αυτή γράφεται στο αρχείο. Εμφανίζει τις εγγραφές με τον όπως φαίνεται παρακάτω:

[1] K. R. Apt and E.-R. Olderog. *Verification of sequential and concurrent programs*. Springer-Verlag, 1991. [[bib](#)]

- **BibBase**¹²: είναι μια βάση με αρχεία βιβλιογραφίας πολλών συγγραφέων και επιτρέπει στους χρήστες να κατηγοριοποιήσουν τις δημοσιεύσεις ανα συγγραφέα, έτος, τύπο. Μπορεί να ενσωματωθεί σε ιστοσελίδες. Υποστηρίζει επίσης τα προγράμματα Zotero, Mendeley και DBLP¹³

Όλα τα παραπάνω(όπως και πολλά ακόμα) είναι εργαλεία με τα οόια μπορεί κανείς να χρησιμοποιήσει αρχεία bibtex αλλά κανένα από αυτά δεν αξιολογεί τις εγγραφές. Κάποια μπορούν να μπουν σε site, άλλα διαθέτουν rss feed, ακόμα και download links αλλά όχι αξιολόγηση.

8 <http://www.monperrus.net/martin/bibtexbrowser/>

9 <http://www.zotero.org/>

10 <http://www.mendeley.com/>

11 <https://www.lri.fr/~filliatr/bibtex2html/>

12 <http://bibbase.org/>

13 <http://www.dblp.org/search/index.php>

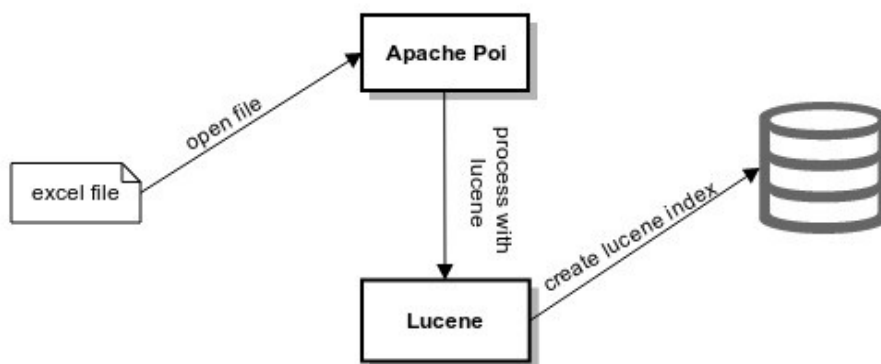
3 ΣΧΕΔΙΑΣΗ ΣΥΣΤΗΜΑΤΟΣ

Το σύστημα αποτελείται από δύο επιμέρους προγράμματα, ένα για την δημιουργία βάσεων/ευρετηρίων Lucene και το κυρίως πρόγραμμα που είναι στην ουσία ένας ιστότοπος. Σε γενικές γραμμές, η σχεδίαση και των δύο είναι απλή.

3.1 Δημιουργία ευρετηρίων

Αρχικά, για να δημιουργηθεί η βάση/ευρετήριο Lucene για τα συνέδρια και τα περιοδικά, έπρεπε να “διαβαστούν” τα αρχεία που ήταν σε μορφή excel και να εγγραφούν με τρόπο τέτοιο που να επιτρέπει στις βιβλιοθήκες του Lucene να αξιοποιούν τις δυνατότητές τους. Άρα, το πρόγραμμα ανοίγει το αρχείο με την βοήθεια των βιβλιοθηκών του Apache Poi και παίρνει τα πεδία που χρειάζεται. Μετά δημιουργεί τα αντίστοιχα ευρετήρια Lucene με τα πεδία που έχει πάρει από το αρχείο excel. Πλέον, έχουν δημιουργηθεί δυο ξεχωριστά ευρετήρια, ένα για τα περιοδικά και ένα για τα συνέδρια.

Στην εικόνα 1 βλέπουμε τον σχεδιασμό του πρώτου μέρους:



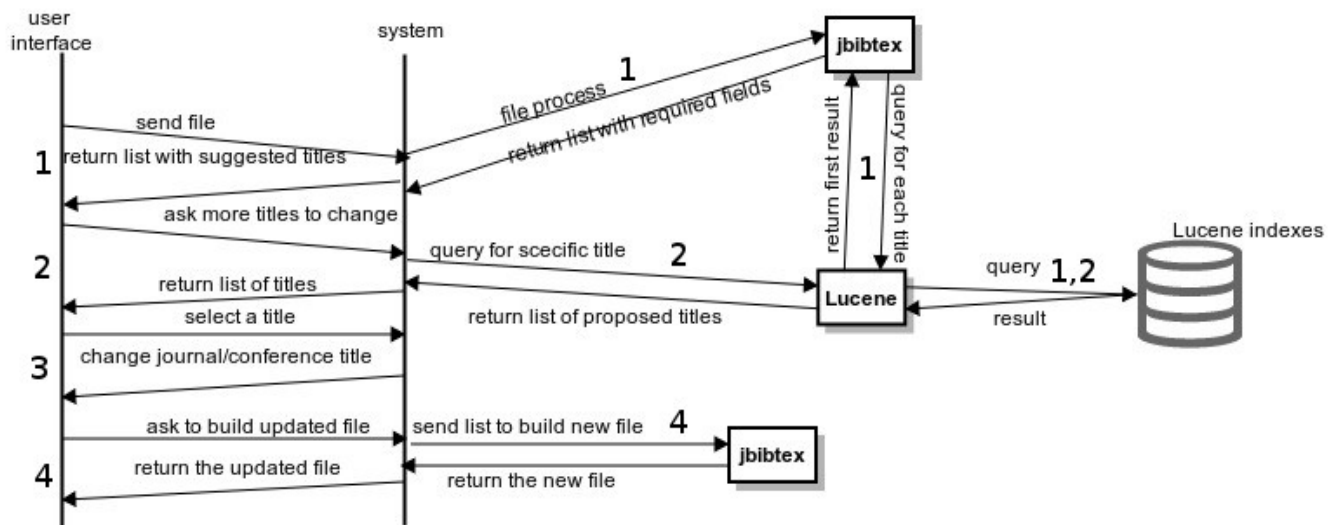
Εικόνα 1: Φόρτωση της λίστας συνεδρίων/περιδικών από το αρχείο excel xls στο το Lucene index directory

3.2 Κυρίως πρόγραμμα

Το κυρίως πρόγραμμα έχει στην ουσία τρεις λειτουργίες. Αρχικά διαβάσει ένα αρχείο σε `bibtex format` που ανεβάζει ο χρήστης στον `server` και περνάει τα δεδομένα που χρειάζεται σε μια λίστα για να είναι πιο εύκολη η διαχείριση. Έπειτα, για κάθε άρθρο της λίστας που είναι δημοσιευμένο σε περιοδικό ή συνέδριο, βρίσκει τους αντίστοιχους τίτλους περιοδικών ή συνεδρίων που ταιριάζουν πιο πολύ στην κάθε εγγραφή και κρατάει τον πρώτο. Το πρόγραμμα δίνει στον χρήστη την δυνατότητα να επιλέξει άλλο τίτλο ή αν θεωρεί ότι δεν ταιριάζει κανένας από τους προτεινόμενους, υπάρχει η επιλογή “Not Found”. Τέλος, αφού έχουν γίνει οι απαραίτητες αλλαγές από τον χρήστη, υπάρχει η επιλογή “build” όπου το σύστημα δημιουργεί ένα νέο ανανεωμένο αρχείο με επιπλέον πεδία αυτά του τίτλου συνεδρίου/περιοδικού και την αντίστοιχη αξιολόγηση. Αφού δημιουργηθεί το νέο αρχείο, το σύστημα εμφανίζει ένα μέσο όρο των αξιολογήσεων του αρχείου καθώς και ένα ιστόγραμμα με τις αξιολογήσεις και την συχνότητα με την οποία αυτές εμφανίζονται στο αρχείο. Το αρχείο αυτό είναι πλέον διαθέσιμο στον χρήστη να το κατεβάσει.

Στην εικόνα 2 βλέπουμε τον σχεδιασμό του συστήματος και πώς αυτό αλληλεπιδρά με τα διάφρα εργαλεία για να καλύψει τις ανάγκες του χρήστη. Πρόκειται στην ουσία για ένα διάγραμμα ακολουθίας που δείχνει και το βασικό σενάριο χρήσης όπου ο χρήστης αξιοποιεί όλες τις δυνατότητες του συστήματος κάνοντας αλλαγές και κατεβάζοντας το αρχείο.

Εδώ βέβαια αξίζει να σημειωθεί ότι ο χρήστης μπορεί να χρησιμοποιήσει το σύστημα και με το ανανεωμένο πλέον αρχείο. Στην περίπτωση αυτή το πρόγραμμα δεν κάνει αυτόματη κατηγοριοποίηση αλλά παίρνει τα δεδομένα που έχει ήδη το αρχείο και απλά δίνει την δυνατότητα αλλαγών.



Εικόνα 2: Σχεδιασμός λειτουργίας του κυρίως συστήματος

Στην ενέργεια 1 βλέπουμε ότι ο χρήστης ανεβάζει το αρχείο, αυτό περνάει από το jbibtex για να αποθηκευθούν τα απαιτούμενα δεδομένα σε μια λίστα και επικοινωνεί με το Lucene για να βρει τον τίτλο που ταιριάζει πιο πολύ(εφόσον δεν είναι ήδη αξιολογημένο) και να καταχωρήσει την αξιολόγηση.

Στην ενέργεια 2, ο χρήστης επιθυμεί να αλλάξει τον προεπιλεγμένο τίτλο και το σύστημα ζητάει από το Lucene να επιστρέψει τις δέκα πρώτες επιλογές για τον τίτλο αυτό καθώς και την επιλογή “Not Found”.

Στην ενέργεια 3 βλέπουμε ότι το σύστημα δεν αλληλεπιδρά με το jbibtex για την αλλαγή και αυτό γιατί όπως είπαμε από την στιγμή που ανεβάζει ο χρήστης το αρχείο μέχρι την στιγμή που ζητάει να φτιάξει το νέο αρχείο, το πρόγραμμα χρησιμοποιεί για ευκολία την λίστα και ό,τι αλλαγές γίνουν απλά τις περνάει μετά στο αρχείο.

Η τελευταία ενέργεια που μπορεί να κάνει ο χρήστης είναι, αφού έχει κάνει τις αλλαγές που επιθυμεί, να κατεβάσει το αρχείο ανανεωμένο. Βλέπουμε ότι εδώ το σύστημα πάλι χρησιμοποιεί το jbibtex, αυτήν την φορά για να δημιουργήσει το αρχείο με τα επιπλέον δεδομένα της λίστας.

3.3 Το αρχείο πριν και μετά

Σε αυτό το σημείο θα δείξουμε πώς διαμορφώνεται μια εγγραφή του αρχείου όταν περάσει από το σύστημα.

Πριν:

```
@article{smeulders00contentbased,  
  author = {Arnold W. M. Smeulders and Marcel Worring and Simone Santini and Amarnath Gupta and Ramesh Jain},  
  title = {Content-Based Image Retrieval at the End of the Early Years},  
  journal = {IEEE Trans. Pattern Anal. Mach. Intell.},  
  volume = {22},  
  number = {12},  
  year = {2000},  
  issn = {0162-8828},  
  pages = {1349--1380},  
  doi = {dx.doi.org/10.1109/34.895972},  
  publisher = {IEEE Computer Society},  
  address = {Washington, DC, USA},  
}
```

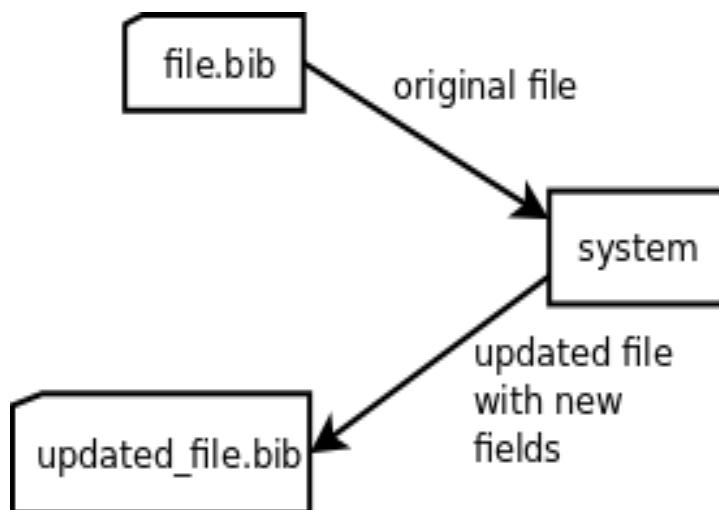
Μετά:

```
@article{smeulders00contentbased,  
  author = {Arnold W. M. Smeulders and Marcel Worring and Simone Santini and Amarnath Gupta and Ramesh Jain},  
  title = {Content-Based Image Retrieval at the End of the Early Years},  
  journal = {IEEE Trans. Pattern Anal. Mach. Intell.},  
  volume = {22},  
  number = {12},  
  year = {2000},  
  issn = {0162-8828},  
  pages = {1349--1380},  
  doi = {dx.doi.org/10.1109/34.895972},  
  publisher = {IEEE Computer Society},  
  address = {Washington, DC, USA},
```

```
journal-title = {IEEE Transactions on Pattern Analysis and Machine Intelligence},  
rank = {A*}
```

Με γρι φόντο φαίνονται τα επιπλέον στοιχεία που προστέθηκαν στο αρχείο. Αν επρόκειτο για συνέδριο, τότε αντί για journal-title θα έγραφε conference-title. Επίσης, αν δεν βρεθεί ο τίτλος(και εφόσον πρόκειται για περιοδικό ή συνέδριο) στα conference-title και rank θα υπάρχουν οι τιμές “Not Found” και “Not Ranked” αντίστοιχα. Αν το συγκρίνει κανείς με το journal (με ανοικτό γκρι) που είναι αυτό που έχει δώσει ο συγγραφέας του bib θα δούμε ότι υπάρχουν μικρές διαφορές.

Στο παρακάτω σχήμα(εικόνα 3) φαίνεται η λειτουργία του συστήματος όπως την αντιλαμβάνεται ο χρήστης:



Εικόνα 3: Η λειτουργία όπως την αντιλαμβάνεται ο χρήστης

Ο χρήστης έχει επικοινωνία μόνο με το interface του συστήματος οπότε δεν βλέπει και δεν τον ενδιαφέρει πώς λειτουργεί αυτό.

4 ΥΛΟΠΟΙΗΣΗ

4.1 Από τα xls στα Lucene indexes

Για να δημιουργηθεί ευρετήριο Lucene από τα αρχεία xls, χρησιμοποιήθηκαν οι βιβλιοθήκες Apache Poi, Lucene και προφανώς τα xls αρχεία με τις αξιολογήσεις των συνεδρίων και των περιοδικών.

4.1.1 Διάβασμα xls

Αρχικά, με την αξιοποίηση του HSSF στοιχείου του Apache Poi, το πρόγραμμα διαβάζει τα δεδομένα από το xls αρχείο και τα περνάει σε μια λίστα, όπως φαίνεται από τον παρακάτω κώδικα:

```
private static ArrayList<conference> ReadConf(String fileName) {
    ArrayList<conference> conf_ar = new ArrayList<conference>();
    try {
        FileInputStream myInput = new FileInputStream(fileName);
        POIFSFileSystem myFileSystem = new POIFSFileSystem(myInput);
        HSSFWorkbook myWorkBook = new HSSFWorkbook(myFileSystem);
        HSSFSheet mySheet = myWorkBook.getSheetAt(0);
        Iterator rowIter = mySheet.rowIterator();
        while (rowIter.hasNext()) {
            HSSFRow myRow = (HSSFRow) rowIter.next();
            Iterator cellIter = myRow.cellIterator();
            conference conf = new conference();
            if ( myRow.getRowNum() == 0 ){
                continue;
            }
            for ( int i = 0; i < 4; i++ ) {
                HSSFCell myCell = (HSSFCell) cellIter.next();
```

```
        if ( i == 0 ){
            String k = myCell.toString();
            char[] e = k.toCharArray();
            String k1 = new String(e);
            float num = Float.parseFloat(k1);
            int id = (int) num;
            conf.setConf_id( id );
        }else if ( i == 1 ){
            conf.setTitle(myCell.toString());
        }else if ( i == 2 ){
            conf.setAcronym(myCell.toString());
        }else{
            conf.setRank(myCell.toString());
        }
    }
    conf_ar.add(conf);
}
} catch (Exception e) {
}
return conf_ar;
}
```

Έχουν χρησιμοποιηθεί δυο διαφορετικές μέθοδοι, μια για συνέδρια(όπως ο παραπάνω) και μια για περιοδικά αντίστοιχα. Ο λόγος είναι ότι για τα συνέδρια αποθηκεύεται ένα επιπλέον στοιχείο, το ακρονύμιο, μιας και το Lucene τα αναγνωρίζει και βοηθάει στην αναζήτηση.

4.1.2 Δημιουργία ευρετηρίων

Στην συνέχεια, δημιουργούνται δυο ευρετήρια Lucene, ένα για κάθε κατηγορία(συνέδρια-περιοδικά), και παίρνουν τα στοιχεία από τις αντίστοιχες λίστες. Παρακάτω φαίνεται ο κώδικας που βάζει τα στοιχεία από την λίστα των συνεδρίων στο αντίστοιχο ευρετήριο:

```
File conf_file = new File("files/conference_index"); //Δημιουργία φακέλου για το νέο ευρετήριο
Analyzer analyzer = new StandardAnalyzer(Version.LUCENE_35);
Directory Cdirectory = FSDirectory.open(conf_file, null);
    IndexWriterConfig Cconfig = new IndexWriterConfig(
        Version.LUCENE_35, analyzer); //Περνάει τα δεδομένα με την βοήθεια του analyzer
    IndexWriter Cwriter = new IndexWriter(Cdirectory, Cconfig);
    List<Document> Cdocs = new ArrayList<Document>();
    for (int i = 0; i < conf_ar.size(); i++) { //conf_ar είναι η λίστα με τα συνέδρια
        Document Cdoc = new Document();
        int k = conf_ar.get(i).getConf_id();
        String k1 = Integer.toString(k);
        Cdoc.add(new Field("ERAID", k1 ,
            Field.Store.YES, Field.Index.ANALYZED));
        Cdoc.add(new Field("Title", conf_ar.get(i).getTitle(),
            Field.Store.YES, Field.Index.ANALYZED));
        Cdoc.add(new Field("Acronym", conf_ar.get(i).getAcronym(),
            Field.Store.YES, Field.Index.ANALYZED));
        Cdoc.add(new Field("Rank", conf_ar.get(i).getRank(),
            Field.Store.YES, Field.Index.ANALYZED));
        Cdocs.add(Cdoc);
    }
    Cwriter.addDocuments(Cdocs);
    Cwriter.close();
    Cdirectory.close();
```

Το Lucene έχει πολλούς τρόπους αποθήκευσης και αναζήτησης δεδομένων και τα ονομάζει Analyzers. Στην εργασία αυτή χρησιμοποιήθηκε για την εισαγωγή των στοιχείων ο StandardAnalyzer για τον λόγο ότι “κόβει” τα λεγόμενα “stopwords” από την αναζήτηση και διατηρεί και τις ιδιότητες που είχαμε αναφέρει στο 2ο κεφάλαιο.

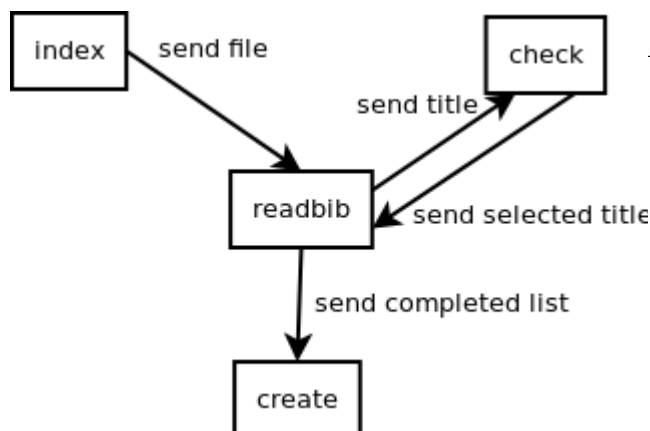
4.2 Ιστοτόπος

4.2.1 Γενικά

Για την δημιουργία του ιστοτόπου προτιμήθηκε η γλώσσα JsP(JavaServer Pages) που είναι συνδυασμός html και java. Επίσης, χρησιμοποιήθηκαν κάποια στοιχεία από css για την μορφοποίηση των σελίδων. Ο ιστοτόπος δεν περιέχει πολλές σελίδες, αφού όπως είδαμε και στον σχεδιασμό, έχει λίγες λειτουργίες. Οι σελίδες που περιέχει είναι οι παρακάτω και θα αναληθούν στην συνέχεια:

- index.jsp
- readbib.jsp
- check.jsp
- create.jsp

Στην εικόνα 4 φαίνεται η επικοινωνία μεταξύ των σελίδων:



Εικόνα 4: Επικοινωνία μεταξύ των σελίδων

Όπως φαίνεται και από την εικόνα, η κύρια σελίδα που περνάει τον περισσότερο χρόνο ο χρήστης είναι η readbib.jsp, αφού φεύγει από αυτήν μόνο όταν θέλει να αλλάξει κάποιον τίτλο ή όταν έχει τελειώσει με τις αλλαγές και θέλει να κατεβάσει το αρχείο.

4.2.2 Index.jsp

Η index.jsp είναι η αρχική σελίδα και δεν περιέχει κάτι το ιδιαίτερο. Εδώ υπάρχει απλά μια φόρμα για να επιλέξει ο χρήστης το αρχείο που θα ανεβάσει. Αξίζει να σημειωθεί ότι γίνεται έλεγχος του ονόματος του αρχείου ώστε και να έχει επιλεγθεί και να έχει την κατάληξη “bib”. Ο έλεγχος αυτός γίνεται μέσω ενός javascript και φαίνεται στο παρακάτω κομμάτι κώδικα:

```
function Checkfiles(){
    var fup = document.getElementById('filename');
    var fileName = fup.value;
    var ext = fileName.substring(fileName.lastIndexOf('.') + 1);
    if(ext == "bib" ){ //Απλά ελέγχει την κατάληξη και όχι το περιεχόμενο
        return true;
    } else{
        alert("Upload bib files only");
        fup.focus();
        return false;
    }
}
```

Όπως φάνηκε στον κώδικα, ο έλεγχος είναι μόνο για την κατάληξη και όχι για το περιεχόμενο. Επίσης, εμφανίζει το ίδιο μήνυμα και σε περίπτωση που δεν επιλεγθεί κάποιο αρχείο από τον χρήστη. Το περιεχόμενο ελέγχεται στην readbib.jsp που περιγράφεται αμέσως μετά.

4.2.3 Readbib.jsp

Η `readbib.jsp` περιέχει το μεγαλύτερο μέρος του κώδικα του προγράμματος αφού από την στιγμή που ανέβει το αρχείο στον `server`, αυτή είναι πλέον η κεντρική σελίδα. Αρχικά, παίρνει το αρχείο `bib` και το αποθηκεύει στο σύστημα. Εδώ, για λόγους ασφαλείας ότι δεν θα μπερδευτούν αρχεία με ίδιο όνομα από διαφορετικούς χρήστες, το αρχείο αποθηκεύεται μαζί με ένα `timestamp` εντός μιας `session`. Από το αρχείο, το σύστημα κρατάει μια λίστα με όλα τα στοιχεία που χρειάζεται για την λειτουργία του έτσι ώστε να μην κάνει αλλαγές κάθε φορά στο ίδιο το αρχείο. Εν συνεχεία, εμφανίζει στον χρήστη έναν πίνακα με τους τίτλους των δημοσιεύσεων από το αρχείο που ανέβασε και όποτε ο χρήστης επιλέγει έναν τίτλο, δείχνει από κάτω τον προτεινόμενο τίτλο συνεδρίου ή περιοδικού και την αντίστοιχη βαθμολογία. Οι μέθοδοι που χρησιμοποιούνται φαίνονται αργότερα.

Όπως αναφέρθηκε προηγουμένως, σε αυτήν την σελίδα ελέγχεται και το περιεχόμενο του αρχείου. Αφού λοιπόν “περάσει” από τον έλεγχο της αρχικής σελίδας(απλά ελέγχει το τελείωμα του τίτλου), εδώ πρέπει να διαβαστεί και από το σύστημα. Αν το αρχείο έχει κάποιο πρόβλημα ή απλά δεν πληροί τις προδιαγραφές του `bibtex format`, εμφανίζεται αντίστοιχο μήνυμα στον χρήστη και το πρόγραμμα δεν συνεχίζει παρακάτω.

4.2.4 Check.jsp

Στην σελίδα αυτή ο χρήστης μπορεί να αλλάξει τον προεπιλεγμένο τίτλο. Εδώ του επιστρέφονται οι δέκα τίτλοι συνεδρίων/περιοδικών που ταιριάζουν περισσότερο (σύμφωνα με την βαθμολόγηση του `Lucene`) ώστε να μπορεί να διαλέξει. Επίσης, υπάρχει η επιλογή “`Not Found`”, αφού είναι αρκετά πιθανό κάποια περιοδικά ή συνέδρια να μην υπάρχουν στην κατηγοριοποιημένη λίστα του συστήματος.

4.2.5 Create.jsp

Αυτή είναι η τελευταία σελίδα που θα επισκευθεί ο χρήστης αφού για να βρεθεί εδώ έχει ζητήσει να “φτιαχτεί” το καινούριο ανανεωμένο αρχείο για να το κατεβάσει. Πέρα όμως από αυτό, του εμφανίζεται το ποσοστό δημοσιεύσεων ανά κατηγορία, ένας μέσος όρος αξιολόγησης των δημοσιεύσεων του αρχείου καθώς και ένα ιστόγραμμα αξιολογήσεων. Ο μέσος όρος αξιολόγησης αφορά τις δημοσιεύσεις όπου βρέθηκε αντίστοιχος τίτλος και έχουν αξιολογηθεί, δηλαδή δεν είναι “Not Ranked”, και χρησιμοποιεί ειδικά βάρη ($A^*=10$, $A=9$, $B=7$, $C=5$) οπότε αναλόγως του τελικού μέσου όρου, ειπλέγεται και η ανάλογη βαθμολόγηση.

4.3 Κλάσεις

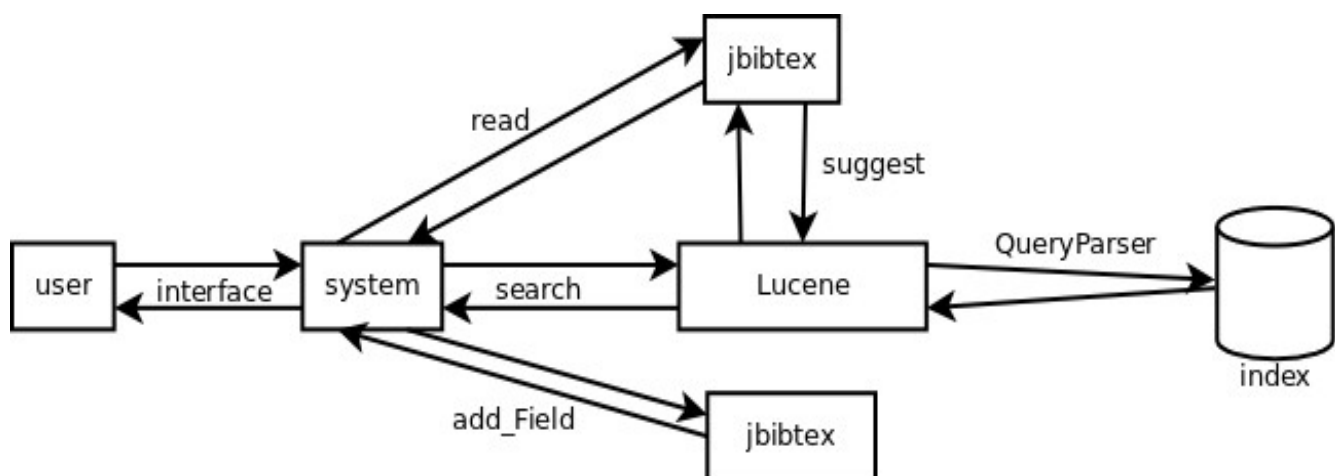
Οι κλάσεις που χρησιμοποιήθηκαν είναι οι ακόλουθες:

- **user** : Η κλάση user υπάρχει για να υποστηρίζει το session και κρατάει το όνομα του αρχείου στο session αυτό
- **methods** : περιέχει σχεδόν όλες τις μεθόδους που χρησιμοποιούνται στο σύστημα
- **results** : περιέχει τα πεδία title και rank και χρησιμοποιείται για την δημιουργία λίστας αποτελεσμάτων όταν καλείται η check.jsp
- **item** : περιέχει key, entry_title, article_title, result, rank και status και χρησιμοποιείται για την δημιουργία της κυρίως λίστας που επεξεργάζεται το σύστημα μέχρι να φτιάξει το ανανεωμένο αρχείο. Το status δείχνει αν ο τίτλος που φαίνεται είναι ο προεπιλεγμένος, ο επιλεγμένος ή ο ήδη υπάρχων

4.4 Η κλάση *methods*

Η κλάση *methods* περιέχει σχεδόν όλες τις μεθόδους που χρησιμοποιούνται από το σύστημα. Ο σκοπός της δημιουργίας της είναι να ελαχιστοποιεί τα κομμάτια κώδικα που βρίσκονται στις σελίδες *jsp* και να επιτρέπει την επαναχρησιμοποίηση των μεθόδων. Οι μέθοδοι που υπάρχουν μέσα σε αυτήν την κλάση περιγράφονται παρακάτω.

Στην παρακάτω εικόνα(εικόνα 5) βλέπουμε πώς αλληλεπιδρούν όλες οι μέθοδοι με το σύστημα και τα διάφορα εργαλεία:



Εικόνα 5: Αλληλεπίδραση όλων των μεθόδων με το σύστημα, τις βιβλιοθήκες και την βάση

4.4.1 Μέθοδοι *read* και *parseBibTeX*

Η μέθοδος *read* είναι αυτή που παίρνει ένα αρχείο μορφοποίησης *bib* και περνάει τα δεδομένα που χρειάζεται το πρόγραμμα σε μια λίστα. Για να το κάνει αυτό, αρχικά χρειάζεται να πάρει μια βάση με εγγραφές *bibtex* εκτελώντας την εντολή:

```
BibTeXDatabase database = parseBibTeX(input);
```

όπου το *input* είναι το όνομα του αρχείου που ανέβασε ο χρήστης.

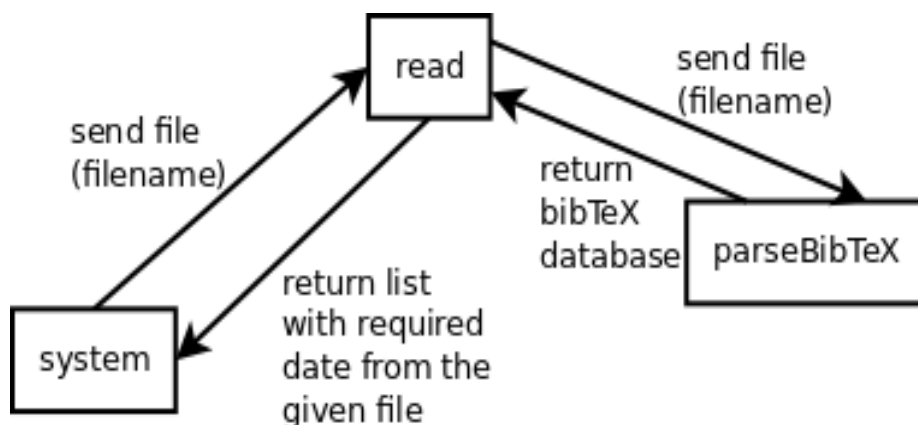
Στην συνέχεια, με την παρακάτω εντολή:

```
Collection<BibTeXEntry> entries = (database.getEntries()).values();
```

δημιουργεί μια συλλογή με τις τιμές των εγγραφών της βάσης που πήρε από την μέθοδο `parseBibTeX`.

Τέλος, τα δεδομένα της συλλογής πλέον μπορεί να τα επεξεργαστεί και τα περνάει σε μια λίστα την οποία επιστρέφει στο πρόγραμμα. Εδώ βέβαια γίνεται και ένας έλεγχος αν έχει ήδη βαθμολογηθεί η εγγραφή για να ξέρει αν χρειάζεται να κάνει την αυτόματη αναζήτηση.

Στην εικόνα 6 φαίνεται η λειτουργία των δυο αυτών μεθόδων:



Εικόνα 6: Επικοινωνία `system-read-parseBibTeX`

4.4.2 Μέθοδοι `add_Field`, `parseBibTeX` και `formatBibTeX`

Όταν ο χρήστης τελειώσει με την επεξεργασία του αρχείου και θέλει πλέον να το κατεβάσει με τα πρόσθετα στοιχεία του, πηγαίνοντας στην σελίδα `create.jsp`, αυτή καλεί την `add_Field` για να φτιάξει το νέο αρχείο.

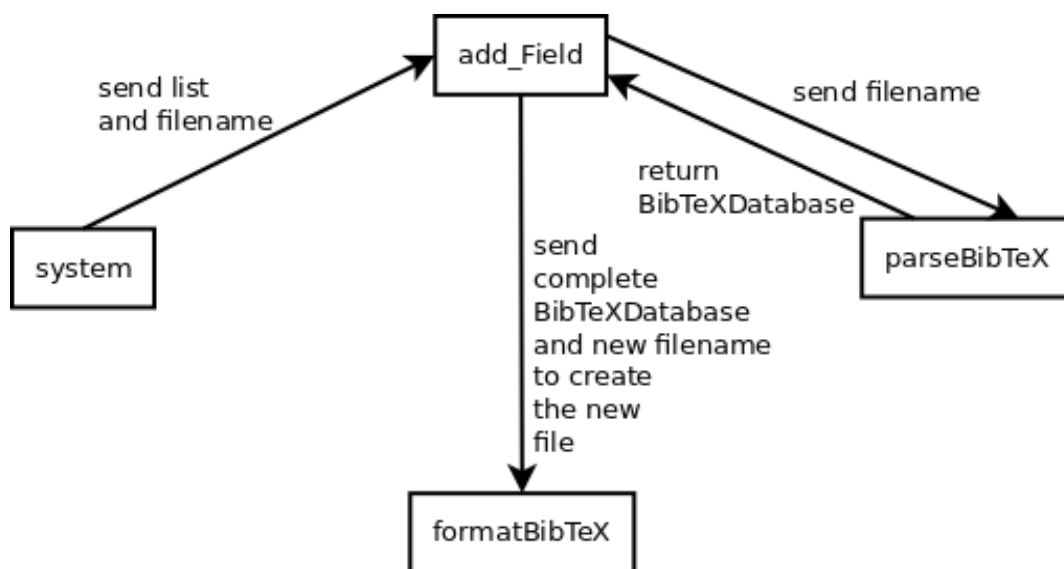
Το σύστημα στέλνει στην μέθοδο `add_Field` την λίστα με τις εγγραφές καθώς και το αρχικό αρχείο(το όνομα μόνο). Η `add_Field` αρχικά κάνει ό,τι και η `read`(στέλνει στην `parseBibTeX` το αρχείο και παίρνει πίσω μια `BibTeXDatabase`) και σε συνδυασμό

με την λίστα που διαθέτει το σύστημα δημιουργεί τα επιπλέον πεδία όπου χρειάζεται. Έπειτα, έχοντας ολοκληρώσει τις αλλαγές στην BibTeXDatabase, την στέλνει στην μέθοδο formatBibTeX μαζί με το όνομα του καινούριου αρχείου.

Ο κώδικας της formatBibTeX φαίνεται παρακάτω:

```
Writer writer = (file != null ? new FileWriter(file) : new OutputStreamWriter(System.out));  
try {  
    BibTeXFormatter formatter = new BibTeXFormatter();  
    formatter.format(database, writer); //Περνάει την BibTeXDatabase στο αρχείο  
} finally {  
    writer.close();  
}
```

Η εικόνα 7 δείχνει την επικοινωνία μεταξύ των μεθόδων και του συστήματος:



Εικόνα 7: Επικοινωνία του συστήματος με τις μεθόδους add_Field, parseBibTeX και formatBibTeX

Όπως φαίνεται και στο διάγραμμα, η μέθοδος formatBibTeX δεν επιστρέφει κάτι ούτε στο σύστημα ούτε στην μέθοδο add_Field αφού είναι μια void μέθοδος.

4.4.3 Μέθοδοι suggest και search

Οι μέθοδοι suggest και search είναι σχεδόν ίδιες. Από τη μία, η suggest χρησιμοποιείται όταν το αρχείο που ανέβασε ο χρήστης δεν έχει ήδη κάποια αξιολόγηση(το σύστημα υποστηρίζει και αρχεία που έχουν ήδη το πεδίο rank). Έτσι, στην αναζήτηση που κάνει για να κάνει την αυτόματη κατηγοριοποίηση, απλά παίρνει το πρώτο αποτέλεσμα, δηλαδή αυτό που ταιριάζει περισσότερο στον τίτλο του περιοδικού ή συνεδρίου αντίστοιχα. Από την άλλη, η μέθοδος search χρησιμοποιείται όταν ο χρήστης θέλει να αλλάξει τον προεπιλεγμένο ή τον ήδη υπάρχων τίτλο και θέλει να δει τις διαθέσιμες επιλογές. Επιλέχθηκε να εμφανίζονται τα δέκα πρώτα αποτελέσματα καθώς και μια επιλογή “Not Found” για τον τίτλο με αξιολόγηση “Not Ranked”. Ο κώδικας της αναζήτησης φαίνεται παρακάτω:

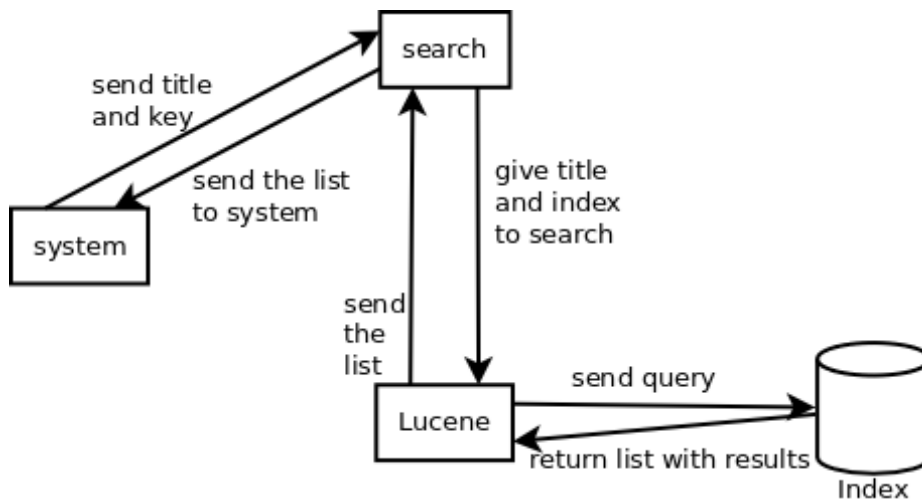
```
Directory directory = FSDirectory.open(file, null); //Ανοίγει τον φάκελο του αντίστοιχου index
Analyzer analyzer = new StandardAnalyzer(Version.LUCENE_35);
IndexReader iRead = IndexReader.open(directory);
IndexSearcher iSearch = new IndexSearcher(iRead);
QueryParser parser = new QueryParser(Version.LUCENE_35, "Title", analyzer); //Μέθοδος
                                                                    αναζήτησης QueryParser τύπου
                                                                    StandardAnalyzer στο πεδίο του τίτλου
String title2=MultiFieldQueryParser.escape(title);
Query query = parser.parse(title2); //Περνάει τον τίτλο στο query
ScoreDoc[] hits = iSearch.search(query, 10).scoreDocs; //Κρατάει τα 10 πρώτα αποτελέσματα
```

Στον κώδικα φαίνεται ότι ανοίγει το directory και δημιουργεί έναν searcher με την βοήθεια ενός reader. Στην συνέχεια δείχνει σε ποιο πεδίο θα ψάξει και με ποιόν τρόπο(StandardAnalyzer). Περνάει τον επιλεγμένο τίτλο με βάση του οποίου θα γίνει αναζήτηση στο ευρετήριο και κρατάει τα πρώτα δέκα αποτελέσματα. Αυτά στην συνέχεια τα περνάει σε μια ArrayList τύπου result, όπως αναφέραμε και πιο πάνω, η οποία περιέχει τους τίτλους που εμφανίζονται στην σελίδα check.jsp του συστήματος

όταν ο χρήστης έχει επιλέξει να αλλάξει την αυτόματη επιλογή τίτλου του συνεδρίου ή περιοδικού. Μαζί με τον τίτλο, περιέχει(αλλά δεν εμφανίζει) την αξιολόγηση του καθενός..

Η διαφορά με την μέθοδο suggest είναι ότι στην suggest δεν επιστρέφονται δέκα αποτελέσματα αλλά μόνο το πρώτο και όπως είναι λογικό δεν χρησιμοποιείται λίστα αφού τα αποτελέσματα(τίτλος και αξιολόγηση) αποθηκεύονται κατ' ευθείαν στην λίστα με τα περιεχόμενα του αρχείου που χρησιμοποιείται από το σύστημα.

Η εικόνα 8 απεικονίζει την λειτουργία της search:



Εικόνα 8: Επικοινωνία του συστήματος με την μέθοδο search και αυτό με τα Lucene indexes

Το Lucene δεν είναι ξεχωριστή μέθοδος αλλά επειδή είναι ξεχωριστό και περιέχει τα απαραίτητα εργαλεία για την αναζήτηση, το βάλαμε με αυτόν τον τρόπο για να τονίζεται η λειτουργία του.

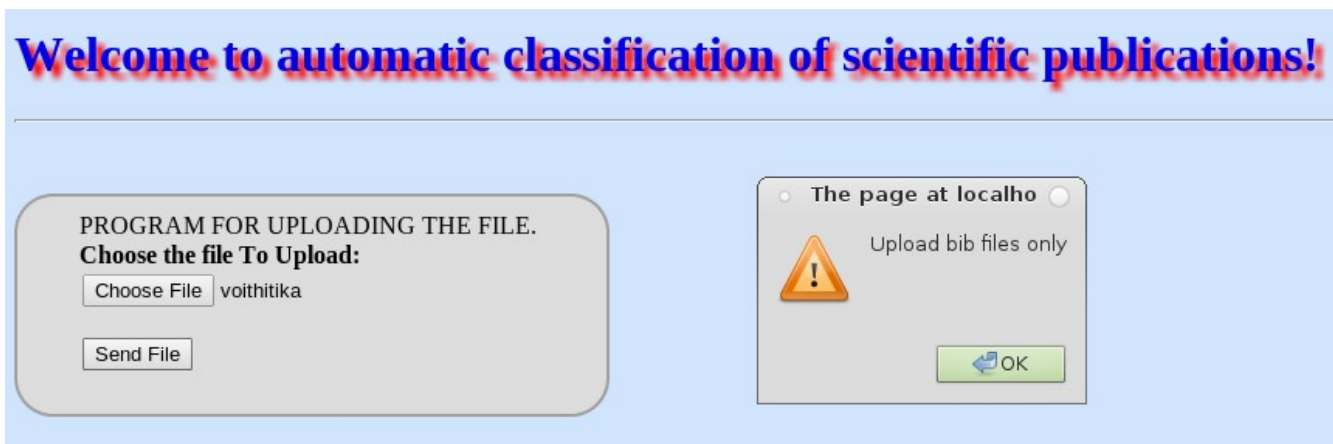
Επίσης, το key που στέλνει το σύστημα στην search του δείχνει αν πρόκειται για συνέδριο ή περιοδικό, άρα χρησιμεύει για να ξέρει η μέθοδος σε ποιο ευρετήριο να ψάξει για τον τίτλο.

5 ΑΞΙΟΛΟΓΗΣΗ – ΣΥΜΠΕΡΑΣΜΑΤΑ

5.1 Ενδεικτική χρήση

Στο κομμάτι αυτό θα δούμε πώς λειτουργεί το σύστημα. Για τον σκοπό αυτό χρησιμοποιήθηκε ένα αρχείο με όλα τα references που έχει το βιβλίο Introduction to Information Retrieval των Manning, Raghavan και Schütze το οποίο μπορεί κανείς να βρει [εδώ](#).

Στην εικόνα 9 φαίνεται η αρχική σελίδα:



Εικόνα 9: Αρχική σελίδα

Εδώ, για να δείξουμε τον έλεγχο βάσει ονόματος, προσπαθήσαμε να ανεβάσουμε ένα αρχείο άλλης μορφοποίησης από κάποιον text editor. Όπως φαίνεται, εμφανίζεται ένα pop-up που επισημαίνει τι είδους αρχεία υποστηρίζει το σύστημα. Αφού λοιπόν επιλέξουμε το αρχείο, πατώντας το κουμπί “Send File”, πηγαίνουμε στην σελίδα readbib.jsp που είναι και η κεντρική σελίδα του συστήματος.

Η readbib.jsp φαίνεται παρακάτω στις εικόνες 10 και 11:

Automatic classification of scientific publications!

Welcome visitor

You have successfully uploaded the file by name: 1380121110518_test.bib

Now you are free to modify it

Titles:

"Proc. National Conference on Artificial Intelligence and Proc. Conference Innovative Applications of Artificial Intelligence", A Unified E
 "IP\&M", Query evaluation: strategies and optimizations
 "Proc. International Conference on Cooperative Information Systems", {P-Grid}: A Self-Organizing Access Structure for {P2P} Informat
 "Advances in information retrieval", Distributed information retrieval
 "Proc. CIKM", Simple {BM25} extension to multiple weighted fields
 "IEEE Trans. Pattern Anal. Mach. Intell.", Content-Based Image Retrieval at the End of the Early Years
 "Proc. SIGIR", Advantages of query biased summaries in information retrieval
 "Not Found", A survey of multilingual text retrieval
 "Not Found", The Turn: Integration of Information Seeking and Retrieval in Context
 "CACM", Natural language processing for information retrieval
 "Not Found", Text Information Retrieval Systems
 "CACM", Trading {MIPS} and memory for knowledge engineering

Build file

Εικόνα 10: Η readbib.jsp δείχνει την λίστα εγγραφών

Automatic classification of scientific publications!

Welcome visitor

You have successfully uploaded the file by name: 1380121110518_test.bib **timestamp** **the saved name in the system**

Now you are free to modify it

Titles:

"Proc. National Conference on Artificial Intelligence and Proc. Conference Innovative Applications of Artificial Intelligence", A Unified E
 "IP\&M", Query evaluation: strategies and optimizations
 "Proc. International Conference on Cooperative Information Systems", {P-Grid}: A Self-Organizing Access Structure for {P2P} Informat
 "Advances in information retrieval", Distributed information retrieval
 "Proc. CIKM", Simple {BM25} extension to multiple weighted fields
 "IEEE Trans. Pattern Anal. Mach. Intell.", Content-Based Image Retrieval at the End of the Early Years
 "Proc. SIGIR", Advantages of query biased summaries in information retrieval
 "Not Found", A survey of multilingual text retrieval
 "Not Found", The Turn: Integration of Information Seeking and Retrieval in Context
 "CACM", Natural language processing for information retrieval
 "Not Found", Text Information Retrieval Systems
 "CACM", Trading {MIPS} and memory for knowledge engineering

Type: conference **type of the article**

Conference: Proc. National Conference on Artificial Intelligence and Proc. Conference Innovative Applications of Artificial Intelligence

Article Title: A Unified Bias-Variance Decomposition for Zero-One and Squared Loss

Suggested Conference Title: Conference on Artificial Intelligence for Applications **suggested conference title by the system**

Rank: B **rank of the suggested title**

Change **button to change conference title**

Build file **button to build the new file**

Εικόνα 11: Η read.jsp δείχνει τον προεπιλεγμένο τίτλο για κάθε εγγραφή

Στο πάνω μέρος της σελίδας, βλέπουμε ένα μήνυμα ότι το αρχείο μεταφορτώθηκε επιτυχώς και εμφανίζεται και το όνομα με το οποίο αποθηκεύθηκε στο σύστημα. Το όνομα περιέχει ένα timestamp για να μην υπάρχει πρόβλημα με άλλα αρχεία ίδιου τίτλου που διαχειρίζονται από το σύστημα την ίδια ώρα.

Πατώντας σε κάποιον τίτλο, βλέπουμε αυτά τα στοιχεία κάτω από το selectbox. Πρόκειται για τα στοιχεία που περιέχει η λίστα και αφού σε αυτό το αρχικό στάδιο δεν έχουμε αλλάξει κάποιον τίτλο, ο επιλεγμένος τίτλος εμφανίζεται σαν “suggested”. Αν ήταν κάποιος τίτλος που προϋπήρχε στο αρχείο, θα εμφανιζόταν σαν “default”.

Πατώντας το κουμπί “change”, πηγαίνουμε στην σελίδα check.jsp που φαίνεται στην εικόνα 12:

Automatic classification of scientific publications!

Title: Proc. International Conference on Cooperative Information Systems
Key: conference

Εδώ εμφανίζεται μια λίστα για να διαλέξει ο χρήστης τίτλο

- International Conference on Cooperative Information Systems
- International Conference on Grid and Cooperative Computing
- International Symposium on Cooperative Database Systems for Advanced Applications
- International Conference on Information Systems
- Cooperative Research Centre for Construction Innovation International Conference
- International Conference on Computer Supported Cooperative Work in Design
- International Conference on Business Information Systems
- International Conference on Enterprise Information Systems
- International Conference on Information Systems Development
- International Conference on Information Systems Security

Choose one: Not Found Επιλογή εάν δεν ταιριάζει κανένας τίτλος Change

*first is default Υποσημείωση ότι ο πρώτος τίτλος είναι ο προεπιλεγμένος

Εικόνα 12: Προτεινόμενοι τίτλοι για το επιλεγμένο συνέδριο ή περιοδικό

Εδώ βλέπουμε την λίστα με τους δέκα προτεινόμενους τίτλους για το επιλεγμένο συνέδριο ή περιοδικό(στην περίπτωση αυτή συνέδριο). Στην ουσία οι επιλογές είναι έντεκα επειδή υπάρχει και η επιλογή “Not Found”, άρα και χωρίς αξιολόγηση(“Not Ranked”). Πατώντας να αλλάξει ο τίτλος, επιστρέφουμε στην σελίδα readbib.jsp όπως φαίνεται στην επόμενη εικόνα(εικόνα 13):

Automatic classification of scientific publications!

Welcome visitor

You have successfully uploaded the file by name: 1380121739589_test.bib

Now you are free to modify it

Titles:

"Proc. National Conference on Artificial Intelligence and Proc. Conference Innovative Applications of Artificial Intelligence", A Unified E
"IP\&M", Query evaluation: strategies and optimizations
"Proc. International Conference on Cooperative Information Systems", {P-Grid}: A Self-Organizing Access Structure for {P2P} Informa
"Advances in information retrieval", Distributed information retrieval
"Proc. CIKM", Simple {BM25} extension to multiple weighted fields
"IEEE Trans. Pattern Anal. Mach. Intell.", Content-Based Image Retrieval at the End of the Early Years
"Proc. SIGIR", Advantages of query biased summaries in information retrieval
"Not Found", A survey of multilingual text retrieval
"Not Found", The Turn: Integration of Information Seeking and Retrieval in Context
"CACM", Natural language processing for information retrieval
"Not Found", Text Information Retrieval Systems
"CACM", Trading {MIPS} and memory for knowledge engineering

Type: conference

Conference: Proc. International Conference on Cooperative Information Systems

Article Title: {P-Grid}: A Self-Organizing Access Structure for {P2P} Information Systems

Selected Conference Title: International Conference on Grid and Cooperative Computing

Rank: C

→ Ο επιλεγμένος από τον χρήστη τίτλος

Change

Build file

Εικόνα 13: Η readbib.jsp δείχνει τον επιλεγμένο από τον χρήστη τίτλο

Στο τέλος, όταν έχουν γίνει οι αλλαγές που επιθυμεί ο χρήστης(αν επιθυμεί), με το κουμπί “Build File” πηγαίνει στην τελευταία σελίδα, την create.jsp, όπου πλέον δημιουργείται το νέο αρχείο και μπορεί να το κατεβάσει.

Επιπλέον, του εμφανίζονται κάποια στατιστικά για τις αξιολογήσεις του, όπως επίσης και ένας μέσος όρος αυτών και ένα ιστόγραμμα με τον αριθμό των δημοσιεύσεων ανά κατηγορία.

Το περιεχόμενο της σελίδας create.jsp φαίνεται στην εικόνα 14:

Automatic classification of scientific publications!

The updated file is successfully built

Your new file is ready to download:

Download

Εδώ ο χρήστης πατάει για να κατεβάσει το αρχείο

Μερικά στατιστικά

Ranking A* : 48 >> 14.20%

Ranking A : 53 >> 15.68%

Ranking B : 51 >> 15.09%

Ranking C : 186 >> 55.03%

Μέση αξιολόγηση(όσων έχουν)

Average Ranking : B



Εικόνα 14: Η σελίδα create.jsp με τα στατιστικά και το γράφημα

Τα στατιστικά αφορούν μόνο τις δημοσιεύσεις οι οποίες έχουν αξιολογηθεί. Πηγαίνοντας τον δείκτη(του ποντικιού) πάνω στις μπλε μπάρες, εμφανίζεται αντίστοιχος αριθμός δημοσιεύσεων(εδώ είναι screenshot και δεν φαίνεται).

5.2 Συμπεράσματα

Από το παραπάνω παράδειγμα ενδεικτικής χρήσης του προγράμματος βλέπουμε είναι πολύ πιθανό να υπάρχει μεγάλο ποσοστό των δημοσιεύσεων που δεν έχουν κατηγοριοποιηθεί. Αυτό βέβαια συμβαίνει επειδή τα αρχεία βιβλιογραφίας περιέχουν διάφορους τύπους δημοσιεύσεων που δεν είναι δυνατό να αξιολογηθούν από το πρόγραμμα όπως για παράδειγμα οι εγγραφές βιβλίου. Γενικά δηλαδή ο αριθμός των αξιολογημένων εγγραφών ποικίλει και έτσι μπορεί να αξιολογηθούν από καμία μέχρι όλες οι εγγραφές.

5.3 Μελλοντικές επεκτάσεις

Το πρόγραμμα αυτό σε καμία περίπτωση δεν μπορεί να θεωρηθεί τελειωποιημένο αφού πάντα υπάρχει και κάτι ακόμα που μπορούμε να κάνουμε. Κάποιες πιθανές και χρήσιμες επεκτάσεις είναι οι παρακάτω:

- Λειτουργία με όλες τις “υγιείς” εγγραφές ενός αρχείου διότι σε αυτό το στάδιο το πρόγραμμα δεν διαβάζει αρχεία που έχουν έστω και μία εγγραφή που δεν πληροί τις προδιαγραφές του bibtex format
- Αυτόματη κατηγοριοποίηση ήδη κατηγοριοποιημένου αρχείου. Όταν ένα αρχείο είναι ήδη κατηγοριοποιημένο, το πρόγραμμα δεν το αλλάζει και ο χρήστης μπορεί να κάνει απλά κάποιες αλλαγές χειροκίνητα. Θα μπορούσε λοιπόν να μπει ένα κουμπί για αυτόματη αξιολόγηση διότι το αρχείο μπορεί να είχε αξιολογηθεί με βάση μια παλαιότερη λίστα και ο χρήστης να θέλει να δει την νέα αξιολόγηση
- Φόρτωση επιπλέον λιστών από άλλους οργανισμούς διότι μπορεί να περιέχουν και άλλους τίτλους περιοδικών
- Ενσωμάτωση της διαδικασίας αξιολόγησης στα project που αναφέραμε στο κεφάλαιο 2

ΒΙΒΛΙΟΓΡΑΦΙΑ

Simon Cooper & Anna Poletti. vol. 53, no. 1, 2011 , The new ERA of journal ranking.

Kim Carr, 31 March 2008. [In search of research excellence](#)

Kim Carr, 15 July 2009. "[New era in research will cut the red tape](#)". [The Australian](#).

Tim Mazzarol , Geoffrey Norman Soutar, 11 July 2011). "[Why the ERA had to change and what we should do next](#)". The Conversation.

Margaret Sheil, 23 November 2009. "[Scopus announced as citation provider for ERA](#)". Australian Research Council.

Grant Ingersoll, 19 November 2007. "[Better Search with Apache Lucene and Solr](#)".

McCandless, Michael; Hatcher, Erik; Gospodnetić, Otis (2010). Lucene in Action, Second Edition. Manning. p. 8. ISBN 1933988177.

Perner, Petra (2007). Machine Learning and Data Mining in Pattern Recognition: 5th International Conference. Springer. p. 387. ISBN 978-3-540-73498-7.

Sundaram, Elango (2004-03-22), [Excelling in Excel with Java](#), Java World.

ΧΡΗΣΙΜΟΙ ΣΥΝΔΕΣΜΟΙ

1. <http://www.arc.gov.au/era/>
2. <http://www.tex.ac.uk/tex-archive/bibliography/bibtex/base/bibtex.web>
3. <https://code.google.com/p/java-bibtex/>
4. <http://www.elsevier.com/online-tools/scopus/content-overview>
5. <http://lucene.apache.org/>
6. <http://lucene.apache.org/core/>
7. <http://poi.apache.org/>
8. <http://www.monperrus.net/martin/bibtexbrowser/>
9. <http://www.zotero.org/>
10. <http://www.mendeley.com/>
11. <https://www.lri.fr/~filliatr/bibtex2html/>
12. <http://bibbase.org/>
13. <http://www.dblp.org/search/index.php>
14. <https://developers.google.com/chart/>