Κατάταξη κειμένων: Δημιουργία υποψηφίων φράσεων-κλειδιών από υπάρχουσες μικρότερες

Νικήτας Ν. Καρανικόλας, Χρήστος Σκουρλάς

Τμήμα Πληροφορικής, ΤΕΙ Αθήνας
nnk@teiath.gr, cskourlas@teiath.gr

Περίληψη

Το πρόβλημα της κατάταξης κειμένων (Text Classification) έχει πολλές όψεις και αντίστοιχες δυνητικές λύσεις. Στην παρούσα εργασία εξετάζουμε δύο ερευνητικές κατευθύνσεις για κατάταξη εγγράφων. Η πρώτη από αυτές βασίζεται στην εξόρυξη γνώσης και στις τεχνικές εξαγωγής (συμπερασμού) κανόνων. Η δεύτερη συνδυάζει παραδοσιακές τεχνικές ανάκτησης κειμένων (όπως χρήση του μοντέλου διανυσματικής αναπαράστασης κειμένων - vector space model, όρων δεικτοδότησης - index terms και μέτρα ομοιότητας - similarity measures), Επεξεργασία Φυσικής Γλώσσας και τεχνικές εκμάθησης με βάση τα παρατηρηθέντα στιγμιότυπα (Instance based Learning). Στις δύο προσεγγίσεις, οι φράσεις κλειδιά (Key-phrases) μπορούν να χρησιμοποιηθούν ως χαρακτηριστικές ιδιότητες (attributes) κατάλληλες για την εξαγωγή κανόνων αλλά και ως βάση για την μέτρηση της ομοιότητας νέων (μη κατεταγμένων) εγγράφων με τα προϋπάρχοντα (κατεταγμένα) έγγραφα. Συνεπώς, η επιλογή των καταλληλότερων φράσεων-κειδιών, όταν αυτές προορίζονται να χρησιμοποιηθούν ως χαρακτηριστικές ιδιότητες για κατάταξη κειμένων, είναι μία ιδιαίτερα σημαντική διαδικασία. Για την διαδικασία αυτή παρουσιάζουμε ένα νέο αλγόριθμο ο οποίος βασίζεται στην ιδέα ότι οι φράσεις-κλειδιά ν συνθετικών (ν λέξεων) μπορούν να προκύψουν από συνδυασμούς φράσεων-κλειδιών λιγότερο «συνθετικών».

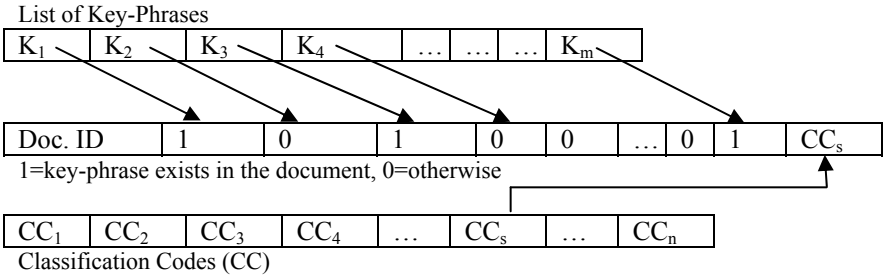
Keywords: text classification, key-phrase extraction, text indexing, information retrieval, document management

1. Εισαγωγή

Ως Κατάταξη Κειμένων μπορούμε να ορίσουμε την εφαρμογή μεθόδων (ημι-) αυτόματης επιλογής της ομάδας, από ένα σύνολο προκαθορισμένων ομάδων, στην οποία ανήκει ένα έγγραφο. Για παράδειγμα αν μιλήσουμε για ιατρικά εξιτήρια, τότε με τον όρο κατάταξη εννοούμε την αυτόματη ή ημιαυτόματη επιλογή του κωδικού ασθενείας και διάγνωσης που αντιστοιχεί στο περιεχόμενο (κείμενο) του εξιτηρίου [8]. Δηλαδή, μιλάμε για την επιλογή ενός κωδικού από το σύνολο των διαθεσίμων κωδικών ενός συστήματος ταξινόμησης ασθενειών και διαγνώσεων όπως είναι το σύστημα ICD (International Classification of Deceases and Diagnoses).

Η Helena Ahonen [2], το 1997, τόνισε τη σημασία που μπορούν να έχουν οι φράσεις για τη περαιτέρω επεξεργασία και οργάνωση εγγράφων (κειμένων). Από τότε πολλές μελέτες έχουν εστιάσει στη δημιουργία μοντέλου (Κανόνων ή Δένδρων αποφάσεων)

που βασίζεται στην παρουσία φράσεων-κλειδιά για να αποφασίσει την κλάση (κατηγορία) στην οποία κατατάσσεται ένα νέο (μη κατεταγμένο) έγγραφο. Όλες αυτές οι μεθόδοι συχνά χρησιμοποιούν εκπαιδευτικά σύνολα (training sets) από προ-κατεταγμένα έγγραφα και μία προκαθορισμένη λίστα από φράσεις-κλειδιά. Οι μέθοδοι αυτές δημιουργούν ένα διάνυσμα, για κάθε έγγραφο του εκπαιδευτικού συνόλου, το οποίο αναπαριστά την ύπαρξη ή μη της κάθε φράσεως-κλειδί από την προκαθορισμένη λίστα, στο έγγραφο. Το τελευταίο στοιχείο του κάθε διανύσματος είναι ο κωδικός της κλάσης (η ετικέτα - label) του εγγράφου. Η επόμενη εικόνα (1) απεικονίζει το διάνυσμα και τη σχέση του με την προκαθορισμένη λίστα φράσεων-κλειδιών και με τη λίστα των διαθέσιμων κλάσεων (κατηγοριών).



Εικόνα 1. Διάνυσμα αναπαράστασης κατεταγμένων εγγράφων

Τα «ονομασθέντα» διανύσματα (labeled vectors) των εγγράφων συνιστούν ένα πίνακα από m+1 χαρακτηριστικά γνωρίσματα (ιδιότητες – attributes) στον οποίο μπορούν να εφαρμοσθούν αλγόριθμοι εξόρυξης γνώσης και να παραχθούν κανόνες κατάταξης (Classification Rules). Μία ιδέα που προτάθηκε πρόσφατα [11] και βασίζεται στα «ονομασθέντα» διανύσματα (labeled vectors) των εγγράφων παράγει αρχικά τις λίστες των αντιπροσωπευτικών φράσεων-κλειδιών (μία λίστα από αντιπροσωπευτικές φράσεις-κλειδιά για κάθε κλάση) και στη συνέχεια τις αξιοποιεί για να δημιουργήσει κανόνες που υπολογίζουν την ομοιότητα κάθε νέου εγγράφου με την κάθε κλάση CL_i. Η ομοιότητα του εγγράφου με μια κλάση CL_i υπολογίζεται ως ο αριθμός των στοιχείων της λίστας των αντιπροσωπευτικών φράσεων-κλειδιών της κλάσης τα οποία πράγματι εμφανίζονται στο νέο έγγραφο, κανονικοποιώντας (διαιρώντας) με το πλήθος στοιχείων της λίστας.

Ο επόμενος συμβολισμός υποδηλώνει τη λίστα των αντιπροσωπευτικών φράσεων-κλειδιών της κλάσεως CL_i:

$$RCL_i = \{kpCL_{i1}, kpCL_{i2}, ..., kpCL_{ir}\}$$

όπου $kpCL_{ij}$ είναι η *j*-(φράση-κλειδί) της λίστας των αντιπροσωπευτικών φράσεων-κλειδιών της κλάσης CL_i. Το μέτρο $|RCL_i|$ υποδηλώνει το πλήθος των αντιπροσωπευτικών φράσεων-κλειδιών στη κλάση CL_i.
Μια ιδέα για τη δημιουργία διαισθητικών / απλοϊκών (naïve) κανόνων που μετρούν την ομοιότητα ενός νέου, μη κατεταγμένου, εγγράφου με την κλάση CL_i ορίζεται στην επόμενη ισότητα:

$$S^{'}(D_{new}, CL_i) = \frac{count(exist(kpCL_{ij}, D_{new}))}{|RCL_i|}$$

όπου η λειτουργία (function) "exists" υποδηλώνει αν η αντίστοιχη φράση-κλειδί $kpCL_{ij}$ υπαρχει στο νέο έγγραφο και η λειτουργία "counts" υπολογίζει το πλήθος τους.

Ένα ενδιαφέρον χαρακτηριστικό της λίστας φράσεων-κλειδιών (Key-phrases Authority list) που επιλέγονται για το σύνολο των κλάσεων (άρα και για το σύνολο των εγγράφων του εκπαιδευτικού συνόλου), είναι ότι δημιουργείται με τέτοιο τρόπο ώστε να περιλαμβάνει φράσεις-κλειδιά οι οποίες είναι συχνές στα έγγραφα μόνο μίας ή, το πολύ, λίγων κλάσεων. Κατά συνέπεια, επιλέγοντας όλες τις φράσεις της Authority list που εμφανίζονται σε κάποια συγκεκριμένη κλάση, έστω την CL_i, έχουμε δημιουργήσει και τη λίστα των αντιπροσωπευτικών φράσεων-κλειδιών της κλάσης CL_i.

Μια άλλη προσέγγιση για την κατάταξη εγγράφων βασίζεται στην ομοιότητα μεταξύ προ-κατεταγμένων κειμένων (του «εκπαιδευτικού» συνόλου) και των νέων (μη-κατεταγμένων) εγγράφων (βλέπε [9]). Αυτή η μέθοδος εκμάθησης με βάση τα παρατηρηθέντα στιγμιότυπα **(**Instance based learning method) βασίζεται στην παραδοχή ότι παρόμοια έγγραφα πρέπει να κατατάσσονται στην ίδια κατηγορία (κλάση), δηλαδή πρέπει να μοιράζονται τον ίδιο κωδικό κατάταξης. Σε αυτή τη μέθοδο, η λίστα των φράσεων-κλειδιών αποτελεί ένα σύνολο από «υποσχόμενα» χαρακτηριστικά γνωρίσματα (attributes) τα οποία μπορούν να χρησιμοποιηθούν για να περιγράψουν και να διακρίνουν (discriminate) μεταξύ κατεταγμένων και νέων (μη-κατεταγμένων) εγγράφων. Αυτή η προσέγγιση εκμεταλεύεται τη γνώση από το πεδίο την Ανάκτησης Πληροφορίας με σκοπό να ορισθεί ένα μέτρο ομοιότητας μεταξύ των νέων εγγράφων και των εγγράφων του εκπαιδευτικού συνόλου.

Το γνωστό πρόβλημα της «ομοιότητας κειμένου με ερώτηση» ("Document versus Query similarity"), επίσης γνωστό ως «πλησιέστεροι γείτονες» ("nearest neighbours") αποτέλεσε αντικείμενο ερευνητικής δραστηριότητας για αρκετά χρόνια. Αρκετές συναρτήσεις, μέτρα και αλγόριθμοι ομοιότητας έχουν προταθεί. Ένα από αυτά (βλέπε [7, 13]) έχει προσαρμοσθεί με σκοπό να μετρά την ομοιότητα νέων (μη-κατεταγμένων) εγγράφων έναντι εγγράφων ενός εκπαιδευτικού συνόλου. Το προσαρμοσμένο μέτρο παρουσιάζεται στην επόμενη ισότητα:

$$S(D_i, D_{new}) = \frac{\sum_{j=1}^{m} q_j k_{ij}}{\sqrt{\sum_{j=1}^{m} q_j^2 \cdot \sum_{j=1}^{m} k_{ij}^2}} = \frac{\sum_{j=1}^{m} q_j k_{ij}}{\sqrt{\sum_{j=1}^{m} q_j^2} \cdot \sqrt{\sum_{j=1}^{m} k_{ij}^2}} = \frac{\sum_{j=1}^{m} q_j k_{ij}}{L_{D_{new}} \cdot L_{D_i}}$$

όπου *m* είναι ο αριθμός των φράσεων-κλειδιών που χρησιμοποιούνται στη συλλογή, k_{ij} είναι ίσο με 1 αν η φράση-κλειδί *j* υπάρχει στο έγγραφο D_i (του εκπαιδευτικού συνόλου), αλλιώς είναι ίσο με 0 και q_j είναι το βάρος (weight) της φράσης-κλειδί *j* στο νέο έγγραφο. Η επόμενη εξίσωση μπορεί να χρησιμοποιηθεί για τον υπολογισμό του όρου q_j:

$$q_j = \log_2\left(\frac{ClassCount}{ClassFreq_j}\right)$$

όπου η *ClassCount* περιέχει το πλήθος των κλάσεων του εκπαιδευτικού συνόλου και η $ClassFreq_j$ περιέχει τον αριθμό των κλάσεων που περιλαμβάνουν τη φράση-κλειδί *j*.

Η επιλογή των καταλληλότερων φράσεων-κλειδιών είναι μία θεμελιώδης πλευρά της εργασίας μας. Η έρευνα επηρεάστηκε από την έρευνα για υψίσυχνα σύνολα στοιχείων (frequent itemset) και τους αλγορίθμους για εύρεση υψίσυχνων συνόλων ή επεισοδίων (episodes) [14, 1, 15]. Υπάρχουν βέβαια ουσιαστικές διαφορές μεταξύ των δύο ερευνητικών περιοχών. Συγκεκριμένα, στην έρευνα μας ενδιαφερόμαστε για ακολουθίες από λέξεις (φράσεις-κλειδιά) που θα χρησιμοποιηθούν ως χαρακτηριστικά γνωρίσματα (features) για κανόνες κατάταξης και όχι για την εξαγωγή κανόνων συσχετισμού (extraction of association rules). Συνεπώς, οι φράσεις κλειδιά που θα προτείνουν οι αλγόριθμοι, που προκύπτουν από την έρευνα μας, πρέπει να είναι υψίσυχνες εντός των εγγράφων μίας ή το πολύ λίγων κλάσεων, αλλά θα πρέπει να μην είναι υψίσυχνες στις υπόλοιπες κλάσεις. Αντίθετα, στην περίπτωση των υψίσυχνων συνόλων στοιχείων ή στοιχειοσυνόλων, τα επιλεγμένα στοιχειοσύνολα θα πρέπει να είναι υψίσυχνα στο σύνολο των συναλαγών. Διευκρινίζουμε ότι, στην ορολογία των υψίσυχνων στοιχειοσυνόλων χρησιμοποιείται ο όρος συναλλαγές (transactions) και όχι έγγραφα.

Στους αλγορίθμους υψίσυχνων στοιχειοσυνόλων, οι συναλλαγές (transactions) είναι σύνολα από στοιχεία χωρίς καμία διάταξη μεταξύ των στοιχείων (για παράδειγμα μπορεί να είναι προϊόντα μέσα σε ένα καλάθι αγορών), ενώ στην περίπτωση που εμείς εξετάζουμε, οι λέξεις που συνθέτουν τις φράσεις-κλειδιά είναι πάντα διατεταγμένες εντός των εγγράφων.

Μία τρίτη διαφορά είναι ότι στην μελέτη των υψίσυχνων στοιχειοσυνόλων, τα υποψήφια στοιχειοσύνολα δεν υπόκεινται σε κανένα περιορισμό απόστασης μεταξύ των στοιχείων τους, ενώ αντίθετα οι λέξεις που συνθέτουν φράσεις-κλειδιά πρέπει να μη παραβιάζουν περιορισμούς απόστασης (δηλαδή, πρέπει να συνυπάρχουν σε συγκεκριμένου μεγέθους παραθύρου). Επιπλέον το μέγεθος του παραθύρου δεν είναι σταθερό αλλά εξαρτάται από το πλήθος των λέξεων που απαρτίζουν τη φράση-κλειδί. Για παράδειγμα το μέγεθος του παραθύρου για μια φράση-κλειδί δύο λέξεων μπορεί να έχει ορισθεί ότι είναι πέντε, ενώ το μέγεθος παραθύρου για φράση-κλειδί τριών λέξεων να έχει ορισθεί ότι είναι επτά.

Η πρώτη από τις προαναφερθέντες διαφορές αποτέλεσε το κίνητρο για να κατασκευάσουμε ένα νέο αλγόριθμο [10]. Η νέα, βελτιωμένη, μορφή του αλγόριθμου μας παρουσιάζεται στην επόμενη ενότητα. Οι άλλες δύο διαφορές μας οδήγησαν στην μορφοποίηση μιας νέας μεθόδου η οποία δημιουργεί φράσεις-κλειδιά μήκους χ-λέξεων από υψίσυχνες φράσεις-κλειδιά μήκους (χ-1)-λέξεων. Η μέθοδος αυτή παρουσιάζεται στην τρίτη ενότητα και χρησιμοποιείται ως ένα βήμα του αλγορίθμου που παρουσιάζουμε στην επόμενη (δεύτερη) ενότητα.

2. Επιλογή υποψήφιων φράσεων-κλειδιών (Selection of candidate key-phrases)

Οι παραπάνω περιγραφείσες προσεγγίσεις βασίζονται στην ύπαρξη λίστας φράσεων-κλειδιών ή Λίστας Καθιερωμένου Περιεχομένου (list of key-phrases or Authority List). Στην ενότητα αυτή συζητάται η κατασκευή της λίστας αυτής.

Ο Turney [19] συγκρίνει τέσσερα (4) συστήματα. Χρησιμοποιεί έναν ενδιαφέροντα συντελεστή F-measure για να συγκρίνει την αποτελεσματικότητα των μεθόδων. Το F-measure τεκμηριώνεται στο σύγγραμα του Rijsbergen [18] και βασίζεται στο precision και στο recall των μεθόδων. Στην πρώτη μέθοδο, για την οποία δίδονται αρκετές λεπτομέρειες, χρησιμοποιείται χαρακτηριστής μερών του λόγου (part of speech tagger) και αναζητώνται συγκεκριμένες συντακτικές μορφές για υποψήφιες φράσεις-κλειδιά (κυρίως ονοματικές φράσεις - noun phrases). Επιλέγονται οι N1 πιο συχνές απλές λέξεις (single words), οι N2 πιο συχνές δυάδες λέξεων (double words) και οι N3 πιο συχνές ακολουθίες τριων λέξεων (sequences of 3 words). Στη δεύτερη μέθοδο για την οποία δίνει επίσης λεπτομερή περιγραφή χρησιμοποιείται το "summarize feature" του προϊόντος Verity χωρίς όμως να γνωρίζουμε πως ακριβώς το Verity επιλέγει τις N προτάσεις που επιστρέφει. Το N προσδιορίζεται από το χρήστη.

Ο Γεωργαντόπουλος [5] παρουσιάζει μια παρόμοια μέθοδο με τον Turney (κάνει χαρακτηρισμό μερών του λόγου, συντακτική ανάλυση και ληματοποίηση) για να επιλέξει φράσεις. Ο Γεωργαντόπουλος επιλέγει όλες τις φράσεις που ικανοποιούν τις συντακτικές μορφές (patterns) χωρίς κανένα φιλτράρισμα. Επειδή δεν κάνει κανένα φιλτράρισμα (δεν επιλέγει τις N1, N2 και N3 πολυπληθέστερες φράσεις) είναι λογικό

να μην μπορεί να κάνει βελτιστοποίηση (των N1, N2 και N3) για να αυξήσει το F-measure.

Ο Mannila [15] βρίσκει συχνά επεισόδια σε μια 'telecommunication network alarm database'. Τα κυριότερα ενδιαφέροντα σημεία είναι: α) τα δεδομένα είναι μια και μόνο ακολουθία από γεγονότα (faults) και δεν ομαδοποιούνται σε ενότητες όπως είναι οι λέξεις και οι φράσεις που ομαδοποιούνται στα έγγραφα (κείμενα) μιας συλλογής εγγράφων (κειμένων), β) βασίζεται στην ιδέα ότι για κάθε συχνή ακολουθία όλες οι υποακολουθίες της θα είναι το ίδιο ή περισσότερο συχνές. Έτσι για να δημιουργήσει τις υποψήφιες ακολουθίες μήκους n+1 (C_{n+1}) βασίζεται στις συχνές ακολουθίες μήκους n (L_n). Με τον τρόπο αυτό μειώνει το χώρο αναζήτησης (search space), γ) τα γεγονότα μιας ακολουθίας δεν είναι κατ' ανάγκη το ένα μετά το άλλο αλλά μπορούν να μεσολαβούν και άλλα.

Η επιλογή των φράσεων (ή επεισοδίων) όταν βασίζεται στην υψηλή συχνότητα εμφανισής τους εκτιμούμε ότι αποτελεί αξιόπιστη μέθοδο που μπορεί να χρησιμοποιηθεί για την πρόβλεψη των λέξεων (γεγονότων) που ακολουθούν (π.χ. type ahead wizard). Έτσι στην περίπτωση που οι φράσεις προορίζονται για text abstraction / summarization εκτιμούμε ότι έχουμε μία ικανοποιητική μέθοδο. Αν όμως οι φράσεις προορίζονται για ευρετηρίαση / ανάκτηση (text indexing / retrieval) τότε οι πολύ υψίσυχνες φράσεις δεν βοηθούν στην διάκριση μεταξύ των κειμένων και καλό θα ήταν να μην χρησιμοποιούνται. Για παράδειγμα αν ο χρήστης υποβάλει μια ερώτηση που περιλαμβάνει φράση (keyphrase) που απαντάται στο 70% των κειμένων της συλλογής τότε η λίστα των αποτελεσμάτων που θα λάβει θα είναι τόσο μεγάλη που θα τον κουράσει και θα τον απογοητεύσει.

Για την περίπτωση που οι φράσεις (keyphrases) προορίζονται για ευρετηρίαση κειμένου (text indexing) και ανάκτηση (retrieval) είναι «αξιόπιστη» η επιλογή τους όταν βασίζεται σε μέτρα που προτιμούν φράσεις που εμφανίζονται σε λίγα κείμενα αλλά εντός των κειμένων αυτών εμφανίζονται πολλές φορές. Στην πιο απλή του μορφή ένα τέτοιο μέτρο είναι το εξής:

$$freq(P,D)\cdot\frac{N}{docfreq(P)}$$

Ένα τέτοιο μέτρο σε κανονικοποιημένη μορφή χρησιμοποιεί και το εργαλείο KEA [4, 20] για ένα από τα δύο χαρακτηριστικά γνωρίσματα (features) στα οποία βασίζεται για να χτίσει ένα μοντέλο πρόγνωσης (prediction model) με βάση ένα εκπαιδευτικό σύνολο εγγράφων. Το μέτρο του KEA είναι:

$$TF\cdot IDF=\frac{freq(P,D)}{size(D)}\cdot-\log_2\left(\frac{docfreq(P)}{N}\right)$$

Η δεύτερη συνιστώσα του μέτρου αυτού είναι 0 για μια φράση (keyphrase) που υπάρχει σε όλα τα έγγραφα, είναι 0,25 για μια φράση που υπάρχει στο 84% των εγγράφων, είναι 0,50 όταν υπάρχει στο 71%, είναι 1 όταν υπάρχει στο 50%, είναι 2 όταν υπάρχει στο 25%, είναι 3 όταν υπάρχει 12,5%, είναι 4 όταν υπάρχει στο ένα δέκατο-έκτο των εγγράφων, είναι 5 όταν υπάρχει στο ένα τριακοστό-δεύτερο των εγγράφων, …, είναι 10 όταν υπάρχει στο ένα χιλιοστό-εικοστό-τέταρτο των εγγράφων.

Στην περίπτωση που οι φράσεις προορίζονται για να χρησιμοποιηθούν ως χαρακτηριστικά γνωρίσματα (features) στα οποία θα βασίζεται η κατηγοριοποίηση / ταξινόμηση των κειμένων [8, 9], τότε οι υψίσυχνες φράσεις (frequent keyphrases) σε καμία περίπτωση δεν θα πρέπει να χρησιμοποιηθούν ως χαρακτηριστικά για την διάκριση των κειμένων.

Μπαίνει το ερώτημα αν θα μπορούσαν, οι φράσεις που λαμβάνουν μεγάλα βάρη από μέτρα σαν και αυτά που χρησιμοποιεί του ΚΕΑ, να επιλεγούν ως οι καταλληλότερες φράσεις-κλειδιά για text classification. Με μια πρώτη σκέψη θα μπορούσαμε να ισχυρισθούμε ότι τέτοια μέτρα δείχνουν προτίμηση (ενισχύουν) τις φράσεις που διακρίνουν / διαχωρίζουν τα κείμενα μιας συλλογής και συνεπώς οι φράσεις με τα μεγαλύτερα βάρη θα ήταν αξιόπιστα χαρακτηριστικά για text classification. Αν εμβαθύνουμε περισσότερο θα διαπιστώσουμε τα προβλήματα στα οποία μπορεί να οδηγηθεί μια μέθοδος κατάταξης εγγράφων (text classification) όταν η επιλογή των φράσεων γίνεται με μέτρα σαν του ΚΕΑ. Ο πρώτος κίνδυνος είναι να απορρίψουμε φράση (keyphrase) η οποία εμφανίζεται σε όλα τα έγγραφα μιας κλάσης και πουθενά αλλού στη συλλογή, όταν το πλήθος των εγγράφων αυτής της κλάσης είναι μεγάλο, εν αντιθέσει με το πλήθος εγγράφων άλλων κλάσεων. Ο δεύτερος κίνδυνος είναι να επιλέξουμε φράση (keyphrase) που εμφανίζεται σε ένα μικρό υποσύνολο των κειμένων μιας πολυπληθούς κλάσης και τα οποία διαπραγματεύονται ένα εξειδικευμένο υπόθεμα της κλάσης. Ο τρίτος κίνδυνος είναι να επιλέξουμε φράση (keyphrase) που εμφανίζεται συνολικά λίγες φορές στη συλλογή και είναι διάσπαρτη σε αρκετές κλάσεις της συλλογής. Επομένως (ένα πρώτο συμπέρασμα είναι) όταν επιλέγουμε φράσεις για text classification δεν πρέπει να βασιζόμαστε ούτε σε υψίσυχνες σε ολόκληρη τη συλλογή φράσεις ούτε σε μέτρα σαν του ΚΕΑ.

Σε αντίθεση με τις υψίσυχνες σε ολόκληρη τη συλλογή φράσεις, οι φράσεις που εμφανίζονται με υψηλή συχνότητα εντός μίας ή ελάχιστων κατηγοριών, ενός εκπαιδευτικού συνόλου, είναι ίσως τα καλύτερα χαρακτηριστικά (features) για την κατάταξη (classification) των κειμένων μιας συλλογής. Εκτιμούμε επίσης ότι η επιλογή των φράσεων-κλειδιών (keyphrases) από φράσεις που ικανοποιούν ορισμένες συντακτικές κατηγορίες επιβαρύνει το σύστημα, το οποίο σπαταλά πόρους σε μορφολογικούς, γραμματικούς και συντακτικούς αναλυτές (morphological, part of speech and syntactic analyzers) και λειτουργεί περιοριστικά για την επιλογή των καταλληλότερων φράσεων-κλειδιά.

Τα τελευταία δύο συμπεράσματα σε συνδυασμό με το δεύτερο ενδιαφέρον σημείο που παρατηρήσαμε στη δουλειά του Mannila [15] μας ώθησαν στην κατασκευή του αλγόριθμου που παρουσιάζεται στη συνέχεια. Ο αλγόριθμος αυτός αντιμετωπίζει το πρόβλημα επιλογής φράσεων-κλειδιών, όταν αυτές προορίζονται να χρησιμοποιηθούν για κατάταξη κειμένων. Ο αλγόριθμος αυτός δέχεται ως δεδομένα μια συλλογή (collection) εγγράφων κατηγοριοποιημένων σε κλάσεις (classes), το μέγεθος ενός παραθύρου (window size / width) και ένα όριο συχνότητας (frequency threshold) και εντοπίζει όλες τις φράσεις κλειδιά (keyphrases) που συναντώνται αρκετά συχνά σε μία ή λίγες κλάσεις και δεν συναντώνται αρκετά συχνά στις υπόλοιπες κλάσεις. Ο αλγόριθμος έχει δύο φάσεις: Μία για την κατασκευή νέων υποψήφιων φράσεων (new candidate key-phrases) και μία για τον υπολογισμό του πόσο συχνά αυτές οι φράσεις συναντώνται στην εξεταζόμενη κλάση της συλλογής. Η ιδέα της κατασκευής υποψήφιων φράσεων από μικρότερες αξιοποιείται στα βήματα 7 και 10 του αλγορίθμου. Η ιδέα αυτή έχει χρησιμοποιηθεί με επιτυχία και για την ανακάλυψη κανόνων συνάφειας (discovery of association rules [1, 14]) και συναντάται και σε άλλου τύπου εφαρμογές [15, 16].

Αλγόριθμος

1 Για κάθε κλάση (CL_i) του εκπαιδευτικού συνόλου κάνε
2 Για κάθε έγγραφο της κλάσης (DCL_i) κάνε
3 εύρεση ριζών λέξεων
4 αφαίρεση τετριμένων λέξεων
5 Τέλος {Για κάθε έγγραφο της κλάσης}
6 Επέλεξε τις πλέον υψίσυχνες ρίζες λέξεων της κλάσης (P_0 παράμετρος)
7 Δημιούργησε τις υποψήφιες φράσεις δύο λέξεων (C_2) από τις υψίσυχνες ρίζες λέξεων (L_1)
8 Επέλεξε τις πλέον υψίσυχνες φράσεις δύο λέξεων (L_2) (W_1 και P_1 παράμετροι)
9 Για x να παίρνει τιμές από 3 μέχρι *mpc* κάνε
10 Δημιούργησε τις υποψήφιες φράσεις x – λέξεων (C_x) από τις υψίσυχνες φράσεις (x-1) – λέξεων (L_{x-1})
11 Επέλεξε τις πλέον υψίσυχνες φράσεις x – λέξεων (L_x) (P_{x-1} και W_{x-1} παράμετροι)
12 Τέλος {Για x να παίρνει τιμές από 3 μέχρι *mpc* κάνε}
13 Συνέθεσε συνολική λίστα, συνενώνοντας τις L_x (για x=2,3,…,*mpc*). Η σύνθεση, ορίζει τη λίστα υψίσυχνων φράσεων της κλάσης (FCL_i)
14 Τέλος {Για κάθε κλάση του εκπαιδευτικού συνόλου}
15 Συνένωσε τις λίστες υψίσυχνων φράσεων από όλες τις κλάσεις του εκπαιδευτικού συνόλου
16 Αφαίρεσε εκείνες τις φράσεις οι οποίες εμφανίζονται σε πολλές κλάσεις (P_t παράμετρος). Οι εναπομείναντες υψίσυχνες φράσεις

συνιστούν το σύνολο φράσεων-κλειδιά ή *Authority list*
17 Δημιούργησε το Λεξικό Όρων ή *Terms*. Πρόκειται για τη λίστα με τις ρίζες λέξεων οι οποίες εμφανίζονται στις φράσεις κλειδιά της *Authority list*.

Παράμετροι:

mpc μέγιστος αριθμός λέξεων φράσης-κλειδί,
P_0 ποσοστό κειμένων της κλάσης τα οποία πρέπει να περιέχουν τη ρίζα,
W_1 πλάτος παραθύρου εντός του οποίου εμφανίζονται οι φράσεις 2-λέξεων,
P_1 ποσοστό των κειμένων της κλασης στα οποία πρέπει να περιέχεται η φράση 2-λέξεων,
W_2 πλάτος παραθύρου εντός του οποίου εμφανίζονται οι φράσεις 3-λέξεων,
P_2 ποσοστό των κειμένων της κλασης στα οποία πρέπει να περιέχεται η φράση 3-λέξεων,
…
W_{mpc-1} πλάτος παραθύρου εντός του οποίου εμφανίζονται οι φράσεις mpc-λέξεων,
P_{mpc-1} ποσοστό των κειμένων της κλασης στα οποία πρέπει να περιέχεται η φράση mpc-λέξεων,
Pt Μεγιστο ποσοστό κλάσεων που μπορούν να περιέχουν μια αποδεκτή φράση-κλειδί.

Στη φυσική γλώσσα εμφανίζεται το φαινόμενο των γραμματικών κλίσεων (grammatical inflections) και σαν συνέπεια του φαινομένου η εμφάνιση λέξεων σε διαφορετικές μορφές. Η ανάγκη για στατιστική ανάλυση των κειμένων σχετίζεται με μια ποικιλία από διεργασίες, όπως η επιμέτρηση της συχνότητας των λέξεων και η ανάκτηση επαναλαμβανόμενων ακολουθιών λέξεων. Η εύρεση των ριζών των λέξεων (Stemming) και η αφαίρεση των τετριμμένων λέξεων (stop-words' removal) είναι βασικές διεργασίες στο πεδίο της Ανάκτησης Πληροφορίας (Information Retrieval) [12]. Η διεργασία εύρεσης ριζών των λέξεων ή η αφαίρεση των επιθεμάτων [17, 6] χρησιμοποιείται για να κανονικοποιήσει τις εμφανίσεις των λέξεων. Η διεργασία αφαίρεσης τετριμμένων λέξεων επιδιώκει την αφαίρεση των λέξεων με μικρή συνεισφορά στην πληροφορία που επικοινωνείται με τις προτάσεις. Έτσι, τα άρθρα, οι προθέσεις, κλπ, αντιμετωπίζονται ως «θόρυβος» και αφαιρούνται.

Ο αλγόριθμος βασίζεται σε δύο κύρια επαναληπτικά σχήματα. Το εξωτερικό επαναληπτικό σχήμα (βήματα 1 έως 14) εξετάζει διαδοχικά όλες τις κλάσεις, ενώ το εσωτερικό επαναληπτικό σχήμα (βήματα 9 μέχρι 12) εξετάζει διαδοχικά κάθε πιθανό μήκος φράσης-κλειδιού για μήκος 3 λέξεων και άνω. Το εσωτερικό επαναληπτικό σχήμα σε συνδυασμό με τα βήματα αρχικοποίησης του (βήματα 2 μέχρι 8) δημιουργούν τελικά τη λίστα των υψίσυχνων φράσεων της εξεταζόμενης κλάσης. Το εσωτερικό επαναληπτικό σχήμα, σε κάθε του βήμα, εκτελεί τη φάση της δόμησης (βήμα 10) και τη φάση της αναγνώρισης (βήμα 11). Στη φάση δόμησης κάθε βήματος

επανάληψης (έστω x=j), μια συλλογή C_j από νέες υποψήφιες φράσεις-κλειδιά δημιουργείται, χρησιμοποιώντας την πληροφορία που είναι διαθέσιμη για τις κατά μία λέξη μικρότερες υψίσυχνες φράσεις-κλειδιά. Στη συνέχεια, οι υποψήφιες φράσεις-κλειδιά αναζητούνται στα έγγραφα της εξεταζόμενης κλάσης και τελικά υπολογίζονται οι συχνότητες εμφάνισης τους. Με βάση τις συχνότητες εμφάνισης δημιουργείται η συλλογή L_j των υψίσυχνων φράσεων-κλειδιά η οποία είναι ένα υποσύνολο της συλλογής των υποψηφίων φράσεων-κλειδιά C_j. Στην επόμενη επανάληψη (όπου x=j+1), οι υποψήφιες φράσεις-κλειδιά του C_{j+1} δημιουργούνται με βάση τις υψίσυχνες φράσεις-κλειδιά του L_j. Τα βήματα αρχικοποίησης του εσωτερικού επαναληπτικού σχήματος μπορούν να ομαδοποιηθούν σε δύο τάξεις. Η τάξη φράσεων-κλειδιών μίας λέξεως αρχικά δημιουργεί το C_1 (βήματα 2 μέχρι 5) και στη συνέχεια το L_1 (βήμα 6) και η τάξη φράσεων-κλειδιών δύο λέξεων που δημιουργεί το C_2 (βήμα 7) και το to L_2 (βήμα 8). Μετά την ολοκλήρωση του εσωτερικού επαναληπτικού σχήματος, δημιουργείται η λίστα των υψίσυχνων φράσεων-κλειδιών της εξεταζόμενης κλάσης (βήμα 13). Μετά την ολοκλήρωση και του εξωτερικού επαναληπτικού σχήματος δημιουργείται το σύνολο των φράσεων-κλειδιών που είναι κατάλληλες να χρησιμοποιηθούν ως χαρακτηριστικά γνωρίσματα (features) για την αυτόματη κατάταξη εγγράφων (βήματα 15 και 16).

Το σύνολο των υποψηφίων φράσεων-κλειδιών 2-λέξεων (C_2) πρέπει να περιέχει φράσεις για τις οποίες ισχύει ότι κάθε μια από αυτές αποτελείται από διαφορετικές ρίζες λέξεων. Για να πραγματοποιηθεί αυτό, στο βήμα 7, δημιουργείται το καρτεσιανό γινόμενο των υψίσυχνων ριζών λέξεων και στη συνέχεια αφαιρούνται οι πλειάδες (ζεύγη ριζών λέξεων) που έχουν δύο ίδιες ρίζες λέξεων. Στην επόμενη ενότητα εξετάζουμε το βήμα 10, το οποίο δημιουργεί υποψήφιες φράσεις-κλειδιά μήκους x – λέξεων (C_x) από υψίσυχνες φράσεις-κλειδιά μήκους (x-1) – λέξεων (L_{x-1}), για 3≤x≤mpc.

3. Κατασκευάζοντας υποψήφιες φράσεις-κλειδιά μήκους x-λέξεων από υψίσυχνες φράσεις-κλειδιά μήκους (x-1)-λέξεων.

Η υλοποίηση του βήματος 10, του αλγορίθμου της προηγούμενης ενότητας, παρουσιάζεται και συζητείται στη συνέχεια.

Μια απλοϊκή (brute force) κατασκευή του C_x μπορεί να βασισθεί στον υπολογισμό του Καρτεσιανού γινομένου: $L_1 \cdot \ldots \cdot L_1$ (x-φορές). Μια τέτοια διαδικασία είναι χρονοβόρα γιατί απαιτεί τον έλεγχο εμφάνισης (και τον υπολογισμό συχνοτήτων εμφάνισης) ενός πολύ μεγάλου αριθμού από υποψήφιες φράσεις. Για το λόγο αυτό, μια «ευφυής» βελτίωση κρίθηκε απαραίτητη [10]. Η αποτελεσματικότητα του αλγορίθμου υποδηλώνεται από την επόμενη σκέψη: "Ο χώρος αναζήτησης μπορεί να μειωθεί δραστικά αν οι μεγαλύτερες φράσεις-κλειδιά δημιουργηθούν από άλλες μικρότερες. Με άλλα λόγια, είναι αρκετό να εξετάσουμε φράσεις κλειδιά των οποίων οι υπό-φράσεις είναι υψίσυχνες".

Αυτή η σκέψη βασίζεται και στην ερευνητική εργασία στον τομέα των αλγορίθμων εύρεσης υψίσυχνων στοιχειοσυνόλων [14, 1, 15] η οποία χαρακτηρίζεται από δύο διακεκριμένες φάσεις. Η φάση *γένεσης (generation phase)* συνδυάζει ζευγάρια από υψίσυχνα στοιχειοσύνολα μεγέθους k στοιχείων (ζευγάρια του L_k) τα οποία έχουν $k-1$ κοινά στοιχεία προκειμένου να παράγει νέα υποψήφια στοιχειοσύνολα με $k+1$ στοιχεία (υποψήφια στοιχεία του C_{k+1}). Η φάση *κλαδέματος* (ελάττωσης, απαλοιφής - *prune phase*) απορίπτει υποψήφια στοιχειοσύνολα $k+1$ στοιχείων (υποψήφια στοιχεία του C_{k+1}) τα οποία περιέχουν υποσύνολα μεγέθους k τα οποία δεν είναι υψίσυχνα (δεν περιέχονται στο L_k).

Όπως έχουμε ήδη αναφέρει τα υψίσυχνα στοιχειοσύνολα (frequent itemsets) είναι σύνολα στοιχείων χωρίς καμία ταξινόμηση μεταξύ των επιμέρους στοιχείων τους. Συνεπώς, αν υποθέσουμε ότι το L_2 περιέχει τα στοιχειοσύνολα AB, AC, BC και BD, τότε η φάση γένεσης (σύμφωνα με το [14]) θα προτείνει ως υποψήφια για το C_3 τα στοιχειοσύνολα ABC, ABD και BCD. Η φάση «κλαδέματος» θα απορίψει το στοιχειοσύνολο ABD επειδή το υποσύνολό του AD δεν είναι μέλος του L_2 και θα απορίψει επίσης το BCD επειδή το υποσύνολό του CD δεν είναι μέλος του L_2. Στο παράδειγμα αυτό η διάταξη των στοιχείων (A, B, C και D) εντός των στοιχειοσυνόλων του L_2 είναι χωρίς καμία αξία και η παρουσίαση (διατήρηση) τους σε λεξικογραφική διάταξη γίνεται μόνο για τεχνικούς λόγους.

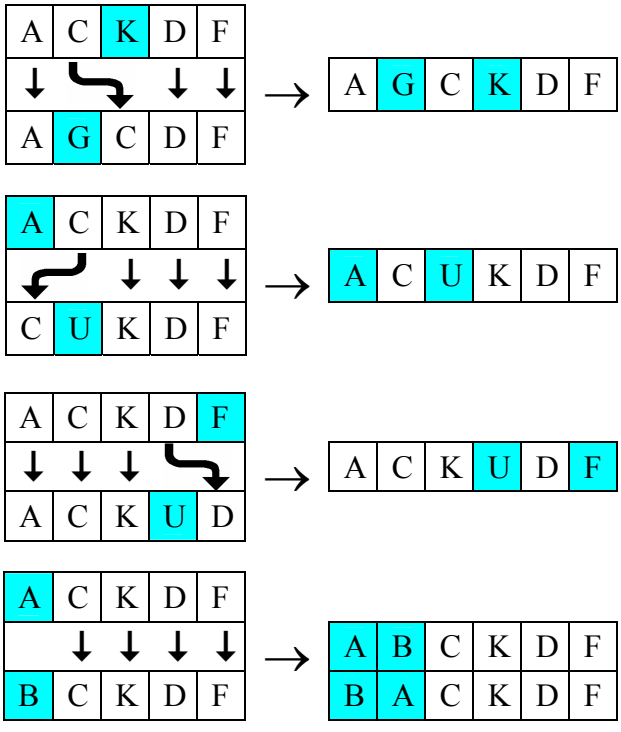
Αντιθέτως, στην περίπτωση των υψίσυχνων φράσεων-κλειδιά η διάταξη των λέξεων είναι σημαντικό στοιχείο. Έτσι, αν χρησιμοποιήσουμε το ίδιο παράδειγμα και υποθέσουμε ότι τα γραμματα A, B, C και D αναπαριστούν λέξεις, τότε η φάση γένεσης (στο δικό μας αλγόριθμο) θα πρότεινε τις ακόλουθες, υποψήφιες για το C_3, φράσεις-κλειδιά: ABC, ACB, ABD, BAC, BCD και BDC. Στον αλγόριθμο μας δεν υπάρχει η ανάγκη της φάσης «κλαδέματος». Η αιτία στην οποία βασίζεται η μή ύπαρξη ανάγκης «κλαδέματος» είναι η τρίτη διαφορά που αναφέρθηκε στην ενότητα 1. Για να γίνει περισσότερο αντιληπτό, μπορούμε να επεκτείνουμε το προηγούμενο παράδειγμα μας και να υποθέσουμε ότι: το μέγεθος του παραθύρου λέξεων εντός του οποίου αναζητούνται οι φράσεις-κλειδιά 2-λέξεων είναι 5, είναι αντίστοιχα 7 για φράσεις-κλειδιά 3-λέξεων και το κατώφλι για να δεχθούμε ότι μια ακολουθία λέξεων είναι υψίσυχνη φράση-κλειδί είναι 3 εμφανίσεις (σε 3 έγγραφα). Ας υποθέσουμε επίσης ότι τα δεδομένα μας (test base) είναι τα ακόλουθα τέσσερα κείμενα:

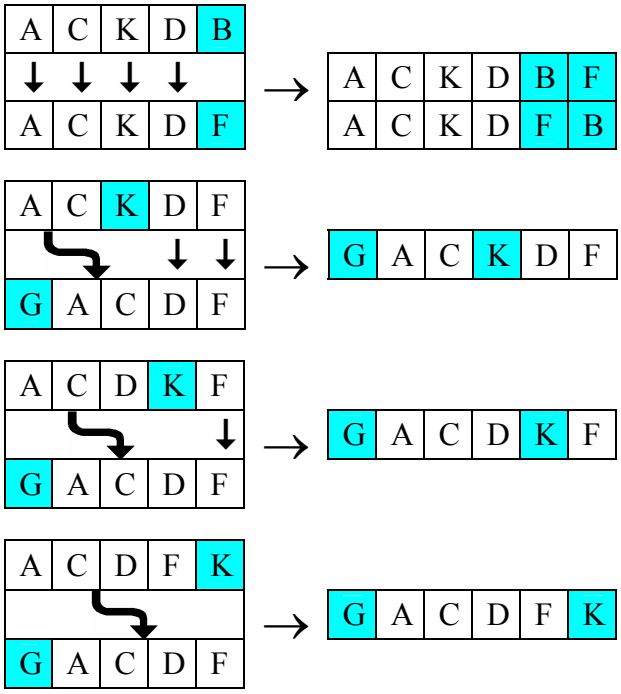
… A ? ? B ? D …
… A ? ? ? B ? D …
… A ? ? ? B D …
… A ? B ? D …

όπου ? αναπαριστά μια απλή λέξη, η οποία δεν είναι καμία από τις A, B ή D,
και … αναπαριστά μια ή περισσότερες λέξεις εκ των οποίων καμία δεν είναι A, B ή D.

Αν ο αλγόριθμος μας χρησιμοποιούσε τη φάση «κλαδέματος» τότε η παραγωγή ABD θα έπρεπε να απορριφθεί, καθώς το AD δεν περιλαμβάνεται στο L_2. Αυτό θα αποτελούσε λάθος, καθώς η παραγωγή ABD εμφανίζεται σε τέσσερα κείμενα (δηλαδή έχει συχνότητα μεγαλύτερη από το κατώφλι αποδοχής) όταν το μέγεθος του παραθύρου είναι 7. Συνεπώς, στην περίπτωση μας, όπου το μέγεθος του παραθύρου ποικίλει ανάλογα με την τάξη των φράσεων-κλειδιών (αριθμός λέξεων που τις συνιστούν), είναι δυνατό μία υψίσυχνη φράση-κλειδί να έχει μη συχνές υπό-φράσεις-κλειδιά. Συνεπώς, δεν υπάρχει χώρος για φάση «κλαδέματος» σε ένα τέτοιο αλγόριθμο και η μέθοδος για τη δημιουργία φράσεων-κλειδιά μήκους x-λέξεων από υψίσυχνες φράσεις κλειδιά μήκους (x-1)-λέξεων πρέπει να περιλαμβάνει μόνο τη φάση γένεσης. Σε αυτή τη φάση εστιάζουμε στη συνέχεια της παρούσας ενότητας.

Στα επόμενα παραδείγματα (εικόνα 2) παρουσιάζουμε με γραφικό τρόπο τη γένεση υποψηφίων φράσεων-κλειδιών έκτης τάξης (x=6) από υψίσυχνες φράσεις κλειδιά πέμπτης τάξης. Σε όλα τα παραδείγματα συνδυάζουμε ζευγάρια του L_5 που έχουν x-2 (=4) κοινά στοιχεία. Σε κάθε μία φράση-κλειδί του κάθε εξεταζόμενου ζευγαριού υπάρχει και από ένα αταίριαστο στοιχείο. Τα αταίριαστα στοιχεία εμφανίζονται σε γκρίζο φόντο. Το ταίριασμα των στοιχείων υποδηλώνεται με βέλη, με φορά από πάνω προς τα κάτω. Τα οριζόντια βέλη υποδηλώνουν την παραγωγή.





Εικόνα 2. Γένεση στοιχείων του C_6

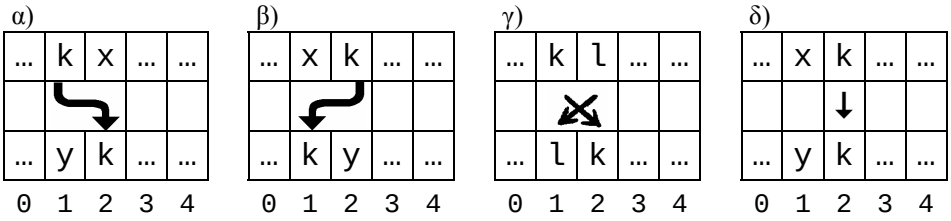
Από τα προηγούμενα παραδείγματα (που δημιουργούν στοιχεία του C_6 από στοιχεία του L_5) παρατηρούμε ότι όταν δύο φράσεις (key-phrases) ταιριάζουν σε πλήθος ριζών λέξεων ίσο με το δείκτη (τάξη) του C μείον δύο (x-2, 4) τότε όλα τα ταιριάσματα έχουν μία το πολύ θέση διαφορά (επόμενη, προηγούμενη ή ίδια θέση). Αντίθετα οι ρίζες (stem) που δεν ταιριάζουν (γκρι φόντο) μπορεί να βρίσκονται σε οποιαδήποτε απόσταση (από 0 μέχρι x-2). Επίσης παρατηρούμε ότι αν εντοπίσουμε τις θέσεις των δύο ριζών (stem) που δεν ταιριάζουν (μία ρίζα από κάθε φράση που συνδυάζουμε) τότε μπορούμε εύκολα να κατασκευάσουμε τη νέα υποψήφια φράση-κλειδί (στοιχείο του C_x).

Για να βρούμε αν ταιριάζουν x-2 ρίζες των φράσεων που συνδυάζουμε, κάνουμε ένα επαναληπτικό βρόχο (while), ο οποίος χρησιμοποιεί δύο δείκτες (i1 και i2, ένα για την τρέχουσα ρίζα της κάθε φράσης) και ολοκληρώνεται όταν έστω ένας δείκτης ξεπεράσει το άνω όριο του. Σε κάθε βήμα της επανάληψης εξετάζεται το ζεύγος ριζών λέξεων που υποδεικνύεται από τους δείκτες. Αν οι συγκρινόμενες ρίζες είναι ίδιες αυξάνονται αμφότεροι οι δείκτες και ο μετρητής των ταιριασμάτων.

Αν οι συγκρινόμενες ρίζες δεν είναι ίδιες τότε οι δείκτες μπορεί να είναι ίσοι ή να διαφέρουν κατά ένα. (Το όλο επαναληπτικό σχήμα είναι τέτοιο που δεν υπάρχει περίπτωση οι δείκτες να διαφέρουν περισσότερο από ένα.)

Αν οι συγκρινόμενες ρίζες (stems) δεν είναι ίδιες και οι δείκτες διαφέρουν κατά ένα τότε αποθηκεύουμε το μικρότερο δείκτη ως θέση ρίζας (της αντίστοιχης φράσης) που δεν ταιριάζει και αυξάνουμε το μικρότερο δείκτη κατά ένα. Γεννιέται το ερώτημα γιατί να μην αποθηκεύουμε το μεγαλύτερο δείκτη ως θέση ρίζας που δεν ταιριάζει (στην άλλη φράση) και να αυξάνουμε το μεγαλύτερο δείκτη κατά ένα. Αυτή η ενέργεια θα σήμαινε ότι στο επόμενο βήμα τις επανάληψης θα ελέγχαμε το ταίριασμα δυο ριζών με διαφορά θέσεων δύο (2) και συνεπώς θα παραβιαζόταν η παρατήρηση ότι όλα τα ταιριάσματα είναι με μία το πολύ θέση διαφορά προκειμένου να έχουμε x-2 ταιριάσματα. Συνεπώς αν πράγματι οι δύο συγκρινόμενες φράσεις ταιριάζουν σε x-2 ρίζες τότε ο μόνος τρόπος να βρεθεί αυτό το ταίριασμα είναι με τον τρόπο που επιλέξαμε (αύξηση του μικρότερου δείκτη κατά ένα).

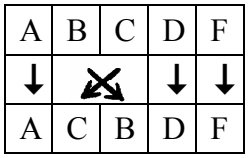
Όταν οι συγκρινόμενες ρίζες δεν είναι ίδιες και οι δείκτες είναι ίσοι (έστω i1=i2=1) τότε τα επόμενα ταιριάσματα ριζών (χωρίς να παραβιάζεται ο κανόνας μιας το πολύ θέσης διαφορά) μπορεί να είναι (εικόνα 3):



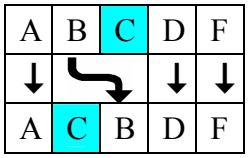
Εικόνα 3. Δυνατότητες όταν οι ρίζες λέξεων με ίσους δείκτες δεν ταιριάζουν

Ο αλγόριθμος σε αυτή την περίπτωση δοκιμάζει το (α) και αν επαληθεύεται, αποθηκεύει το δείκτη της δεύτερης φράσης ως θέση ρίζας (της δεύτερης φράσης) που δεν ταιριάζει και αυξάνει το δείκτη της δεύτερης φράσης κατά ένα (1). Αν δεν επαληθεύεται το (α) τότε δοκιμάζει το (β). Αν επαληθεύεται το (β) κάνει τα ανάλογα. Δυστυχώς η υλοποίησή μας δεν επιτρέπει προς το παρόν backtracking και αντιμετωπίζει την (γ) περίπτωση σαν να ήταν (α). Έτσι χάνονται κάποιες πιθανές φράσεις-κλειδιά του C_x (εικόνα 4).

Για παράδειγμα αν το (γ) ήταν:

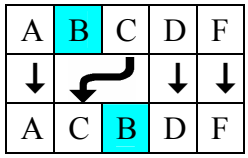


θα θεωρήσει

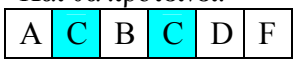


Ενώ

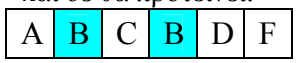
δεν θα θεωρήσει



Και θα προτείνει:



και δε θα προτείνει:



Εικόνα 4. Δυνητικές φράσεις του C_x που χάνονται

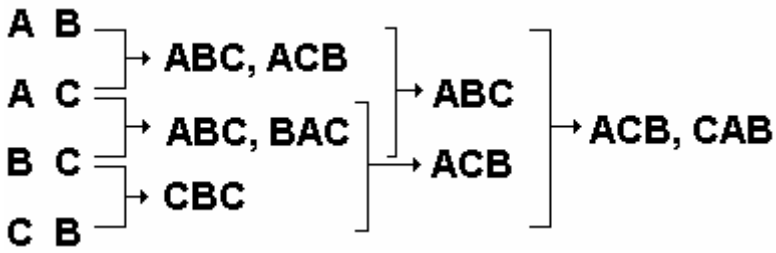
Αν δεν επαληθεύεται ούτε το (α), ούτε το (β) τότε αποθηκεύει αμφότερες τις θέσεις ως θέσεις ριζών (των αντίστοιχων φράσεων) που δεν ταιριάζουν και αυξάνει αμφότερους τους δείκτες κατά ένα. Η ενέργεια αυτή στην ουσία αναβάλει για το επόμενο βήμα της επανάληψης τον έλεγχο του ταιριάσματος των ριζών - stem στις νέες θέσεις που δείχνουν οι δείκτες.

Όπως προαναφέραμε το επαναληπτικό σχήμα τερματίζεται όταν έστω ένας δείκτης ξεπεράσει το όριο του. Όμως αν σκεφθούμε καλύτερα θα δούμε ότι υπάρχει και άλλη συνθήκη εξόδου. Για παράδειγμα όταν ελέγχουμε το ταίριασμα της $3^{ης}$ ρίζας από την πρώτη φράση (i1=2) με την 3η ρίζα από τη δεύτερη φράση (i2=2) τότε αν τα προηγούμενα ταιριάσματα είναι τουλάχιστον ένα (matches >= 1) και επειδή μπορεί και οι τρέχουσες ρίζες να ταιριάζουν πρέπει να συνεχίσουμε. Όταν ελέγχουμε το ταίριασμα της $3^{ης}$ ρίζας από την πρώτη φράση (i1=2) με την $4^{η}$ ρίζα από τη δεύτερη φράση (i2=3) τότε αν τα προηγούμενα ταιριάσματα είναι τουλάχιστον δύο (matches >= 2) και επειδή μπορεί και οι τρέχουσες ρίζες - stem να ταιριάζουν πρέπει να συνεχίσουμε. Άρα προσθέτουμε τη συνθήκη:

(matches >= max(i1,i2)-1)

Μετά την ολοκλήρωση του επαναληπτικού σχήματος (while), ελέγχουμε και αν τα matches είναι x-2 τότε εξετάζουμε τις θέσεις των ριζών που δεν ταιριάζουν. Αν οι θέσεις είναι ίδιες κατασκευάζουμε δύο νέες υποψήφιες φράσεις (στοιχεία του C_x), διαφορετικά κατασκευάζουμε μια νέα υποψήφια φράση.

Η συνδυαστική λειτουργία του αλγορίθμου (παρότι εξετάζει κάθε φράση του L_{x-1} μόνο με τις φράσεις που έπονται) μπορεί να οδηγήσει στην πολλαπλή παραγωγή της ιδιας υποψήφιας φράσης (στοιχείο του C_x). Για παράδειγμα, έστω ότι το L_2 αποτελείται από: {ΑΒ, ΑΓ, ΒΓ, ΓΒ}, τότε μπορεί να παραχθούν τα (εικόνα 5):



Εικόνα 5. Πολλαπλή παραγωγή των ιδίων υποψηφίων φράσεων

Δηλαδή παράγει: ΑΒΓ 3 φορές, ΑΓΒ 3 φορές, ΒΑΓ 1 φορά, ΓΑΒ 1 φορά, ΓΒΓ 1 φορά. Για το λόγο αυτό το C_x πρέπει να είναι ταξινομημένο και να ελέγχεται κάθε παραγωγή πριν μπει στο C_x.

Για τη διατήρηση της λίστας C_x σε διατεταγμένη μορφή απαιτείται η διάσχιση της μέχρι να βρεθεί κόμβος (key-phrase) μεγαλύτερος ή ίσος με την προς εισαγωγή φράση. Αυτή η διάσχιση απαιτεί πολλές συγκρίσεις string και συνεπώς έχει μεγάλη επιβάρυνση. Για να μειωθεί η επιβάρυνση από τις συγκρίσεις θα μπορούσαν οι ρίζες από string να αντικατασταθούν από τον αύξοντα αριθμό τους στη λίστα των πιο υψίσυχνων ριζών της κλάσης (L_1).

4. Συμπεράσματα

Στην εργασία αυτή παρουσιάστηκαν συνοπτικά δύο κύριες ερευνητικές κατευθύνσεις για την αυτόματη κατηγοριοποίηση εγγράφων (automatic classification of documents) βασιζόμενες στην ανάκτηση πληροφορίας και την ανακάλυψη γνώσης σε βάσεις δεδομένων (based on information retrieval and knowledge discovery in databases). Και οι δύο προσεγγίσεις βασίζονται στη χρήση φράσεων κλειδιών από ελεγχόμενο λεξιλόγιο (use of key-phrases from a controlled list). Επειδή η κατασκευή λίστας φράσεων κλειδιών είναι ένα σημαντικό και χρήσιμο αντικείμενο έρευνας για τον τομέα της κατηγοριοποίησης εγγράφων (domain of document classification) παρουσιάσαμε έναν αλγόριθμο που βασίζεται στην ιδέα ότι οι κατάλληλες φράσεις κλειδιά (the appropriate key-phrases for text classification) είναι εκείνες που είναι υψίσυχνες (frequent) στα έγγραφα μίας μόνο ή γενικά λίγων κλάσεων του εκπαιδευτικού συνόλου (training set). Ο αλγόριθμός μας ελαττώνει το χώρο αναζήτησης (search space) κατασκευάζοντας μεγαλύτερες φράσεις κλειδιά (key-phrases) από μικρότερες.

Ευχαριστίες

Η έρευνα αυτή συγχρηματοδοτήθηκε κατά 75% από την Ευρωπαϊκή Ένωση και κατά 25% από το Ελληνικό Δημόσιο στο πλαίσιο του ΕΠΕΑΕΚ - Αρχιμήδης.

Αναφορές

1. Agrawal Rakesh and Srikant Ramakrishnan, Fast Algorithms for Mining Association Rules, in Proc. of the 20th International Conf. On Very Large Databases, Santiago, Chile, September 1994.
2. Ahonen H., Heinonen O., Klemettinen M. and Verkamo A. I., Mining in the phrasal frontier, in Proc. Principles of Knowledge Discovery in Databases Conference, Trondheim, Norway, June 1997. Lecture Notes in Computer Science, Springer Verlag, 1997.
3. Veal D. C., Techniques of Document Management: A review of Text Retrieval and related technologies, Journal of Documentation, vol. 57, 2001, pp. 192-217.
4. Frank E., Paynter G. W., Witten I. H., Gutwin C. and Nevill-Manning C. G., Domain-Specific Keyphrase Extraction, in Proc. International Joint Conference of Artificial Intelligence, 1999.
5. Georgantopoulos Byron and Piperidis Stelios, Automatic Term Extraction Based on Pattern Grammars, LogoNavigation, Issue 5, May 1999.
6. Kalamboukis T., Suffix stripping in Greek, Program, vol. 29, 1995, pp. 313-321.
7. Karanikolas N. N. and Skourlas C., Computer Assisted Information Resources Navigation, Medical Informatics & the Internet in Medicine, vol. 25, 2000, pp. 133-146.
8. Karanikolas N. N. and Skourlas C., Automatic Diagnosis Classification of patient discharge letters in MIE'2002: XVIIth International Congress of the European Federation for Medical Informatics, Budapest, Hungary, August, 2002. IOS Press, ISBN: 1-58603-279-8, ISSN: 0926-9630.
9. Karanikolas N. N., Skourlas C., Christopoulou A. and Alevizos T., Medical Text Classification based on Text Retrieval techniques, in Proc. MEDINF 2003: 1st International Conference on Medical Informatics & Engineering, Craiova, Romania, October 2003.
10. Karanikolas N. N. and Skourlas C., Key-Phrase Extraction for Classification in MEDICON and HEALTH TELEMATICS 2004: X Mediterranean Conference on Medical and Biological Engineering, Ischia, Italy, July 31 - August 5, 2004.
11. Karanikolas N. N. and Skourlas C., Naive Rule Induction for Text Classification based on Key-Phrases in Proc. of the 6TH International Conference on Data Mining, Text Mining and their Business Applications, Skiathos, Greece, May 2005, pp. 175-182. WIT Transactions on Information and Communication Technologies, vol. 35, Data Mining VI: Data Mining, Text Mining and their Business Applications, WIT Press, ISBN 1-84564-017-9, ISSN 1764-4463.

12. Kowalski G., Information Retrieval Systems. Theory and Implementation, 1st edn, Kluwer Academic Publishers, Printed in the USA, 1997, ISBN 0-7923-9899-8.
13. Lucarella D., A document retrieval system based on nearest neighbour searching, Journal of Information Science, vol. 14, 1988, pp. 25-33.
14. Mannila Heikki, Toivonen Hannu and Verkamo A. Inkeri, Efficient algorithms for discovering association rules, in Proc. KDD-94: AAAI Workshop on Knowledge Discovery in Databases, Seattle, Washington, July 1994.
15. Mannila Heikki, Toivonen Hannu and Verkamo A. Inkeri, Discovering frequent episodes in sequences, in Proc. KDD-95: First International Conference on Knowledge Discovery and Data Mining, Montreal, Canada, August 1995.
16. Mannila Heikki and Toivonen Hannu, Discovering generalized episodes using minimal occurrences, in Proc. KDD-96: Second International Conference on Knowledge Discovery and Data Mining, Portland, Oregon, August, 1996, AAAI Press.
17. Porter M., An algorithm for suffix stripping, Program, vol. 14, 1980, pp. 130-137.
18. Van Rijsbergen C. J., Information Retrieval, 2nd edn, Butterworths, London, UK, 1979.
19. Turney P., Extraction of Keyphrases from Text: Evaluation of Four Algorithms. National Research Council of Canada, Technical Report ERB-1051, October 23, 1997.
20. Witten Ian and Frank Eibe, Data Mining: Practical Machine Learning tools and Techniques with Java implementation, Morgan Kaufmann, 1999, ISBN: 1-55860-552-5.