ΜΕΘΟΔΟΙ ΚΑΤΑΤΜΗΣΗΣ ΣΕ ΡΟΕΣ ΚΕΙΜΕΝΩΝ

Η
ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ ΕΞΕΙΔΙΚΕΥΣΗΣ

Υποβάλλεται στην

ορισθείσα από την Γενική Συνέλευση Ειδικής Σύνθεσης
του Τμήματος Μηχανικών Η/Υ & Πληροφορικής
Εξεταστική Επιτροπή

από την

ΠΑΡΑΣΚΕΥΗ ΚΟΣΜΑ

ως μέρος των Υποχρεώσεων

για τη λήψη

του

ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΔΙΠΛΩΜΑΤΟΣ ΣΤΗΝ ΠΛΗΡΟΦΟΡΙΚΗ

ΜΕ ΕΞΕΙΔΙΚΕΥΣΗ ΣΤΟ ΛΟΓΙΣΜΙΚΟ

Φεβρουάριος 2016

ΕΥΧΑΡΙΣΤΙΕΣ

Κατά κύριο λόγο, οφείλω να εκφράσω τις θερμές ευχαριστίες μου στον επιβλέποντα καθηγητή μου κ. Αριστείδη Λύκα, για την εμπιστοσύνη που μου έδειξε δίνοντάς μου την δυνατότητα να εκπονήσω την διατριβή μου στο συγκεκριμένο επιστημονικό τομέα. Τον ευχαριστώ επίσης για την πολύτιμη βοήθεια και καθοδήγησή του, καθώς επίσης και για τις γνώσεις και συμβουλές που μου παρείχε καθόλη τη διάρκεια της μεταπτυχιακής μου εργασίας.

Τέλος θα ήθελα να ευχαριστήσω την οικογένειά μου για όλα όσα έχει κάνει για εμένα και τους φίλους μου για την ηθική υποστήριξη, την συμπαράσταση και την κατανόησή τους.

ΠΕΡΙΕΧΟΜΕΝΑ

ΕΥΡΕΤΗΡΙΟ ΠΙΝΑΚΩΝ

ΕΥΡΕΤΗΡΙΟ ΣΧΗΜΑΤΩΝ

ΠΕΡΙΛΗΨΗ

Παρασκευή Κοσμά του Χαρίλαου και της Βασιλικής.
MSc, Τμήμα Μηχανικών Η/Υ & Πληροφορικής, Πανεπιστήμιο Ιωαννίνων.
Φεβρουάριος 2016.
Τίτλος: Μέθοδοι Κατάτμησης σε ροές κειμένων.
Επιβλέπων: Αριστείδης Λύκας.

Στην παρούσα διατριβή εξετάζεται το πρόβλημα της κατάτμησης σε ροές κειμένων. Στόχος των μεθόδων που πραγματοποιούν κατάτμηση σε ροές κειμένων, είναι ο διαχωρισμός των κειμένων σε τμήματα, καθένα από τα οποία αποτελεί μια ξεχωριστή θεματική ενότητα. Μια προσέγγιση για την αναγνώριση σημείων αλλαγής θέματος είναι αυτή που χρησιμοποιεί λεξικολογική ανάλυση, βασισμένη στην βαρύτητα όρων προκειμένου να χωριστούν τα κείμενα σε εννοιολογικές ομάδες όπως θα έκρινε ένας αναγνώστης των κειμένων.

Ένα βασικό ζήτημα κατά την κατάτμηση κειμένων σε εννοιολογικές ομάδες, σχετίζεται με την εκτίμηση του αριθμού των ορίων, ο οποίος δεν είναι γνωστός εκ των προτέρων. Αυτό ακριβώς το πρόβλημα αποτελεί και το αντικείμενο μελέτης της συγκεκριμένης εργασίας.

Στην εργασία αυτή καταρχήν, παρουσιάζονται δύο αλγόριθμοι οι Texttiling και Tsf, οι οποίοι εφαρμόζονται για τον εντοπισμό των σημείων αλλαγής θέματος από ένα σύνολο κειμένων διαφορετικών κατηγοριών. Ο αλγόριθμος Tsf μάλιστα, παρουσιάζει πολύ καλά αποτελέσματα όταν ο σωστός αριθμός των ορίων δίνεται από τον χρήστη. Στη συνέχεια παρουσιάζεται ένας αλγόριθμος ο SegmentDip, ο οποίος προτείνει ένα κριτήριο για τον στατιστικό έλεγχο της μονοτροπικότητας ενός συνόλου δεδομένων και χρησιμοποιεί τη μεθοδολογία αυτή για την κατάτμηση κειμένων. Η συγκεκριμένη μέθοδος υποδεικνύει με αυτόματο τρόπο τον αριθμό των

ορίων μεταξύ των τμημάτων. Τέλος περιγράφεται η προτεινόμενη μεθοδολογία αυτής της εργασίας, ο αλγόριθμος Dip – Tsf, ο οποίος συνδιάζει το πλεονέκτημα του αλγορίθμου SegmentDip για την αυτόματη εκτίμηση του αριθμού των ορίων με τα πολύ καλά αποτελέσματα του αλγορίθμου Tsf όταν είναι γνωστός ο αριθμός των ορίων.

Πειραματικά, η απόδοση του αλγορίθμου Dip – Tsf παρουσιάζει πολύ καλά αποτελέσματα σημειώνοντας υψηλά ποσοστά επιτυχίας στην ανίχνευση των ορίων μεταξύ των τμημάτων μιας ροής κειμένων, χωρίς να απαιτεί τον προκαθορισμό του αριθμού των ορίων.

EXTENDED ABSTRACT IN ENGLISH

Kosma, Paraskevi, C.,V.
MSc, Department of Computer Science & Engineering, University of Ioannina, Greece.
February 2016.
Thesis Title: Segmentation Methods in text streams.
Thesis Supervisor: Aristidis Likas.

This thesis studies the problem of segmentation in text streams. The aim of segmentation in text streams is to split a stream into segments, each of which constitutes a different topic. An approach for identifying topic shifts is the one that uses lexical cohesion based on term weighting so as texts are divided into subtopic units in a similar way as human's judge.

A key issue in subdividing text streams into segments is related to the estimation of the number of boundaries, which is usually unknown beforehand. This important problem is the object of our study in this thesis.

Initially, in this thesis two algorithms are studied and implemented, namely the TextTiling and Tsf algorithms. Those are used for the detection of topic shifts in a text stream of different categories. In particular, the Tsf algorithm shows good performance when the correct number of boundaries is given by the user. Next, the SegmentDip algorithm is presented, which proposes the use of a criterion for statistical testing the unimodality of a dataset, and uses this methodology for text segmentation. This particular method automatically estimates the number of boundaries between segments of a text stream. Finally, the Dip – Tsf algorithm which is the proposed methodology of this thesis, is described. This algorithm combines the advantage of SegmentDip algorithm for the automatic estimation of the number of

boundaries with the very good results of Tsf algorithm when the number of boundaries is known.

In the experimental evaluation, Dip – Tsf shows very good performance achieving high success rates in identifying boundaries between segments in a text stream, without requiring the apriori specification of the number of text segments.

ΚΕΦΑΛΑΙΟ 1. ΕΙΣΑΓΩΓΗ

1.1. Γενικά

Η συνεχής ανάγκη για ακριβέστερη ανάκτηση και εξόρυξη πληροφορίας από ένα σύνολο κειμένων, οδήγησε στην εύρεση και ανάπτυξη μεθόδων, που προσεγγίζουν με μεγαλύτερη αποτελεσματικότητα τον βαθμό συσχέτισης μεταξύ των κειμένων. Πιο συγκεκριμένα, οι εν λόγω μέθοδοι, αξιοποιώντας την συσχέτιση των όρων-εννοιών που εμφανίζονται σε κάθε κείμενο, πλησιάζουν τον τρόπο με τον οποίο ο ανθρώπινος νους πραγματοποιεί αυτή τη συσχέτιση. Η εφαρμογή τέτοιας φύσης μεθόδων, εντοπίζεται σε πολλά πεδία ανάλυσης κειμένων, όπως είναι αυτό της ανάκτησης πληροφορίας και της εξαγωγής περίληψης.

Έτσι λοιπόν, για την ανάπτυξη μεθόδων, που στοχεύουν στην περαιτέρω ανάλυση, επεξεργασία και στον διαχωρισμό κειμένων με βάση το περιεχόμενό τους, η επιστημονική κοινότητα, στρέφει το ενδιαφέρον της στο αντικείμενο της Κατάτμησης Κειμένου (“Text Segmentation”). Με τον όρο Κατάτμηση ενός κειμένου, εννοούμε τον χωρισμό αυτού σε ομοιογενή τμήματα, καθένα από τα οποία αναφέρεται σε ένα συγκεκριμένο θέμα, ενώ συνεχόμενα τμήματα αντιστοιχούν σε διαφορετικά θέματα. Μια πολύ γνωστή προσέγγιση είναι αυτή που χρησιμοποιεί λεξιλογική ανάλυση, βασισμένη στην βαρύτητα όρων προκειμένου να χωριστούν τα έγγραφα σε θεματικές ενότητες, όπως θα έκρινε ένας αναγνώστης των κειμένων.

Για την πραγματοποίηση της κατάτμησης κειμένων, απαιτείται συνδυασμός τεχνικών εύρεσης της ομοιότητας μεταξύ των κειμένων και καθορισμού των ορίων μεταξύ των τμημάτων. Τέλος για την αξιολόγηση του αποτελέσματος της κατάτμησης, απαιτούνται τρόποι υπολογισμού του αποτελέσματος των μεθόδων κατάτμησης, εφόσον φυσικά είναι γνωστές εξαρχής οι θέσεις των ορίων μεταξύ των διαφόρων τμημάτων.

Η δυσκολία στο πεδίο της κατάτμησης κειμένου έγκειται κυρίως στα χαρακτηριστικά των κειμένων που πρόκειται να διαχωριστούν, (π.χ επιστημονικά κείμενα, ειδησεογραφικά νέα, κ.τ.λ) καθώς επίσης και από το διαχωρισμό που επιθυμούμε (εννοιολογικές ομάδες - topics, παράγραφοι, προτάσεις, κ.τ.λ).

Κατά το στάδιο ανάλυσης της απόδοσης διαφόρων μεθόδων κατάτμησης κειμένων με βάση το περιεχόμενο, εντοπίζονται δυσκολίες ως προς την αναγνώριση των αληθινών σχέσεων μεταξύ των κειμένων κάθε τμήματος και μεταξύ των τμημάτων, λαμβανομένης υπόψη της φυσικής συνεκτικότητας και συνοχής σε επίπεδο περιεχομένου. Τέτοιες δυσκολίες είναι για παράδειγμα: δημιουργία τμημάτων με ημιτελή πληροφορία, λάθος όρια (“κοψίματα”) που ανιχνεύονται σε τμήματα που ανήκουν στην ίδια εννοιολογική ομάδα, παράλειψη κειμένων επειδή ανήκουν σε άλλα τμήματα. Επίσης πολλές από τις μεθόδους εξαρτώνται από διάφορες παραμέτρους που δίνει ο χρήστης, όπως είναι μια τιμή κατωφλίου, προκειμένου να αποφασίσουν εάν υπάρχει συνεκτικότητα μεταξύ δύο κειμένων. Αυτή η εξάρτηση ενός αλγορίθμου από τον καθορισμό κατωφλίου, κάνει δύσκολη την χρήση του. Επιπλέον το να επιλέξουμε το κατάλληλο κατώφλι, για να μπορέσουμε να αποφασίσουμε αν δύο κείμενα για παράδειγμα σχετίζονται, είναι αρκετά δύσκολο, πολύπλοκο και σχετίζεται με τα χαρακτηριστικά των κειμένων κάθε φορά.

1.2. Οργάνωση της Διατριβής

Η παρούσα διατριβή επικεντρώνεται στην περιγραφή μεθόδων κατάτμησης ενός συνόλου κειμένων σε μικρότερα τμήματα, καθένα από τα οποία αντιστοιχεί και σε ένα διαφορετικό θέμα, καθώς επίσης και στην αναγνώριση των κατάλληλων σημείων αλλαγής νοήματος.

Η διάθρωση της διατριβής έχει ως εξής: στο κεφάλαιο 2 εξετάζεται το πρόβλημα της κατάτμησης κειμένων και γίνεται μια μικρή αναφορά στις διαφορετικές προσεγγίσεις όπως αυτές παρουσιάζονται στη βιβλιογραφία. Έπειτα, παρουσιάζεται η διαδικασία της προεπεξεργασίας των κειμένων πριν την εφαρμογή κάποιου αλγορίθμου κατάτμησης κειμένων, καθώς επίσης και τα διάφορα μέτρα αξιολόγησης των μεθόδων κατάτμησης κειμένων. Στο κεφάλαιο 3 παρουσιάζονται δύο αλγόριθμοι κατάτμησης κειμένων. Στο κεφάλαιο 4 παρουσιάζονται δύο μέθοδοι που βασίζονται στο στατιστικό κριτήριο Dip – dist. Τέλος στο κεφάλαιο 5 ακολουθεί η πειραματική διαδικασία και η αξιολόγηση της προτεινόμενης μεθοδολογίας, ενώ στο τελευταίο κεφάλαιο συζητούνται συνοπτικά τα αποτελέσματα και συμπεράσματα της εργασίας.

ΚΕΦΑΛΑΙΟ 2. ΚΑΤΑΤΜΗΣΗ ΚΕΙΜΕΝΟΥ – ΣΧΕΤΙΚΗ ΔΟΥΛΕΙΑ

2.1. Θεωρία των Halliday και Hasan

Η βασική θεωρία πάνω στην οποία βασίζονται όλες οι μέθοδοι κατάτμησης κειμένου είναι αυτή των Halliday και Hasan [19]. Οι Halliday και Hasan στο βιβλίο τους "Cohesion in English" υποστηρίζουν ότι κάθε κείμενο διέπεται από δύο στοιχεία: τη *συνεκτικότητα* και τη σ*υνοχή*. Οι Halliday και Hasan ορίζουν τη *συνοχή* ως μια ποιοτική ιδιότητα του κειμένου η οποία συνεισφέρει στον συνολικό χαρακτήρα αυτού, ενώ τη *συνεκτικότητα* ως μια κατάσταση στην οποία όλα τα τμήματα ή ιδέες που αναφέρονται μέσα στο κείμενο ταιριάζουν τόσο καλά μεταξύ τους ώστε να σχηματίζουν μια ολότητα, υπάρχει δηλαδή λογική σύνδεση σε επίπεδο ιδεών. Η *συνοχή* "*cohesion*" αφορά τη μορφή εξωτερικής σύνδεσης των νοημάτων μεταξύ τους με τη χρήση διαρθρωτικών λέξεων, όχι το περιεχόμενο. Πιο αναλυτικά η συνοχή του κειμένου εξασφαλίζεται με: α) επανάληψη μιας λέξης ή φράσης που προηγήθηκε β) αντικατάσταση μιας λέξης ή φράσης που προηγήθηκε με άλλη λέξη ισοδύναμή της

(π.χ αντωνυμία) γ) παράλειψη μιας λέξης επειδή εννοείται εύκολα από τα προηγούμενα δ) ερωταπόκριση (τίθεται δηλαδή ένα ερώτημα και δίνεται μετά η απάντηση) ε) οργάνωση του λόγου στον άξονα του χρόνου ή του χώρου ζ) διατήρηση ενιαίου ύφους στο λόγο η) διαρθρωτικές λέξεις ή φράσεις (επειδή, όμως, ύστερα, τελικά, δηλαδή, με άλλα λόγια, αν, πρώτον). Συνεπώς, διάφοροι μηχανισμοί συνοχής συνδέουν τα διάφορα κομμάτια του κειμένου και δημιουργούν συνεκτικότητα στα κείμενα, λογικό ειρμό. Η *συνεκτικότητα* "*coherence*" είναι η εσωτερική σύνδεση των νοημάτων που αφορά το περιεχόμενο και σχετίζεται με τη μορφή του λόγου, περιεχομένου ή στυλ όπως αυτά είχε την πρόθεση να τα παραθέσει ο συγγραφέας καταλήγοντας σε λογική διάταξη ιδεών.

Οι Halliday και Hasan αναφέρθηκαν στην έννοια της *λεξιλογικής συνοχής*, και επεσήμαναν πως η επανάληψη λέξεων και φράσεων δίνουν συνεκτικότητα σε ένα κείμενο. Συγκεκριμένα η λεξιλογική συνοχή αναφέρεται στον τρόπο με τον οποίο λέξεις που σχετίζονται μεταξύ τους, επιλέγονται για να συνδέουν τμήματα του κειμένου. Επίσης, οι Halliday και Hasan υποστήριξαν πέντε σημασιολογικές σχέσεις που αποδεικνύουν λεξιλογική συνοχή: την επανάληψη με ομοιότητα, την επανάληψη χωρίς ομοιότητα, την επανάληψη μέσω αναφοράς σε ανώτερη κατηγορία στην οποία η προαναφερθείσα οντότητα ανήκει, τη συστηματική σημασιολογική σχέση - η οποία εμφανίζεται όταν μια λέξη ή μια ομάδα λέξεων έχει μια ευκρινώς ορισμένη σχέση με μια προηγούμενα αναφερόμενη λέξη ή πρόταση -, και τέλος τη μη συστηματική σημασιολογική σχέση η οποία εμφανίζεται μεταξύ δύο λέξεων ή προτάσεων όταν αυτές ανήκουν στο ίδιο θέμα, η φύση όμως της μεταξύ τους σχέσης είναι δύσκολο να οριστεί.

Οι παραπάνω σχέσεις αποτελούν ενδείξεις ή αλλιώς κριτήρια συσχέτισης των διαφόρων τμημάτων ενός κειμένου ως προς την ομοιότητα τους, ενώ ταυτόχρονα σκιαγραφούν τη δομή αυτού. Από την άλλη πλευρά βέβαια, υπάρχουν και ερευνητές που ορίζουν διαφορετικές ενδείξεις για την εύρεση των σχέσεων μεταξύ των υπό εξέταση τμημάτων. Οι Raskin και Weiser [32] για παράδειγμα, υποστηρίζουν ότι οι μορφές που υποδεικνύουν *λεξιλογική συνοχή*, είναι η η επανάληψη και η ιδιωματική φράση – συνδιασμός λέξεων. Η επανάληψη χρησιμοποιεί την ίδια λέξη, ή συνώνυμα για να δηλώσει την αναφορά στο ίδιο στοιχείο. Η ιδιωματική φράση χρησιμοποιεί

λέξεις που κατά κάποιο τρόπο σχετίζονται μεταξύ τους, δηλαδή συνήθως πηγαίνουν μαζί ή τείνουν να επαναλαμβάνουν την ίδια έννοια.

Τέλος, βασιζόμενοι σε αυτή τη θεωρία οι μετέπειτα ερευνητές για τον υπολογισμό της ομοιογένειας ή εναλλακτικά της ετερογένειας τμημάτων σε ένα κείμενο, ανέπτυξαν διάφορους αλγορίθμους κατάτμησης, οι οποίοι διαχωρίζονται σε δύο μεγάλες οικογένειες: τη *γλωσσολογική* και τη *στατιστική*.

2.2. Γλωσσολογική προσέγγιση της εύρεσης της δομής ενός κειμένου

Η προσέγγιση αυτή επιδιώκει τη χρήση *γλωσσολογικών ενδείξεων* ή αλλιώς κριτηρίων για την εύρεση της ομοιότητας μεταξύ διαφορετικών τμημάτων ενός κειμένου. Ως τέτοιες ενδείξεις χρησιμοποιήθηκαν από τους Hirschberg και Litman [20] οι λέξεις και προτάσεις "σινιάλο" ("cue words and phrases"), οι οποίες είναι λέξεις και προτάσεις που χρησιμοποιούνται για να σηματοδοτήσουν τη δομή του κειμένου όπως π.χ επιρρήματα ή ρήματα ή άλλες λέξεις που υποδηλώνουν την πιθανή αλλαγή θεματικής ενότητας. Λέξεις όπως π.χ (now, anyway , meanwhile, on the other hand, that reminds me) , θεωρούνται λέξεις "σινιάλο", επιπλέον, η λέξη «anyway» μπορεί και να σημαίνει επιστροφή στο θέμα, ενώ η λέξη «that reminds me» μπορεί να σημαίνει παρέκβαση, απομάκρυνση από το θέμα. Οι εν λόγω λέξεις και οι προτάσεις «σινιάλο» ("cue words and phrases"), συλλέχθηκαν από διάφορες πηγές οι οποίες ήταν ανεξάρτητες θεματικής κατηγορίας. Αντίθετα από τον Reynar [21] χρησιμοποιήθηκαν συνώνυμα, λέξεις και οι προτάσεις "σινιάλο" ("cue words and phrases"), οι οποίες εξαρτώνται σημαντικά από την εκάστοτε θεματική περιοχή ή περιείχαν ακολουθίες λέξεων συγκεκριμένου τύπου όπως για παράδειγμα ονόματα περιοχών και ατόμων.

Από την άλλη πλευρά, οι Passonneau και Litman [8] [9] ανέπτυξαν αλγορίθμους κατάτμησης βασιζόμενους πέρα από λέξεις σινιάλο ("cue words"), σε αναφορικές ονοματικές προτάσεις και την ύπαρξη σημείων στίξης. Τέλος, οι Kan, Klavans και McKeown [38] για την εύρεση της δομής ενός κειμένου, επικέντρωσαν

την προσοχή σε αντωνυμίες καθώς επίσης και σε ονοματικές προτάσεις, υποστηρίζοντας πως οι κύριες ονοματικές προτάσεις, οι κοινές ονοματικές προτάσεις και οι προσωπικές και κτητικές αντωνυμίες αντανακλούν το θεματικό περιεχόμενο του κειμένου.

2.3. Στατιστική προσέγγιση εύρεσης της δομής ενός κειμένου

Η προσέγγιση αυτή επιδιώκει την εύρεση της δομής ενός κειμένου με τη χρήση στατιστικών κριτηρίων. Για την εν λόγω εύρεση, οι ερευνητές βασίστηκαν στην παραδοχή ότι η δομή ενός κειμένου είναι είτε γραμμική είτε ιεραρχική. Βασιζόμενοι στη γραμμική ή ιεραρχική παραδοχή της δομής του κειμένου, ανέπτυξαν στη συνέχεια διάφορους αλγόριθμους (οι οποίοι βασίζονται κυρίως στη συχνότητα εμφάνισης των λέξεων μέσα στο κείμενο) και διάφορες τεχνικές για τον υπολογισμό της ομοιότητας είτε μεταξύ γειτονικών είτε μεταξύ όλων των τμημάτων των κειμένων. Στην διατριβή της Fragkou P. [44] προτείνεται η παρακάτω κατηγοροποίηση μεθόδων.

2.4. Γραμμική κατάτμηση κειμένου

Στην γραμμική κατάτμηση κειμένου, ένα μεγάλο κείμενο χωρίζεται σε μικρότερα τμήματα, συνήθως σε μπλόκ διαδοχικών προτάσεων. Οι αλγόριθμοι οι οποίοι πραγματοποιούν γραμμική κατάτμηση βασίζονται στην παραδοχή ότι η δομή του λόγου έχει γραμμική μορφή. Σαν αποτέλεσμα αυτής της παραδοχής τόσο ο τρόπος με τον οποίο υπολογίζεται η ομοιότητα μεταξύ των διαφόρων τμημάτων ενός κειμένου όσο και οι τεχνικές εύρεσης μεταξύ των ορίων των τμημάτων είναι γραμμικοί. Στις επόμενες υποενότητες περιγράφονται τόσο οι τεχνικές που χρησιμοποιήθηκαν για την εύρεση των στοιχείων εκείνων που υποδηλώνουν συνεκτικότητα μέσα στο κείμενο, όσο και οι τεχνικές βάσει των οποίων υπολογίζεται η ομοιότητα μεταξύ μερικών μερών ή όλων των μερών του κειμένου.

2.4.1. Εξέταση της εμφάνισης των λέξεων μέσα σε ένα κείμενο

Σύμφωνα με την πλειοψηφία των ερευνητών, η αναλυτική μελέτη της εμφάνισης των λέξεων μέσα σε ένα κείμενο μπορεί να οδηγήσει σε σημαντικά συμπεράσματα για την ομοιογένεια ή ετερογένεια τμημάτων ενός κειμένου και εν γένει για την εύρεση της συνεκτικότητας κάθε κειμένου. Η σχέση ανάμεσα στην συνεκτικότητα και στην εμφάνιση των λέξεων μέσα σε ένα κείμενο μοντελοποιήθηκε από τον κάθε ερευνητή με διαφορετικό τρόπο.

Ο Youmans [22] [23] υποστήριξε ότι η πρώτη χρήση λέξεων μέσα σε ένα κείμενο συνήθως συνοδεύει αλλαγές στο θέμα, δεδομένου ότι γίνεται αναφορά σε καινούρια πρόσωπα, μέρη και γεγονότα, οδηγώντας στη χρήση λέξεων οι οποίες δεν εμφανίζονται προηγούμενα μέσα σε αυτό. Αντίστοιχα, λέξεις οι οποίες επαναλαμβάνονται συχνά αποτελούν ισχυρή ένδειξη για την παρουσία του ίδιου θέματος στο εύρος του κειμένου που εξετάζουμε. Από την άλλη πλευρά ο Philips [24] εξέτασε τη σχέση μεταξύ των λέξεων που έχουν την τάση να εμφανίζονται μαζί και χρησιμοποίησε τα στατιστικά συχνότητας εμφάνισης τέτοιων λέξεων μαζί με ανάλυση ομάδων για να αναγνωρίσει δίκτυα λέξεων σε κεφάλαια επιστημονικών βιβλίων. Η Hearst [2] [3] [4] [5] χρησιμοποίησε το παραδοσιακό μοντέλο διανύσματος χώρου σε επίπεδο προτάσεων, χρησιμοποιώντας την συχνότητα εμφάνισης των λέξεων που εμφανίζονται σε καθεμία από αυτές. Η τεχνική κατάτμησης κειμένου που πρότεινε, στηρίζεται στην παραδοχή ότι μια αλλαγή στο θέμα συνοδεύεται από αλλαγή στο λεξιλόγιο. Την ίδια ακριβώς προσέγγιση ακολούθησαν και ο Kern & Granitzer [7]. Επίσης ο Yaari [25] [26] εξέτασε την επανάληψη λέξεων ή αλυσίδων λέξεων που αναφέρονται στο ίδιο θέμα.

Οι Ponte και Croft [27] πρότειναν μια τεχνική κατάτμησης κειμένου ανά θεματική ενότητα η οποία βασίζεται στην *τοπική ανάλυση περιεχομένου* (Xu & Croft [28]), επιτρέποντας την αντικατάσταση κάθε πρότασης με λέξεις και φράσεις που σχετίζονται με αυτή. Στη συνέχεια υπολόγιζαν την ομοιότητα μεταξύ των γειτονικών τροποποιημένων πια προτάσεων, μετρώντας τον αριθμό των κοινών λέξεων και φράσεων ανάμεσα στα σύνολα των επιλεγμένων λέξεων και φράσεων που εμφανίζονταν σε κάθε πρόταση. Από την άλλη πλευρά οι Ahonen, Heikkinen,

Heinonen and Klemettinen [29] χρησιμοποίησαν μόνο ένα προκαθορισμένο αριθμό από λέξεις (π.χ 50 λέξεις περίπου σε επίπεδο παραγράφου), οι οποίες εμφάνιζαν τη μεγαλύτερη συχνότητα, για να υπολογίσουν την ομοιότητα μεταξύ των παραγράφων με τη βοήθεια της συνημιτονοειδούς ομοιότητας.

Οι Beeferman, Berger και Laffety [11] [12] [13] πρότειναν ένα πιθανοτικό μοντέλο το οποίο ονόμασαν "μοντέλο επαγωγής χαρακτηριστικών όρων από τυχαία πεδία και εκθετικά μοντέλα". Η όλη ιδέα έχει να κάνει με την ανάθεση μιας πιθανοτικής κατανομής σε μια συγκεκριμένη θέση στην ροή των δεδομένων συνδυάζοντας διαφορετικούς χαρακτηριστικούς όρους και αναθέτοντας σε καθένα από αυτούς ένα βάρος σε ένα εκθετικό μοντέλο. Το μοντέλο χρησιμοποίησε δύο κατηγορίες χαρακτηριστικών όρων: τους τ*οπικούς*, οι οποίοι χρησιμοποιούσαν προσαρμοσμένα γλωσσικά μοντέλα και τις λέξεις "σινιάλο" ("cue words") χαρακτηριστικούς όρους για την ανίχνευση ειδικών λέξεων οι οποίες μπορεί να εξαρτώνται από τη θεματική περιοχή.

Όλες οι παραπάνω μέθοδοι χρησιμοποιούνταν συχνά στον υπολογισμό της ομοιότητας μεταξύ γειτονικών ή όλων των τμημάτων ενός κειμένου. Οι μηχανισμοί που κατά καιρούς προτάθηκαν στη βιβλιογραφία για τον εν λόγω υπολογισμό περιγράφονται στις ενότητες που ακολουθούν.

2.4.2. Μηχανισμοί εύρεσης της ομοιότητας μεταξύ γειτονικών τμημάτων ενός κειμένου

Η συγκεκριμένη οικογένεια αλγορίθμων εξετάζει μόνο την πιθανή εμφάνιση του ίδιου θέματος σε γειτονικά τμήματα του κειμένου, δηλαδή κάθε φορά εξετάζεται ένα μικρό τμήμα του κειμένου με τα γειτονικά του για τυχόν αλλαγές στην ομοιότητα. Όπως έχει ήδη αναφερθεί, οι αλλαγές στην ομοιότητα μεταξύ των γειτονικών τμημάτων, σηματοδοτούν την ύπαρξη ορίων και κατά συνέπεια την εμφάνιση διαφορετικής θεματικής ενότητας μέσα στο κείμενο.

Η εμφάνιση παρόμοιου λεξιλογίου το οποίο παραμένει αμετάβλητο μέσα σε τμήματα κειμένου που πραγματεύονται το ίδιο θέμα, οδήγησε σε μεθόδους που προχωρούν σε διαχωρισμό του κειμένου, βασιζόμενες στον υπολογισμό της ομοιότητας του λεξιλογίου μεταξύ δύο γειτονικών παραθύρων κειμένου. Δηλαδή, οι μέθοδοι αυτοί, χρησιμοποιούν κατά κύριο λόγο την επανάληψη λέξεων μέσα σε ένα κινούμενο παράθυρο ως έναν μηχανισμό λεξιλογικής συνοχής.

Η παραπάνω προσέγγιση αυτή ακολουθήθηκε από την Hearst [2] [3] [4] [5] η οποία χρησιμοποίησε την τεχνική του κινούμενου παραθύρου "sliding window" και υπολόγιζε την ομοιότητα ενός μπλοκ με τα γειτονικά του. Εδώ ένας σημαντικός παράγοντας του αλγορίθμου ήταν το μέγεθος του μπλόκ, δηλαδή το πλήθος των ψευδοπροτάσεων (οι οποίες αποτελούνταν κατά μέσο όρο από 20 λέξεις) οι οποίες το συνιστούσαν. Ως μέγεθος του μπλοκ Κ ορίστηκε ο μέσος όρος των παραγράφων ενός κειμένου (μετρούμενο ως προς τις λέξεις). Στα πειράματα τα οποία πραγματοποίησε, η ομοιότητα μεταξύ μπλόκ υπολογίστηκε με τη βοήθεια του συνημιτόνου.

Οι Roman Kern και Michael Granitz [7] υιοθέτησαν τον ισχυρισμό ότι η λεξιλογική συνοχή αποτελεί ισχυρή ένδειξη αλλαγής νοήματος σε ένα κείμενο και χρησιμοποίησαν επίσης την τεχνική του κινούμενου παραθύρου "sliding window", για τον υπολογισμό της ομοιότητας κάθε τμήματος με τα γειτονικά του. Παρομοίως με την τεχνική της Hearst, για κάθε πλευρά δημιουργούνται δύο μπλόκ, τα οποία προηγούνται και αντιστοίχως διαδέχονται το υπό εξέταση τμήμα κειμένου. Το μέγεθος του μπλόκ δίνεται από τον χρήστη και υποδηλώνει το ελάχιστο μέγεθος για ένα τμήμα ώστε να θεωρηθεί έγκυρο. Για τον υπολογισμό της ομοιότητας μεταξύ των τμημάτων χρησιμοποιήθηκε η συνημιτονοειδής ομοιότητα.

Από την άλλη πλευρά οι Richmond, Smith και Amitay [30] αφού μέτρησαν την σημαντικότητα των λέξεων με βάση τη συχνότητα εμφάνισής τους μέσα στο κείμενο και την απόσταση των επαναλήψεων των λέξεων, καθόρισαν την ομοιότητα μεταξύ γειτονικών περιοχών του κειμένου αθροίζοντας τα βάρη των λέξεων οι οποίες εμφανίζονταν και στις δύο περιοχές και στη συνέχεια αφαιρώντας τα αθροισμένα βάρη των λέξεων οι οποίες εμφανίζονταν μόνο σε ένα από τα δύο τμήματα. Στη συνέχεια κανονικοποιούσαν διαιρώντας με τον αριθμό των λέξεων σε κάθε τμήμα.

Τέλος ο Choi [31], υπολόγισε την ομοιότητα μεταξύ γειτονικών τμημάτων κάνοντας χρήση της τεχνικής "*Σειρά σε τοπικό Επίπεδο*", η οποία αποτελεί ένα σχήμα ταξινόμησης σε σειρά που βασίζεται στην ομοιότητα που εμφανίζεται σε τοπική περιοχή. Πιο συγκεκριμένα, κάθε τιμή του πίνακα ομοιότητας μεταξύ των προτάσεων του κειμένου (όπου κάθε πρόταση έχει παρασταθεί με τη βοήθεια ενός διανυσματικού μοντέλου) αντικαθίσταται από τη σειρά της σε τοπική περιοχή. Η τιμή που λαμβάνει η σειρά είναι ο αριθμός των γειτονικών περιοχών οι οποίες εμφανίζουν μικρότερη τιμή ομοιότητας. Για την αποφυγή προβλημάτων κανονικοποίησης, κάθε τιμή αντικαθίσταται από το λόγο του πλήθους των στοιχείων τα οποία παρουσιάζουν μικρότερη τιμή ομοιότητας δια το πλήθος των εξεταζόμενων στοιχείων.

2.4.3. Μηχανισμοί εύρεσης της ομοιότητας μεταξύ όλων των τμημάτων ενός κειμένου

Η εν λόγω οικογένεια αλγορίθμων, από την άλλη πλευρά, εξετάζει πιθανές εμφανίσεις του ίδιου θέματος σε όλη την έκταση του κειμένου, δηλαδή βασίζεται στην παραδοχή ότι περισσότερα από ένα τμήματα του κειμένου τα οποία απαντώνται σε τυχαία – δηλαδή όχι κατ'ανάγκη σε συνεχόμενα - σημεία αυτού, είναι δυνατό να αναφέρονται στο ίδιο θέμα. Έτσι υπολογίζουν την ομοιότητα όλων των μερών του κειμένου μεταξύ τους με σκοπό την εύρεση και τέτοιων επαναλήψεων.

Σε αυτή την οικογένεια αλγορίθμων ανήκει και η μέθοδος που ανέπτυξε ο Philips [24] με σκοπό την εύρεση της συνολικής δομής των θεμάτων, χρησιμοποιώντας τη μέθοδο ομαδοποίησης για να αναγνωρίσει δίκτυα λέξεων σε κεφάλαια επιστημονικών βιβλίων. Η εν λόγω μέθοδος αρχικά εξήγαγε από τα συμπλέγματα λέξεων που παρήχθησαν ως αποτέλεσμα της ομαδοποίησης "κεντρικές" λέξεις , τις οποίες θεωρούσε ότι ήταν οι πιο βασικές στο κείμενο με σκοπό την αναγνώριση της δομής των υποθεμάτων. Στη συνέχεια συνέκρινε σύνολα από "κεντρικές" λέξεις από διαφορετικά κεφάλαια και στην περίπτωση που ήταν επαρκώς όμοια σημείωνε ομοιότητα μεταξύ των κεφαλαίων.

Ο Reynar [21] [35] για τον υπολογισμό της ομοιότητας μεταξύ όλων των μερών του κειμένου βασίστηκε στο διανυσματικό μοντέλο κατασκευάζοντας τα αντίστοιχα διανύσματα για κάθε πρόταση του κειμένου. Στη συνέχεια υπολόγιζε την ομοιότητα κάθε πρότασης με όλες τις υπόλοιπες, με τη βοήθεια του συνημιτόνου. Με αυτόν τον τρόπο κατασκεύαζε κάθε φορά έναν τετραγωνικό πίνακα με τις εν λόγω ομοιότητες. Παρατήρησε ότι η απόδοση όλων των παραμετρικοποιήσεων της εν λόγω τεχνικής βελτιωνόταν όταν οι λέξεις αντικαθίστανται από το αντίστοιχό τους λήμμα και αγνοούνται αυτές που ανήκαν στην stop list.

Οι Utiyama και Isahara [34] κατασκεύασαν ένα πιθανοτικό μοντέλο το οποίο υπολόγιζε την μέγιστη πιθανότητα κατάτμησης για ένα δοσμένο κείμενο. Ο αλγόριθμος αυτός επέλεγε τη βέλτιστη κατάτμηση βασιζόμενος στην πιθανότητα που οριζόταν από ένα στατιστικό μοντέλο. Το στατιστικό μοντέλο υπολόγιζε τις πιθανότητες των λέξεων να ανήκουν σε ένα θέμα. Έκαναν την παραδοχή ότι ένα θέμα καθορίζεται από την κατανομή των λέξεων οι οποίες περιέχονται σε αυτό, με διαφορετικά θέματα να εμφανίζουν διαφορετική κατανομή λέξεων.

Οι Choi,Wiemer-Hastings και Moore [33] πρότειναν μια καινούρια μετρική για τον υπολογισμό της ομοιότητας η οποία ονομάζεται Latent Semantic Analysis (LSA). Εδώ η σημασία μιας λέξης παρίσταται με βάση τη συσχέτισή της με τις άλλες λέξεις. Συγκεκριμένα, η τεχνική LSA εφαρμόζει την τεχνική ανάλυσης σε συνιστώσες που καταλαμβάνουν την πρώτη θέση στον πίνακα ομοιότητας μεταξύ των λέξεων με σκοπό τον προσδιορισμό των καλύτερων χαρακτηριστικών όρων για τη διάκριση ανομοιογενών λέξεων. Η σημασία κάθε λέξης υπολογίζεται ως το άθροισμα των διανυσμάτων των χαρακτηριστικών όρων. Η ομοιότητα των κειμένων στη συνέχεια, υπολογίζεται με τη βοήθεια του συνημιτόνου της γωνίας των αντίστοιχων διανυσμάτων των χαρακτηριστικών όρων.

Ο Sardinha [45] [46] αναφέρθηκε στον υπολογισμό της ομοιότητας μεταξύ *συνδέσμων* οι οποίοι εμφανίζονται ανάμεσα στις προτάσεις του κειμένου. Συγκεκριμένα, θεώρησε ότι ένας σύνδεσμος εμφανίζεται όταν λαμβάνει χώρα επανάληψη μιας λέξης σε δύο διαφορετικές προτάσεις. Έτσι εισήγαγε την έννοια του *συνόλου συνδέσμων,* όπου θεώρησε ότι το σύνολο των προτάσεων με το οποίο

συνδέεται μια πρόταση μέσω συνδέσμου (δηλ. έχει κοινές λέξεις) σχηματίζει το σύνολο των συνδέσμων γι'αυτή. Κατά αυτόν τον τρόπο, η ομοιότητα μέσα στο κείμενο μπορεί να καθοριστεί εξετάζοντας το κοινό λεξιλόγιο μεταξύ των προτάσεων του κειμένου.

2.5. Ιεραρχική κατάτμηση κειμένου

Στην ιεραρχική κατάτμηση κειμένου, τα κείμενα χωρίζονται επαναληπτικά σε τμήματα. Συγκεκριμένα η έξοδος ενός αλγορίθμου που υλοποιεί την διαδικασία αυτή προσπαθεί κατά κάποιον τρόπο να αναγνωρίσει την δομή ενός εγγράφου, συνήθως τα κεφάλαια και τα υποκεφάλαια αυτού. Η όλη ιδέα έχει να κάνει με την υπόθεση ότι ένα κείμενο έχει ιεραρχική δομή.
Οι τεχνικές που πραγματοποιούν ιεραρχική κατάτμηση βασίζονται σε δύο κυρίως θεωρίες:

Η πιο γνωστή ιεραρχική θεωρία του λόγου είναι αυτή της attentional/ intentional δομής η οποία διατυπώθηκε από τους Grosz και Sider [37] και σύμφωνα με την οποία η δομή του λόγου συνίσταται από τμήματα αυτού και μια σχέση εμπέδωσης η οποία υπάρχει μεταξύ τους. Πιο συγκεκριμένα, η εν λόγω θεωρία συνίσταται από τρία τμήματα: την attentional state, τη γλωσσολογική δομή και τη δομή πρόθεσης ("intentional state"). Η attentional state επικεντρώνεται στην εστίαση προσοχής των συνομιλητών και στην πρόσβαση των οντοτήτων του λόγου που βρίσκονται σε «περίοπτη θέση» ("salience"). Η γλωσσολογική δομή «συλλαμβάνει» τις σχέσεις ανάμεσα σε διαδοχικές λέξεις ή εκφράσεις ("utterances") και διαιρεί το κείμενο σε τμήματα του λόγου. Αυτά τα τμήματα σχηματίζουν μια ιεραρχική δομή. Η γλωσσολογική δομή περιορίζει αλλαγές στην attentional state. Τέλος, η δομή πρόθεσης ("intentional state"), μοντελοποιεί τους στόχους και τους υποστόχους των συνομιλητών. Ο στόχος των τμημάτων του λόγου είναι η πρόθεση η οποία σχετίζεται με ένα τμήμα του λόγου. Οι Grosz και Sidver κατηγοριοποιούν τις συσχετίσεις μεταξύ των προθέσεων οι οποίες είναι δυνατό να αναγνωριστούν από τη γλωσσολογική δομή. Έτσι όταν μια πρόθεση κυριαρχεί έναντι κάποιας άλλης στη δομή πρόθεσης, τότε η κυριαρχούμενη πρόθεση αντιστοιχεί σε ένα τμήμα του λόγου

στη γλωσσολογική δομή η οποία είναι απόγονος του τμήματος λόγου που σχετίζεται με την κυριαρχούσα δομή, σχηματίζοντας έτσι μια ιεραρχική περιγραφή του λόγου.

Μια άλλη θεωρία πάνω στην οποία βασίζονται οι τεχνικές που πραγματοποιούν ιεραρχική κατάτμηση είναι η ρητορική δομή η οποία διατυπώθηκε από τους Mann και Thomson [36]. Η θεωρία της Ρητορικής δομής είναι μια συναρτησιακή ανάλυση βασιζόμενη σε ένα περιγραφικό μοντέλο, της ρητορικής δομής του κειμένου.

Οι Morris και Hirst [39] [40] υλοποίησαν έναν αλγόριθμο κατάτμησης, ο οποίος χώριζε το κείμενο σε τμήματα τα οποία σχημάτιζαν ιεραρχική δομή. Η τεχνική που χρησιμοποιήθηκε είναι αυτή της δημιουργίας λεξιλογικών αλυσίδων (“lexical chains”) για την εύρεση της δομής ενός κειμένου. Αρχικά, γίνεται η σύνδεση ακολουθιών λέξεων – σχετικών μεταξύ τους – με σκοπό τον σχηματισμό αλυσίδων. Για την εύρεση των συσχετίσεων που δηλώνουν λεξιλογική συνοχή μεταξύ των λέξεων, χρησιμοποιήθηκε ο θησαυρός όρων Roget [41] [42]. Μετά την αναγνώριση και τον σχηματισμό αλυσίδων λέξεων στο κείμενο, γίνεται σύγκριση των στοιχείων των αλυσίδων λέξεων για τον καθορισμό του κατά πόσο μια αλυσίδα αποτελεί συνέχεια μιας άλλης η οποία εμφανίστηκε προηγουμένως. Σε τέτοιες αλυσίδες έδιναν την ετικέτα “επιστρεφόμενες αλυσίδες” επειδή “επανα-επισκέπτονταν” το θέμα το οποίο είχε αναφερθεί από μια άλλη αλυσίδα λέξεων, κατασκευάζοντας με αυτόν τον τρόπο μια ιεραρχία από αλυσίδες λέξεων.

Ο Yaari [25] [26] πραγματοποίησε ιεραρχική κατάτμηση, κάνοντας χρήση της τεχνικής *Hierarchical Agglomerative Clustering*. Αρχικά, έγινε η αφαίρεση λέξεων, οι οποίες ανήκαν σε μια κλειστή κατηγορία και έπειτα με τη βοήθεια του αλγόριθμου του Porter [43] πραγματοποιήθηκε η μείωση των λέξεων στη ρίζα τους. Στη συνέχεια υπολογίστηκε η ομοιότητα μεταξύ παραγράφων χρησιμοποιώντας την συνημιτονοειδή ομοιότητα με το αντίστροφο βάρος συχνότητας εμφάνισης, το οποίο δίνει μεγαλύτερη βαρύτητα σε σπάνιες λέξεις του κειμένου παρά σε πολύ συχνές. Η τεχνική *Hierarchical Agglomerative Clustering* αξιοποίησε αυτές τις τιμές της ομοιότητας και προχώρησε στην ομαδοποίηση γειτονικών τμημάτων και πιο συγκεκριμένα παραγράφων. Κατόπιν, ο Yaari δημιούργησε ένα δενδρόγραμμα

δείχνοντας τη σειρά με την οποία πραγματοποιήθηκαν οι συγχωνεύσεις και στη συνέχεια, ακολούθησε κάποιους κανόνες για την εύρεση των ορίων μεταξύ των διαφόρων τμημάτων με σκοπό τη μετατροπή της ιεραρχικής δομής σε μια γραμμική κατάτμηση ούτως ώστε να γίνεται ευκολότερη η σύγκριση του αποτελέσματος της κατάτμησης.

2.6. Προεπεξεργασία αρχείων κειμένου

Είναι πολύ αναγκαία η σωστή οργάνωση και η προετοιμασία των δεδομένων που πρόκειτε να επεξεργαστούμε, πριν εφαρμόσουμε οποιονδήποτε αλγόριθμο κατάτμησης κειμένου. Σε αυτό το σημείο θα πρέπει να σημειωθεί, πως υπάρχουν στην διάθεσή μας λογισμικά, που υλοποιούν την παρακάτω διαδικασία προεπεξεργασίας κειμένων.

Στο πρώτο στάδιο, γίνεται η ανάλυση του κειμένου, δηλαδή, όλα τα κείμενα πρέπει να έρθουν σε μια πρότυπη μορφή (π.χ τίτλος, συγγραφέας, κείμενο), ανάλογα με το είδος του κειμένου ή του τύπου δεδομένων. Στη συνέχεια, στο επόμενο στάδιο, πρέπει να οριστούν ποιοι όροι έχουν σημασιολογική βαρύτητα και ποιοι όχι. Αυτοί οι όροι που δεν έχουν σημασιολογική βαρύτητα θα πρέπει να αποκλειστούν, τέτοιοι όροι είναι συνήθως άρθρα, προθέσεις, κάποια συνηθισμένα επίθετα αλλά και κάποια ουσιαστικά,έτσι δημιουργείτε μια λίστα με τους όρους που θέλουμε να αποκλείσουμε (stop list), π.χ ‘is’, ‘the’, ‘for’, ‘of’, ‘to’, ‘and’. Συνήθως οι όροι με μεγαλύτερη σημασιολογική βαρύτητα είναι τα ρήματα. Επίσης υποψήφιες για αποκλεισμό, μπορεί να είναι και άλλες λέξεις που εμφανίζονται σε μικρό αριθμό εγγράφων, αν δηλαδή η συχνότητα εμφάνισής τους (“document frequency – df”) , στο σύνολο των εγγράφων, είναι κάτω από το κατώφλι που έχουμε ορίσει, τότε μπορούμε να τις αφαιρέσουμε. Αυτή η προσέγγιση ονομάζεται document frequency thresholding (DFT).

Έπειτα, δεδομένου ότι οι όροι ενός κειμένου, αντιμετωπίζονται σαν λεκτικές μονάδες, είναι μέρος της σχεδίασης ο χαρακτηρισμός τους (λέξεις, αριθμοί, σημεία στίξης κλπ) αλλά και η αφαίρεση του επιθέματος τους (Stemming). Για παράδειγμα οι λέξεις ‘player’, ‘playing’, ‘played’, δεν έχουν σημασιολογική διαφορά η μία από

την άλλη, δηλαδή σχετίζονται όλες με το ρήμα ‘play’, επομένως κρατάμε μόνο το θέμα του όρου (στο συγκεκριμένο παράδειγμα play) και αφαιρούμε το επίθεμα.

Τέλος, γίνεται η τελική επιλογή του λεξιλογίου, η εξαγωγή των δεδομένων και η κανονικοποίηση των όρων.

2.7. Μοντέλο Διανυσματικού Χώρου – Vector Space Model

Το μοντέλο διανυσματικού χώρου (“Vector Space Model”) αποτελεί μια μέθοδο αναπαράστασης των όρων σε μία συλλογή κειμένων. Σε ένα τέτοιο μοντέλο, κάθε στοιχείο του διανύσματος μπορεί να χρησιμοποιηθεί για να αναπαραστήσει λέξεις, φράσεις ή θέμα ενός κειμένου. Κάθε τιμή που παίρνει αυτό το στοιχείο αντανακλά στην σπουδαιότητα του όρου και αναπαριστά την σημαντικότητα (“semantic”) του κειμένου, αλλά σε αυτό θα αναφερθούμε αναλυτικότερα παρακάτω.

Μια συλλογή κειμένων που αποτελείται από n κείμενα τα οποία περιέχουν m όρους, οι οποίοι έχουν καθοριστεί κατά το στάδιο της προεπεξεργασίας, μπορούν να αναπαρασταθούν σαν ένας nxm πίνακας A. Οι γραμμές του πίνακα A αποτελούν τα διανύσματα του κειμένου (“document vectors”) και οι στήλες του πίνακα A αποτελούν τα διανύσματα όρου (“term vectors”). Επομένως το στοιχείο A_{ij} αναπαριστά την συχνότητα εμφάνισης του όρου j στο κείμενο i.

Επιπλέον, δεδομένου ότι ένα έγγραφο μπορεί να είναι μεγάλο σε μέγεθος και άλλο να είναι μικρό, για να έχει κάθε έγγραφο ίση σημασία – αναλογία με τα υπόλοιπα, και για να δοθεί μεγαλύτερη σημασία στα μικρά κείμενα, γίνεται κανονικοποίηση των όρων. Συγκεκριμένα, σε κάθε όρο εκχωρείται ένα βάρος, που δηλώνει την σπουδαιότητά του, για ένα συγκεκριμένο έγγραφο. Η πιο γνωστή μέθοδος είναι η Tf-idf term weighting, η οποία αναθέτει ένα υψηλό βάρος σε έναν όρο, εάν αυτός εμφανίζεται συχνά στο έγγραφο, αλλά σπάνια σε όλο το σύνολο των εγγράφων. Στην αντίθετη περίπτωση, ένας όρος που εμφανίζεται σχεδόν σε όλα τα κείμενα δεν αποτελεί ισχυρή ένδειξη για διάκριση και του αποδίδεται χαμηλό βάρος.

Για να υπολογιστεί το tf-idf βάρος ενός όρου σε ένα συγκεκριμένο έγγραφο, είναι απαιραίτητο να γνωρίζουμε: α) Πόσο συχνά εμφανίζεται αυτός ο όρος στο έγγραφο (term frequency=tf) και β) σε πόσα έγγραφα από το σύνολο των εγγράφων εμφανίζεται (document frequency=df) . Στην συνέχεια, βρίσκουμε τον αντίστροφο του df (idf – Inverse document frequency) που ορίζεται ως ο λογάριθμος (με βάση το 10) του πηλίκου του συνολικού αριθμού εγγράφων και της συχνότητας εμφάνισης του όρου στο σύνολο των εγγράφων. Τέλος, το γινόμενο tf x idf υπολογίζει το βάρος κάθε όρου.

$$\text{tf x idf=tf x } \log\left(\frac{n}{df}\right) \qquad \text{Εξ. 2.7.1}$$

Η ομοιότητα μεταξύ των κειμένων μπορεί να υπολογιστεί με διάφορα μέτρα πάνω σε διανύσματα. Παραδείγματα τέτοιων μέτρων ομοιότητας είναι η συνημιτονοειδής ομοιότητα (cosine measure), η Ευκλείδεια απόσταση (Euclidian measure) κ.α. Όταν χρησιμοποιείται η tfidf αναπαράσταση, η συνημιτονοειδής ομοιότητα (cosine similarity) έχει αποδειχτεί πως είναι ένας αποτελεσματικός τρόπος μέτρησης της ομοιότητας μεταξύ δύο εγγράφων στο πεδίο της εξόρυξης κειμένων.

$$\text{Similarity (A,B)} = \cos(\theta) = \frac{\mathbf{A}\cdot\mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}} \qquad \text{Εξ. 2.7.2}$$

όπου A_i το βάρος του όρου i του εγγράφου A και B_i το βάρος του όρου i του εγγράφου B. Η συνημιτονοειδής ομοιότητα, δίνει τιμές μεταξύ 0 και 1 και όσο πιο πολλές κοινές λέξεις έχουν τα κείμενα τόσο πιο μεγάλη είναι η τιμή της.

2.8. Μέτρα αξιολόγησης της κατάτμησης

Υπάρχουν πολλοί τρόποι με τους οποίους θα μπορούσε κανείς να αξιολογήσει μια μέθοδο κατάτμησης κειμένου. Καταρχήν, αν υποτεθεί πως είναι γνωστή η πραγματική κατάτμηση, (“ground truth”) , αν γνωρίζουμε δηλαδή εκ των προτέρων

τα σημεία αλλαγής κάθε θεματικής κατηγορίας, τότε η διαφορά μεταξύ αυτής της σωστής λύσης και μιας υποθετικής κατάτμησης θεωρείται ότι μετράει το ποσοστό λάθους που κάνει μια μέθοδος κατάτμησης κειμένου, ως προς την ανίχνευση των ορίων.

Οι Passonneau και Litman [8] [9] υπολόγισαν κάποιες μετρικές που ανήκουν στον τομέα της *ανάκτησης πληροφορίας*, αυτές είναι οι *Precision, Recall, Fallout, Error*. Για το πρόβλημα της κατάτμησης κειμένων, το μέτρο του *Precision* ορίστηκε ως το πλήθος των εκτιμώμενων ορίων που είναι πραγματικά όρια, διαιρούμενο προς το συνολικό πλήθος των εκτιμώμενων ορίων. Εντελώς αντίστοιχα, το μέτρο του *Recall* ορίστηκε ως το πλήθος των εκτιμώμενων ορίων που είναι πραγματικά όρια διαιρούμενα δια του συνολικού πλήθους των πραγματικών ορίων. Το μέτρο του *Fallout* υπολογίζει πόσο συχνά μη σωστά όρια αναγνωρίζονται από μια υποθετική κατάτμηση ενώ το *Error* μετράει τον συνολικό αριθμό των λάθος ορίων που αναγνωρίζονται και των σημείων που δεν θεωρούνται σημεία αλλαγής νοήματος.

$$Precision = \frac{TP}{TP + FP} \qquad \text{Εξ. 2.8.1}$$

$$Recall = \frac{TP}{TP + FN} \qquad \text{Εξ. 2.8.2}$$

$$Fallout = \frac{FP}{FP + TN} \qquad \text{Εξ. 2.8.3}$$

$$Error = \frac{FP+FN}{TP + TN+FP+FN} \qquad \text{Εξ. 2.8.4}$$

Για τον υπολογισμό των παραπάνω μετρικών χρησιμοποιήθηκε ο αριθμός των *TP* (“true positives”), των *FP* (“false positives”), των *FN* (“false negatives”), και των *TN* (“true negatives”). *TP* είναι ο αριθμός των ορίων που εντοπίζει σωστά μια μέθοδος κατάτμησης, η υποθετική κατάτμηση δηλαδή, συμφωνεί με την πραγματική (“ground truth”). Από την άλλη, *FP* είναι τα λάθος όρια που βρίσκει μια μέθοδος. *FN* είναι ο αριθμός των ορίων που χάνει μια μέθοδος κατάτμησης και *TN* είναι ο αριθμός των σημείων που δεν θεωρούνται σημεία αλλαγής νοήματος ούτε από την πραγματική κατάτμηση ούτε από την υποθετική κατάτμηση.

Παρατηρήθηκε όμως, πως οι τιμές των *Precision* και *Recall* μπορούν να διαμορφώνονται η μια σε βάρος της άλλης, έτσι οι αλγόριθμοι μπορούν να

ρυθμιστούν με τέτοιο τρόπο ώστε να επιτυγχάνουν υψηλή απόδοση στο *Recall* αλλά ταυτόχρονα υψηλή και στο *Precision.* Επίσης ένα σημαντικό μειονέκτημα είναι το ότι κάθε λανθασμένα εκτιμώμενο όριο μεταξύ τμημάτων «τιμωρείται ισάξια» ανεξάρτητα αν αυτό είναι κοντά ή όχι στο πραγματικό όριο. Αυτό σημαίνει ότι για την πλήρη γνώση της απόδοσης ενός αλγορίθμου χρειαζόμαστε ένα μέτρο που να συνδιάζει αυτές τις δύο ποσότητες, το *Precision* και το *Recall.* Μια τέτοια εναλλακτική λύση είναι το *F-measure.*

$$\begin{aligned} \mathrm{F}_\beta\text{-measure} &= (1+\beta^2)\cdot\frac{\mathrm{Precision}\cdot\mathrm{Recall}}{(\beta^2\cdot\mathrm{Precision})+\mathrm{Recall}} \\ &= \frac{(1+\beta^2)\cdot\mathrm{TP}}{(1+\beta^2)\cdot\mathrm{TP}+\beta^2\cdot\mathrm{FN}+\mathrm{FP}} \end{aligned} \qquad \text{Εξ. 2.8.5}$$

Όπου β μια παράμετρος.

Το πιο γνωστό μέτρο είναι το *F1*, που δίνει ίση βαρύτητα στο *Precision* και στο *Recall* (β=1) και εκφράζει το ποσοστό επιτυχίας ανίχνευσης των ορίων μεταξύ των τμημάτων.

$$F_1 = \frac{2\cdot\mathrm{Precision}\cdot\mathrm{Recall}}{\mathrm{Precision}+\mathrm{Recall}} \qquad \text{Εξ. 2.8.6}$$

Ένας άλλος τρόπος για την μέτρηση του αποτελέσματος της κατάτμησης, αφορά την χρησιμοποίηση κριτών. Οι κριτές μετά την ανάγνωση του προς κατάτμηση κειμένου προσδιόριζαν, κατά την κρίση τους, τα όρια μεταξύ των τμημάτων. Επειδή όμως, όπως είναι φυσικό, ο τρόπος αυτός είναι υποκειμενικός, έπρεπε να βρεθούν τρόποι μέτρησης της συμφωνίας μεταξύ των σχολιαστών.

Η Carletta J. [10] εισήγαγε την μετρική *kappa-statistic* η οποία εφαρμόζεται στον τομέα της *ανάλυσης περιεχομένου*, για την μέτρηση της συμφωνίας μεταξύ των σχολιαστών. Ο υπολογισμός της φαίνεται από την παρακάτω σχέση:

$$K = \frac{P(A) - P(E)}{1 - P(E)}$$ Εξ. 2.8.7

όπου *P(A)* το ποσοστό των φορών που οι σχολιαστές συμφωνούν μεταξύ τους και *P(E)* το ποσοστό των φορών που οι σχολιαστές αναμένεται να συμφωνήσουν μεταξύ τους τυχαία. Όταν υπάρχει συνολική συμφωνία το K είναι ένα, ενώ όταν δεν υπάρχει καμία συμφωνία πέρα από αυτή που αναμένονταν τυχαία, το K είναι μηδέν. Αναλυτικότερα, στον τομέα της *ανάλυσης περιεχομένου* όταν η τιμή του μέτρου *kappa-statistic* είναι μεγαλύτερη από 0.8 τότε θεωρούμε ότι έχουμε υψηλή αξιοπιστία ενώ όταν η τιμή του μέτρου *kappa-statistic* είναι μεταξύ 0.67 και 0.8 τότε θεωρούμε ότι έχουμε αβέβαιη αξιοπιστία.Τέλος, όταν η τιμή αυτού του μέτρου είναι μικρότερη από 0.67 θεωρούμε ότι ο σχολιασμός που έγινε από τους διάφορους σχολιαστές δεν είναι αξιόπιστος.

Ένα επιπλέον μέτρο P_k , προτάθηκε από τους Beeferman, Berger και Laffety [13] για την εκτίμηση της απόδοσης κατάτμησης, το οποίο μετρά την ανακρίβεια της κατάτμησης. Στην ουσία, μια υποθετική κατάτμηση η οποία χάνει πολλά (ακόμα και όλα) τα πραγματικά όρια των τμημάτων συγκρίνεται με την πραγματική κατάτμηση ως προς τα σημεία αλλαγής που εντοπίζει σε σχέση με τα πραγματικά όρια τμημάτων. Διαισθητικά, η μετρική P_k υπολογίζει το ποσοστό των κειμένων (αν έχουμε να κάνουμε με ένα text stream) ή αντίστοιχα το ποσοστό των λέξεων, ή προτάσεων ή παραγράφων (αν έχουμε να κάνουμε με ένα κείμενο), που α) είτε λανθασμένα προβλέφτηκαν ότι ανήκουν στο ίδιο τμήμα, ενώ κανονικά ανήκουν σε διαφορετικά τμήματα β) ή λανθασμένα προβλέφτηκαν ότι ανήκουν σε διαφορετικά τμήματα, ενώ πραγματικά ανήκουν στο ίδιο.

Αναλυτικά, δοθέντος ενός κειμένου το οποίο αποτελείται από T προτάσεις, πρώτα ορίζουμε για κάθε s, t =1,2,....T, τις ποσότητες $\delta_{tru}(s,t)$ και $\delta_{hyp}(s,t)$ όπως ακολούθως.

$\delta_{tru}(s,t) =$ 1, αν και μόνο αν οι προτάσεις *s, t* ανήκουν στο ίδιο τμήμα στην πραγματική τμηματοποίηση,

0, αλλιώς

$\delta_{hyp}(s,t) =$ 1, αν και μόνο αν οι προτάσεις *s, t* ανήκουν στο ίδιο τμήμα στην υποθετική τμηματοποίηση,

0, αλλιώς

Έπειτα, εισάγουμε μια κατανομή πιθανότητας D(s,t), {s,t = 1,2,…,T} στο σύνολο των τυχαία επιλεγμένων ζευγών (s,t) και ορίζουμε την P_k.

$$P_k = \sum_{1 \le s \le t \le T} D(s,t) \cdot 1(\delta_{tru}(s,t) \neq \delta_{hyp}(s,t)) \qquad \text{Εξ. 2.8.8}$$

όπου 1(a≠b) ισούται με 1 όταν a≠b και 0 αλλιώς. Παρατηρούμε ότι η P_k εξαρτάται από την πιθανότητα D(s,t) επιλογής δύο συγκεκριμένων προτάσεων και ισούται με μηδέν (P_k=0) μόνο όταν η πραγματική και η υποθετική κατάτμηση είναι πανομοιότυπες. Συνεπώς μικρές τιμές του P_k υποδεικνύουν υψηλό ποσοστό ακρίβειας κατάτμησης.

Έτσι λοιπόν αν υποτεθεί ότι μας δίνεται ένα κείμενο, η πραγματική καθώς και η υποθετική κατάτμηση αυτού, τότε θα λέγαμε ότι το P_k είναι η πιθανότητα δύο τυχαία επιλεγμένα δείγματα, από το σύνολο των προτάσεων του κειμένου, να είναι λανθασμένα κατατμημένα δηλαδή, η υποθετική κατάτμηση εσφαλμένα υποστηρίζει είτε ότι τα δύο δείγματα ανήκουν στο ίδιο τμήμα, ή ότι ανήκουν σε διαφορετικά τμήματα. Τέλος, αποδεικνύεται ότι το P_k τιμωρεί σφάλματα κοντά στα όρια, λιγότερο απ'οτι σφάλματα μακρύτερα από τα όρια των τμημάτων.

Ένα άλλο μέτρο, το *WindowDiff*, προτάθηκε επίσης από τους Pevzner και Hearst [14] το οποίο χρησιμοποιεί ένα κινούμενο παράθυρο σαρώνοντας το σύνολο των δεδομένων ενός text stream και "τιμωρεί" μια υποθετική κατάτμηση όταν ο αριθμός των ορίων που εντοπίζει μέσα στο παράθυρο δεν ταιριάζει με τον πραγματικό αριθμό των ορίων, γι' αυτό το παράθυρο του text stream. Η μετρική *WindowDiff* δουλεύει ως εξής: Για κάθε σύνολο κειμένων που βρίσκονται μέσα σε ένα παράθυρο μεγέθους *k*, γίνεται σύγκριση μεταξύ του αριθμού των πραγματικών ορίων, που τυχαίνει να πέφτουν μέσα σε αυτό το διάστημα (r_i), όπως αυτά ορίζονται από την πραγματική κατάτμηση, και του αριθμού των ορίων που δίνει η υποθετική κατάτμηση (α_i). Μια μέθοδος λοιπόν, "τιμωρείται" αν $r_i \neq \alpha_i$, (το οποίο υπολογίζεται ως $| r_i - \alpha_i | > 0$). Μετέπειτα όλες οι "ασυμφωνίες" μεταξύ της υποθετικής και της πραγματικής κατάτμησης προσθέτονται, στη συνέχεια η τιμή κανονικοποιείται και η μετρική παίρνει τιμές μεταξύ του 0 και του 1. Συγκεκριμένα, η μετρική παίρνει την τιμή 0 αν όλα τα όρια έχουν εντοπιστεί σωστά και 1 αν από την εκάστοτε μέθοδο έχουν εντοπιστεί εντελώς διαφορετικά σημεία αλλαγής νοήματος σε σχέση με την πραγματική κατάτμηση.

Πιο συγκεκριμένα,

$$WindowDiff(ref, hyp) = \frac{1}{N-k} \sum_{i=1}^{N-k} (|b(ref_i, ref_{i+k}) - b(hyp_i, hyp_{i+k})| > 0) \qquad \text{Εξ. 2.8.9}$$

όπου *b(i,j)* ο αριθμός των ορίων μεταξύ των θέσεων i και j στο κείμενο , *N* ο αριθμός των κειμένων ενός text stream και *k* το μέγεθος του παραθύρου.

Οι Pevzner και Hearst [14] υποστηρίζουν πως παρόλου που η P_k μετρική κερδίζει έδαφος σε ότι αφορά την αξιολόγηση αλγορίθμων κατάτμησης κειμένου, τα προβλήματα που προκύπτουν είναι αρκετά. Συγκεκριμένα, η μετρική "τιμωρεί" πολύ πιο αυστηρά όρια που μια μέθοδος χάνει (*FN*), από όρια που βρίσκει λάθος (*FP*). Επίσης "τιμωρεί" υπερβολικά, σφάλματα κοντά στα όρια ("near misses"), και επηρεάζεται από μεταβολή στην κατανομή του μεγέθους του τμήματος. Από την άλλη πλευρά, η προσέγγιση WindowDiff εξαλείφει την ασυμμετρία μεταξύ των *FP* ("false positive") και *FN* ("false negative") "κυρώσεων", που παρουσιάζονται στην P_k

μετρική. Επίσης εντοπίζει τα FP και FN μέσα σε τμήματα μεγέθους μικρότερου από k.

ΚΕΦΑΛΑΙΟ 3. ΑΛΓΟΡΙΘΜΟΙ TEXTTILING ΚΑΙ TSF

3.1. Αλγόριθμος TextTiling

Ο αλγόριθμος TextTiling όπως αυτός παρουσιάστηκε από την Hearst το 1997, [2] [3] [4] [5], αποτελεί μια μέθοδο γραμμικής κατάτμησης, σύμφωνα με την οποία αρχεία κειμένων χωρίζονται σε μεγάλες παραγραφικές μονάδες. Η βαθύτερη υπόθεση αυτού του αλγορίθμου είναι ότι υπάρχει μεγάλη πιθανότητα, λέξεις που έχουν να κάνουν με ένα συγκεκριμένο θέμα να επαναληφθούν όποτε γίνει ξανά αναφορά στο θέμα. Άλλη μια σημαντική υπόθεση είναι ότι όταν εμφανίζεται ένα καινούριο θέμα, η επιλογή του λεξιλογίου θα αλλάξει και θα διατηρηθεί αμετάβλητη μεταξύ των ορίων που περικλείουν το συγκεκριμένο θέμα, μέχρι την επόμενη αλλαγή θέματος. Συνεπώς, η προσέγγιση αυτή χρησιμοποιεί την επανάληψη των όρων που περιέχουν τα κείμενα ως μηχανισμό λεξιλογικής συνοχής, για να εντοπίσει τα σημεία στα οποία αλλάζει το νόημα. Η μοντελοποίηση των όρων γίνεται στον διανυσματικό χώρο (Vector Space Modeling) χρησιμοποιώντας την μετρική tf.idf ως μέτρο ανάκτησης πληροφοριών. Η Hearst χρησιμοποίησε την τεχνική του κινούμενου παραθύρου "sliding window" και υπολόγιζε την ομοιότητα ενός μπλοκ με τα γειτονικά του. Για την εύρεση της ομοιότητας μεταξύ των τμημάτων χρησιμοποιείται η ομοιότητα συνημιτόνου ("cosine similarity").

3.1.1. Δομή και λειτουργία του αλγορίθμου TextTiling

Ο αλγόριθμος TextTiling περιλαμβάνει τρία κύρια ζητήματα: α) τον διαχωρισμό του κειμένου σε λεκτικές μονάδες β) τον καθορισμό λεκτικής ομοιότητας και γ) την αναγνώριση σημείων αλλαγής θέματος.

Κατά το πρώτο στάδιο όπου το κείμενο χωρίζεται σε λεκτικές μονάδες, γίνεται κάποιο είδος προεπεξεργασίας, πραγματοποιείται δηλαδή, η αφαίρεση των συχνών λέξεων (stop-list), η αποκοπή καταλήξεων (stemming), η μετατροπή όλων των λέξεων σε λέξεις με πεζούς χαρακτήρες, η κατάτμηση σε τμήματα-ψευδοπροτάσεις (οι οποίες αποτελούνταν κατά μέσο όρο από 20 λέξεις) που συνιστούν ένα μπλόκ. Η Hearst χρησιμοποίησε ως μέγεθος του μπλόκ K τον μέσο όρο των παραγράφων ενός κειμένου (μετρούμενο ως προς τις λέξεις). Σε αυτό το σημείο θα πρέπει να επισημανθεί πως όταν γίνεται λόγος για ένα κείμενο, στο οποίο θέλουμε να βρούμε τα σημεία διαχωρισμού του κειμένου, ως μπλόκ θεωρούμε τις προτάσεις - παραγράφους αυτού. Όμως ο αλγόριθμος αυτός, όπως και άλλοι αλγόριθμοι γραμμικής κατάτμησης, μπορούν να εφαρμοστούν και σε μεγάλα text streams, δηλαδή σε ροές κειμένων διαφορετικών κατηγοριών, όπου στόχος είναι ο καθορισμός των ορίων κάθε τμήματος που ανήκει στην ίδια θεματική ενότητα, και σε αυτή τη περίπτωση ως μπλόκ ορίζεται το κάθε κείμενο.

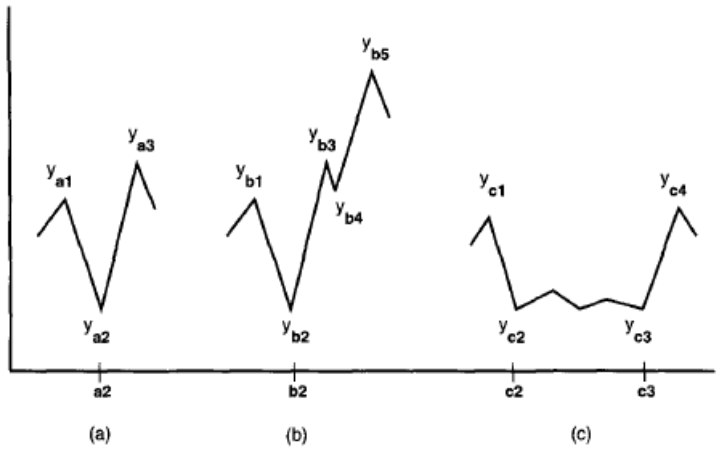
Στο δεύτερο στάδιο γίνεται ο υπολογισμός του μέτρου ομοιότητας μεταξύ κάθε δύο συνεχόμενων μπλόκ με τη βοήθεια της συνημιτονοειδούς ομοιότητας. Σημειώνεται πως κάθε μπλόκ παριστάνεται ως ένα διάνυσμα και κάθε συνιστώσα του διανύσματος είναι η συχνότητα εμφάνισης μιας λέξης, σε κανονικοποιημένη μορφή, από το επιλεγμένο λεξιλόγιο. Η συνημιτονοειδής ομοιότητα μεταξύ δύο συνεχόμενων μπλόκ, υπολογίζεται από την παρακάτω σχέση:

$$Score(b_1,b_2) = \frac{\sum_t w_{t,b_1} w_{t,b_2}}{\sqrt{\sum_t w_{t,b_1}^2 \sum_t w_{t,b_2}^2}} \qquad \text{Εξ. 3.1.1}$$

όπου b1, b2, τα δύο μπλόκ κειμένου σε μορφή διανύσματος, t το εύρος όλων των όρων από το επιλεγμένο λεξιλόγιο, το οποίο έχει καθοριστεί κατά το στάδιο της

διάσπασης του κειμένου σε λεκτικές μονάδες, $w_{t,b}$ είναι το βάρος που έχει δοθεί στον όρο t του μπλόκ b.Τα βάρη είναι οι κανονικοποιημένες συχνότητες εμφάνισης των λέξεων μέσα στο εκάστοτε μπλόκ και παίρνουν τιμές ανάμεσα από το μηδέν και το ένα.

Στο τρίτο στάδιο όπως φαίνεται και παρακάτω στο Σχήμα (3.1.1), γίνεται ο εντοπισμός των σημείων αλλαγής θέματος αυτή η μέθοδος χρησιμοποιεί ένα άλλο παράθυρο, συνήθως μικρότερο, για να αποφασίσει το μικρότερο μέγεθος ενός έγκυρου τμήματος. Στόχος είναι η αναγνώριση "κοιλάδων" δηλαδή σημείων όπου πέφτει απότομα η ομοιότητα, γι'αυτόν τον λόγο, για κάθε τέτοιο σημείο a_2 υπολογίζεται η πτώση $(y_{a1} - y_{a2}) + (y_{a3} - y_{a2})$ από τις δύο γειτονικές κορυφές, η οποία ονομάζεται depth score. Έπειτα, επιλέγονται εκείνα τα σημεία, η τιμή των οποίων υπερβαίνει ένα κατώφλι π.χ τη μέση τιμή πλήν την τυπική απόκλιση. Τέλος, τα depth scores ταξινομούνται κατά φθίνουσα σειρά για να γίνει η τελική επιλογή των ορίων: όσο πιο μεγάλη η τιμή του depth score τόσο μεγαλύτερη η πιθανότητα για διαχωρισμό σε εκείνο το σημείο.



Σχήμα 3.1.1 Υπολογισμός των depth scores σε τρεις διαφορετικές περιπτώσεις.

Αυτή η μέθοδος όμως παρουσιάζει ένα πιθανό πρόβλημα, στο παραπάνω σχήμα 3.1.1(b) βλέπουμε μια μικρή "κοιλάδα" στο σημείο b_4, η οποία "παρεμβαίνει" στον υπολογισμό της ομοιότητας του σημείου b2: Σε αυτή την περίπτωση ο αλγόριθμος χρησιμοποιεί την *εξομάλυνση* ("*smoothing"*), η οποία περιγράφεται παρακάτω, για να εξαλείψει τέτοιες τυχόν μικρές παρεμβολές.

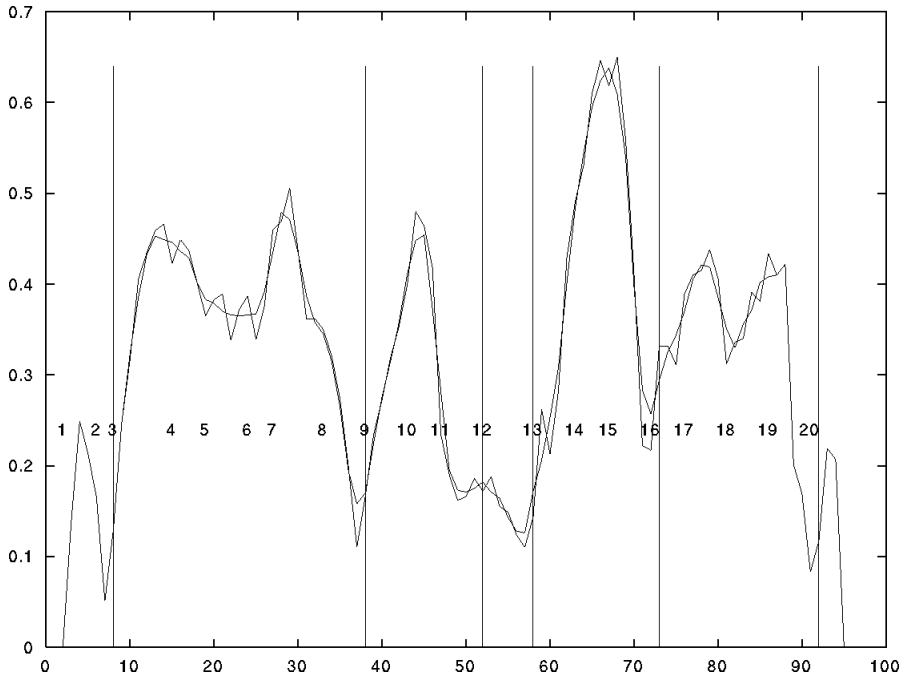
Επιπλέον, επειδή η απόσταση ανάμεσα στο y_{b3} και στο y_{b4} είναι μικρή, σε τέτοιες περιπτώσεις αυτό το σημείο (το y_{b4}) δεν θεωρείται σημείο αλλαγής νοήματος, σε σχέση με το b_2 που έχει μεγάλη απόσταση από τις γειτονικές κορυφές του, τόσο από δεξιά όσο και από αριστερά. Έτσι λοιπόν γεννιέται η ανάγκη, να ληφθεί υπόψη το μήκος και από τις δύο πλευρές της "κοιλάδας", αφού μια "κοιλάδα" που έχει μεγάλη απόσταση και από τις δύο κορυφές της, δείχνει πως η ομοιότητα των μπλόκ σε επίπεδο λεξιλογίου, μειώνεται από τα αριστερά , και αυξάνεται αντίστοιχα από τα δεξιά, καθιστώντας την κοιλάδα-σημείο, ως το ιδανικό χώρισμα που σηματοδοτεί μια μεγάλη αλλαγή στο νόημα. Στο Σχήμα 3.1.1(c) φαίνεται ένα άλλο πρόβλημα, συγκεκριμένα δύο μεγάλες κορυφές πλαισιώνουν μια μεγάλη και επίπεδη "κοιλάδα" και το ερώτημα είναι ποια από τα δύο πιθανά "κοψίματα" c_2, c_3, ή και τα δύο, αποτελούν σημεία αλλαγής θέματος. Σε αυτή τη περίπτωση ο αλγόριθμος κάνει μια τυχαία επιλογή αν πρόκειται για μια μικρή σε έκταση κοιλάδα, διαφορετικά για μια μεγαλύτερη, επιλέγει και τα δύο σημεία ως "κοψίματα". Το παραπάνω φαινόμενο παρατηρείται σε περιπτώσεις όπου το λεξιλόγιο αλλάζει σταδιακά και οι διαφορές στο συγκεκριμένο τμήμα του κειμένου είναι μεγάλες, η ομοιότητα δηλαδή μεταξύ των μπλόκ είναι μικρή.

Η διαδικασία της εξομάλυνσης έχει ως εξής: Για κάθε τιμή ομοιότητας (score), μεταξύ δύο συνεχόμενων μπλόκ, υπολογίζεται εκ νέου η τιμή της με τη βοήθεια ενός αριθμού s. Συγκεκριμένα η νέα ομοιότητα (score) ορίζεται ως η μέση τιμή του παλιού score, των s/2 score αριστερά από αυτό και των s/2 score δεξιά από αυτό και η διαδικασία επαναλαμβάνεται για όλα τα ζεύγη από συνεχόμενα τμήματα.

Επίσης, μπορεί να γίνει επιπλέον εξομάλυνση της γραφικής παράστασης στο τελικό στάδιο, αφού θα έχει γίνει ο τελικός καθορισμός των πιθανών ορίων, για να αποφευχθεί το φαινόμενο της ύπαρξης ορίων μεταξύ κοντινών μπλόκ του κειμένου.

Στο παρακάτω σχήμα (Σχήμα 3.1.2) παρουσιάζεται μια γραφική παράσταση με και χωρίς εξομάλυνση, η οποία απεικονίζει τα πιθανά σημεία αλλαγής του νοήματος ενός κειμένου. Οι εσωτερικοί αριθμοί υποδηλώνουν τους αριθμούς των παραγράφων, ο άξονας x δείχνει την αλληλουχία των προτάσεων που αποτελούν ένα μπλόκ, (δηλαδή το μέγεθος μπλόκ K=10), ο άξονας y δείχνει την ομοιότητα μεταξύ των μπλόκ. Οι

κατακόρυφες γραμμές απεικονίζουν τα τελικά όρια (“κοψίματα”) που επιλέχτηκαν από τον αλγόριθμο. Για παράδειγμα η πρώτη κάθετη γραμμή από τα αριστερά αναπαριστά ένα “κόψιμο” μετά την παράγραφο 3.



Σχήμα 3.1.2 Αποτελέσματα του αλγορίθμου TextTiling.

3.1.2. Συμπέρασμα

Ο αλγόριθμος TextTiling, παρουσιάζει καλή απόδοση, όμως έχει σχεδιαστεί για την εύρεση ομοιογενών τμημάτων μέσα σε ένα μόνο κείμενο, καθιστώντας δύσκολη την σύγκρισή του με μεθόδους που προσπαθούν να εντοπίσουν “κοψίματα” σε συνεχόμενα διαφορετικά έγγραφα. Επίσης έχει αρκετά μειονεκτήματα. Πρώτον, μπορεί να δημιουργούνται πολλά παραπάνω “κοψίματα”, αυτό συμβαίνει για παράδειγμα, σε περιπτώσεις όπου μια μικρή παράγραφός ή φράση παρεμβαίνει σε ένα τμήμα του κειμένου το οποίο παρουσιάζει ισχυρή συνεκτικότητα, με αποτέλεσμα να αναγνωρίζεται από τον αλγόριθμο ως αλλαγή στο νόημα και επομένως ως διαφορετικό τμήμα. Δεύτερον, ο συγκεκριμένος αλγόριθμος, εξαρτάται από ένα κατώφλι, προκειμένου να αποφασιστεί εάν υπάρχει συνεκτικότητα μεταξύ δύο συνεχόμενων μπλόκ και κατά συνέπεια να επιλεχθούν τα τελικά όρια του κειμένου.

Κάτι τέτοιο όμως είναι αρκετά δύσκολο, πολύπλοκο, και έχει να κάνει με τα χαρακτηριστικά του εγγράφου κάθε φορά.

3.2. Αλγόριθμος Tsf

Ο TSF είναι ένας αλγόριθμος γραμμικής κατάτμησης , ο οποίος προτάθηκε από τους Roman Kern και Michael Granitzer το 2009 [7]. Όπως ακριβώς ο αλγόριθμος TextTiling, έτσι και αυτός, στοχεύει στον διαχωρισμό ομοιογενών τμημάτων στα κατάλληλα σημεία, με βάση την συχνότητα εμφάνισης λέξεων. Αναλυτικότερα, στηρίζεται στην παραδοχή ότι λέξεις που επαναλαμβάνονται συχνά, αποτελούν ισχυρή ένδειξη για την παρουσία του ίδιου θέματος στο εύρος των κειμένων που εξετάζονται. Και εδώ χρησιμοποιείται το μοντέλο διανύσματος χώρου , όπου κάθε πρόταση ή κείμενο ανάλογα με την περίπτωση που εξετάζεται, αναπαρίσταται με τη μορφή διανύσματος με κάθε συνιστώσα του διανύσματος να είναι η συχνότητα εμφάνισης μιας λέξης. Επιπλέον, χρησιμοποιείται η τεχνική του κινούμενου παραθύρου “sliding window”, και ως μέτρο υπολογισμού της ομοιότητας ενός μπλόκ με τα γειτονικά του, χρησιμοποιείται η cosine similarity.

3.2.1. Δομή και λειτουργία του αλγορίθμου Tsf

Ο αλγόριθμος TSF περιλαμβάνει τρία κύρια σημεία: α) Προεπεξεργασία β) Υπολογισμός του μέτρου ομοιότητας μεταξύ των μπλοκ γ) Καθορισμός των τελικών σημείων αλλαγής του νοήματος.

Στο πρώτο στάδιο γίνεται η προεπεξεργασία, δηλαδή η αφαίρεση των συχνών λέξεων (stop-list) , η αποκοπή καταλήξεων και η επιστροφή αυτών των λέξεων στην ρίζα τους (stemming), καθώς επίσης και η τελική επιλογή των λέξεων που θα αποτελούν το λεξιλόγιο για το συγκεκριμένο data set. Στη συνέχεια γίνεται ο καθορισμός των μπλόκ. Όπως αναφέρθηκε και στην προηγούμενη ενότητα, ανάλογα

με τη φύση του προβλήματος, αν στόχος μας είναι ο εντοπισμός των σημείων αλλαγής νοήματος σε ένα κείμενο, τότε ως μπλόκ ορίζεται μια πρόταση-παράγραφος, ενώ αντίστοιχα, όταν έχουμε να κάνουμε με ένα text stream που αποτελείται από κείμενα διαφόρων κατηγοριών, ως μπλόκ ορίζεται κάθε μεμονωμένο κείμενο. Έτσι λοιπόν, ένα κείμενο στην τελική του μορφή, αναπαρίσταται με τη μορφή διανύσματος, όπου κάθε συνιστώσα αυτού, είναι η συχνότητα εμφάνισης μιας λέξης από το επιλεγμένο λεξιλόγιο. Θα πρέπει επίσης να σημειωθεί, πως αμέσως μετά την δημιουργία του τελικού πίνακα με τα επιλεγμένα κείμενα στην μορφή διανύσματος, γίνεται η κανονικοποίηση των όρων με την χρήση της μετρικής tf idf, έτσι ώστε να αποδοθεί σε κάθε έγγραφο ίση σημασία – αναλογία σε σχέση με τα υπόλοιπα, δεδομένου ότι ένα έγγραφο μπορεί να είναι πολύ μεγάλο σε μέγεθος και άλλο να είναι πολύ μικρό.

Στο δεύτερο στάδιο γίνεται ο υπολογισμός του μέτρου ομοιότητας μεταξύ κάθε δύο συνεχόμενων μπλόκ με τη βοήθεια της συνημιτονοειδούς ομοιότητας.

Στο τρίτο στάδιο γίνεται ο εντοπισμός των σημείων αλλαγής θέματος. Και εδώ ακολουθείται, όπως και στον αλγόριθμο TextTiling, η τεχνική του κινούμενου παραθύρου. Η διαδικασία έχει ως εξής: Για κάθε θέση i, δημιουργούνται δύο γειτονικά μπλόκ: ένα μπλόκ B^{pre}_i, περιέχει έναν αριθμό κειμένων που προηγούνται από τη συγκεκριμένη θέση που εξετάζουμε κάθε φορά, βρίσκονται δηλαδή στα αριστερά του κειμένου και ένα άλλο μπλόκ B^{post}_i, ακολουθεί τη συγκεκριμένη θέση και εντοπίζεται στα δεξιά του. Το μέγεθος του μπλόκ είναι μια παράμετρος που πρέπει να δίνεται από τον χρήστη και πρέπει να αντικατοπτρίζει το μικρότερο μέγεθος του προς εξέταση τμήματος κειμένων.

Έτσι λοιπόν, για κάθε ένα από τα δύο μπλόκ κειμένου, από τα δεξιά και από τα αριστερά κάθε θέσης του κινούμενου παραθύρου, υπολογίζεται η ομοιότητα όλων των κειμένων ανά ζεύγος και στην συνέχεια, η μέση τιμή αυτών των ομοιοτήτων χρησιμοποιείται για να υπολογιστεί η μέση *εσωτερική ομοιότητα* αυτής της θέσης. Αυτή η εσωτερική ομοιότητα μπορεί να ερμηνευτεί ως ο βαθμός ομοιότητας των κειμένων που περιβάλλουν την συγκεκριμένη θέση του παραθύρου, και υπολογίζεται από την παρακάτω σχέση:

$$sim_i^{inner} = \frac{\mu(B_i^{pre}, B_i^{pre}) + \mu(B_i^{post}, B_i^{post})}{2} \quad \text{Εξ. 3.2.1}$$

όπου B^{pre}_i , η ομοιότητα των κειμένων ανά ζεύγος στα αριστερά της εκάστοτε θέσης του παραθύρου, B^{post}_i η ομοιότητα των κειμένων ανά ζεύγος στα δεξιά και όπου μ, ο μέσος όρος των ομοιοτήτων ανά ζεύγος, για κάθε μπλόκ.

Έπειτα, υπολογίζονται οι ομοιότητες μεταξύ κειμένων από το ένα μπλόκ, με τις ομοιότητες από το άλλο μπλόκ. Πάλι, παίρνουμε την μέση τιμή αυτών των τιμών για να υπολογίσουμε την *εξωτερική ομοιότητα*, η οποία εξετάζει το κατά πόσο είναι όμοιο το ένα μπλόκ με το άλλο.

Η εξωτερική ομοιότητα sim_i^{outer} δίνεται από την παρακάτω σχέση:

$$sim_i^{outer} = \mu(B_i^{pre}, B_i^{post}) \quad \text{Εξ. 3.2.2}$$

Τέλος, αυτές οι δύο ομοιότητες η sim_i^{inner} και η sim_i^{outer} συνθέτουν ένα μέτρο για τον υπολογισμό της "ανομοιότητας" των δύο μπλόκ που περιβάλλουν την εκάστοτε θέση του παραθύρου, που αποτελεί και υποψήφιο "χώρισμα" του text stream.

Η ανομοιότητα αυτή ονομάζεται dissimilarity και δίνεται από την παρακάτω σχέση:

$$dissimilarity_i = \frac{sim_i^{inner} - sim_i^{outer}}{sim_i^{inner}} \quad \text{Εξ. 3.2.3}$$

Η τιμή της dissimilarity για κάθε πιθανό χώρισμα, είναι θετική, αν η μέση ομοιότητα των κειμένων και των δύο μπλόκ είναι μικρότερη από τις μέσες ομοιότητες κάθε μπλόκ. Αυτή η περίπτωση αποτελεί ισχυρή ένδειξη της ύπαρξης

σημείου αλλαγής της θεματικής ενότητας. Με την ίδια λογική, η dissimilarity σε ένα σημείο είναι αρνητική, αν η μέση ομοιότητα των κειμένων και των δύο μπλόκ είναι μεγαλύτερη από τις μέσες ομοιότητες κάθε μπλόκ ξεχωριστά, αυτό αυτόματα συνεπάγεται πως πρόκειται για κείμενα που ανήκουν στην ίδια θεματική ενότητα, άρα δεν έχουμε διαχωρισμό σε εκείνο το σημείο. Η μέγιστή τιμή 1 επιτυγχάνεται αν η εξωτερική ομοιότητα είναι 0, το οποίο μπορεί μόνο να συμβεί, αν τα μπλόκ δεν έχουν καμία κοινή λέξη, συνεπώς, ένα τέτοιο σημείο θεωρείται ως όριο (“κόψιμο”) για τον αλγόριθμο.

Στην συνέχεια, κάθε dissimilarity μπορεί να συγκριθεί με ένα κατώφλι, που δίνεται από τον χρήστη οι τιμές που ξεπερνούν την τιμή που έχει το κατώφλι θεωρούνται “κοψίματα”, οι άλλες αγνοούνται. Κατόπιν, οι dissimilarities ταξινομούνται κατά φθίνουσα σειρά, για να γίνει η τελική επιλογή των ορίων, όσο πιο μεγάλη η τιμή της dissimilarity τόσο μεγαλύτερη η πιθανότητα για διαχωρισμό σε εκείνο το σημείο.

Τέλος, για να αποφευχθεί το φαινόμενο της ύπαρξης συχνών ορίων μεταξύ κοντινών σημείων υψηλής dissimilarity, ένα υποψήφιο όριο (με την υψηλότερη dissimilarity) επιλέγεται ως τελικό όριο και τα υπόλοιπα σημεία που βρίσκονται σε σχετικά μικρό εύρος με αυτό, αγνοούνται. Σε αυτό το σημείο πρέπει να αναφερθεί πως η τελική επιλογή του αριθμού των ορίων μεταξύ των τμημάτων γίνεται χειροκίνητα.

3.2.2. Συμπέρασμα

Ο αλγόριθμος Tsf παρουσιάζει σε γενικές γραμμές πολύ καλή απόδοση, εξαρτάται βέβαια και από το προς εξέταση σύνολο δεδομένων που έχουμε στην διάθεσή μας κάθε φορά, παρουσιάζει όμως και κάποιες δυσκολίες. Σύμφωνα με τα προηγούμενα, το μέγεθος του μπλόκ είναι η πρώτη παράμετρος που δίνεται από τον χρήστη , σύμφωνα με την οποία, καθορίζεται το ελάχιστο μέγεθος του παραθύρου για τον υπολογισμό των dissimilarities. Δεύτερη παράμετρος που απαιτεί η συγκεκριμένη

μέθοδος, είναι ένα κατώφλι, ο καθορισμός του οποίου είναι δύσκολος, ώστε να αποφασιστεί εν τέλει αν δυο συνεχόμενα κείμενα σχετίζονται.

Στα πλαίσια της παρούσας διατριβής, εξετάστηκε η περίπτωση όπου ο αριθμός των segments ή καλύτερα ο αριθμός των σημείων αλλαγής θέματος σε ροές κειμένων (“text streams”), δίνεται από τον χρήστη, έτσι ώστε να γίνει εκτίμηση για το αν ο αλγόριθμος συμφωνεί με την ανθρώπινη κρίση. Τα αποτελέσματα έδειξαν πως ο αλγόριθμος είναι ιδιαίτερα αποτελεσματικός και μπορεί να εντοπίσει με μεγάλη ακρίβεια τα ίδια σημεία, όπως θα έκρινε ένας αναγνώστης των κειμένων. Έτσι λοιπόν, λαμβάνοντας υπόψη τα πολύ καλά αποτελέσματα του αλγορίθμου TSF όταν είναι γνωστός ο αριθμός των ορίων, σε μεγάλα text stream, κρίναμε πως θα ήταν ιδιαίτερα χρήσιμο αν υπήρχε κάποιος αυτόματος τρόπος που θα υποδείκνυε στον αλγόριθμο Tsf τον αριθμό των ορίων και στη συνέχεια να γινόταν η εκτίμηση της απόδοσής του ως προς τα πραγματικά σημεία αλλαγής του νοήματος. Γι ’αυτόν ακριβώς τον λόγο επιλέξαμε να χρησιμοποιήσουμε στη μέθοδο που αναπτύξαμε, το στατιστικό κριτήριο Dip - dist. Η ανάλυση της προτεινόμενης μεθοδολογίας περιγράφεται στην επόμενη ενότητα.

ΚΕΦΑΛΑΙΟ 4. ΜΕΘΟΔΟΙ ΠΟΥ ΒΑΣΙΖΟΝΤΑΙ ΣΤΟ DIP – DIST ΚΡΙΤΗΡΙΟ

4.1. Στατιστικό κριτήριο Dip – dist

Το στατιστικό κριτήριο Dip – dist προτάθηκε από τους Καλογεράτο και Λύκα το 2012 [16], στην περίπτωση της ομαδοποίησης δεδομένων (cluster analysis), για την εκτίμηση της δομής των ομάδων ενός συνόλου δεδομένων ως προς την ομοιογένειά τους. Στην ουσία το στατιστικό κριτήριο Dip - dist, ελέγχει εάν ένα σύνολο δεδομένων αποτελεί μια μοναδική ομάδα ή πρέπει να γίνει διαχωρισμός του σε δύο ή περισσότερα μέρη. Αυτό όμως είναι το ζητούμενο και στην περίπτωση της κατάτμησης κειμένου "text segmentation" που μελετάμε στα πλαίσια της παρούσας διατριβής, γι 'αυτό ακριβώς, αποφασίσαμε να χρησιμοποιήσουμε αυτό το κριτήριο στην μέθοδο που αναπτύξαμε. Εν ολίγοις, το κριτήριο Dip - dist αναλαμβάνει τον ρόλο της αναγνώρισης των σημείων αλλαγής θεματικής ενότητας σε ένα text stream και του διαχωρισμού κειμένων σε τμήματα που ανήκουν στην ίδια κατηγορία και σε τμήματα που ανήκουν σε διαφορετικές κατηγορίες.

Το στατιστικό κριτήριο Dip - dist πετυχαίνει την αναγνώριση ομοιογενών ή ετερογενών τμημάτων, εξετάζοντας αν η κατανομή της πυκνότητας των δεδομένων είναι *μονοτροπική (με μια κορυφή) ή αλλιώς unimodal*. Πιο συγκεκριμένα, κοιτάζοντας το ιστόγραμμα των αποστάσεων ή της ομοιότητας κάθε σημείου από τα

υπόλοιπα σημεία, αποφασίζει για το αν υπάρχει μονοτροπική κατανομή. Μονοτροπικές κατανομές είναι για παράδειγμα οι student-t, η Cauchy, η σφαιρική Γκαουσιανή, η ελλειπτική που είναι μια περίπτωση Γκαουσιανής, η ομοιόμορφη, η οποία αποτελεί ακραίας μορφής unimodal κατανομής. Αναλυτικότερα, ως unimodal ορίζεται μια συνάρτηση κατανομής F(t) η οποία παρουσιάζει κορυφή στην περιοχή της $s_m = \{(t_L, t_U): t_{L\leq} t_U\}$, εάν είναι κυρτή στην περιοχή $s_{L(-\infty}$, $t_{L)}$, σταθερή στην περιοχή $[t_L, t_U]$ και κοίλη στην περιοχή $s_U = [t_U, \infty)$.

Το Dip - dist κριτήριο χρησιμοποιεί ένα στατιστικό τέστ που ονομάζεται Hartigan's dip test [17] για να προχωρήσει στον έλεγχο για μονοτροπικότητα κάθε ομάδας ενός συνόλου δεδομένων. Το Hartigan's dip test προτιμήθηκε από άλλα στατιστικά κριτήρια όπως τη μέθοδο του Silverman [18], λόγω της απλότητας και της αποτελεσματικότητας του. Επίσης έχει αποδειχθεί ιδιαίτερα ισχυρό κυρίως σε μονοδιάστατα δεδομένα.

4.2. Αλγόριθμος SegmentDip

Ο αλγόριθμος SegmentDip αναπτύχθηκε από τους Χασάνη, Ιωαννίδη και Λύκα το 2014 [15] και εφαρμόστηκε σε ακολουθίες βίντεο. Η τεχνική στην οποία βασίζεται είναι αυτή του κυλιόμενου παραθύρου "sliding window", την οποία επίσης χρησιμοποιούν και άλλοι αλγόριθμοι κατάτμησης κειμένου όπως αναφέρθηκε στην προηγούμενη ενότητα, καθώς επίσης και η χρήση του κριτηρίου Dip - dist. Στα πλαίσια της παρούσας διατριβής η εν λόγω μέθοδος εφαρμόστηκε σε ροές κειμένων ("text streams").

Διαισθητικά, ένα κινούμενο παράθυρο μεγέθους w, μετακινείται σαρώνοντας όλη την ακολουθία από κείμενα. Το σύνολο των κειμένων που περιέχονται σε κάθε παράθυρο ελέγχεται για ομοιογένεια ως προς το περιεχόμενο, χρησιμοποιώντας το κριτήριο Dip - dist, το οποίο βασίζεται στο Hartigan's dip test [17]. Συνεπώς, κάθε παράθυρο χαρακτηρίζεται ως unimodal ("μονοτροπικό") ή multimodal ("πολυτροπικό") και η ακολουθία των κειμένων κάθε πολυτροπικού παραθύρου

χωρίζεται σε δύο (πιθανώς μονοτροπικά) τμήματα. Επιπλέον, ένα πολύ σημαντικό σημείο της μεθόδου που εξετάζουμε είναι ότι ο αριθμός των σημείων διαχωρισμού της ακολουθίας κειμένων υπολογίζεται αυτόματα, χωρίς να απαιτείται να καθοριστεί εκ των προτέρων.

4.2.1. Δομή και λειτουργία αλγορίθμου SegmentDip

Η λειτουργία του αλγορίθμου SegmentDip έχει ως εξής: Αρχικά, θεωρούμε ένα κινούμενο παράθυρο μεγέθους w, που μετακινείται μέχρι να καλύψει όλο το φάσμα των κειμένων του "text stream". Για κάθε τέτοιο παράθυρο, το κριτήριο dip θεωρεί κάθε σημείο μιας ομάδας του παραθύρου ως έναν θεατή ("*viewer"*). Στην περίπτωση που εξετάζουμε εμείς σε αυτήν την διατριβή, ένα σημείο είναι ένα κείμενο από το σύνολο δεδομένων που έχουμε στην διάθεσή μας. Έτσι λοιπόν, αρχικά, δημιουργείται ένας πίνακας wxw, κάθε φορά που μετακινείται το παράθυρο. Αυτός ο πίνακας περιέχει την ομοιότητα κάθε κειμένου στο παράθυρο με τα υπόλοιπα κείμενα του παραθύρου. Στη συνέχεια για κάθε γραμμή, για κάθε διάνυσμα ομοιοτήτων του κειμένου με τα υπόλοιπα κείμενα του παραθύρου δηλαδή, εφαρμόζεται το Hartigan's dip test [17]. Αυτό το test εξετάζει το ιστόγραμμα των τιμών του διανύσματος για να αποφασίσει αν για κάθε τέτοιο διάνυσμα υπάρχει μονοτροπική κατανομή, με άλλα λόγια, η κατανομή των τιμών κάθε διανύσματος, φανερώνει τη δομή της ομάδας που ελέγχεται. Αν τώρα υπάρχει μια κορυφή ("unimodal ομάδα"), υποδηλώνεται η ύπαρξη μιας μόνο ομάδας, αντίστοιχα, εάν βρεθούν δύο κορυφές ("bimodal") τότε γίνεται φανερή η ύπαρξη δύο ξεχωριστών ομάδων, αν βρεθούν περισσότερες από δύο κορυφές (multimodal) τότε μιλάμε για περισσότερες των δύο ξεχωριστές ομάδες.

Για να αποφασιστεί εν τέλει αν μια κατανομή είναι μονοτροπική ή πολυτροπική, για κάθε διάνυσμα ομοιοτήτων, υπολογίζεται μια τιμή dip ("dip-value"), και μια τιμή p ("p-value"), ο υπολογισμός των οποίων, περιγράφεται αναλυτικότερα παρακάτω. Η τιμή dip είναι στην ουσία η απόσταση του ιστογράμματος από τη πλησιέστερη μονοτροπική κατανομή και η τιμή p ($0 \leq p \leq 1$)

είναι ένα στατιστικό κατώφλι, μάλιστα αυτό το κατώφλι καθορίζει αν ένα παράθυρο είναι multimodal ("πολυτροπικό") ή unimodal ("μονοτροπικό").

Σε αυτό το σημείο είναι σημαντικό να αναφερθεί πως υπάρχει εξάρτηση ως προς τη θέση του κάθε θεατή. Αυτό σημαίνει πως θεατές που βρίσκονται στα όρια των ομάδων ανιχνεύουν ευκολότερα την ύπαρξη δύο κορυφών. Μπορεί να απαιτείται η ύπαρξη ενός ποσοστού (π.χ 1% τουλάχιστον) θεατών σε σχέση με το πλήθος των σημείων που εξετάζονται κάθε φορά, οι οποίοι να προτείνουν διαχωρισμό για να θεωρηθεί η ομάδα πολυτροπική (multimodal). Αυτοί οι θεατές καλούνται *θεατές διαχωρισμού ("split viewers").*

Για τον ορισμό της τιμής dip, θεωρούμε δύο φραγμένες συναρτήσεις (κατανομές) F, G. Έστω $\rho(F,G) = \max|F(t) - G(t)|$ η απόσταση μεταξύ των δύο κατανομών F, G και U^* η κλάση όλων των unimodal κατανομών, τότε η τιμή dip μιας συνάρτησης κατανομής F θα δίνεται από τον παρακάτω ορισμό.

$$dip(F) = \min_{G \in U^*} \rho(F, G) \qquad \text{Εξ. 4.2.1}$$

Μια σημαντική ιδιότητα του κριτηρίου dip είναι πως εάν F_w είναι ένα δείγμα από w παρατηρήσεις της κατανομής F, τότε θα ισχύει η ισότητα $\lim_{w\to\infty} dip(F_w) = dip(F)$. Επιπλέον έχει αποδειχθεί από τους J.A Hartigan και P.M Hartigan [17] πως η κλάση των ομοιόμορφων κατανομών U είναι η καταλληλότερη για μηδενική υπόθεση H_0 στο dip test, καθώς οι dip τιμές που εμφανίζει είναι μεγαλύτερες από άλλες unimodal κατανομές.

Καθότι λοιπόν γνωρίζουμε ότι η ακραία μορφή μονοτροπικότητας είναι η ομοιόμορφη κατανομή, χρησιμοποιείται αυτή η κατανομή ως κατανομή αναφοράς για να ελεγχθεί αν η τιμή dip για κάθε διάνυσμα ομοιοτήτων κάθε σημείου με τα υπόλοιπα, είναι καλύτερη από την τιμή dip της ομοιόμορφης κατανομής με βάση κάποια πιθανότητα. Αυτό επιτυγχάνεται κατασκευάζοντας b bootstrap σύνολα από w δείγματα από ομοιόμορφες κατανομές U στο διάστημα [0,1], $\{U^r_w\}$, r=1,…,b. Αξίζει

να σημειωθεί πως ο υπολογισμός των bootstrap αυτών δειγμάτων παίρνει ως είσοδο δύο μεταβλητές, το μέγιστο πλήθος στοιχείων που μπορεί να έχει μια κατανομή, δηλαδή w, καθώς και το πλήθος των κατανομών που θα παραχθούν. Στη συνέχεια, εφαρμόζεται το dip test σε κάθε ένα από αυτά τα b διανύσματα και επιστρέφεται μια τιμή dip, $dip(U_w^r)$, r=1,…,b. Ο υπολογισμός της p-value για την F_w εκφράζει την πιθανότητα η τιμή dip της F_w να είναι μεγαλύτερη από την τιμή dip της U_w^r ενός ομοιόμορφου δείγματος από w στοιχεία, όπως φαίνεται και στον παρακάτω τύπο.

$$P=\#[dip(F_w)\geq dip(U_w^r)]/b,\ r=1,\ldots.,b \qquad \text{Εξ. 4.2.2}$$

Είναι προφανές πως όσο μεγαλύτερο είναι το πλήθος των bootstrap κατανομών που χρησιμοποιούνται τόσο πιο έγκυρη και ακριβής θα είναι η p-value που θα υπολογιστεί. Για την τελική απόφαση για το αν υπάρχει διαχωρισμός σε κάποιο σημείο, χρησιμοποιείται όμως και ένα επίπεδο εμπιστοσύνης *α* που δίνεται ως είσοδος αρχικά στον αλγόριθμο, καθώς επίσης και το ποσοστό των θεατών/κειμένων που αναγνωρίζουν πολυτροπικότητα.

Αναλυτικότερα, οι δύο εναλλακτικές υποθέσεις H_0 και H_1 του dip τεστ που χρησιμοποιούνται για να καθοριστεί αν το δείγμα των δεδομένων που ελέγχεται ακολουθεί ή όχι unimodal κατανομή θα είναι:

- H_0 : Η F_w περιγράφεται από μια unimodal κατανομή. Αυτό ισχύει όταν η p-value είναι μεγαλύτερη από ένα επίπεδο εμπιστοσύνης *α* , p-value > α.

- H_1 : Τα στοιχεία της F_w δεν υποδεικνύουν την ύπαρξη μίας και μόνο κορυφής. Αυτό σημαίνει πως υπάρχουν θεατές διαχωρισμού και κατά συνέπεια η F_w είναι multimodal και υποδηλώνει την ύπαρξη διαχωρισμού.

Εάν το ποσοστό θεατών/κειμένων που αναγνωρίζουν πολυτροπικότητα είναι υψηλότερο από ένα κατώφλι (π.χ 1%), τότε το παράθυρο χαρακτηρίζεται ως πολυτροπικό, διαφορετικά θεωρείται μονοτροπικό.

Αν υποτεθεί πως ένα παράθυρο μεγέθους w έχει χαρακτηριστεί ως πολυτροπικό, η διαδικασία που ακολουθείται για τον καλύτερο διαχωρισμό του σε τμήματα είναι η εξής: Καταρχήν, θεωρούμε πως ένα τμήμα που παρουσιάζει ομοιογένεια, ξεκινάει από το πρώτο μονοτροπικό παράθυρο και συνεχίζει για πολλά συνεχόμενα κείμενα, των οποίων τα αντίστοιχα παράθυρα παραμένουν μονοτροπικά, μέχρι να εμφανιστεί ένα πολυτροπικό παράθυρο. Αυτό το παράθυρο τότε χωρίζεται σε δύο κομμάτια σε ένα σημείο διαχωρισμού *s*, και το κείμενο που βρίσκεται στην θέση *s* θεωρείται ως το τέλος του τμήματος που παρουσιάζει μονοτροπικότητα. Έτσι λοιπόν υπολογίζεται για κάθε υποψήφιο σημείο διαχωρισμού μέσα στο πολυτροπικό παράθυρο, το άθροισμα των διασπορών των δύο μικρότερων παραθύρων που προκύπτουν (δεξιά και αριστερά από το σημείο διαχωρισμού). Ως καλύτερο σημείο *s* θεωρείται αυτό που έχει το ελάχιστο άθροισμα.

Το επόμενο τμήμα που ακολουθεί, δεν σημαίνει απαραίτητα πως ξεκινάει ακριβώς μετά το τέλος του προηγούμενου, δηλαδή ψάχνουμε πάλι για τον εντοπισμό του πρώτου επόμενου μονοτροπικού παραθύρου το οποίο τελειώνει σε κάποιο άλλο αντίστοιχο σημείο *s*.

Τέλος, όπως και σε κάθε αλγόριθμο κατάτμησης κειμένου, για να αποφύγουμε μικρά τμήματα, μπορούμε να τα αγνοήσουμε, για παράδειγμα, τμήματα που περιέχουν πολύ μικρό αριθμό κειμένων (π.χ λιγότερα από πέντε).

4.3. Αλγόριθμος Dip – Tsf

Στα πλαίσια της παρούσας διατριβής υλοποιήσαμε έναν αλγόριθμο τον οποίο ονομάσαμε Dip – Tsf, που εντοπίζει τα σημεία αλλαγής θεματικής ενότητας σε ροές κειμένων (“text streams”), όταν χρησιμοποιείται ο αλγόριθμος TSF, αξιοποιώντας το σημαντικό πλεονέκτημα που εμφανίζει ο αλγόριθμος SegmentDip. Συγκεκριμένα, έπειτα από πειράματά μας, διαπιστώθηκε η πολύ καλή απόδοση του αλγορίθμου Tsf όταν είναι γνωστός ο αριθμός των ορίων, ο οποίος όμως δίνεται χειροκίνητα. Από την άλλη πλευρά, ο αλγόριθμος SegmentDip όπως αναφέρθηκε και στην προηγούμενη

ενότητα, κάνοντας χρήση του στατιστικού κριτηρίου dip, είναι σε θέση να εντοπίζει με αυτόματο τρόπο τον αριθμό των “κοψιμάτων” σε ένα text stream. Έτσι λοιπόν, στη μέθοδο που αναπτύξαμε, συνδιάσαμε το πλεονέκτημα του αλγορίθμου SegmentDip για την αυτόματη εκτίμηση του αριθμού των ορίων με τα πολύ καλά αποτελέσματα του αλγορίθμου Tsf όταν είναι γνωστός ο αριθμός των ορίων. Αναλυτικότερα, σε ένα σύνολο από διαφορετικές κατηγορίες κειμένων ενός text stream, πρώτα εφαρμόζεται ο αλγόριθμος SegmentDip και στη συνέχεια εφαρμόζεται ο αλγόριθμος Tsf, έχοντας γνωστό πλέον τον αριθμό των ορίων μεταξύ των τμημάτων, τον οποίο θα παρέχει ο αλγόριθμος SegmentDip.

ΚΕΦΑΛΑΙΟ 5. ΠΕΙΡΑΜΑΤΙΚΗ ΜΕΛΕΤΗ

5.1. Μεθοδολογία

Η εφαρμογή της προτεινόμενης μεθοδολογίας, δηλαδή του αλγορίθμου Dip – Tsf, έχει ως εξής: Έστω ένα συγκεκριμένο σύνολο δεδομένων D, με γνωστά τα ακόλουθα στοιχεία:

1. Μια ροή κειμένων $x_1, x_2, \ldots\ldots, x_M$.
2. Μια συλλογή από K θεματικές ενότητες $c_1, c_2, \ldots., c_K$.
3. Η πληροφορία της θεματικής ενότητας στην οποία ανήκει το κείμενο x_m (για m=1,2,…..,M, $x_m \in \{ c_1, c_2, \ldots., c_K \}$).

Καθένα από τα κείμενα είναι ένα διάνυσμα λέξεων, δηλαδή για m=1,2,....,M, έχουμε $X_m = [x_{m1}, x_{m2}, \ldots.., x_{mj},\ldots., x_{mJm}]$

όπου m=1,2,….,M, και j=1,2,….,J_m.

Επίσης, J_m είναι ο συνολικός αριθμός των λέξεων οι οποίες εμφανίζονται στο m–στο κείμενο και x_{mj} είναι η j-στη λέξη η οποία εμφανίζεται στο m-στο κείμενο. Το x_{mj} λαμβάνει τιμές από το λεξιλόγιο το οποίο ορίζεται από το διάνυσμα:

$W=[w_{1,}w_{2,}\ldots,w_{Nw}]$

όπου N_w είναι ο συνολικός αριθμός των λέξεων που εμφανίζονται σε όλα τα κείμενα.

Σε αυτό το σημείο είναι σημαντικό να αναφέρουμε ότι, η παράσταση του κειμένου είναι ένα διάνυσμα σταθερού μήκους N_w και ας τοποθετούνται τα περιεχόμενα του m-στου κειμένου στο διάνυσμα $x_{m,}$ με μεταβλητό μήκος J_m . Χρησιμοποιούμε για την αναπαράσταση ενός κειμένου m με βάση τις λέξεις το *Διάνυσμα Συχνοτήτων Λέξεων:*

$f_m = [f_{m1,} f_{m2,} \ldots, f_{mn,}\ldots f_{mNw}]$

όπου (για $n=1,2,\ldots.,N_w$) έχουμε f_{mn} = το πλήθος των φορών που η n-λέξη εμφανίζεται στο m-στο κείμενο με τις τιμές τους σε κανονικοποιημένη μορφή.

Σε κάθε πείραμα υλοποιήθηκαν τα εξής:

Στο πρώτο στάδιο, επιλέγεται τυχαία ο αριθμός των κατηγοριών – θεματικών ενοτήτων, που θα αποτελέσουν το υπό εξέταση text stream, από ένα εύρος C, π.χ $5\leq C\leq 10$.

Έπειτα κατά τρόπο τυχαίο, ορίζεται η αλληλουχία των κατηγοριών του text stream, μεγέθους C, με τέτοιο τρόπο ώστε να εξασφαλίζεται ότι δεν εμφανίζονται δύο ίδιες κατηγορίες διαδοχικά η μια μετά τη άλλη.

Στο επόμενο βήμα, για την επιλεγμένη αλληλουχία, ορίζεται τυχαία ο αριθμός των κειμένων ανά κατηγορία, από ένα εύρος L, π.χ $50\leq L\leq 100$.

Στη συνέχεια, αποφασίζονται τα πραγματικά όρια της κατάτμησης (“ground truth boundaries”). Εφόσον από το προηγούμενο στάδιο έχει καθοριστεί ο αριθμός των εγγράφων κάθε κατηγορίας, εύκολα αναγνωρίζονται τα πραγματικά σημεία αλλαγής κάθε κατηγορίας των εγγράφων. Σε αυτό το σημείο πρέπει να σημειωθεί, πως ο υπολογισμός των πραγματικών ορίων μεταξύ των διαφορετικών τμημάτων του text stream, χρειάζεται για την τελική αξιολόγηση των αλγορίθμων ως προς τα “κοψίματα” που αναγνωρίζουν αυτές οι δύο μέθοδοι.

Στο επόμενο βήμα, εφαρμόζεται ο αλγόριθμος SegmentDip, λαμβάνοντας υπόψη διάφορες παραμέτρους, οι οποίες είναι οι w, voting, p_th, ss: w το μέγεθος του παραθύρου, voting το ποσοστό των θεατών διαχωρισμού (default=0.01), p_th το επίπεδο εμπιστοσύνης (default=0), ss ένα κατώφλι π.χ. 20, σύμφωνα με το οποίο αγνοούνται μικρά τμήματα, δηλαδή “κοψίματα” που η απόστασή τους από το προηγούμενο όριο είναι μικρότερη από τη τιμή που ορίζει το κατώφλι.

Εφαρμόζοντας τον αλγόριθμο SegmentDip στην παραπάνω ακολουθία κειμένων (“text stream”), προκύπτουν τα όρια διαχωρισμού κάθε τμήματος του text stream, ακολουθώντας εν συντομία την εξής διαδικασία: Αρχικά, θεωρούμε ένα κινούμενο παράθυρο μεγέθους w, που μετακινείται μέχρι να καλύψει όλο το φάσμα των κειμένων του “text stream”. Κάθε φορά που μετακινείται το παράθυρο δημιουργείται ένας πίνακας ομοιότητας wxw, ο πίνακας αυτός περιέχει την ομοιότητα κάθε κειμένου - θεατή με τα υπόλοιπα κείμενα στο παράθυρο. Στη συνέχεια για κάθε γραμμή του πίνακα ομοιότητας εφαρμόζεται το Hartigan’s dip test [17]. Αυτό το test κοιτάζει το ιστόγραμμα των στοιχείων της γραμμής για να αποφασίσει αν υπάρχει μονοτροπική κατανομή, ή αντίστοιχα πολυτροπική. Τέλος, με βάση κάποια κριτήρια, τα οποία έχουν αναλυθεί στην προηγούμενη ενότητα αποφασίζεται ο τελικός καθορισμός των ορίων μεταξύ των τμημάτων του “text stream”.

Κατόπιν, αμέσως μετά από κάθε υπολογισμό του αριθμού των “κοψιμάτων” που καταλήγει ο αλγόριθμος SegmentDip, για κάθε ζεύγος τιμών p_th και w, εφαρμόζεται ο αλγόριθμος Tsf. Ο αλγόριθμος Tsf κατά την υλοποίησή του όπως έχει ήδη αναφερθεί και σε προηγούμενη ενότητα, χρησιμοποιεί την τεχνική του κινούμενου παραθύρου και υπολογίζει τις τιμές ανομοιότητας (“dissimilarities”) οι

οποίες ταξινομούνται κατά φθίνουσα σειρά, για να γίνει η τελική επιλογή των ορίων. Σημειώνεται πως όσο πιο μεγάλη η τιμή της dissimilarity για κάποιο σημείο της ακολουθίας, τόσο μεγαλύτερη η πιθανότητα για διαχωρισμό σε εκείνο το σημείο. Με τη διαφορά όμως, σύμφωνα με τον αλγόριθμο Dip – Tsf που παρουσιάζουμε, ότι ο αριθμός των ορίων δεν δίνεται χειροκίνητα, δηλαδή από τον χρήστη, αλλά αντίθετα δίνεται ως είσοδος από τον αλγόριθμο SegmentDip.

5.2. Μέτρο αξιολόγησης αλγορίθμου Dip – Tsf

Στα πλαίσια της παρούσας διατριβής εκτελέσαμε καταρχήν τον αλγόριθμο SegmentDip και εξετάσαμε την απόδοσή του για κάθε ζεύγος τιμών w, p_th, κρατώντας τις τιμές voting, ss αμετάβλητες. Συγκεκριμένα, ως p_th χρησιμοποιήθηκαν οι τιμές 0.01, 0.03, 0.05 και ως w οι τιμές 20, 30, 40 και για κάθε τιμή p_th με κάθε μία από τις κατά σειρά τιμές w τρέξαμε τον αλγόριθμο SegmentDip, σημειώνεται πως οι τιμές voting, ss παρέμεναν αμετάβλητες, δηλαδή ss=20, voting=0.01. Στη συνέχεια, με βάση τον αριθμό των ορίων μεταξύ των τμημάτων του text stream που δίνει ο αλγόριθμος SegmentDip για κάθε ζεύγος τιμών w, p_th, εφαρμόσαμε τον αλγόριθμο Tsf. Τέλος εξετάσαμε τα ποσοστό επιτυχίας ανίχνευσης ορίων των δύο αλγορίθμων, SegmentDip και Tsf.

Αναλυτικότερα, προκειμένου να δούμε κατά πόσο συμφωνούν αυτές οι δύο μέθοδοι με την πραγματική κατάτμηση συγκρίνονται τα αποτελέσματα του αλγορίθμου SegmentDip και του αλγορίθμου Tsf αντίστοιχα, με τα πραγματικά όρια της κατάτμησης (“ground truth boundaries”). Έτσι λοιπόν, για να αποφασιστεί ποια όρια θεωρούνται σωστά και ποια θεωρούνται λάθος, χρησιμοποιούμε μια τιμή ως *ανοχή,* π.χ 20. Σύμφωνα με αυτήν την τιμή, κάθε όριο που βρίσκεται μέσα στο εύρος των ορίων της ground truth όπως αυτό ορίζεται από την *ανοχή*, για παράδειγμα, είναι μεγαλύτερο ή μικρότερο κατά 20 μονάδες, ή ίσο με το όριο της ground truth, θεωρείται σωστό, διαφορετικά θεωρείται λάθος. Σε περίπτωση που βρεθούν δύο ή περισσότερα όρια στο ίδιο εύρος τιμών της ground truth τότε θεωρείται μόνο το ένα σωστό.

Συνεπώς, βασιζόμενοι στην παραπάνω τεχνική, για κάθε έναν από τους δύο αλγορίθμους γίνεται: α) εύρεση του αριθμού των ορίων (TP), που ο εκάστοτε αλγόριθμος εντοπίζει σωστά, δηλαδή υπάρχει συμφωνία με την πραγματική κατάτμηση (“ground truth”), β) εύρεση του αριθμού των ορίων (FP), που ο εκάστοτε αλγόριθμος εντοπίζει λάθος, γ) εύρεση του αριθμού των ορίων (FN), που ο εκάστοτε αλγόριθμος χάνει.

Στο τελικό στάδιο, γίνεται η αξιολόγηση των αποτελεσμάτων που παρουσιάζουν οι δύο αλγόριθμοι ξεχωριστά. Ως μέτρο αξιολόγησης της κατάτμησης για κάθε μέθοδο, χρησιμοποιήθηκε το μέτρο *F1*, το οποίο δίνει ίση βαρύτητα στο *Precision* και στο *Recall* και εκφράζει το ποσοστό επιτυχίας ανίχνευσης των ορίων μεταξύ των τμημάτων.

$$F_1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \qquad \text{Εξ. 5.2.1}$$

5.3. Data – sets

Στα πειράματά μας χρησιμοποιήθηκαν δύο διαφορετικά σύνολα δεδομένων που περιέχουν κείμενα (“text datasets”). Το πρώτο σύνολο δεδομένων που χρησιμοποιήσαμε είναι ένα υποσύνολο του TDT2 corpus. Το TDT2 corpus (Nist Topic Detection and Tracking) αποτελείται από δεδομένα που συλλέχθηκαν κατά την διάρκεια του πρώτου μισού του 1998 προερχόμενα από 6 πηγές: 2 υπηρεσίες μετάδοσης ειδησεογραφικών νέων (APW,NYT), 2 ραδιοφωνικά προγράμματα (VOA, PRI) και 2 τηλεοπτικά προγράμματα (CNN, ABC). Επίσης, αποτελείται από 11201 έγγραφα χωρισμένα σε 96 κατηγορίες. Στο υποσύνολο του TDT2 corpus με το οποίο ασχοληθήκαμε εμείς στα πλαίσια της παρούσας διατριβής, εκείνα τα κείμενα που εμφανίζονται σε δύο ή περισσότερες κατηγορίες αφαιρούνται και μένουν μόνο οι 30 μεγαλύτερες κατηγορίες, με συνολικό αριθμό 9394 κειμένων.

Το δεύτερο σύνολο δεδομένων που χρησιμοποιήθηκε είναι το BBC dataset.Το BBC dataset περιέχει 2225 κείμενα από την σελίδα των ειδησεογραφικών νέων του BBC, που αντιστοιχούν σε πέντε διαφορετικές θεματικές ενότητες από το 2004-2005. Οι θεματικές ενότητες είναι οι εξής: Business, Entertainment, Politics, Sport, Tech.

5.4. Πειραματικά αποτελέσματα

5.4.1. Πρώτη σειρα πειραμάτων

Στη πρώτη σειρά πειραμάτων που πραγματοποιήσαμε, εφαρμόσαμε τον αλγόριθμο Dip – Tsf 100 φορές, δημιουργώντας 100 διαφορετικά σύνολα δεδομένων (“datasets”). Αρχικά, για κάθε dataset, η επιλογή του αριθμού των κατηγοριών – θεματικών ενοτήτων που θα αποτελέσουν το προς εξέταση text stream, γίνεται κατά τρόπο τυχαίο, από ένα εύρος C, $5 \leq C \leq 10$. Επίσης, τυχαία επιλέγεται η αλληλουχία των κατηγοριών του text stream με τέτοιο τρόπο ώστε να εξασφαλίζεται ότι δεν εμφανίζονται δύο ίδιες κατηγορίες διαδοχικά η μια μετά την άλλη. Κατόπιν, με τυχαίο τρόπο επιλέγεται ο αριθμός των κειμένων ανά κατηγορία, από ένα εύρος L, $50 \leq L \leq 100$. Στο επόμενο βήμα, για κάθε ένα από αυτά τα σύνολα δεδομένων, πήραμε όλους τους δυνατούς συνδυασμούς, δηλαδή για κάθε ζεύγος τιμών w, p_th, με τις τιμές ss=20, voting=0.01 αμετάβλητες, εντοπίσαμε τα όρια μεταξύ των τμημάτων του text stream που βρίσκει ο αλγόριθμος SegmentDip και Tsf αντίστοιχα. Στη συνέχεια, υπολογίσαμε το ποσοστό επιτυχίας ανίχνευσης των ορίων μεταξύ των τμημάτων κάθε αλγορίθμου ξεχωριστά, με τη βοήθεια του μέτρου *F1*. Τέλος, υπολογίσαμε τους μέσους όρους της συνολικής απόδοσης του αλγορίθμου Tsf και SegmentDip αντίστοιχα, για κάθε ζεύγος τιμών p_th, w, από το σύνολο των *F1* τιμών του συνολικού πλήθους των datasets, για τον κάθε αλγόριθμο ξεχωριστά.

BBC dataset

Πίνακας 5.4.1. BBC dataset: Μέσοι όροι της απόδοσης (F1) των αλγορίθμων Dip-Tsf και SegmentDip για κάθε ζεύγος τιμών p_th, w, για 100 διαφορετικά datasets, με τυχαίο αριθμό κατηγοριών να συνθέτουν το text stream κάθε φορά.

p_th	w	MO_Dip - Tsf	MO_SegmentDip
0.01	20	0.6438	0.3465
0.01	30	0.5362	0.3043
0.01	40	0.3757	0.2563
0.03	20	0.7954	0.5424
0.03	30	0.8018	0.5205
0.03	40	0.6618	0.4160
0.05	20	0.7669	0.5673
0.05	30	0.7761	0.6019
0.05	40	0.8033	0.5565

TDT2 corpus

Πίνακας 5.4.2. TDT2 corpus: Μέσοι όροι της απόδοσης (F1) των αλγορίθμων Dip-Tsf και SegmentDip για κάθε ζεύγος τιμών p_th, w, για 100 διαφορετικά datasets, με τυχαίο αριθμό κατηγοριών να συνθέτουν το text stream κάθε φορά.

p_th	w	MO_Dip - Tsf	MO_SegmentDip
0.01	20	0.8391	0.7434
0.01	30	0.8338	0.7691
0.01	40	0.8685	0.7691
0.03	20	0.7886	0.6288
0.03	30	0.7883	0.6650
0.03	40	0.8536	0.6720
0.05	20	0.8325	0.5492
0.05	30	0.7902	0.6161
0.05	40	0.8666	0.6260

5.4.2. Δεύτερη σειρά πειραμάτων

Στη δεύτερη σειρά πειραμάτων που πραγματοποιήσαμε, εφαρμόσαμε τον αλγόριθμο Dip – Tsf 100 φορές, δημιουργώντας 100 διαφορετικά σύνολα δεδομένων ("datasets"). Αρχικά για κάθε dataset, γίνεται η επιλογή του αριθμού των κατηγοριών – θεματικών ενοτήτων που θα αποτελέσουν το προς εξέταση text stream. Σε αυτή τη σειρά πειραμάτων όμως, η επιλογή του αριθμού των κατηγοριών – θεματικών ενοτήτων που θα αποτελέσουν το προς εξέταση text stream δεν επιλέγεται κατά τρόπο τυχαίο από ένα εύρος C, $5 \leq C \leq 10$. Αντίθετα, επιλέξαμε ο αριθμός αυτός να είναι ίδιος για κάθε dataset, συγκεκριμένα επιλέχθηκαν 6 κατηγορίες. Στη συνέχεια, η αλληλουχία των κατηγοριών του text stream επιλέγεται τυχαία με τέτοιο τρόπο ώστε να εξασφαλίζεται ότι δεν εμφανίζονται δύο ίδιες κατηγορίες διαδοχικά η μια μετά την άλλη. Επίσης στο επόμενο βήμα, με τυχαίο τρόπο επιλέγεται ο αριθμός των κειμένων ανά κατηγορία, από ένα εύρος L, $50 \leq L \leq 100$. Η μετέπειτα διαδικασία που ακολουθείται είναι ακριβώς ίδια με αυτήν της πρώτης σειράς πειραμάτων.

BBC dataset

Πίνακας 5.4.3. BBC dataset: Μέσοι όροι της απόδοσης (F1) των αλγορίθμων Dip-Tsf και SegmentDip για κάθε ζεύγος τιμών p_th, w, για 100 διαφορετικά datasets, με 6 επιλεγμένες κατηγορίες να συνθέτουν το text stream κάθε φορά.

p_th	w	MO_Dip - Tsf	MO_SegmentDip
0.01	20	0.6056	0.2711
0.01	30	0.5079	0.2621
0.01	40	0.3787	0.2377
0.03	20	0.7742	0.5224
0.03	30	0.7715	0.4894
0.03	40	0.6493	0.4001
0.05	20	0.7504	0.5645
0.05	30	0.7682	0.5697
0.05	40	0.7815	0.5329

TDT2 corpus

Πίνακας 5.4.4. TDT2 corpus: Μέσοι όροι της απόδοσης (F1) των αλγορίθμων Dip-Tsf και SegmentDip για κάθε ζεύγος τιμών p_th, w, για 100 διαφορετικά datasets, με 6 επιλεγμένες κατηγορίες να συνθέτουν το text stream κάθε φορά.

p_th	w	MO_Dip - Tsf	MO_SegmentDip
0.01	20	0.8574	0.7529
0.01	30	0.8437	0.7485
0.01	40	0.8658	0.7764
0.03	20	0.7827	0.6287
0.03	30	0.7824	0.6617
0.03	40	0.8429	0.6971
0.05	20	0.7953	0.5434
0.05	30	0.7772	0.5947
0.05	40	0.8301	0.6143

5.4.3. Τρίτη σειρα πειραμάτων

Στη τρίτη σειρά πειραμάτων που πραγματοποιήσαμε, εφαρμόσαμε τον αλγόριθμο Dip – Tsf 100 φορές, δημιουργώντας 100 διαφορετικά σύνολα δεδομένων (“datasets”). Αρχικά για κάθε dataset, η επιλογή του αριθμού των κατηγοριών – θεματικών ενοτήτων που θα αποτελέσουν το προς εξέταση text stream είναι η ίδια, όπως και στη δεύτερη σειρά πειραμάτων. Σε αυτή τη περίπτωση όμως, έγινε η επιλογή μεγαλύτερου αριθμού, συγκεκριμένα επιλέχθηκαν 9 κατηγορίες. Η αλληλουχία των κατηγοριών του text stream επιλέγεται τυχαία με τέτοιο τρόπο ώστε να εξασφαλίζεται ότι δεν εμφανίζονται δύο ίδιες κατηγορίες διαδοχικά η μια μετά την άλλη. Επίσης, με τυχαίο τρόπο επιλέγεται ο αριθμός των κειμένων ανά κατηγορία από ένα εύρος L, $50 \leq L \leq 100$. Η μετέπειτα διαδικασία που ακολουθείται είναι ακριβώς ίδια με τις προηγούμενες σειρές πειραμάτων.

BBC dataset

Πίνακας 5.4.5. BBC dataset: Μέσοι όροι της απόδοσης (F1) των αλγορίθμων Dip-Tsf και SegmentDip για κάθε ζεύγος τιμών p_th, w, για 100 διαφορετικά datasets, με 9 επιλεγμένες κατηγορίες να συνθέτουν το text stream κάθε φορά

p_th	w	MO_Dip - Tsf	MO_SegmentDip
0.01	20	0.6345	0.3152
0.01	30	0.5410	0.3147
0.01	40	0.3945	0.2154
0.03	20	0.7953	0.5361
0.03	30	0.7927	0.4909
0.03	40	0.7045	0.4280
0.05	20	0.7604	0.5726
0.05	30	0.7748	0.6039
0.05	40	0.8025	0.5663

TDT2 corpus

Πίνακας 5.4.6. TDT2 corpus: Μέσοι όροι της απόδοσης (F1) των αλγορίθμων Dip-Tsf και SegmentDip για κάθε ζεύγος τιμών p_th, w, για 100 διαφορετικά datasets, με 9 επιλεγμένες κατηγορίες να συνθέτουν το text stream κάθε φορά

p_th	w	MO_Dip - Tsf	MO_SegmentDip
0.01	20	0.8678	0.7330
0.01	30	0.8347	0.7623
0.01	40	0.8645	0.7534
0.03	20	0.8057	0.6245
0.03	30	0.7842	0.6764
0.03	40	0.8548	0.6875
0.05	20	0.8362	0.5556
0.05	30	0.7904	0.6268
0.05	40	0.8655	0.6330

5.5. Συμπεράσματα

Ερμηνεύοντας τα παραπάνω πειραματικά αποτελέσματα μπορούμε να ισχυριστούμε ότι η απόδοση του αλγορίθμου Dip – Tsf και στις τρεις σειρές πειραμάτων που

πραγματοποιήθηκαν, είναι αρκετά καλή και μάλιστα σημειώνει υψηλά ποσοστά επιτυχίας ανίχνευσης των ορίων μεταξύ των τμημάτων. Αυτό σημαίνει ότι η αξιοποίηση του αλγορίθμου SegmentDip από τον Tsf σε ότι έχει να κάνει την αυτόματη εκτίμηση του αριθμού των ορίων, μπορεί να χαρακτηριστεί ως πολύ καλή επιλογή. Αξίζει επίσης να αναφερθεί, πως στο σύνολο δεδομένων TDT2 corpus ο αλγόριθμος Dip – Tsf καθώς επίσης και ο αλγόριθμος SegmentDip σημειώνει καλύτερα αποτελέσματα σε σχέση με το BBC dataset, αυτό βέβαια έχει να κάνει με τα χαρακτηριστικά των κειμένων του συγκεκριμένου dataset. Άλλωστε η δυσκολία στο πεδίο της κατάτμησης κειμένου, έχει να κάνει κυρίως με τα χαρακτηριστικά των κειμένων που πρόκειται να διαχωριστούν, από το data set δηλαδή που έχουμε στην διάθεσή μας π.χ επιστημονικά κείμενα, ειδησεογραφικά νέα, κ.τ.λ.

ΚΕΦΑΛΑΙΟ 6. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΗ ΕΡΓΑΣΙΑ

6.1. Σύνοψη συμπερασμάτων

Στην εργασία αυτή μελετήθηκε και παρουσιάστηκε το πρόβλημα της κατάτμησης σε ροές κειμένων. Παρουσιάστηκαν και υλοποιήθηκαν, δύο αλγόριθμοι γραμμικής κατάτμησης κειμένου, οι TextTiling και Tsf, που βασίζονται στην τεχνική του κυλιόμενου παραθύρου (“sliding window”), ένας αλγόριθμος που κάνει χρήση του στατιστικού κριτηρίου dip-dist ο SegmentDip, καθώς επίσης και μια προτεινόμενη μεθοδολογία η οποία δίνει λύση στο πρόβλημα του αυτόματου καθορισμού του αριθμού των ορίων μεταξύ των τμημάτων ενός text stream.

Στα πλαίσια της παρούσας διατριβής, διαπιστώθηκε η πολύ καλή απόδοση του αλγορίθμου Tsf σε ροές κειμένων όταν ο αριθμός των σημείων αλλαγής θέματος δίνεται από τον χρήστη, γι΄αυτό και κρίναμε πως θα ήταν ιδιαίτερα χρήσιμο αν υπήρχε κάποιος αυτόματος τρόπος που θα υποδείκνυε στον αλγόριθμο Tsf των αριθμό των ορίων και στη συνέχεια να γινόταν η εκτίμηση της απόδοσής του ως προς τα πραγματικά σημεία αλλαγής του νοήματος. Έτσι εφαρμόσαμε τον αλγόριθμος Dip – Tsf που αποτελεί την προτεινόμενη μέθοδο στην παρούσα εργασία, ο οποίος συνδυάζει το πλεονέκτημα του αλγορίθμου SegmentDip για την αυτόματη εκτίμηση του αριθμού των ορίων με τα πολύ καλά αποτελέσματα του αλγορίθμου Tsf όταν είναι γνωστός ο αριθμός των ορίων.

Έπειτα, μελετήσαμε πειραματικά την απόδοση του αλγορίθμου SegmentDip σε δύο σύνολα δεδομένων, καθώς επίσης και του αλγορίθμου Dip – Tsf. Παρατηρήσαμε ότι η απόδοση του αλγορίθμου Dip – Tsf είναι πολύ καλή και μάλιστα σημειώνει υψηλά ποσοστά επιτυχίας ανίχνευσης των ορίων μεταξύ των τμημάτων, καθιστώντας την μέθοδο που αναπτύξαμε ως αρκετά ικανοποιητική επιλογή.

6.2. Κατευθύνσεις μελλοντικής εργασίας

Μια αρχική κατεύθυνση για μελλοντική εργασία στις μεθόδους κατάτμησης που αναπτύχθηκαν θα μπορούσε να είναι ο συνδυασμός νέων τεχνικών – μετρικών ομοιότητας κατά τη φάση της προεπεξεργασίας των κειμένων, για την εύρεση εκείνων των στοιχείων που υποδεικνύουν ομοιογένεια ή ετερογένεια μεταξύ των διαφόρων τμημάτων σε ένα text stream. Στην παρούσα εργασία, οι μέθοδοι που μελετήσαμε χρησιμοποιούσαν την συχνότητα εμφάνισης λέξεων ως βασική ένδειξη της ομοιότητας των κειμένων, δηλαδή βασίζονταν στην παραδοχή ότι κείμενα που έχουν παρόμοιο λεξιλόγιο, έχουν μεγάλη πιθανότητα να πραγματεύονται το ίδιο θέμα και συνεπώς να ανήκουν στην ίδια θεματική ενότητα. Όμως, η αναλυτικότερη μελέτη της εμφάνισης των λέξεων μέσα σε ένα text stream, καθώς επίσης και η αξιοποίηση διαφόρων γλωσσολογικών ενδείξεων, όπως είναι οι λέξεις και οι προτάσεις σινιάλο (“cue words and phrases”) καθώς επίσης και τα συνώνυμα, θα μπορούσαν να περιέχουν περισσότερη χρήσιμη πληροφορία για την παραγωγή “ορθότερης” λύσης κατάτμησης.

Επιπρόσθετα, θα είχε ενδιαφέρον να εξεταστούν οι μεθοδολογίες που αναπτύχθηκαν σε περισσότερα σύνολα δεδομένων. Τέλος, στην παρούσα εργασία χρησιμοποιήθηκε το μέτρο *F1* για την αξιολόγηση των μεθόδων κατάτμησης σε ροές κειμένων. Μπορούν επίσης να εξεταστούν εναλλακτικά μέτρα αξιολόγησης, και θα ήταν ιδιαίτερα χρήσιμο να γίνει σύγκριση των αποτελεσμάτων των μεθόδων κατάτμησης σε ροές κειμένων για διαφορετικά μέτρα αξιολόγησης.

ΑΝΑΦΟΡΕΣ

[1] Abella-Pérez, R., & Medina-Pagola, J. E. (2010). "An Incremental Text Segmentation by Clustering Cohesion". The International Workshop on Handling Concept Drift in Adaptive Information Systems: Importance, Challenges and Solutions (HaCDAIS 2010). Workshop del ECML-PKDD 2010, pp. 65-72.

[2] Hearst, M. (1997). "TextTiling: Segmenting Text into Multi-paragraph Subtopic Passages". Computational Linguistics, vol, 23(1).

[3] Hearst, M. (1993). "TextTiling: A quantitative approach to discourse segmentation". Technical Report Sequoia 93/24, Computer Science Division, University of California, Berkeley.

[4] Hearst, M. (1994). "Multi-paragraph segmentation of expository texts". In Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistic, pp. 9-16, Las Cruses, New Mexico.

[5] Hearst, M. and Plaunt, C. (1993). "Subtopic structuring for full-length document access". In Proceedings of the Special Interest Group on Information Retrieval, pp. 59-68.

[6] Fragkou P., Petridis V., Kehagias A. (2004). "Linear Text Segmentation using a Dynamic Programming Algorithm". To appear in Journal of Intelligent Information Systems, Kluwer Academic Publishers.

[7] Kern, R., & Granitzer, M. (2009). "Efficient Linear Text Segmentation Based on Information Retrieval Techniques". MEDES 2009 Lyon, France.

[8] Passonneau, Rebecca J. and Diane J. Litman. (1993). "Intention – based segmentation: human reliability and correlation with linguistic cues". In Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, Stroudsburg, PA, USA.

[9] Passonneau, Rebecca J. and Diane J. Litman. (1995). "Combining multiple knowledge sources for discourse segmentation". In Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, Stroudsburg, PA, USA.

[10] Carletta J. (1994). “Assessing agreement on classification tasks: The kappa statistic”. Computational Linguistics, vol 22(2), pp.249-254.

[11] Beeferman D., Berger A. and Laffety J. (1997(a)). “A model of lexical attraction and repulsion”. In Proceedings of the 35th Annual Meeting of the Association of Computational Linguistics, pp. 373-380, Madrid.

[12] Beeferman D., Berger A. and Laffety J. (1997(b)). “Text Segmentation using exponential models”. In Proceedings of the Second Conference on Empirical Methods in Natural Language Processing, pp. 35-46, Providence, Rhode Island.

[13] Beeferman D., Berger A, and Laffety J. (1999). “Statistical Models for Text Segmentation”. Machine Learning, Special Issue on Natural Language Processing.

[14] Pevzner L. and Hearst M. (2002). “A Critique and Improvement of an Evaluation Metric for Text Segmentation”. Computational Linguistics, 2002.

[15] V. Chasanis and A. Ioannidis and A.Likas, (2014). “Efficient Key-frame Extraction Based on Unimodality of Frame Sequences”. Signal Processing (ICSP), 2014 12th IEEE International Conference on, 1133-1138.

[16] A. Kalogeratos and A.Likas, (2012). “Dip-mean: an incremental clustering method for estimating the number of clusters.” Proc. Neural Information Processing Systems (NIP’ 12), Lake Tahoe, Nevada, USA, 2012.

[17] J.A Hartigan and P.M Hartigan, (1985). “The dip test of unimodality.” The Annals of Statistics, 13(1), pp. 70-84, 1985.

[18] B.W Silverman, (1981). “Using Kernel density estimates to investigate multimodality.” Journal of Royal Statistis Society B, 43(1), pp.97-99, 1981.

[19] Halliday M. and Hasan R. (1976). “Cohesion in English”. Longman Group, New York.

[20] Hirschberg J. and Litman D. (1993). “Empirical studies on the disambiguation and cue phrases”. Computational Linguistics, vol.19(3), pp. 501-530.

[21] Reynar J.C (1998). “Topic Segmentation Algorithms and Applications”. Phd Thesis, Philadelphia.

[22] Youmans G. (1990). “Measuring lexical style and competence: The type-token vocabulary curve”. Style, vol 24, pp.584-599.

[23] Youmans G. (1991). “A new tool for discourse analysis: The vocabulary management profile”. Language, 67(4): 763-789.

[24] Philips M. (1985). “Aspects of text structure: An investigation of the lexical organization of text”. North Holland Linguistic Series. North Holland, Amsterdam.

[25] Yaari Y. (1997). “Segmentation of expository texts by hierarchical agglomerative clustering”. In Proceedings of Recent Advances in Natural Language Processing Bulgaria.

[26] Yaari Y. (1999). “Intelligent exploration of expository texts”. Phd thesis. Bar-Ilan University, Ramat-Gan, Israel.

[27] Ponte J.M and Croft W.B (1997). “Text Segmentation by topic”. In European Conference on Digital Libraries, Pisa, Italy.

[28] Xu J. and Croft W.B (1996). “Query expansion using local and global document analysis”. In Proceedings of the Nineteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Zurich, Switzerland.

[29] Ahonen H., Heikkinen B., Heinonen O. and Klemettinen M. (1997). “Discovery of Reasonably – sized Fragments Using Inter-paragraph Similarities”. Technical Report C-1997-67. University of Helsinki, Department of Computer Science.

[30] Richmond K., Smith A., Amitay E. (1997). “Detecting subject boundaries within text: A language independent statistical approach.” In Exploratory Methods in Natural Language Processing, Rhode Island.

[31] Choi F.Y.Y (2000). “Advances in domain independent linear text segmentation”. In Proceedings of the North American Chapter of the Association for Computational Linguistics, Seattle, USA, May, ACL.

[32] Raskin V, and Weiser I, (1987). “Language and writing: Applications of linguistics to rhetoric and composition”. Norwood, New Jersey: ABLEX: Publishing Corporation.

[33] Choi F.Y.Y, Wiemer – Hastings P, Moore J. (2001). “Latent Semantic Analysis for Text Segmentation”. In Proceedings of the 6th Empirical Methods of Natural Language Processing.

[34] Utiyama M., Isahara H. (2001). “A statistical model for domain – independent text segmentation”. In Proceedings of the ACL’ 2001, Toulouse , France.

[35] Reynar J.C (1994). “An automatic method of finding topic boundaries”. In Proceedings of the 32th Annual Meeting of the Association for Computational Linguistics, Student Session, Las, Cruses, New Mexico.

[36] Mann W, Thomson S.A (1987). “Rhetorical structure theory: A theory of text organization”. Technical Report ISI/RS 87-190, ISI.

[37] Grosz B.J, Sidner C.L (1986). “Attention Intentions and the structure of discourse”. Computational Linguistics.

[38] Kan M., Klavans J.L, McKeown, K.R (1998). “Linear Segmentation and Segment Significance”. In Proceedings of the 6th International Workshop of Very Large Corpora, Montreal, Quebec, Canada.

[39] Morris J. (1998). “Lexical cohesion, the thesaurus and the structure of the text”. Technical Report CSRI-219, Computer Systems Resarch Institute, University of Toronto.

[40] Morris J., Hirst G. (1991). “Lexical cohesion computed by thesaural relations as an indicator of the structure of text”. Computational Linguistics.

[41] Roget P.M (1911). “Roget’s International Thesaurus”. Cromwell, New York, first edition.

[42] Roget P.M (1977). “Roget’s International Thesaurus”. Harper and Row, New York, fourth edition.

[43] Porter M.F (1980). “An algorithm for suffix stripping”. Program 14,3 pp. 130-137.

[44] Fragkou P. (2004). Doctoral dissertation. “Classification and Segmentation of Texts using methods of Computational Linguistics”.

[45] Sardinha B. (1993). “Lexis in annual reports: Text Segmentation and lexical threads”. Technical Report 8, Development of International Research in English for Commerce and Technology.

[46] Sardinha B. (1999). “Looking at discourse in a corpus: The role of lexical cohesion”. In Proceedings of AILa ’99, Tokyo, Japan.

ΣΥΝΤΟΜΟ ΒΙΟΓΡΑΦΙΚΟ

Η Παρασκευή Κοσμά γεννήθηκε στα Ιωάννινα στις 24 Ιουνίου 1983. Αποφοίτησε από το 7^{ο} Ενιαίο Λύκειο της ίδιας πόλης και εισήχθη στο προπτυχιακό πρόγραμμα σπουδών του Τμήματος Τηλεπληροφορικής & Διοίκησης του Α.Τ.Ε.Ι Άρτας το 2001 από όπου αποφοίτησε το 2006. Το 2013 εισήχθη στο Μεταπτυχιακό πρόγραμμα σπουδών του τμήματος Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής του πανεπιστημίου Ιωαννίνων.