ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ

ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Πρόγραμμα Μεταπτυχιακών Σπουδών στις Τεχνολογίες Υπολογισμού και Δικτύων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Ανάλυση Μεθόδων Εξόρυξης Μαζικών Δεδομένων (Big Data) Από Κείμενα Και Εικόνες

Βασίλειος Νικόλαος Ματσάγγος
Α.Μ. 16004

Εισηγητής: Δρ Βασίλειος Μάμαλης, Καθηγητής

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Ανάλυση Μεθόδων Εξόρυξης Μαζικών Δεδομένων (Big Data) Από Κείμενα Και Εικόνες

Βασίλειος Νικόλαος Ματσάγγος
Α.Μ. 16004

Εισηγητής:

Δρ Βασίλειος Μάμαλης, Καθηγητής

Εξεταστική Επιτροπή:

Βασίλειος Μάμαλης, Καθηγητής
Γραμματή Πάντζιου, Καθηγήτρια
Ιωάννα Καντζάβελου, Επίκουρη Καθηγήτρια

Vasileios Mamalis
Digitally signed by Vasileios Mamalis
Date: 2021.10.18 13:25:24 +03'00'

Grammati Pantziou
Digitally signed by Grammati Pantziou
Date: 2021.10.28 00:09:36 +03'00'

Ioanna Kantzavelou
Digitally signed by Ioanna Kantzavelou
Date: 2021.10.29 13:04:08 +03'00'

Ημερομηνία εξέτασης 08/10/2021

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΜΕΤΑΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Ο/η κάτωθι υπογεγραμμένος Βασίλειος Νικόλαος Ματσάγγος του Γεωργίου, με αριθμό μητρώου 16004 φοιτητής/τρια του Προγράμματος Μεταπτυχιακών Σπουδών στις Τεχνολογίες Υπολογισμών και Δικτύων του Τμήματος Μηχανικών Πληροφορικής και Υπολογιστών της Σχολής Μηχανικών του Πανεπιστημίου Δυτικής Αττικής, δηλώνω ότι:
«Είμαι συγγραφέας αυτής της μεταπτυχιακής εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της, είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, οι όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε ακριβώς είτε παραφρασμένες, αναφέρονται στο σύνολό τους, με πλήρη αναφορά στους συγγραφείς, τον εκδοτικό οίκο ή το περιοδικό, συμπεριλαμβανομένων και των πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο. Επίσης, βεβαιώνω ότι αυτή η εργασία έχει συγγραφεί από μένα αποκλειστικά και αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο δικής μου, όσο και του Ιδρύματος.
Παράβαση της ανωτέρω ακαδημαϊκής μου ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του πτυχίου μου».

Ο/Η Δηλών/ούσα

ΕΥΧΑΡΙΣΤΙΕΣ

Η παρούσα διπλωματική εργασία ολοκληρώθηκε μετά από επίμονες προσπάθειες, σε ένα ενδιαφέρον γνωστικό αντικείμενο, όπως αυτό της εμβάθυνσης στην γνώση των αυτοματοποιημένων μεθόδων εξόρυξης δεδομένων καθώς και στις τεχνολογίες που χρησιμοποιούνται δια αυτήν. Την προσπάθειά μου αυτή υποστήριξε ο επιβλέπων καθηγητής μου, τον οποίο θα ήθελα να ευχαριστήσω.

Ακόμα θα ήθελα να ευχαριστήσω τη σύζυγο μου και την οικογένεια μου για την αμέριστη συμπαράσταση που έδειξαν κατά τη διάρκεια της συγγραφής αυτής της εργασίας καθώς και κατά τη διάρκεια της ολοκλήρωσης του κύκλου σπουδών μου στο ίδρυμα.

ΠΕΡΙΛΗΨΗ

Η παρούσα πτυχιακή εργασία αφορά τη διερεύνηση, μαθηματική και αλγοριθμική ανάλυση των μεθόδων εξόρυξης δομημένων δεδομένων από αδόμητα μαζικά δεδομένα (big data), και τη μελέτη των τρόπων επεξεργασίας και ομαδοποίησης αυτών. Θα γίνει μελέτη των περιγραφικών χαρακτηριστικών που ορίζουν τα δομημένα, αδόμητα, μαζικά δεδομένα εν γένει, και θα δοθεί έμφαση σε αυτά που εμφανίζονται με τις μορφές των κειμένων και των εικόνων. Θα παρουσιαστούν και θα αναλυθούν τα στάδια εξόρυξης των δεδομένων (εν γένει και κατά περίπτωση) οι αλγόριθμοι που χρησιμοποιούνται σε κάθε στάδιο, καθώς και η υλοποίηση των παραπάνω σε περιβάλλον κατανεμημένων (παράλληλων) συστημάτων. Σκοπός της παρούσας εργασίας είναι η βιβλιογραφική και μαθηματική διερεύνηση των παραπάνω τεχνολογιών (κατανεμημένα συστήματα, αλγόριθμοι εξόρυξης δεδομένων) , καθώς και η επιλογή των βέλτιστων συνδυασμών αυτών βάσει των αποδόσεων τους.

ABSTRACT

The present thesis concerns the exploration, mathematical and algorithmic analysis of the mining, processing and clustering methods of structured data from unstructured big data. The descriptive features that define structured, unstructured, big data in general will be studied, with emphasis on those in the form of text and images. The stages of data mining (in general and on a case-by-case basis), the algorithms used in each stage, as well as the implementation of the aforementioned in the environment of distributed (parallel) systems will be presented and analyzed. The purpose of this thesis is the bibliographic and mathematical investigation of the above technologies (distributed systems, data mining algorithms), as well as the choice of the optimal combinations based on their performance.

ΕΠΙΣΤΗΜΟΝΙΚΗ ΠΕΡΙΟΧΗ: Εξόρυξη Δεδομένων
ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Εξόρυξη δεδομένων, μαζικά δεδομένα, κατανεμημένα συστήματα, παράλληλοι αλγόριθμοι, mapreduce, ομαδοποίηση δεδομένων, δομημένα δεδομένα, αδόμητα δεδομένα, κείμενο, εικόνες

Περιεχόμενα

ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΗ

Την τελευταία δεκαπενταετία, η χρήση του διαδικτύου χαρακτηρίζεται από ραγδαία αύξηση. Βάσει της Διεθνούς Ένωσης Τηλεπικοινωνιών, του πρακτορείου Πληροφορίας και Τηλεπικοινωνιών των Ηνωμένων Εθνών, οι χρήστες του διαδικτύου αυξήθηκαν από 1.1 δισεκατομμύρια το έτος 2005 (17% του παγκόσμιου πληθυσμού) σε 3.7 δισεκατομμύρια το έτος 2018 (49% του παγκόσμιου πληθυσμού) [1]. Απόρροια αυτού, είναι πως η χρονική περίοδος της τελευταίας δεκαετίας χαρακτηρίστηκε από τον όρο “εποχή του zettabyte”. Αυτό έγινε επειδή αφενός το έτος 2012 καταγράφηκε για πρώτη φορά όγκος αποθηκευμένων δεδομένων στον παγκόσμιο ιστό που ξεπερνούσε το ένα zettabyte (10^{21} bytes)[2] και αφετέρου επειδή τον Σεπτέμβριο του 2016 ήταν η πρώτη φορά που καταγράφηκε από εταιρεία τηλεπικοινωνιών (Cisco) ετήσια κίνηση δεδομένων της αντίστοιχης τάξης[3].

Όπως είναι αναμενόμενο, η εκτεταμένη αυτή χρήση του διαδικτύου χαρακτηρίζεται από ανομοιογένεια των δεδομένων που διακινούνται και αποθηκεύονται, καθώς τέτοια δεδομένα μπορεί να είναι μηνύματα ηλεκτρονικού ταχυδρομείου, φωτογραφίες, σχόλια και περιεχόμενο ανεβασμένο σε λογαριασμούς μέσων κοινωνικής δικτύωσης, βίντεο σε αντίστοιχες πλατφόρμες φιλοξενίας, προσωπικά ιστολόγια, προσωπικές και επαγγελματικές ιστοσελίδες κ.α. Επίσης, τα τελευταία χρόνια, με την ραγδαία ανάπτυξη των ενσωματωμένων συστημάτων, όπου συσκευές καθημερινής χρήσης (αυτοκίνητα, ψυγεία, κινητά τηλέφωνα, GPS, συστήματα συναγερμού) αποκτούν μέσω μικροεπεξεργαστών τη δυνατότητα εκτέλεσης υπολογισμών σε πραγματικό χρόνο, καθώς και τη δυνατότητα πρόσβασης στο διαδίκτυο, δημιουργείται μια νέα κατηγορία κίνησης δεδομένων στον παγκόσμιο ιστό: η κατηγορία “Internet of Things”.

Ήδη από την δεκαετία του 1990, για να περιγραφεί η παραπάνω (παρούσα και αναμενόμενη) αύξηση του όγκου δεδομένων, δημιουργήθηκε από ακαδημαϊκούς και ερευνητές ο όρος “Big Data” (“Μαζικά Δεδομένα”)[4]. Σήμερα, αυτός ο όρος περιγράφει τον κλάδο της Επιστήμης των Υπολογιστών που ασχολείται με την εξόρυξη, αποθήκευση, περισυλλογή, κατηγοριοποίηση, ανάλυση, ευρετηρίαση πληροφορίας από μεγάλες και πολύπλοκες συλλογές δεδομένων[5]. Μερικές από τις εφαρμογές επεξεργασίας και εξόρυξης μαζικών δεδομένων εμφανίζονται γενικότερα στους τομείς της υγείας, των επιχειρήσεων, της εκπαίδευσης, αλλά και ειδικότερα στον τομέα της πληροφορίας, με παραδείγματα όπως τις μηχανές αναζήτησης, τις

στοχευμένες διαφημίσεις, τα μέσα κοινωνικής δικτύωσης και τις συσκευές πλοήγησης. Γίνεται κατανοητό πως πλέον, είναι αναγκαία η έρευνα και η ανάπτυξη νέων τεχνολογιών και μεθόδων επεξεργασίας των δεδομένων πέραν των παραδοσιακών, καθώς απαιτείται ταχύτερη επεξεργασία μεγάλου όγκου ετερόκλητων δεδομένων.

ΚΕΦΑΛΑΙΟ 2

ΜΑΖΙΚΑ ΔΕΔΟΜΕΝΑ (BIG DATA)

2.1. Ορισμός Μαζικών Δεδομένων

Ο όρος “Μαζικά Δεδομένα” (Big Data) για πολύ καιρό ήταν αρκετά ασαφής. Χρησιμοποιούταν κατά βούληση βιβλιογραφικά για να περιγράψει διάφορες περιπτώσεις καταστάσεων και επεξεργασίας δεδομένων. Η πρώτη εμφάνιση ενός σαφούς περιγραφικού ορισμού, βασισμένου σε συγκεκριμένα χαρακτηριστικά για τα Μαζικά Δεδομένα έγινε το 2011 και οι κατηγορίες αυτών των χαρακτηριστικών επεκτάθηκαν αργότερα από διαφόρους ερευνητές τα επόμενα χρόνια[6]. Το 2016 έγινε πρόταση για επίσημο ορισμό , η οποία ακολουθείται έως σήμερα από πολλούς ερευνητές και ακαδημαϊκούς, βάσει της οποίας τα Μαζικά Δεδομένα χαρακτηρίζονται από[6]:

- τα χαρακτηριστικά των ίδιων των δεδομένων
- τις τεχνολογικές ανάγκες επεξεργασίας των δεδομένων
- τα “κατώφλια” των συμβατικών απαιτήσεων που ξεπερνούν τα Μαζικά Δεδομένα
- τον κοινωνικό αντίκτυπο

2.1.1. Χαρακτηριστικά των δεδομένων

Όπως προαναφέρθηκε, ο κλάδος των Μαζικών Δεδομένων για πολύ καιρό είχε αρκετά ασαφή ορισμό. Για αυτόν τον λόγο, διάφοροι ερευνητές και ακαδημαϊκοί πρότειναν διάφορα χαρακτηριστικά που κατά τη γνώμη τους όριζαν τα Μαζικά δεδομένα. Συναθροίζοντας τις κατά καιρούς προτάσεις, και βάσει της λόγιας αποδοχής αυτών, τα δεδομένα θεωρούνται μαζικά όταν περιγράφονται από τα παρακάτω χαρακτηριστικά[7]:

Όγκος

Ο όγκος των δεδομένων που δημιουργείται, επεξεργάζεται, μεταφέρεται και αποθηκεύεται. Στην περίπτωση των Μαζικών Δεδομένων είναι της τάξης των exabytes (10^{18} bytes) ή zettabytes (10^{21} bytes). Εταιρείες όπως η Google διαχειρίζονται δεδομένα της τάξης των petabytes σε ημερήσια βάση (10^{15} bytes).

Ταχύτητα

Πολλές φορές, η επεξεργασία των Μαζικών Δεδομένων γίνεται (και αναμένονται αποτελέσματα) σε πραγματικό χρόνο. Έτσι, η ταχύτητα διεκπεραίωσης κάποιου αιτήματος σχετικό με Μαζικά Δεδομένα πρέπει να ανταποκρίνεται στις απαιτήσεις του όγκου των δεδομένων ή της καθυστέρησης λόγω της διακίνησης της πληροφορίας. Παραδείγματα Μαζικών Δεδομένων που απαιτείται υψηλή ταχύτητα και επιστροφή αποτελεσμάτων σε πραγματικό χρόνο είναι οι μηχανές αναζήτησης, η αναγνώριση προσώπων, η αντίστροφη αναζήτηση ηχητικών αποσπασμάτων κ.α.

Ποικιλομορφία

Τα δεδομένα εν γένει, μπορούν να χωριστούν στις εξής κατηγορίες: Δομημένα, ημιδομημένα και αδόμητα. Ως δομημένα ορίζονται τα δεδομένα τα οποία χαρακτηρίζονται από την οργανωμένη πινακοειδή μορφή τους, όπου κάθε κελί του πίνακα αντιστοιχεί σε κάποια διακριτή τιμή. Χαρακτηριστικότερο παράδειγμα δομημένων δεδομένων είναι μια σχεσιακή βάση δεδομένων. Ως αδόμητα αντιθέτως, χαρακτηρίζονται τα δεδομένα τα οποία εμφανίζονται σε πλήρως ανεπεξέργαστη μορφή, όπως ένα κείμενο, μια φωτογραφία ή ένα αρχείο βίντεο. Τέλος, ως ημιδομημένα χαρακτηρίζονται τα δεδομένα τα οποία ναι μεν δεν διαθέτουν πινακοειδή μορφή, αλλά περιέχουν στοιχεία κατηγοριοποίησης και ιεράρχησης αντικειμένων (όπως για παράδειγμα ένα αρχείο JSON). Η ποικιλομορφία των Μαζικών Δεδομένων, προϋποθέτει την ύπαρξη και των τριών τύπων σε μία συλλογή δεδομένων.

Εγκυρότητα

Σε μία συλλογή δεδομένων, ειδικότερα σε μία συλλογή μεγάλου όγκου, είναι στατιστικά αναμενόμενο πως θα υπάρχει και θόρυβος. Για παράδειγμα, στην περίπτωση επεξεργασίας και κατηγοριοποίησης ενός συνόλου ειδησεογραφικών

άρθρων, είναι πολύ πιθανόν να βρεθούν και άρθρα μη ειδησεογραφικής εγκυρότητας. Για αυτό το λόγο τα πηγαία δεδομένα θα πρέπει να φιλτράρονται, ώστε να μπορεί να επιτευχθεί η μέγιστη δυνατή εγκυρότητα.

Αξία

Η αξία μιας συλλογής μαζικών δεδομένων είναι συνδεδεμένη με τη χρησιμότητα των δεδομένων, καθώς και των εξαγώγιμων πληροφοριών από αυτά, αφού ο οποιοσδήποτε οργανισμός χρειάζεται να επενδύσει σε υποδομές, ανθρώπινο δυναμικό και τεχνογνωσία ώστε να μπορεί να επεξεργαστεί μαζικά δεδομένα. Για ένα κοινωνικό δίκτυο, είναι μεγάλης αξίας τα δεδομένα που μπορούν να χρησιμοποιηθούν για στοχευμένες διαφημίσεις, για μία μηχανή αναζήτησης είναι μεγάλης αξίας τα δεδομένα τα οποία βοηθούν στην ευρετηρίαση ιστοσελίδων, ενώ για ένα ιατρικό ερευνητικό κέντρο μεγάλης αξίας είναι τα δεδομένα που μπορούν να βοηθήσουν στην ανάπτυξη ενός φαρμάκου ή στην κατηγοριοποίηση καρκινικών κυττάρων.

2.1.2. Τεχνολογικές ανάγκες

Η αποθήκευση, μεταφορά και επεξεργασία Μαζικών Δεδομένων απαιτεί υπολογιστικές δυνατότητες και μεθόδους ανάλυσης πέραν των συμβατικών. Μία μικρότερη συλλογή δεδομένων μπορεί να αναλυθεί από ένα απλό υπολογιστικό σύστημα (προσωπικός υπολογιστής, μηχάνημα εξυπηρετητή), χρησιμοποιώντας συμβατικές μεθόδους επεξεργασίας (σειριακός αλγόριθμος, εκτέλεση σε σύστημα μοναδικού επεξεργαστή με πρόσβαση σε μοναδικό αποθηκευτικό μέσο). Αν και έχουν αναπτυχθεί μέθοδοι επεξεργασίας Μαζικών Δεδομένων σε μοναδικούς υπολογιστές, πλέον η ανάλυση τους γίνεται κατά κόρον συστοιχίες κατανεμημένων υπολογιστικών συστημάτων, με χρήση παραλλήλων αλγορίθμων έναντι των σειριακών, ανάπτυξη αλγόριθμων ομαδοποίησης δεδομένων, καθώς και με χρήση ειδικού λογισμικού ταυτόχρονης και παράλληλης εκμετάλλευσης των υπολογιστικών συστοιχιών. Τέλος, οι συστοιχίες επεξεργασίας και ανάλυσης Μαζικών Δεδομένων (καθώς και οι αλγόριθμοι που εκτελούνται σε αυτές) θα πρέπει να είναι δυναμικώς κλιμακούμενες και επεκτάσιμες, καθώς οι απαιτήσεις σε επεξεργαστική ισχύ, αποθηκευτικό χώρο και ταχύτητα μεταφοράς μέσω δικτύου μεταβάλλονται συνεχώς και σε πραγματικό χρόνο[8].

2.1.3. Κατώφλια

Βάσει των τεχνολογικών απαιτήσεων που αναπτύχθηκαν παραπάνω, μπορούν να ορισθούν και τα κατώφλια από τα οποία χαρακτηρίζονται τα Μαζικά Δεδομένα. Για

παράδειγμα η επεξεργασία μίας φωτογραφίας, η οποία γίνεται με τη βοήθεια μιας εφαρμογής εγκατεστημένης σε συγκεκριμένο αποθηκευτικό μέσο, και η οποία εκτελείται από έναν πυρήνα επεξεργασίας, δεν εμπίπτει στον κλάδο των Μαζικών Δεδομένων. Αντίθετα, η αντίστροφη αναζήτηση μιας εικόνας, βάσει της ομοιότητας της με ένα τεράστιο σύνολο εικόνων, η οποία απαιτεί πολύπλοκους υπολογισμούς που πρέπει να γίνουν παράλληλα σε μία συστοιχία υπολογισμών, απαιτεί πρόσβαση σε δεδομένα τεράστιου όγκου (υπερπολλαπλάσιου της χωρητικότητας του μέγιστου μοναδιαίου αποθηκευτικού μέσου), εμπίπτει στον παραπάνω κλάδο. Μία καλή προσέγγιση υπολογισμού αλλά και πρόβλεψης των κατωφλιών που συνιστούν τα Μαζικά Δεδομένα είναι ο νόμος του Moore, καθώς ιστορικά το αποδεκτό μέγεθος των Μαζικών Δεδομένων ακολουθεί τη μορφή της καμπύλης αυτού του νόμου, αλλά και επειδή βάσει του νόμου του Moore μπορεί να οριστεί η αγοραστική δυνατότητα επεξεργαστικής ισχύος και χώρου αποθήκευσης για τον μέσο χρήστη[9].

2.1.4 Κοινωνικός αντίκτυπος

Ο κλάδος των Μαζικών Δεδομένων έχει αντίκτυπο σε διάφορους τομείς της κοινωνίας, είτε αυτός ο αντίκτυπος μπορεί να θεωρηθεί θετικός είτε αρνητικός. Βάσει έρευνας που χρηματοδοτήθηκε από την επιτροπή της Ευρωπαϊκής Ένωσης, τα Μαζικά Δεδομένα εκλαμβάνονται ότι ωφελούν την κοινωνία στην λήψη αποφάσεων, τη δημιουργία επιχειρησιακών μοντέλων, τη διαφύλαξη της κοινωνίας και του περιβάλλοντος και την συμμετοχή των πολιτών στα κοινωνικά δρώμενα, όσον αφορά τομείς όπως το σύστημα υγείας, το φυσικό περιβάλλον, την ενέργεια, τη διαχείριση κρίσεων και τη διανομή εμπορευμάτων. Στον αντίποδα, όσον αφορά την ιδιωτικότητα, την ευθύνη και τη λογοδοσία οργανισμών διαχείρισης δεδομένων, την διαχείριση της πνευματικής ιδιοκτησίας και την πιθανή κατάχρηση των δεδομένων για παρακολούθηση, οι πολίτες είναι διστακτικοί για την αποδοχή της ορθής χρήσης των Μαζικών Δεδομένων[10].

2.2 Εφαρμογές των Μαζικών Δεδομένων

Ο κλάδος των Μαζικών Δεδομένων βρίσκει εφαρμογή σε διάφορους κοινωνικούς, επιστημονικούς και τεχνολογικούς τομείς.

Υγεία

Ο τομέας της υγείας είναι ένας από τους πλέον αυτοματοποιημένους και μηχανογραφημένους τομείς της σύγχρονης κοινωνίας. Υπάρχει μεγάλος όγκος δεδομένων που αποθηκεύεται καθημερινά, από ιστορικά ασθενών, καταγραφές επιπλοκών και παρενεργειών στα τμήματα φαρμακοεπαγρύπνησης εταιρειών και

οργανισμών, αποτελέσματα κλινικών δοκιμών, καθώς και ιατρικά στοιχεία μεγάλης πληθυσμιακής κλίμακας, όπως ποσοστιαίες καταγραφές θετικότητας μιας ασθένειας στον γενικότερο πληθυσμό. Με την ανάλυση Μαζικών Δεδομένων, από τα παραπάνω δεδομένα μπορούν να εξαχθούν χρήσιμα συμπεράσματα και προβλέψεις, όπως οι πιθανές παρενέργειες ενός φαρμάκου σε έναν νέο χρήστη (ασθενή), η καταλληλότερη θεραπεία ενός ασθενούς βάσει του ιστορικού του, η ελαχίστου κόστους μέθοδος παραγωγής ενός φαρμάκου, ή η ανάλυση μοτίβων μίας ασθένειας με σκοπό την πρόβλεψη της διασποράς της στον πληθυσμό, αλλά και ανάπτυξη των κατάλληλων γραμμών άμυνας εναντίον αυτής. Μία σχετικά πρόσφατη προσθήκη στον τομέα της υγείας που αφορά την ανάλυση δεδομένων, είναι η συνεργία συσκευών (Internet of Things) όπως ιατρικές εφαρμογές σε smartphones, αισθητήρες βιομετρικών χαρακτηριστικών και συσκευές ιατρικής παρακολούθησης [11]. Η πιο πρόσφατη και χαρακτηριστική περίπτωση ανάλυσης δεδομένων στον τομέα της υγείας συνδέεται με την πανδημία COVID-19, με παραδείγματα όπως της Κίνας όπου αναπτύχθηκε εφαρμογή ανίχνευσης άμεσης επαφής με επιβεβαιωμένο κρούσμα[12] για κινητά τηλέφωνα.

Εκπαίδευση

Για να διατηρηθούν ενημερωμένα τα προγράμματα σπουδών των εκπαιδευτικών ιδρυμάτων (σχολεία, πανεπιστήμια, ιδιωτικοί φορείς εκπαίδευσης) έχουν εντάξει στον οδηγό σπουδών τους τη διδασκαλία των Μαζικών Δεδομένων, με σκοπό την κατανόηση του κλάδου καθώς και την έρευνα σε αυτόν. Όμως, και η ίδια η διδασκαλία έχει ωφεληθεί από τον κλάδο των Μαζικών Δεδομένων, καθώς με την ανάπτυξη και την υιοθέτηση της τηλέ-εκπαίδευσης (είτε ως επίσημο μέσο εκπαιδευτικών οργανισμών είτε ως μέσο αυτοδιδασκαλίας), διάφοροι φορείς που προσφέρουν εκπαίδευση εξ αποστάσεως εφαρμόζουν ανάλυση δεδομένων, ώστε βάσει του προφίλ του χρήστη να προτείνουν νέα μαθήματα, οδηγούς σπουδών κλπ [13]. Επίσης, πάροχοι εκπαιδευτικού περιεχομένου, διαχειρίζονται τεράστιο και δυναμικό όγκο μη-δομημένων δεδομένων, τα οποία με τη βοήθεια της αντίστοιχης ανάλυσης κατηγοριοποιούνται ώστε να είναι ευκολότερα προσβάσιμα από τον τελικό χρήστη και τους διαχειριστές της πλατφόρμας.

Ηλεκτρονικό εμπόριο

Στη σημερινή εποχή, μεγάλο μέρος των αγορών γίνεται διαδικτυακά. Οι αγοραστές αναζητούν και επιλέγουν καταστήματα βάσει προϊόντων και τιμών για να κάνουν αγορές είτε ψηφιακά είτε μέσω φυσικής παρουσίας. Έχουν δημιουργηθεί ιστότοποι που συγκεντρώνουν καταστήματα και προϊόντα ώστε να διευκολύνουν τον τελικό καταναλωτή στην αναζήτησή του (amazon, ebay, skroutz, efood). Για να ανταποκριθούν στον όγκο δεδομένων που απαιτείται να διαχειριστούν αυτοί οι ιστότοποι, χρησιμοποιούν παράλληλα κατανεμημένα συστήματα, αλγόριθμους

παράλληλης επεξεργασίας και ανάλυσης μαζικών δεδομένων καθώς και συστοιχίες υπολογιστών[14][15].

Μέσα κοινωνικής δικτύωσης

Μέσα κοινωνικής δικτύωσης όπως το facebook, το twitter και το instagram, φιλοξενούν περιεχόμενο από δισεκατομμύρια χρήστες. Περιεχόμενο όπως προφίλ, σχόλια, φωτογραφίες, βίντεο, αναρτήσεις. Πέραν της αποθήκευσης όλου του όγκου του περιεχομένου, τα μέσα κοινωνικής δικτύωσης έρχονται αντιμέτωπα και με άλλες προκλήσεις. Προκλήσεις όπως η καταπολέμηση των ψευδών ειδήσεων, η ανίχνευση και διαγραφή ψεύτικων προφίλ[16], ή η ανίχνευση ακατάλληλου περιεχομένου (ακατάλληλη γλώσσα, φωτογραφίες που απεικονίζουν βία κλπ)[17]. Εκτός από τις εξ ορισμού προκλήσεις όμως, τα μέσα αυτά στοχεύουν και σε δευτέρου επιπέδου αναλύσεις, όπως την εξόρυξη δεδομένων από φωτογραφίες, την ανάλυση του προφίλ των χρηστών με σκοπό την προβολή προσωποποιημένων διαφημίσεων[18], ή ακόμα και προτάσεις προς τον χρήστη για περιεχόμενο που ίσως τον ενδιαφέρει. Για την επίλυση των παραπάνω, τα μέσα κοινωνικής δικτύωσης έχουν αναπτύξει δικές τους μεθόδους ανάλυσης και εξόρυξης Μαζικών Δεδομένων, πολλές φορές βασισμένες σε ήδη υπάρχουσες τεχνολογίες[14].

Οικονομία

Στον κλάδο της οικονομίας, υπάρχουν πολλοί τομείς όπου εφαρμόζονται μέθοδοι μαζικής ανάλυσης δεδομένων, όπως το χρηματιστήριο, το παγκόσμιο τραπεζικό σύστημα, οι μεγάλου μεγέθους εταιρείες και οργανισμοί, καθώς και ο τομέας των ασφαλίσεων. Οι συνηθέστερες αναλύσεις δεδομένων που λαμβάνουν χώρα σε αυτόν τον κλάδο έχουν ως στόχο την πρόβλεψη. Από την μελλοντική κίνηση μίας μετοχής, το ρίσκο ενός δανείου βάση του προφίλ του δανειολήπτη, ή την πρόβλεψη αν μία σειρά τραπεζικών κινήσεων μπορεί να καταλήξει σε κάποια απάτη. Αυτές οι αναλύσεις γίνονται συνήθως από πολύπλοκα και ταχύτατα νευρωνικά δίκτυα, καθώς απαιτείται άμεσος υπολογισμών των αποτελεσμάτων και εξαγωγής των προβλέψεων (η αξία μίας μετοχής μπορεί να αλλάξει σε πολύ μικρό χρονικό διάστημα, όπως σε πολύ μικρό χρονικό διάστημα απαιτείται να αποφευχθεί μία ηλεκτρονική τραπεζική απάτη)[19]. Πέραν των προβλέψεων όμως, ο κλάδος της οικονομίας είναι ένας κατ εξοχήν κλάδος συνδεδεμένος με τα Μαζικά Δεδομένα, καθώς ήταν από τους πρώτους που εφαρμόστηκαν αναλύσεις δεδομένων, και, στην σημερινή εποχή που το μεγαλύτερο μέρος των οικονομικών κινήσεων γίνεται ηλεκτρονικά, είναι από τους κλάδους με την μεγαλύτερη κίνηση, αποθήκευση και ανάλυση δεδομένων[19][20].

Ψυχαγωγία

Οι μέθοδοι ανάλυσης δεδομένων στον κλάδο της ψυχαγωγίας, και ειδικότερα σε πλατφόρμες ροής περιεχομένου (youtube, netflix), όπως και στον κλάδο των μέσων κοινωνικής δικτύωσης, χρησιμοποιούνται για εξαγωγή προσωποποιημένων προτάσεων προς τον χρήστη βάσει ανάλυσης των επιλογών του και του προφίλ του[20]. Επίσης, μέσω της ανάλυσης Μαζικών Δεδομένων, εταιρείες παροχής περιεχομένου μπορούν να ελέγχουν και να σταματούν τη διακίνηση πειρατικών πολυμέσων με τη βοήθεια αλγορίθμων καταγραφής και σύγκρισης οπτικοακουστικού περιεχομένου που προστατεύεται από πνευματικά δικαιώματα[21]. Τέλος, τα τελευταία χρόνια έχουν αναπτυχθεί και διαδοθεί ευρέως εφαρμογές που βασίζονται σε τεχνικές εξόρυξης δεδομένων, όπως εφαρμογές που αναλύουν πολυμέσα για τον τελικό χρήστη[22] (εφαρμογές αναγνώρισης μελωδίας, εφαρμογές σύγκρισης φωτογραφιών) ή εφαρμογές που χρησιμοποιούν νευρωνικά δίκτυα για την άμεση δημιουργία περιεχομένου[23][24] (παραγωγή τρισδιάστατων μοντέλων από φωτογραφίες, αλλαγή προσώπων σε βίντεο).

Internet of Things

Με την όλο και αυξανόμενη συνδεσιμότητα συσκευών στο διαδίκτυο, αυξάνονται οι απαιτήσεις σε μετάδοση και επεξεργασία πληροφοριών. Συσκευές όπως έξυπνα αυτοκίνητα, σκούπες, ρολόγια, έξυπνα κτίρια, συστήματα πυρόσβεσης, ιατρικός εξοπλισμός, μεταδίδουν συνεχώς τεράστιο όγκο μη-δομημένων δεδομένων, δημιουργώντας πρακτικά ένα δικό τους οικοσύστημα Μαζικών Δεδομένων. Ερευνητικοί οργανισμοί (όπως το MIT ή το Πανεπιστήμιο του Berkeley) ήδη αναζητούν βέλτιστες λύσεις στις προκλήσεις που δημιουργεί η τεράστια παραγωγή δεδομένων από τις έξυπνες συσκευές[25].

Συγκοινωνίες - Μετακινήσεις

Η εύρεση και απεικόνιση της βέλτιστης διαδρομής μεταξύ δύο σημείων στον χάρτη μέσω κάποιας εφαρμογής πλοήγησης (GPS) απαιτεί ιδιαίτερα πολύπλοκους υπολογισμούς και πρόσβαση σε μεγάλο όγκο δεδομένων, καθώς απαιτείται η απόκτηση πρόσβασης σε όλο το οδικό δίκτυο μεταξύ των δύο σημείων, να υπολογιστούν εκατοντάδες συνδυασμοί διαδρομών και να αναλυθεί σε πραγματικό χρόνο η κίνηση που υπάρχει στους δρόμους. Επίσης, το αποτέλεσμα της ανάλυσης των παραπάνω θα πρέπει να υπολογιστεί σε πολύ μικρό χρονικό διάστημα και να παρουσιαστεί στον χρήστη πριν την αναχώρηση του, ή πολλές φορές, κατά τη διάρκεια της μετακίνησης του (πχ στην περίπτωση μη αναμενόμενης αλλαγής πορείας). Λόγω αυτών των απαιτήσεων αλλά και των περιορισμών, πρέπει να υλοποιηθούν αλγόριθμοι ανάλυσης Μαζικών Δεδομένων με παράλληλη χρήση

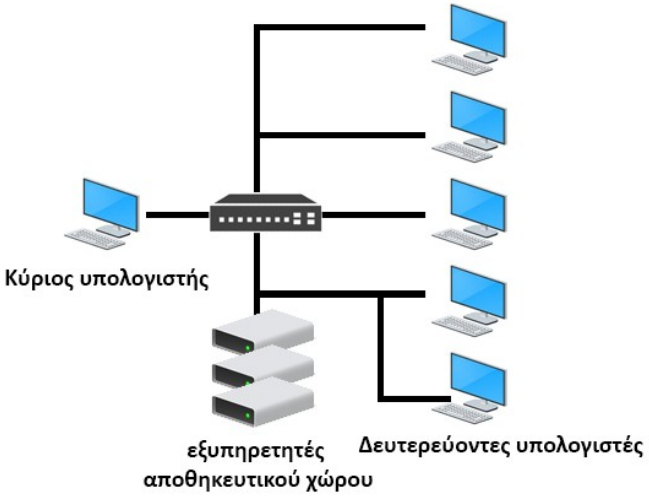
υπολογιστικών συστημάτων για μέγιστη και κλιμακούμενη υπολογιστική ισχύ (ώστε να επιτευχθεί ελάχιστη καθυστέρηση) ,αλλά και άμεση πρόσβαση σε πολύ μεγάλο όγκο δεδομένων[26]. Επίσης, στον τομέα των συγκοινωνιών έχουν αναπτυχθεί εφαρμογές εύρεσης βέλτιστων (χρονικά και οικονομικά) αεροπορικών και ναυτιλιακών διαδρομών, οι οποίες εκμεταλλεύονται τεράστιες βάσεις δεδομένων με πτήσεις και απόπλους, και αναλύουν τα στοιχεία τους με αντίστοιχο τρόπο με τον προαναφερθέν. Τέλος, πολλά συγκοινωνιακά μέσα έχουν προσχωρήσει στον τομέα του “Internet of Things”, όπως για παράδειγμα λεωφορεία που παρέχουν υπηρεσίες τηλεματικής μέσω συστημάτων ιχνηλάτησης (GPS trackers) στον επιβάτη με τη βοήθεια ειδικών εφαρμογών[27].

2.3. Επεξεργασία των Μαζικών Δεδομένων

Βάσει της φύσης των Μαζικών Δεδομένων όπως αυτή περιγράφηκε παραπάνω (παράγραφος 2.1.1), δημιουργούνται προκλήσεις αναφορικά με την επεξεργασία και την ανάλυση τους. Προκλήσεις σχετικές με το μέγεθος των συλλογών δεδομένων, την απαιτούμενη ταχύτητα εξαγωγής αποτελεσμάτων και συμπερασμάτων καθώς και την ετερογένεια των ακατέργαστων δεδομένων. Οι ερευνητές του κλάδου της επιστήμης των δεδομένων έχουν δημιουργήσει ολοκληρωμένα συστήματα διαχείρισης Μαζικών Δεδομένων, τα οποία απαρτίζονται από υποσυστήματα, αλγόριθμους και προγραμματιστικά μοντέλα που έχουν ως στόχο την αντιμετώπιση των παραπάνω προκλήσεων[28]. Κάποια από τα χαρακτηριστικότερα παραδείγματα αυτών των υποσυστημάτων παρουσιάζονται στις επόμενες παραγράφους.

2.3.1. Συστοιχίες υπολογιστών

Ένα σύνολο υπολογιστών συνδεδεμένων μεταξύ τους, μέσω ενός μεταγωγέα δικτύου (network switch), αποτελεί μία συστοιχία υπολογιστών (computer cluster). Ένας (ή κάποιες φορές παραπάνω) από αυτούς τους υπολογιστές λειτουργεί ως κύριος και οι υπόλοιποι ως δευτερεύοντες (κατανομή master / slave). Σε κάποιες περιπτώσεις που απαιτείται επιπλέον αποθηκευτικός χώρος αλλά όχι επιπλέον επεξεργαστική ισχύς, είναι συνδεδεμένοι στον μεταγωγέα εξυπηρετητές αποθηκευτικού χώρου (file servers / network-attached storage)[32]



Εικόνα 2.1: ενδεικτικό γράφημα συστοιχίας υπολογιστών

Οι συστοιχίες υπολογιστών μπορούν να δημιουργηθούν με καθορισμένη δομή και να αφιερωθούν στη διεκπεραίωση συγκεκριμένων διεργασιών, αλλά τις περισσότερες φορές η δομή τους είναι ακαθόριστη και δυναμική. Πάροχοι υπηρεσιών “υπολογισμών χωρίς διακομιστή” (serverless computing) προσφέρουν στον χρήστη τη δυνατότητα δέσμευσης και χρήσης υλικού (Infrastructure as a Service) [33]. Όπως μαρτυρεί το όνομα της κατηγορίας αυτών των υπηρεσιών, ο χρήστης δεν έχει εικόνα και άμεση σχέση με την δόμηση της συστοιχίας, αλλά ανάλογα με τις απαιτήσεις των προγραμμάτων που εκτελεί, αυτή δημιουργείται και αλλάζει δυναμικά.

2.3.2. Κατανεμημένα συστήματα αρχείων

Ένα κατανεμημένο σύστημα αρχείων, σε αντίθεση με τα συστήματα αρχείων μοναδιαίων αποθηκευτικών μέσων (FAT32, NTFS), χρησιμοποιεί μία συστοιχία συνδεδεμένων υπολογιστών για να αποθηκεύει και να επεξεργάζεται αρχεία. Η συστοιχία των υπολογιστών αποτελείται από υπολογιστές-πελάτες και από εξυπηρετητές. Ένας κύριος εξυπηρετητής διαχειρίζεται όλα τα μετά-δεδομένα του συστήματος αρχείων, και οι υπόλοιποι εξυπηρετητές χρησιμοποιούνται για την αποθήκευση κομματιών των αρχείων. Τα αρχεία χωρίζονται σε ισομεγέθη κομμάτια και μπορούν να διαμοιραστούν από τους εξυπηρετητές σε περισσότερους από έναν πελάτες κάθε φορά, βοηθώντας έτσι στην επεξεργασία μεγάλου όγκου δεδομένων ταυτόχρονα (αν επιτρέπεται από τους τεχνικούς περιορισμούς, προτιμάται η αποθήκευση κάθε κομματιού του αρχείου σε διαφορετικό εξυπηρετητή). Ένας

υπολογιστής – πελάτης όταν ζητάει πρόσβαση σε κάποιο αρχείο γνωρίζει μόνο το αντιπροσωπευτικό όνομα και τη διαδρομή του αρχείου, δεν έχει εικόνα της πηγής των κομματιών του, ούτε της ευρετηρίασης των κομματιών των αρχείων που έχει κάνει ο κύριος εξυπηρετητής. Όλα τα κατανεμημένα συστήματα αρχείων δημιουργούν επιπλέον αντίγραφα ασφαλείας των κομματιών των αρχείων σε ξεχωριστούς υπολογιστές, έτσι ώστε να υπάρχει ανοχή σε σφάλματα ανάγνωσης / εγγραφής κάποιου κατεστραμμένου κομματιού. Τα παρακάτω κριτήρια μπορούν να χαρακτηρίσουν ένα κατανεμημένο σύστημα αρχείων[29]:

- Ανοχή σε σφάλματα (κατεστραμμένα αρχεία, απώλεια εξυπηρετητή)
- Διαφάνεια (γνώση μόνο του ονόματος και της διαδρομής αρχείου)
- Ύπαρξη αντιγράφων ασφαλείας
- Συγχρονισμός (αλλαγή σε ένα αρχείο επιφέρει αλλαγή και στα αντίγραφά του)
- Ονοματοδοσία (γίνεται σωστή και μονοσήμαντη ονοματοδοσία σε αρχεία, εξυπηρετητές, χρήστες)

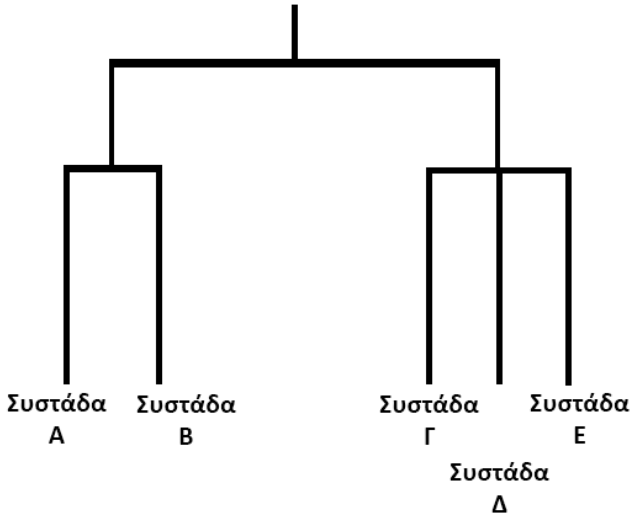
2.3.3. Εφαρμογές παράλληλης επεξεργασίας

Προκειμένου να μπορεί να γίνει η επεξεργασία των αρχείων σε ένα κατανεμημένο σύστημα υπολογιστών, έχουν δημιουργηθεί και τα κατάλληλα προγραμματιστικά μοντέλα. Παραδείγματα αυτών των μοντέλων είναι το MapReduce του οικοσυστήματος Hadoop της Apache (πρωτοκυκλοφόρησε ως εργαλείο της Google), το Spark του ίδιου κατασκευαστή[30], ή το MPI[31] (πρωτόκολλο μετάδοσης πληροφορίας μεταξύ υπολογιστών για παράλληλη επεξεργασία). Στόχος αυτών των προγραμματιστικών μοντέλων είναι η εκμετάλλευση της παραλληλίας των υπολογιστών, αναθέτοντας υπορουτίνες του προγράμματος μεταξύ αυτών, κατανέμοντας τον φόρτο επεξεργασίας και μεταφέροντας πληροφορία μεταξύ τους. Όπως και στην περίπτωση των κατανεμημένων συστημάτων αρχείων, ο χρήστης των προγραμματιστικών αυτών μοντέλων δεν έχει εικόνα της κατανομής του φόρτου και της ανάθεσης στα μέρη της συστοιχίας, αλλά συγγράφει κώδικα όπως θα έκανε σε σύστημα μοναδικού υπολογιστή (λειτουργούν ως μεσολογισμικό / middleware).

2.3.4. Αλγόριθμοι συσταδοποίησης

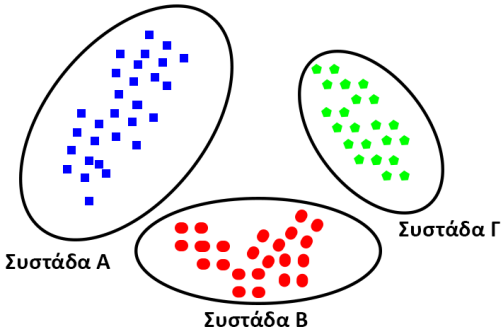
Ως συσταδοποίηση (clustering) ορίζεται η διαδικασία ταξινόμησης στοιχείων μιας συλλογής σε ομάδες (συστάδες) βάσει της ομοιότητάς τους. Οι αλγόριθμοι συσταδοποίησης χωρίζονται στις παρακάτω κατηγορίες[34]:

Αλγόριθμοι ιεραρχικής συσταδοποίησης: Δημιουργούνται ομάδες που σχηματίζουν μία δενδροειδή δομή. Η δημιουργία αυτής της δομής μπορεί να γίνει από πάνω προς τα κάτω, όπου θεωρείται ότι όλα τα στοιχεία ανήκουν σε μία ενιαία ομάδα η οποία διαιρείται σε μικρότερες, ή από κάτω προς τα πάνω, όπου κάθε στοιχείο απαρτίζει από μόνο του μία συστάδα και βάσει συγκεκριμένων κριτηρίων αυτές οι συστάδες συνενώνονται.



Σχήμα 2.1: δημιουργία συστάδων μέσω ιεραρχικού αλγόριθμου

Αλγόριθμοι διαχωριστικής συσταδοποίησης: Στη συσταδοποίηση με τη βοήθεια διαχωριστικών αλγορίθμων χρησιμοποιούνται μετρικές αποστάσεων, όπως η ευκλείδεια απόσταση ή η ομοιότητα συνημιτόνου. Βάσει αυτών των αποστάσεων και ελέγχοντας συγκεκριμένα κριτήρια (ανάλογα με τον αλγόριθμο χρήσης), όπως για παράδειγμα υπολογίζοντας ποια συστάδα διατηρεί τον μέγιστο αριθμό γειτόνων για ένα νέο σημείο εισόδου, τοποθετείται το κάθε σημείο στη συστάδα που αντιστοιχεί.



Σχήμα 2.2: δημιουργία συστάδων μέσω διαχωριστικού αλγόριθμου

Αλγόριθμοι συσταδοποίησης βασισμένοι σε μορφή πλέγματος: Χωρίζεται ο χώρος στον οποίον ανήκουν τα στοιχεία σε ένα πλέγμα, και οι συστάδες υπολογίζονται ανάλογα με τον αριθμό των στοιχείων που ανήκουν σε κάθε κελί του πλέγματος. Αυτό οδηγεί αφενός σε πολύ χαμηλότερη πολυπλοκότητα του αλγορίθμου (ανάλογη με τα κελιά του πλέγματος και όχι με τα στοιχεία), αλλά ταυτόχρονα τον κάνει επιρρεπή σε σφάλματα λόγω χαμηλής ακρίβειας.

Αλγόριθμοι συσταδοποίησης βασισμένοι στην πυκνότητα: Δημιουργούνται συστάδες στις περιοχές όπου υπάρχει μεγαλύτερη πυκνότητα στοιχείων. Αυτοί οι αλγόριθμοι έχουν το πλεονέκτημα της καλής εξουδετέρωσης θορύβου, αφού οι αποκομμένες τιμές (outliers) δεν δημιουργούν περιοχές υψηλής πυκνότητας, οπότε και δεν δημιουργούν συστάδες. Στον αντίποδα όμως, πολλές φορές αδυνατούν να διαχωρίσουν γειτονικές συστάδες, καθώς αναγνωρίζουν την γενική πυκνότητα της περιοχής, και άρα μία ενιαία συστάδα.

Φασματικοί αλγόριθμοι συσταδοποίησης: Συσταδοποίηση που βασίζεται σε λογική ομοιότητας των στοιχείων. Πρακτικά, δημιουργούνται γράφοι, και αν η ομοιότητα μεταξύ των στοιχείων δύο γράφων περνάει ένα δοσμένο κατώφλι, τότε ο αλγόριθμος συνδέει τις κορυφές τους, δημιουργώντας έναν νέο και ενιαίο γράφο. Οι αλγόριθμοι φασματικής συσταδοποίησης έχουν πολύ μεγάλη ανοχή στα σφάλματα, αλλά στον αντίποδα έχουν πολύ υψηλή πολυπλοκότητα (οπότε και είναι ασύμφοροι για συσταδοποίηση μεγάλων συλλογών)[35].

Κλείνοντας, θα πρέπει να τονιστεί, πως για την συσταδοποίηση μιας συλλογής δεδομένων, ανεξαρτήτως του αλγόριθμου που θα επιλεγεί, θα πρέπει τα στοιχεία που την απαρτίζουν να είναι σε δομημένη ή ημι-δομημένη μορφή, ώστε να μπορούν να υποστούν επεξεργασία τα στοιχεία της. Στο επόμενο κεφάλαιο θα γίνει εκτεταμένη

παρουσίαση του τι συνιστά αδόμητα, ημι-δομημένα και δομημένα δεδομένα, καθώς και τις μεθόδους μετάβασης από τη μία κατάσταση στην άλλη.

ΚΕΦΑΛΑΙΟ 3

ΔΟΜΗΜΕΝΑ, ΗΜΙΔΟΜΗΜΕΝΑ ΚΑΙ ΑΔΟΜΗΤΑ ΔΕΔΟΜΕΝΑ

3.1. Δομημένα Δεδομένα

Ως δομημένα ορίζονται τα δεδομένα τα οποία ακολουθούν ένα προκαθορισμένο σχεσιακό σχήμα, συνήθως πινακοειδούς μορφής[36]. Στην κατηγορία αυτή εμπίπτουν οι βάσεις δεδομένων, τα λογιστικά φύλλα, τα αρχεία csv ή μία συλλογή με πίνακες (διανύσματα). Χαρακτηρίζονται από μεγάλο βαθμό οργάνωσης, και σταθερή μορφή από στοιχείο σε στοιχείο. Ως εκ τούτου, και επειδή υπάρχουν (και έχουν εξελιχθεί) πολλά χρόνια στην αγορά εργαλεία ανάλυσης τους, είναι ο ευκολότερα επεξεργάσιμος τύπος δεδομένων από μία εφαρμογή υπολογιστή. Η ανάλυση τους γίνεται βάσει της σύνταξης τους η οποία είναι προκαθορισμένη (και γνωστή στο αντίστοιχο λογισμικό) και του μεγέθους τους (καθώς δηλώνεται στο λογισμικό κατά της αρχικοποίηση και είναι σταθερό). Εξαιτίας αυτής της σταθερής μορφής τους δεν είναι αρκετά ευέλικτα και εμφανίζουν μικρό βαθμό επεκτασιμότητας. Παράδειγμα της έλλειψης ευελιξίας είναι ένας τηλεφωνικός κατάλογος που αποτυπώνει χρήστες με σταθερό και κινητό τηλέφωνο, καθώς αν σε μία νέα καταχώρηση ένας χρήστης έχει και δεύτερο κινητό τηλέφωνο, θα πρέπει να δημιουργηθεί επιπλέον ιδιότητα για όλους τους χρήστες (Πίνακας 3.1)

	A	B	C
1	Όνομα	Σταθερό	Κινητό
2	Γιάννης	2101111111	6901234567
3	Βασίλης	2110000333	6907654321
4	Νίκος	2101231231	6907777777

	A	B	C	D
1	Όνομα	Σταθερό	Κινητό	Κινητό Β
2	Γιάννης	2101111111	6901234567	-
3	Βασίλης	2110000333	6907654321	-
4	Νίκος	2101231231	6907777777	-
5	Τάκης	2103456456	6905555555	6904444444

Πίνακας 3.1: Δημιουργείται επιπλέον ιδιότητα για τους 4 πρώτους χρήστες, παρότι είναι αχρείαστη για αυτούς

3.2. Αδόμητα δεδομένα

Ως αδόμητα, ορίζονται τα δεδομένα που δεν διαθέτουν κάποιο σχεσιακό σχήμα / μοτίβο που να τα χαρακτηρίζει[36]. Ως αδόμητα δεδομένα μπορούν να ταυτοποιηθούν τα αρχεία κειμένου, οι φωτογραφίες, τα αρχεία πολυμέσων. Διατηρούν το πλεονέκτημα της ευκολότερης κατανόησης από τον χρήστη σε αντίθεση με άλλους τύπους δεδομένων, καθώς και της μεγάλης ευελιξίας τους (ένα αρχείο κειμένου μπορεί να επεκταθεί πάρα πολύ εύκολα, ένας φάκελος με φωτογραφίες μπορεί άμεσα να αποκτήσει περισσότερες), αλλά στον αντίποδα έχουν το μειονέκτημα της δύσκολης αυτοματοποιημένης διαχείρισης τους από το λογισμικό υπολογιστών. Σε αντίθεση με τα δομημένα δεδομένα όπου η εξαγωγή πληροφορίας γίνεται βάσει ανάλυσης της σύνταξης, η εξαγωγή πληροφορίας στα αδόμητα δεδομένα γίνεται βάσει σημασιολογίας. Αυτό δε σημαίνει βέβαια πως δε γίνεται εξαγωγή πληροφορίας από αδόμητα δεδομένα μέσω λογισμικού, καθώς πέραν των μεθόδων μετατροπής των αδόμητων σε δομημένα δεδομένα (οι οποίοι θα μελετηθούν αναλυτικά στο πλαίσιο της παρούσας εργασίας), υπάρχουν παραδείγματα όπως η εύρεση κάποιας λέξης σε μια ιστοσελίδα. Τέλος θα πρέπει να τονιστεί, πως κάποιες φορές μία συλλογή ή κάποιο αρχείο μπορεί να διαθέτει δομημένα στοιχεία που το απαρτίζουν, αλλά να συνεχίζει να κατατάσσεται στην κατηγορία των αδόμητων δεδομένων. Παράδειγμα αυτής της περίπτωσης μπορεί να είναι ένα αρχείο κειμένου που διαθέτει ενσωματωμένο ένα λογιστικό φύλλο ή ένας συμπιεσμένος φάκελος που μπορεί να απαρτίζεται από εικόνες, αρχεία ήχου και ένα αρχείο βάσης δεδομένων.

3.3. Ημιδομημένα δεδομένα

Προκειμένου να καμφθούν οι δυσκολίες που προκύπτουν από τα μειονεκτήματα των δομημένων και αδόμητων δεδομένων, αλλά και να χρησιμοποιηθούν τα πλεονεκτήματα τους, τα τελευταία χρόνια έχει αναπτυχθεί ένας “ενδιάμεσος” τύπος, τα ημιδομημένα δεδομένα. Το πιο διαδεδομένα παραδείγματα ημιδομημένων δεδομένων είναι τα αρχεία διαχείρισης αντικειμένων XML και JSON. Διατηρούν ένα είδος δομής (όχι όμως πινακοειδούς) ώστε να είναι αναλύσιμα από το αντίστοιχο λογισμικό επεξεργασίας, όμως η φύση αυτής της δομής τα κάνει πολύ πιο ευέλικτα και επεκτάσιμα, σε αντίθεση με τα πλήρως δομημένα δεδομένα. Για παράδειγμα, ένα JSON αρχείο μπορεί είναι συμβατό με πολλούς τύπους δεδομένων όπως τα ζεύγη κλειδιών-τιμών ή διανύσματα (προσφέροντας τη δυνατότητα προσθήκης καταχωρήσεων χωρίς να απαιτείται η γενική αλλαγή της μορφολογίας, όπως θα γινόταν στην περίπτωση μιας πινακοειδούς βάσης δεδομένων). Στον αντίποδα, η επεξεργασία τους είναι δυσκολότερη σε επίπεδο λογισμικού, καθώς πέραν του μικρότερου χρόνου ύπαρξης τους (σε αντίθεση με τα πλήρως δομημένα δεδομένα) που συνεπάγεται λιγότερη ανάπτυξη του αντίστοιχου λογισμικού, κάθε πλατφόρμα και γλώσσα προγραμματισμού έχει διαφορετικό τρόπο πρόσβασης σε αυτά.

3.4. Ένα απλό παράδειγμα

Ακολουθεί ένα απλουστευμένο παράδειγμα επεξεργασίας και εξαγωγής της ίδιας πληροφορίας από αδόμητα, δομημένα και ημιδομημένα δεδομένα.

Αδόμητα δεδομένα

Έστω το παρακάτω κείμενο:

“Ο Γιάννης και ο Τάκης πήγανε για αγορές στο ίδιο κατάστημα. Ο Γιάννης αγόρασε 2 πακέτα μακαρόνια, 1 μπουκάλι αναψυκτικό, 3 κρουασάν και 2 κουτιά μπισκότα. Ο Τάκης αγόρασε 1 πακέτο μακαρόνια, 3 σαπούνια, 1 κρουασάν και 1 οδοντόβουρτσα.”

Όπως είναι αναμενόμενο, το παραπάνω κείμενο είναι πλήρως αναγνώσιμο και κατανοητό από έναν άνθρωπο, καθώς είναι γραμμένο σε φυσική γλώσσα. Αντιθέτως, είναι πολύ δύσκολο να εξαχθεί η ακριβής πληροφορία από μία μηχανή, καθώς θα πρέπει να γίνει συντακτική και γραμματική ανάλυση, να αφαιρεθούν κοινότυπες λέξεις και σημεία στίξης και τέλος να γίνει σημασιολογική ανάλυση. Η ευελιξία του παραπάνω κειμένου είναι μεγάλη, καθώς μπορεί ανά πάσα στιγμή να προστεθεί κάποιο επιπλέον προϊόν στις παραπάνω λίστες ή μια ολόκληρη υποκατηγορία προϊόντων (για παράδειγμα αν υπήρχαν 2 τύποι μπισκότων).

Δομημένα δεδομένα

Το ίδιο παράδειγμα που παρουσιάστηκε παραπάνω, αν ήταν πλήρως δομημένο θα είχε την παρακάτω μορφή:

	A	B	C	D	E	F	G
1	Όνομα	Μακαρόνια	Αναψυκτικά	Κρουασάν	Μπισκότα	Σαπούνια	Οδοντόβουρτσες
2	Γιάννης	2	1	3	2	0	0
3	Τάκης	1	0	1	0	3	1

Πίνακας 3.2: - Πινακοειδής (δομημένη) απεικόνιση

Η δομή αυτή, είναι η καταλληλότερη για την εξαγωγή τιμών μέσω κάποιου προγράμματος, καθώς αρκεί το όνομα του αγοραστή και το προϊόν. Επιπλέον,

μπορούν προγραμματιστικά να παραμετροποιηθούν περαιτέρω τα αιτήματα, όπως “δείξτε μου πόσα πακέτα μακαρόνια πήραν οι αγοραστές που πήραν 1 κρουασάν”. Όμως, όπως γίνεται αντιληπτό, επειδή οι δύο αγοραστές δεν πήραν προϊόντα από τις ίδιες κατηγορίες, υπάρχουν καταχωρήσεις που είναι περιττές (οι σημειωμένες με μηδέν), και των οποίων ο αριθμός μεγαλώνει κατά πολύ στην περίπτωση που αυξάνονται τα διαθέσιμα προϊόντα (στήλες). Επίσης, στην περίπτωση όπου ένας αγοραστής θελήσει ένα νέο προϊόν, θα πρέπει να προστεθεί ολόκληρη στήλη, ακόμα και αν είναι ο μοναδικός που το προτίμησε. Ακόμη, αν χρειαστεί να δημιουργηθούν υποκατηγορίες προϊόντων η μορφολογία αυξάνει την πολυπλοκότητά της, καθώς θα έπρεπε να προστεθούν πολλές επιπλέον στήλες ή να απαρτίζεται από πολλαπλούς πίνακες που θα συνενώνονται μέσω δεικτών. Τέλος, αν και το παραπάνω παράδειγμα είναι ευανάγνωστο από τον άνθρωπο, σε πολυπλοκότερες περιπτώσεις (όπως πολλαπλών πινάκων) η ανάγνωση γίνεται κατά πολύ δυσκολότερη.

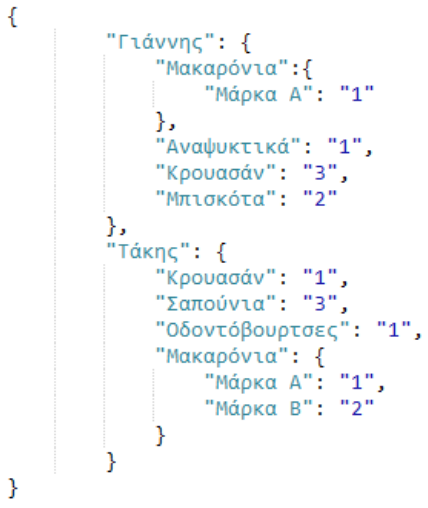
Ημιδομημένα δεδομένα

Όπως αναφέρθηκε παραπάνω, μία περίπτωση ημιδομημένων δεδομένων είναι η μορφή των αρχείων JSON. Το παραπάνω παράδειγμα σε μορφή JSON:

```
{
    "Γιάννης": {
        "Μακαρόνια": "2",
        "Αναψυκτικά": "1",
        "Κρουασάν": "3",
        "Μπισκότα": "2"
    },
    "Τάκης": {
        "Μακαρόνια": "1",
        "Κρουασάν": "1",
        "Σαπούνια": "3",
        "Οδοντόβουρτσες": "1"
    }
}
```

Εικόνα 3.1: Ημιδομημένη μορφή

Αρχικά μπορεί να γίνει η παρατήρηση πως δεν υπάρχουν περιττές καταχωρήσεις. Αν και θα μπορούσαν να γίνουν καταχωρήσεις για μηδενικές ή “άδειες” (null) τιμές, δεν είναι απαραίτητο αν δεν απαιτείται ρητά (για λόγους ομοιομορφίας ή σύνταξης) από τον αλγόριθμο αξιοποίησης των δεδομένων. Η παραπάνω μορφή επίσης είναι ευανάγνωστη, ανεξάρτητα από τον αριθμό καταχωρήσεων ή τον αριθμό των ιδιοτήτων (κατά αντιστοιχία γραμμές και στήλες στην πινακοειδή μορφή). Αυτή την ευκολία ανάγνωσης, τη διατηρεί και στην περίπτωση υποκατηγοριών, καθώς, αν υποτεθεί πως υπάρχουν παραπάνω από μία επιλογές στην κατηγορία “μακαρόνια”, παίρνει την κάτωθι μορφή:



Εικόνα 3.2: υποκατηγορίες σε ένα αρχείο JSON

Πέραν της ευκολίας ανάγνωσης, γίνεται αντιληπτή και η ευελιξία των ημιδομημένων δεδομένων, καθώς όπως αναφέρθηκε, η παραπάνω μετατροπή της μεταβλητής “μακαρόνια” σε αντικείμενο με δικές του μεταβλητές σε μια πινακοειδή μορφή θα αντιστοιχούσε είτε σε προσθήκη επιπλέον στηλών, είτε σε χρήση δεύτερου πίνακα. Στον αντίποδα, αν και οι διεπαφές προγραμματισμού (APIs) των ημιδομημένων δεδομένων έχουν εξελιχθεί τα τελευταία χρόνια, δεν προσφέρουν προγραμματιστικά τη σταθερότητα μίας ισχυρής και ευρετηριασμένης βάσης δεδομένων.

Εν κατακλείδι, τα χαρακτηριστικά των παραπάνω τριών τύπων δεδομένων μπορούν να συνοψιστούν ως εξής:

Πίνακας 3.3

	Ευελιξία	Ευκολία Αυτοματοποιημένης Επεξεργασίας	Αναγνωσιμότητα
Δομημένα Δεδομένα	Μεσαία	Υψηλή	Χαμηλή (για μεγάλο όγκο δεδομένων)
Ημιδομημένα Δεδομένα	Υψηλή	Μεσαία	Υψηλή
Αδόμητα Δεδομένα	Υψηλή	Χαμηλή	Υψηλή

ΚΕΦΑΛΑΙΟ 4

ΕΞΟΡΥΞΗ ΔΕΔΟΜΕΝΩΝ

Ως εξόρυξη δεδομένων μπορεί γενικώς να οριστεί η διαδικασία κατά την οποία εξάγεται χρήσιμη πληροφορία και μοτίβα από μεγάλες συλλογές δεδομένων[37]. Η εξόρυξη δεδομένων μπορεί να χωριστεί σε δύο βασικές κατηγορίες[39]:

- Εξόρυξη δεδομένων περιγραφικού χαρακτήρα
- Εξόρυξη δεδομένων προγνωστικού χαρακτήρα

Περιγραφική εξόρυξη δεδομένων: Αναζητούνται περιγραφικά και σχεσιακά χαρακτηριστικά των δεδομένων. Τα χαρακτηριστικά αυτά συνήθως δεν είναι προκαθορισμένα βάσει κάποιας τιμής σύγκλισης, αλλά προκύπτουν από τις σχέσεις και τη σύγκλιση των ίδιων των δεδομένων. Για παράδειγμα, στην περίπτωση τυχαίων κειμένων που ανήκουν σε διαφορετικές κατηγορίες, δεν αναζητείται η ομοιότητα βάσει προκαθορισμού συγκεκριμένων ομάδων, αλλά αντίθετα, αυτές προκύπτουν μετά την τελική ομαδοποίηση των δεδομένων. Πάραυτα, υπάρχουν και ορισμένες περιπτώσεις που η ομαδοποίηση αρχικοποιείται με προκαθορισμένα τα χαρακτηριστικά (και τις αντίστοιχες ομάδες) βάσει των οποίων αυτή θα γίνει. Η κατηγοριοποίηση της πρώτης μορφής ονομάζεται μη εποπτευόμενη, ενώ της δεύτερης εποπτευόμενη[38].

Προγνωστική εξόρυξη δεδομένων: Αναζητούνται σχέσεις μεταξύ ανεξάρτητων μεταβλητών (μεμονωμένων ή συνόλων) καθώς και σχέσεις μεταξύ εξαρτημένων και ανεξάρτητων μεταβλητών. Πρακτικά, η προγνωστική εξόρυξη δεδομένων καλείται να δώσει απαντήσεις σε ερωτήματα της μορφής “Εάν το συμβάν Α (ή η συλλογή συμβάντων Α) έχει το αποτέλεσμα Β, το συμβάν Γ τι αποτέλεσμα θα έχει;” Όπως αναφέρθηκε στην παράγραφο 2.2, η προγνωστική εξόρυξη δεδομένων έχει διάφορες εφαρμογές στην καθημερινότητα, όπως στους τομείς της υγείας και της οικονομίας.

Γίνεται κατανοητό, πως η παραπάνω κατηγοριοποίηση της εξόρυξης δεδομένων, ορίζεται βάσει της αντίληψης αυτής από τον ενδιαφερόμενο χρήστη / ειδικό, και όχι από την εκτέλεση της σε υπολογιστικό επίπεδο, καθώς αυτές πολλές φορές οι δύο κατηγορίες αλληλοκαλύπτονται. Δηλαδή πολλές φορές, για να γίνει εξόρυξη προγνωστικού χαρακτήρα, θα πρέπει να έχει προηγηθεί κατηγοριοποίηση περιγραφικού χαρακτήρα. Για παράδειγμα, αν ζητηθεί από έναν αλγόριθμο να προβλέψει αν η συμπεριφορά ενός πελάτη τραπέζης μπορεί να οδηγήσει σε απάτη, θα πρέπει να έχει ήδη προηγηθεί ομαδοποίηση των ύποπτων και μη συμπεριφορών.

4.1. Διαδικασία εξόρυξης δεδομένων

Τα κοινότερα βήματα που ακολουθούνται κατά τη διαδικασία εξόρυξης δεδομένων είναι τα εξής[37]:

- Εξερεύνηση
- Εύρεση / ταυτοποίηση μοτίβου
- Ανάπτυξη

Εξερεύνηση: Είναι το βήμα, κατά τη διάρκεια του οποίου τα δεδομένα, κυρίως αδόμητα ή μείξη δομημένων και αδόμητων, συλλέγονται και μετατρέπονται στην επιθυμητή προς επεξεργασία μορφή (ημιδομημένα, δομημένα), ώστε να είναι ευκόλως αναλύσιμα από τον αλγόριθμο που έχει επιλεγεί. Πέραν της διαδικασίας της μετατροπής, σε αυτό το βήμα ανήκει και η διαδικασία της εκκαθάρισης, κατά την οποία απαλείφονται στοιχεία των δεδομένων τα οποία έχουν μηδαμινή αξία εξόρυξης (όπως για παράδειγμα άρθρα και σύνδεσμοι σε κείμενα και κενές περιοχές σε εικόνες). Η διαδικασία της εξερεύνησης κατά τη διάρκεια εξόρυξης δεδομένων μπορεί να χωριστεί στις παρακάτω υπο-διαδικασίες: [37][39]

- Ενσωμάτωση δεδομένων
- Επιλογή
- Εκκαθάριση
- Μετατροπή

Εύρεση / ταυτοποίηση μοτίβου: Κατά τη διάρκεια αυτού του βήματος, γίνεται τοποθέτηση των στοιχείων των δεδομένων σε συγκεκριμένα μοτίβα, βάσει επιλεγμένων χαρακτηριστικών γνωρισμάτων τους. Μία από τις πιο διαδεδομένες μορφές ταυτοποίησης μοτίβου είναι η συσταδοποίηση, όπως αυτή παρουσιάστηκε στην παράγραφο 2.4.3. Πέραν της συσταδοποίησης υπάρχουν και άλλες μορφές ομαδοποίησης των δεδομένων, των οποίων η κατά περίπτωση επιλογή εξαρτάται από το είδος της εξόρυξης δεδομένων που προτιμάται (περιγραφικού ή προγνωστικού χαρακτήρα). Δύο από τις πιο κοινές (σε εφαρμογή αλλά και σε βιβλιογραφικές αναφορές) μεθόδους ευρέσεως μοτίβου είναι η κατάταξη και η εξαγωγή ακολουθιακών μοτίβων[37][38].

Η κατάταξη, όπως και η συσταδοποίηση, είναι μέθοδος δημιουργίας ομάδων, με την ειδοποιό διαφορά την χρήση αρχικοποιημένων χαρακτηρισμών βάσει των οποίων γίνεται η σύγκλιση (εποπτευόμενη ομαδοποίηση) σε αντίθεση με τις μεθόδους συσταδοποίησης (μη εποπτευόμενη ομαδοποίηση)[38]. Υπάρχουν διάφορες υποκατηγορίες κατάταξης, όπως η κατάταξη βάσει δένδρων αποφάσεων, οι μηχανές διανυσμάτων υποστήριξης, ή η σχεσιακή κατάταξη[37]. Τόσο η κατάταξη, όσο και η συσταδοποίηση, ανήκουν στην περιγραφική εξόρυξη δεδομένων.

Η εξαγωγή ακολουθιακών μοτίβων αντίθετα, ανήκει στην προγνωστική εξόρυξη δεδομένων. Έχει ως στόχο την ανακάλυψη υποακολουθιών που δύναται να δημιουργήσουν συγκεκριμένα συμπεριφοριστικά μοτίβα σε ακολουθιακά δεδομένα (χρονικά ή μη, η χρήση της έννοιας “ακολουθιακά δεδομένα” υπονοεί απλά τη σειριακή εκτέλεση κάποιων συμβάντων), ώστε να μπορεί να προβλεφθεί σε μελλοντικό χρόνο η συμπεριφορά των στοιχείων ενός συστήματος[37].

Ανάπτυξη: Στο τελικό βήμα της εξόρυξης δεδομένων τα μοτίβα που ταυτοποιήθηκαν προηγουμένως χρησιμοποιούνται για την εξαγωγή συμπερασμάτων (όπως για παράδειγμα η κατηγοριοποίηση ειδησεογραφικών άρθρων).

4.2. Μορφές δεδομένων και εφαρμογές

Ο κλάδος της εξόρυξης βρίσκει εφαρμογή σε διάφορα είδη δεδομένων, με τις πιο διαδεδομένες από αυτές να είναι τα κείμενα, οι εικόνες και οι γράφοι[38], κυρίως λόγω της παλαιότητας τους αλλά και της ευρείας χρήσης τους στην επιστήμη των υπολογιστών.

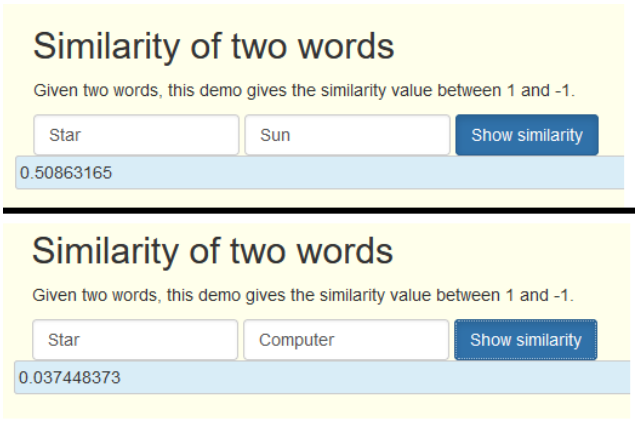
4.2.1. Εξόρυξη δεδομένων σε κείμενα

Η εξόρυξη δεδομένων από ένα κείμενο ή μία συλλογή κειμένων (μη δομημένης μορφής) μπορεί να αποσκοπεί στην εξαγωγή διαφόρων συμπερασμάτων, όπως ακριβώς η ανάλυση του από έναν αναγνώστη[40]. Συμπεράσματα όπως η εξαγωγή πληροφοριών βάσει λέξεων ή φράσεων που θεωρούνται “κλειδιά”, εύρεση θεματολογίας του κειμένου, δημιουργία σύνοψης ή κατηγοριοποίηση. Από τις πιο διαδεδομένες μεθόδους μετατροπής των αδόμητων κειμένων σε δομημένη - και άρα επεξεργάσιμη και αναλύσιμη – μορφή είναι η αναπαράστασή τους σε πολυδιάστατα διανύσματα, με τον αριθμό των διαστάσεων τους να είναι ίσος με τον αριθμό επιλεγμένων λέξεων από κάποιο λεξικό.

Αναλόγως του προς εξαγωγή συμπεράσματος της ανάλυσης χρησιμοποιείται και διαφορετικού είδους μέθοδος εξόρυξης δεδομένων. Συγκεκριμένα, αν ένα σύνολο

κειμένων χρειαστεί ομαδοποίηση των στοιχείων του βάσει ομοιότητας, χωρίς να υπάρχει η ανάγκη για εύρεση της θεματολογίας, θα χρησιμοποιηθούν μέθοδοι περιγραφικής μη εποπτευόμενης εξόρυξης. Αν χρειαστεί η εύρεση της θεματολογίας των στοιχείων που απαρτίζουν ένα σύνολο κειμένων, με τη βοήθεια κειμένων γνωστής θεματολογίας που λειτουργούν ως δεδομένα εκπαίδευσης (training data), τότε θα επιλεχθεί κάποια μέθοδος περιγραφικής εποπτευόμενης εξόρυξης. Τέλος, υπάρχουν και περιπτώσεις ανάλυσης κειμένων βασισμένες σε προγνωστικές μεθόδους εξόρυξης δεδομένων, όπως οι προτάσεις αυτόματης συμπλήρωσης κειμένου σε μηχανές αναζήτησης ή, τελευταία, σε πληκτρολόγια κινητών τηλεφώνων[41].

Μία άλλη μέθοδος εξόρυξης δεδομένων σε μορφή κειμένου, είναι η “ενσωμάτωση λέξεων” (word embedding). Σε αυτή την περίπτωση, δημιουργείται πάλι πολυδιάστατος διανυσματικός χώρος βάσει λεξικού, όμως το κάθε διάνυσμα πλέον αντιπροσωπεύει μία λέξη αντί για ολόκληρο κείμενο. Έτσι προκύπτουν σχεσιακές μετρικές μεταξύ των λέξεων, όπως μετρικές ομοιότητας και αντιστοίχισης[42].



Εικόνα 4.1: Παράδειγμα ενσωμάτωσης λέξεων, Πανεπιστήμιο Turku, Φινλανδία[43]

Η εξόρυξη δεδομένων από κείμενα βρίσκει πολλές εφαρμογές στον τομέα των Μαζικών Δεδομένων, όπως την εκπαίδευση (κατηγοριοποίηση βιβλίων, προτάσεις βιβλιογραφίας προς τους φοιτητές), την ψυχαγωγία (αναζήτηση ταινιών σε πλατφόρμες streaming βάσει της περιγραφής ήδη προβεβλημένων πολυμέσων), την ενημέρωση (θεματική κατηγοριοποίηση αρθρογραφίας), την οικονομία (ανάλυση απόψεων με σκοπό την πρόγνωση της κίνησης της αγοράς, επισκόπηση ερωτηματολογίων των υπαλλήλων ενός οργανισμού) καθώς και αμιγώς την επιστήμη των υπολογιστών (προτάσεις αναζήτησης, αυτόματη συμπλήρωση φράσεων σε μηνύματα, ανάλυση σχολίων και αναρτήσεων σε μέσα κοινωνικής δικτύωσης, ανίχνευση ανεπιθύμητης αλληλογραφίας). Είναι δε τόσο διαδεδομένη η ανάλυση

κειμένων με σκοπό την εξαγωγή χρήσιμης πληροφορίας από αυτά, ώστε να μπορεί και η ίδια να χωριστεί σε περαιτέρω κατηγορίες, βάσει των εφαρμογών της[40]:

- Εξόρυξη πληροφορίας από κείμενα για τη διαχείριση ανθρωπίνου δυναμικού
- Εφαρμογές εξόρυξης πληροφορίας κειμένων στον τομέα των πελατειακών σχέσεων και της ανάλυσης της αγοράς
- Εφαρμογές στον τομέα της τεχνολογίας
- Επεξεργασία φυσικής γλώσσας / πολυγλωσσική ανάλυση

4.2.2. Εξόρυξη δεδομένων σε εικόνες

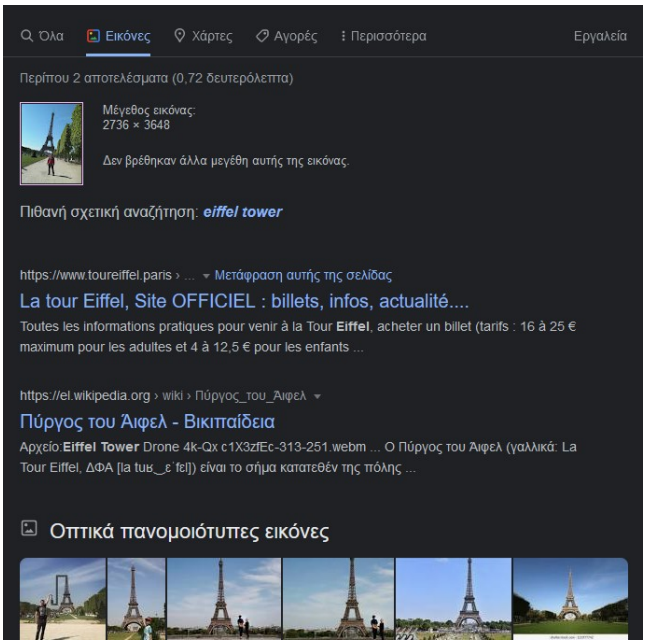
Βάσει των αναμενόμενων αποτελεσμάτων, η εξόρυξη δεδομένων από εικόνες δείχνει τελείως διαφορετική από την αντίστοιχη των κειμένων. Στην πραγματικότητα όμως, τόσο η διαδικασία όσο και η αναλυτική λογική των δύο μεθόδων έχουν πολλά κοινά. Αρχικά, μία από τις πιο διαδεδομένες μορφές δομημένων δεδομένων στην οποία μετατρέπεται μία εικόνα (ή μέρος αυτής) είναι αυτή του ιστογράμματος, από το οποίο προκύπτουν πολυδιάστατα διανύσματα[38][44]. Επίσης, μπορούν να εξαχθούν διάφορα συμπεράσματα από την ανάλυση μιας εικόνας, όπως η άμεση σύγκριση δύο εικόνων, η κατάτμηση μίας εικόνας ώστε να ανιχνευθούν συγκεκριμένα αντικείμενα, ή να βρεθούν κάποια ακολουθιακά μοτίβα τα οποία ακολουθεί η απεικόνιση της.

Γίνεται αντιληπτό πως και σε αυτή την περίπτωση, ανάλογα με το είδος του προς εξαγωγή συμπεράσματος, ποικίλουν οι μορφές εξόρυξης δεδομένων που χρησιμοποιούνται. Στην περίπτωση της θεματικής κατηγοριοποίησης μίας εικόνας συγκριτικά με μία χαρακτηρισμένη συλλογή θα λάβει χώρα περιγραφική εποπτευόμενη εξόρυξη, ενώ στην περίπτωση της ανάλυσης μίας εικόνας μιας συλλογής κυττάρων για να ανιχνευθεί η ανάπτυξη καρκινικού όγκου θα χρησιμοποιηθεί προγνωστική εξόρυξη δεδομένων.

Η ανάλυση εικόνων με σκοπό την εξόρυξη πληροφορίας από αυτές, βρίσκει εφαρμογή σε πολλές κατηγορίες του τομέα των Μαζικών Δεδομένων, όπως η ιατρική (ανάλυση απεικονίσεων από μαγνητικούς τομογράφους και ανίχνευση ανωμαλιών) [38], την ψυχαγωγία και την εκπαίδευση (εύρεση μη εξουσιοδοτημένης χρήσης πολυμέσων στο διαδίκτυο)[45], ή την επεξεργασία δορυφορικών φωτογραφιών (ανίχνευση πυρκαγιών). Επίσης, υπάρχουν εφαρμογές της εξόρυξης από εικόνες που αφορούν την ίδια την επιστήμη της πληροφορικής, όπως η αντίστροφη αναζήτηση εικόνων σε μηχανές αναζητήσεως ή η ρομποτική όραση[46].

Κάποιες φορές, ανάλογα την εφαρμογή και τις απαιτήσεις, μπορεί να λαμβάνουν χώρα ταυτόχρονα παραπάνω από μία μέθοδοι εξόρυξης πληροφοριών από εικόνες. Στο παρακάτω παράδειγμα (εικόνα 4.2) γίνεται ταυτόχρονα αναζήτηση εικόνων με

πλήρη ομοιότητα με αυτή του δείγματος (“δε βρέθηκαν άλλα μεγέθη αυτής της εικόνας”), κατάτμηση της εικόνας και εύρεση αντικειμένων σε αυτή (“πιθανή σχετική αναζήτηση: eiffel tower”), και εύρεση ακολουθιακών μοτίβων σε αυτή (“οπτικά πανομοιότυπες εικόνες”).



Εικόνα 4.2: αντίστροφη αναζήτηση εικόνας με τη βοήθεια της μηχανής αναζήτησης Google

4.2.3. Εξόρυξη δεδομένων σε γράφους

Η εξόρυξη δεδομένων σε γράφους είναι πρακτικά μία υποκατηγορία της εξαγωγής ακολουθιακών μοντέλων[38]. Έχει ως στόχο την εύρεση συγκεκριμένων υπο-γράφων που εμφανίζονται συχνά, είτε σε κάποια μεγάλη συλλογή γράφων είτε σε κάποιο μεγαλύτερο από αυτούς γράφο[38], την μελέτη των μονοπατιών που απαρτίζουν τον γράφο, ή την εγγύτητα των κορυφών του[47]. Επειδή οι γράφοι έχουν φύσει κάποια συγκεκριμένη δομή (ημιδομημένα δεδομένα), είναι πιο εύκολο να δεχθούν επεξεργασία από κάποιον αλγόριθμο εξαγωγής πληροφοριών.

Όπως και στην περίπτωση των κειμένων και εικόνων, υπάρχουν διάφορες μέθοδοι εξόρυξης δεδομένων, ανάλογα με το είδος του προς εξαγωγή συμπεράσματος. Συγκεκριμένα, έχουν αναπτυχθεί αλγόριθμοι εύρεσης ελαχίστου μονοπατιού μεταξύ δύο κορυφών, εύρεσης βέλτιστου μονοπατιού σάρωσης ολόκληρου του γράφου, εύρεσης “βαθμών” απόστασης και ελάχιστης εγγύτητας μεταξύ κορυφών, ή εύρεσης “κοινωνιών” εντός του γράφου (ομαδοποίηση συγκεκριμένων κορυφών εντός του

γράφου οι οποίες έχουν περισσότερες συνδέσεις εντός της “κοινωνίας” που δημιουργούν από ότι εκτός).

Η ανάλυση γράφων με σκοπό την εξαγωγή χρήσιμων πληροφοριών έχει ποικίλες εφαρμογές στον τομέα των Μαζικών Δεδομένων, όπως το Internet of Things (εύρεση ελάχιστου μονοπατιού σε συσκευές πλοήγησης), τα μέσα κοινωνικής δικτύωσης (προτάσεις διασύνδεσης χρηστών βάσει της εγγύτητας των επαφών τους), την οικονομία (εύρεση σχέσεων μεταξύ χρηστών ώστε να αποφευχθεί μία απόπειρα απάτης), τις συγκοινωνίες (υπολογισμών χρόνου αφίξεως ενός μεταφορικού μέσου) ή τη διασκέδαση (προτάσεις παρακολούθησης περιεχομένου βάσει των προτιμήσεων του χρήστη).

Όπως και στην περίπτωση της εξόρυξης εικόνων, έτσι και στην εξόρυξη δεδομένων από γράφους, υπάρχουν περιπτώσεις που μπορεί να χρησιμοποιηθούν παραπάνω από μία μέθοδοι για την εξαγωγή κάποιου συμπεράσματος. Για παράδειγμα, η εφαρμογή κοινωνικής δικτύωσης “Twitter”, προκειμένου να προτείνει νέες σελίδες προς παρακολούθηση σε κάποιον χρήστη, κάνει ταυτόχρονη χρήση αλγορίθμου εύρεσης βαθμών απόστασης και αλγορίθμου ανίχνευσης “κοινωνιών”[47].

Κλείνοντας, θα πρέπει να ειπωθεί πως ένα σύστημα εξόρυξης δεδομένων, δεν επεξεργάζεται σε συγκεκριμένο χρόνο αναγκαστικά μία μόνο κατηγορία δεδομένων, αλλά μπορεί να αναλύει ταυτόχρονα συνδυασμό αυτών (είτε είναι οι συνήθεις κατηγορίες που αναφέρθηκαν παραπάνω είτε εμπεριέχονται περισσότερες). Για παράδειγμα, έχουν αναπτυχθεί συστήματα όπου εξάγονται χαρακτηριστικά από εικόνες με ταυτόχρονη εξαγωγή πληροφορίας από το κείμενο που τις συνοδεύει, ώστε να γίνεται μια πιο πλήρης επισήμανση / προσθήκη ετικετών σε αυτές[48]. Επίσης, στην περίπτωση των στοχευμένων διαφημίσεων στο διαδίκτυο, μπορεί να εξάγονται πληροφορίες από κείμενα (λέξεις κλειδιά, ιστορικό επισκέψεων και αναζητήσεων) και από γράφους (υπολογισμός εγγύτητας, ανίχνευση “κοινωνιών”) ταυτόχρονα ώστε να προβληθεί η βέλτιστη διαφήμιση[49].

ΚΕΦΑΛΑΙΟ 5

ΣΕΙΡΙΑΚΟΙ, ΠΑΡΑΛΛΗΛΟΙ ΚΑΙ ΚΑΤΑΝΕΜΗΜΕΝΟΙ ΥΠΟΛΟΓΙΣΜΟΙ

5.1. Σειριακή επεξεργασία

Σε ένα σύστημα υπολογιστή όπου καλείται να τρέξει ένας αλγόριθμος ή μία σειρά εντολών, η εκτέλεση αυτή σε επίπεδο μηχανής γίνεται διαδοχικά, εντολή προς εντολή. Η ταχύτητα που θα εκτελεστεί ο αλγόριθμος, εξαρτάται από την συχνότητα συγχρονισμού του ρολογιού του επεξεργαστή. Η αντίστροφη έννοια αυτής της συχνότητας είναι ο κύκλος (περίοδος) ρολογιού του επεξεργαστή, δηλαδή ο χρόνος

μεταξύ δύο ηλεκτρονικών παλμών του. Μία εντολή για να εκτελεστεί χρειάζεται παραπάνω από έναν κύκλο ρολογιού, καθώς σε κάθε κύκλο εκτελείται ένα από τα βασικά στάδια που απαρτίζουν την εντολή (πρόσβαση στη μνήμη, εκτέλεση, εγγραφή κλπ). Ανάλογα της αρχιτεκτονικής των εντολών ενός επεξεργαστή και της συχνότητας εμφάνισης των σταδίων που απαρτίζουν τις εντολές, υπολογίζεται ο μέσος αριθμός κύκλων ανά εντολή. Ιστορικά έχουν αναπτυχθεί μέθοδοι μείωσης αυτού του αριθμού, όπως η διοχέτευση (pipelining) όπου εκτελείται η επόμενη εντολή προτού ολοκληρωθεί η προηγούμενη, με την προϋπόθεση ότι τα στάδια των εντολών που εκτελούνται ταυτόχρονα δε χρησιμοποιούν τους ίδιους καταχωρητές και ότι γίνεται εκτίμηση κινδύνου ώστε να επιτυγχάνεται ορθή σειρά στη χρήση των δεδομένων[50] [51]. Για παράδειγμα, ένας επεξεργαστής τύπου RISC, με αριθμό κύκλων ανά εντολή ίσο με 4 και συχνότητα ρολογιού 1GHz μπορεί να εκτελέσει 250 εκατομμύρια εντολές το δευτερόλεπτο.

Η μείωση του μεγέθους των ημιαγωγών που χρησιμοποιούνται για την κατασκευή ενός επεξεργαστή έχει ως αποτέλεσμα την αύξηση του αριθμού των ημιαγωγών ανά επεξεργαστική μονάδα, και ως εκ τούτου την αύξηση της συχνότητας ρολογιού. Όμως αυτή η αύξηση δεν μπορεί να είναι αέναη, καθώς δημιουργούνται προβλήματα φυσικού περιεχομένου, όπως η υπερβολική έκλυση θερμότητας και η αδυναμία αυτής να διοχετευθεί. Αυτό έχει ως αποτέλεσμα τα τελευταία χρόνια να έχει δημιουργηθεί ένα ανώφλι στην συχνότητα χρονισμού των επεξεργαστών, με αυτή να είναι της τάξης των 3-4 GHz (για οικονομικά διαθέσιμους υπολογιστές)[51]. Έτσι, για να παρακαμφθεί αυτή η τροχοπέδη, δημιουργήθηκαν διαφορετικές αρχιτεκτονικές υπολογιστικών συστημάτων, χρησιμοποιώντας όμως την παρούσα τεχνολογία των επεξεργαστών, όπως τα συστήματα παράλληλης και κατανεμημένης επεξεργασίας.

5.2. Παράλληλη επεξεργασία

Μέχρι και τα μέσα της δεκαετίας του 1990 (σε μικρής και μεσαίας βαθμίδας υπολογιστές), η παραλληλία στους υπολογισμούς που καλούταν να φέρει εις πέρας ένας επεξεργαστής βασιζόταν καθαρά στην μεθοδολογική προσέγγιση της εκτέλεσης των εντολών που απάρτιζαν τον εκάστοτε αλγόριθμο. Αυτού του τύπου μέθοδοι παραλληλίας είναι η διοχέτευση (pipelining) που αναφέρθηκε στην προηγούμενη παράγραφο, ή η αρχιτεκτονική “Single Instruction, Multiple Data” (SIMD) όπου όπως υποδηλώνει και το όνομά της, εκτελείται η ίδια εντολή σε πολλαπλά δεδομένα ταυτόχρονα, δημιουργώντας “νήματα” (threads) παράλληλης επεξεργασίας και μειώνοντας έτσι τον χρόνο εκτέλεσης του αλγορίθμου με ρυθμό ανάλογο του αριθμού δεδομένων ανά εντολή[51].Οι παραπάνω μέθοδοι όμως έβρισκαν περιορισμούς, όπως τον μη ικανοποιητικό λόγο απόδοσης προς κατανάλωση ενέργειας, καθώς πάνω από συγκεκριμένες τιμές, μικρή αύξηση στην απόδοση επέφερε δυσανάλογα μεγάλη αύξηση στην κατανάλωση ενέργειας (και ως έμμεσο αποτέλεσμα την έκλυση θερμότητας λόγω απωλειών στο περιβάλλον). Ένας άλλος περιορισμός που υπήρχε στις παραπάνω αρχιτεκτονικές, ήταν πως τα νήματα που εκτελούνταν παράλληλα χρησιμοποιούσαν την ίδια λειτουργική μονάδα, μειώνοντας έτσι την απόδοση της παραλληλίας.

Η λύση που δόθηκε ώστε να καμφθούν τα παραπάνω εμπόδια, και η οποία εφαρμόζεται μέχρι και σήμερα, είναι η δημιουργία πολυπύρηνων επεξεργαστών, δηλαδή επεξεργαστών που αποτελούνται από πολλαπλές υπολογιστικές μονάδες και οι οποίοι διαχειρίζονται μία κοινή μνήμη (μνήμη του συστήματος). Εκτός από την ύπαρξη πλέον ανεξάρτητων λειτουργικών μονάδων για κάθε πυρήνα, επιλύεται και βελτιστοποιείται και η αναλογία απόδοσης προς κατανάλωση ενέργειας. Σε ένα μονοπύρηνο επεξεργαστή, αν διπλασιαστεί η επιφάνεια του (και άρα ο αριθμός των ημιαγωγών) διπλασιάζεται (γραμμική αναλογία) και η ενέργεια που καταναλώνει. Η απόδοση του όμως δεν διπλασιάζεται, καθώς υπάρχουν απώλειες για τη μεταφορά του σήματος εσωτερικά του επεξεργαστή, οι οποίες γίνονται μεγαλύτερες όσο μεγαλώνει η επιφάνεια που πρέπει να διασχίσουν. Προσεγγιστικά, η απόδοση του επεξεργαστή είναι ανάλογη με την τετραγωνική ρίζα της επιφάνειας του[51].

Οπότε, αν υποθετικά ένας επεξεργαστής έχει επιφάνεια Α, απόδοση Χ, κατανάλωση ενέργειας Ε, λόγο απόδοσης προς κατανάλωση ενέργειας λ, και τετραπλασιαστεί η επιφάνειά του, τότε:

$$\lambda_{αρχ}=\frac{X}{E} \quad , \text{ με} \quad X=X_c\cdot\sqrt{A} \quad \text{και} \quad E=E_c\cdot A$$

$$\lambda_{τελ}=\frac{X_{τελ}}{E_{τελ}}=\frac{X_c\cdot\sqrt{4\cdot A}}{E_c\cdot 4\cdot A}=\frac{2\,X_c\cdot\sqrt{A}}{4\,E_c\cdot A}=\frac{1}{2}\,\lambda_{αρχ}$$

Ενώ στην περίπτωση τεσσάρων πυρήνων ανά επεξεργαστή, με το μέγεθος της επιφάνειας του κάθε πυρήνα ίσο με το αρχικό μέγεθος του επεξεργαστή:

$$\lambda_{αρχ}=\frac{X}{E}$$

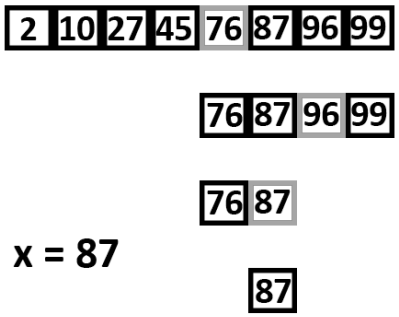
$$\lambda_{τελ}=\frac{X_{τελ}}{E_{τελ}}=\frac{4\,X_c\cdot A}{E_c\cdot 4\cdot A}=\lambda_{αρχ}$$

Στην πρώτη περίπτωση η ζητούμενη αναλογία υποδιπλασιάζεται, ενώ με τη χρήση πολυπύρηνου επεξεργαστή παραμένει σταθερή.

Η κατασκευή ενός αλγόριθμου με σκοπό την σειριακή εκτέλεση του είναι σχετικά σαφής διαδικασία. Η δημιουργία ενός παράλληλου αλγορίθμου ενέχει περισσότερες δυσκολίες, ώστε να βρεθεί η βέλτιστη δομή του. Προφανώς, ο οποιοσδήποτε αλγόριθμος μπορεί να δοθεί σε έναν πολυπύρηνο επεξεργαστή για να εκτελεστεί με

τυχαία κατανομή των υπολογισμών ανά πυρήνα, με αποτέλεσμα η χρονική πολυπλοκότητα του αλγορίθμου να υποβιβάζεται αντιστρόφως ανάλογα με τον αριθμό των πυρήνων. Τις περισσότερες φορές όμως, το πρόβλημα προς επίλυση διαθέτει συγκεκριμένες συμμετρίες, οι οποίες πρέπει να ανιχνευθούν από τον εκάστοτε προγραμματιστή ώστε να προκύψει η υλοποίηση του αλγορίθμου με την ελάχιστη πολυπλοκότητα.

Για παράδειγμα, έστω μία αύξουσα ταξινομημένη λίστα n αριθμών και ένας αλγόριθμος που καλείται να βρει μία συγκεκριμένη τιμή εντός αυτής της λίστας. Η απλούστερη μορφή αυτού του (σειριακού) αλγόριθμου θα ήταν να αναζητήσει αυτόν τον αριθμό σαρώνοντας μία μία τις τιμές έως ότου βρει τη ζητούμενη. Η χρονική πολυπλοκότητα αυτού του αλγορίθμου προφανώς είναι o(n). Η βελτιστοποίηση όμως του σειριακού αλγόριθμου προκύπτει με τη χρήση της μεθόδου της διχοτόμησης. Η λίστα χωρίζεται στα δύο, ελέγχεται αν ο διαχωριστικός αριθμός είναι μικρότερος ή μεγαλύτερος από τον προς αναζήτηση αριθμό, και επιλέγεται αν θα συνεχίσει η αναζήτηση στο άνω ή στο κάτω μισό της λίστας. Η διαδικασία επαναλαμβάνεται, έως ότου βρεθεί η θέση του ζητούμενου αριθμού.



Σχήμα 5.1: αναζήτηση με διχοτόμηση
- με γκρίζο πλαίσιο το σημείο
διαχωρισμού της λίστας

Στο παράδειγμα της εικόνας 5.1, γίνεται αντιληπτό πως ο αλγόριθμος ολοκληρώνεται σε τρία βήματα, για λίστα οκτώ στοιχείων. Με εφαρμογή του κυρίαρχου θεωρήματος υπολογίζεται πως η πολυπλοκότητα του αλγορίθμου είναι $o(\log_2 n)$, το οποίο επαληθεύεται και από το αποτέλεσμα του παραπάνω παραδείγματος ($2^3=8$).

Στην περίπτωση της χρήσης συστήματος παράλληλης επεξεργασίας (όπως είναι ένας πολυπύρηνος επεξεργαστής) πρέπει να γίνουν επιπλέον διερευνήσεις για την εύρεση της βέλτιστης υλοποίησης της παραπάνω αναζήτησης, καθώς υπάρχουν παραπάνω από μία δυνατές υλοποιήσεις.

Έστω ένας επεξεργαστής που διαθέτει τέσσερις πυρήνες και καλείται να κάνει την παραπάνω αναζήτηση. Μία δυνατή προσέγγιση θα ήταν η παραπάνω λίστα να χωριστεί σε 4 τμήματα, και κάθε επεξεργαστής να αναλάβει ένα από αυτά τα κομμάτια, ακολουθώντας την λογική του σειριακού αλγόριθμου. Σε αυτή τη περίπτωση, η χρονική πολυπλοκότητα γίνεται $o(\log_2(n/4))$, η οποία δίνει μια μικρή επιτάχυνση στον αλγόριθμο, αλλά όχι σημαντική (καθώς $\log_2(n/4) = \log_2(n) - \log_2(4) = \log_2(n) - 2$, οπότε από άποψη σύγκλισης παραμένει $o(\log_2 n)$, ειδικά για μεγάλα n). Το μικρό μέγεθος της επιτάχυνσης γίνεται αντιληπτό και λογικά, καθώς μετά τον αρχικό διαχωρισμό, μόνο το κομμάτι που ανήκει σε έναν από τους τέσσερις πυρήνες διαθέτει την σωστή τιμή, ενώ οι υπόλοιποι εκτελούν αχρείαστες συγκρίσεις.

Μία άλλη προσέγγιση, που κάνει ορθή χρήση και των τεσσάρων πυρήνων, θα ήταν να χωριστεί η λίστα σε 4 τμήματα (ένα ανά πυρήνα) και ο κάθε πυρήνας να ελέγχει τη μεσαία τιμή του κομματιού του. Θα υπάρχουν δύο γειτονικά τμήματα, από τα οποία το ένα θα επιστρέψει μεγαλύτερη τιμή, και το άλλο μικρότερη (από την ζητούμενη). Έτσι ανιχνεύεται το τμήμα στο οποίο βρίσκεται εντός η ζητούμενη τιμή, και επαναλαμβάνεται η παραπάνω διαδικασία σε αυτό και επαναλαμβάνεται ο κύκλος. Από άποψη συμμετρίας είναι εύκολα αντιληπτό πως το μέγεθος αυτού του τμήματος είναι το ¼ του τμήματος του προηγούμενου βήματος. Οπότε, πάλι βάσει του θεωρήματος κυριαρχίας, η χρονική πολυπλοκότητα ισούται με

$$T(n)=\log_4(n)=\frac{\log_2(n)}{\log_2(4)}=\frac{\log_2(n)}{2}$$

με γενική αναγωγή της λύσης για p αριθμό επεξεργαστών

$$T(n)=\frac{\log_2(n)}{\log_2(p)}$$

Στη δεύτερη περίπτωση, η επιτάχυνση, αν και δεν είναι η βέλτιστη δυνατή που αναμένεται σε τέτοιου είδους διερευνήσεις, είναι αρκετά σημαντικότερη από αυτή της πρώτης περίπτωσης. Η παραπάνω επιτάχυνση γίνεται περισσότερο αντιληπτή αν συνδυαστεί με πολυνηματική αρχιτεκτονική τύπου SIMD (καθώς σε κάθε παράλληλο τμήμα εκτελούνται οι ίδιες εντολές στον ίδιο χρόνο, αλλά σε διαφορετικά δεδομένα). Για παράδειγμα, σε έναν μεσαίας βαθμίδας επεξεργαστή της Intel, με 6 πυρήνες και 6 SSE SIMD νήματα ανά πυρήνα[52], η παραπάνω επιτάχυνση είναι της τάξεως του 5.16 ($\log_2 36$).

Για την ανάπτυξη παράλληλων αλγορίθμων, έχουν δημιουργηθεί από διάφορους φορείς τα αντίστοιχα προγραμματιστικά πλαίσια. Παραδείγματα αυτών των πλαισίων

είναι το OpenMP, το OpenGL (λογισμικά ανοικτού κώδικα), τα Threading Building Blocks, Cilk Plus και Array Building Blocks της Intel[53].

5.3. Κατανεμημένη Επεξεργασία

Όπως ειπώθηκε στην προηγούμενη παράγραφο, η παράλληλη επεξεργασία αναφέρεται σε υπολογιστικά συστήματα αποτελούμενα από πολυπύρηνους επεξεργαστές κοινής μνήμης. Βάσει αυτής της αρχιτεκτονικής όμως η οποία είναι πεπερασμένη (ανώφλι στον αριθμό των πυρήνων που δύναται να απαρτίζουν έναν επεξεργαστή), μπορούν να λυθούν σε αποδεκτό χρόνο υπολογιστικά προβλήματα τα οποία είναι και αυτά πεπερασμένα. Αναζητήθηκε λοιπόν ένα διαφορετικού τύπου "κατεστημένο" το οποίο να μην έχει κάποιο ανώτατο όριο στην παραλληλία των εργασιών που μπορεί να του ανατεθούν. Όπως αναφέρθηκε στην παράγραφο 2.3.1 αυτή η νέα αρχιτεκτονική ήρθε με τη δημιουργία των συστοιχιών υπολογιστών, και ο κλάδος που αναπτύχθηκε βάσει αυτής είναι ο κλάδος των "συστημάτων κατανεμημένης επεξεργασίας".

Στην κατανεμημένη επεξεργασία, σε αντίθεση με την παράλληλη, το υπολογιστικό σύστημα είναι επεκτάσιμο[54]. Πέραν των στατικών κατανεμημένων συστημάτων τα οποία διαθέτουν προκαθορισμένο αριθμό επεξεργαστικών μονάδων στις οποίες μπορεί ανά πάσα στιγμή να προστεθεί μία νέα μονάδα (ή να αφαιρεθεί μία από τις ήδη υπάρχουσες), υπάρχουν και τα δυναμικά συστήματα. Αυτά τα συστήματα συνήθως δεν αποτελούνται από φυσικά μηχανήματα, αλλά ανάλογα με τις απαιτήσεις των εφαρμογών που καλούνται να εκτελέσουν, δεσμεύουν ή αποδεσμεύουν πόρους από μία μεγάλου μεγέθους συλλογή υπολογιστικών συστημάτων. Ένα παράδειγμα δυναμικού κατανεμημένου συστήματος είναι η παροχή "υποδομής ως υπηρεσία" (infrastructure as a service) όπως αυτή της Amazon[55], όπου δεσμεύεται δυναμικά και βάσει χρήσης για κάθε χρήστη το εκάστοτε μέρος του απαιτούμενου υλικού από μία τεράστια συλλογή συστοιχιών υπολογιστών. Ένα άλλο χαρακτηριστικό παράδειγμα είναι οι υπηρεσίες αποθήκευσης αρχείων στο νέφος. Αν ένας χρήστης έχει κάνει εγγραφή σε μία υπηρεσία που παρέχει χώρο στο νέφος μεγέθους 200GB, δε θα το δεσμεύσει εξαρχής και σε συγκεκριμένη δικτυακή μονάδα αποθήκευσης (καθώς αυτό θα ήταν αντιπαραγωγικό για τον πάροχο), αλλά θα δεσμεύεται επιπλέον χώρος μόνο κατ' απαίτηση. Γίνεται κατανοητό, πως οι κατ' υπηρεσία υποδομές δεν παρέχουν αποκλειστικά συστήματα επεξεργαστικής ισχύος, αλλά και μνήμης, αποθήκευσης, κλπ. Το ίδιο ισχύει και για στατικά κατανεμημένα συστήματα, καθώς σε μία συστοιχία υπολογιστών μπορεί να είναι συνδεδεμένοι υπολογιστές χαμηλής επεξεργαστικής ισχύος αλλά εφοδιασμένο με μεγάλης χωρητικότητας μνήμη ή αποθηκευτικά μέσα (σκληροί δίσκοι).

Η ετερογένεια των κατανεμημένων συστημάτων είναι η μεγαλύτερη αιτία διαφοροποίησης τους από τα συστήματα παράλληλης επεξεργασίας. Για αρχή, ένα κατανεμημένο σύστημα δύναται να εμπεριέχει συστήματα παράλληλης επεξεργασίας, καθώς μπορεί να αποτελείται από υπολογιστές πολυπύρηνων επεξεργαστών. Επίσης, η αρχιτεκτονική ενός παράλληλου συστήματος είναι σαφώς

ορισμένη (πολλαπλοί πυρήνες εντός επεξεργαστή με κοινή μνήμη ή απλή επεξεργαστική μονάδα με δυνατότητα ταυτόχρονης επεξεργασίας διαφορετικών δεδομένων), σε αντίθεση με την αρχιτεκτονική ενός κατανεμημένου συστήματος (ετερογένεια, άμεση επεκτασιμότητα). Αυτό έχει ως αποτέλεσμα να υπάρχουν διαφόρων ειδών κατανεμημένα συστήματα, οι οποίες διαχωρίζονται βάσει της αρχιτεκτονικής και των επιπέδων (διαφάνειας ως προς τον τελικό χρήστη) που διαθέτουν[56]. Συγκεκριμένα διαχωρίζονται στις εξής αρχιτεκτονικές:

- Πελάτη-εξυπηρετητή: Είναι η απλούστερη μορφή, όπου η εφαρμογή που εκτελείται στον υπολογιστή-πελάτη ζητάει την εκτέλεση υπολογισμών από τον υπολογιστή που λειτουργεί ως εξυπηρετητής.
- Πολλαπλών βαθμίδων: Ανάμεσα στο επίπεδο του πελάτη και το επίπεδο που προσφέρει την υποδομή (επεξεργαστική ισχύ, αποθηκευτικό χώρο) υπάρχουν διάφορα μη ορατά επίπεδα, τα οποία αναλαμβάνουν να διαχωρίσουν και να κατανείμουν τις εργασίες. Σε αυτή τη κατηγορία ανήκει η “υποδομή ως υπηρεσία” όπου ο πάροχος μέσω των ενδιάμεσων επιπέδων αντιστοιχεί τις απαιτήσεις των χρηστών σε υλικό που διαθέτει. Η πιο απλουστευμένη μορφή αυτής της κατηγορίας είναι η αρχιτεκτονική τριών επιπέδων (πελάτη, επιχείρησης, υποδομής).
- Στενής ζεύξης: Στην αρχιτεκτονική στενής ζεύξης ένα σύστημα από παράλληλα συνδεδεμένους υπολογιστές εκτελεί την ίδια εργασία χωρισμένη σε ισάριθμα τμήματα. Πρακτικά πρόκειται για συστήματα μεγάλης στατικότητας και μικρής ευελιξίας, όπως για παράδειγμα μία συστοιχία πεπερασμένου αριθμού υπολογιστών ενός ερευνητικού εργαστηρίου.
- Ομότιμων κόμβων: Τα συστήματα ομότιμων κόμβων, ή όπως είναι αγγλιστί γνωστά ως peer-to-peer, είναι συστήματα όπου όλοι οι υπολογιστές λειτουργούν και ως πελάτες αλλά και ως εξυπηρετητές. Στις περιπτώσεις που μπορεί να υπάρχει κάποιος ειδικός εξυπηρετητής λειτουργεί μόνο ως πάροχος ευρετηρίου μεταξύ των κόμβων. Τα συστήματα ομότιμων κόμβων δεν έχουν βαθμίδες αρχιτεκτονικής, καθώς κάθε κόμβος γνωρίζει ανά πάσα στιγμή τους κόμβους στους οποίους αναθέτει (ή του αναθέτουν) διεργασίες προς εκτέλεση.

Αντίστοιχα σε αλγοριθμικό επίπεδο, και συγκεκριμένα στο τμήμα της πολυπλοκότητας και τον υπολογισμό αυτής, η κατανεμημένη επεξεργασία παρουσιάζει διαφοροποιήσεις από την παράλληλη. Πέραν της χρονικής πολυπλοκότητας, η οποία όπως και στην παράλληλη επεξεργασία εξαρτάται από την βελτιστοποίηση του εκάστοτε αλγόριθμου, πρέπει να υπολογιστεί και η πολυπλοκότητα που δημιουργείται λόγω της λήψης και εκπομπής μηνυμάτων[57]. Σε ένα παράλληλο σύστημα, η πολυπλοκότητα λόγω της επικοινωνίας μεταξύ των μονάδων επεξεργασίας (πυρήνες επεξεργαστή) μπορεί να θεωρηθεί αμελητέα, καθώς οι πυρήνες διαθέτουν κοινή μνήμη με την οποία ανταλλάσουν μηνύματα σε επίπεδο μηχανής μέσω διαύλου επικοινωνίας. Σε ένα κατανεμημένο σύστημα αντίθετα, η επικοινωνία γίνεται σε υψηλότερο επίπεδο (λογισμικού) μέσω δικτύου

(τοπικού ή παγκοσμίου) και υπόκειται σε περιορισμούς όπως οι καθυστερήσεις και η περιορισμένη ταχύτητα δικτύου. Η πολυπλοκότητα επικοινωνίας πολλές φορές υπολογίζεται ως συνδυασμός του αριθμού των μηνυμάτων που εκπέμπονται με το αντίστοιχο μέγεθός τους. Έτσι, όταν ένας σειριακός αλγόριθμος μετατρέπεται στον αντίστοιχο κατανεμημένο πρέπει να ερευνηθεί η βελτιστοποίηση του και αναφορικά με την αναλογία της χρονικής πολυπλοκότητας ως προς την πολυπλοκότητα ανταλλαγής μηνυμάτων. Για παράδειγμα, μπορεί ένας αλγόριθμος να κατατμηθεί σε μεγάλο αριθμό κόμβων έτσι ώστε η χρονική πολυπλοκότητα να είναι O(1), αλλά η πολυπλοκότητα επικοινωνίας να είναι τόσο μεγάλη, ώστε συνδυαστικά με τους περιορισμούς του δικτύου να επιβραδύνει τελικώς την εκτέλεσή του.

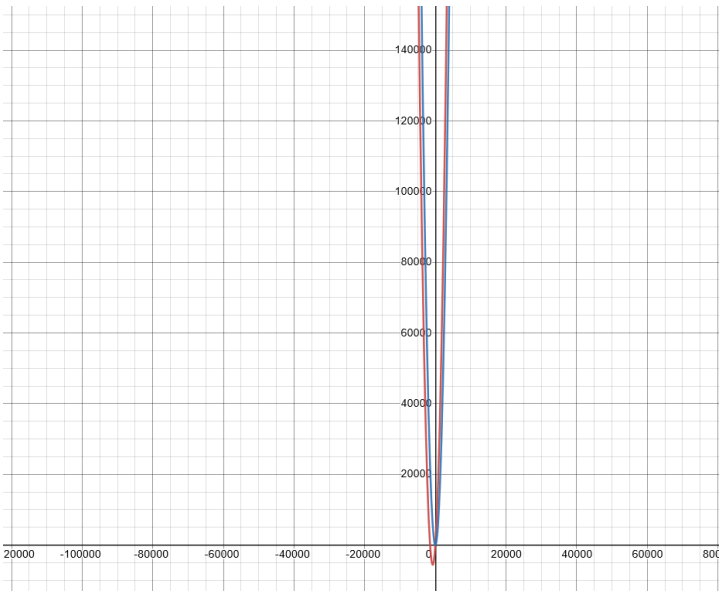
Έστω το πρόβλημα της εύρεσης του μεγίστου σε μία συλλογή αριθμών μεγέθους n, με χρήση κατανεμημένης αρχιτεκτονικής στενής ζεύξης, όπου ένας κόμβος διαθέτει την πλήρη συλλογή. Ο απλούστερος σειριακός αλγόριθμος θα ήταν οι διαδοχικές συγκρίσεις μεταξύ των στοιχείων, κρατώντας κάθε φορά το μέγιστο που προκύπτει καθώς και την θέση του. Αυτή η προσέγγιση είναι και η βέλτιστη, καθώς είναι απαραίτητη η γνώση κάθε στοιχείου, και η πολυπλοκότητα αυτής είναι O(n). Στην περίπτωση της κατανεμημένης αρχιτεκτονικής θα μπορούσε να δοθεί σε κάθε κόμβο ένα μέρος της συλλογής, να βρεθεί το μέγιστο αυτής της συλλογής και να επιστραφούν στον αρχικό κόμβο τα αποτελέσματα για να εκτελέσει τις τελικές συγκρίσεις. Η χρονική πολυπλοκότητα σε αυτή τη περίπτωση είναι O(n/p) και η πολυπλοκότητα επικοινωνίας O(p), όπου p ο αριθμός των κόμβων. Γεννάται όμως το ερώτημα, αν είναι βέλτιστη αυτή η προσέγγιση, καθώς η αποστολή μηνυμάτων έχει μεγαλύτερο “κόστος” από την σύγκριση. Συγκεκριμένα, έστω ένα τοπικό δίκτυο της τάξης του Gigabit, το οποίο μεταφέρει Ethernet πακέτα τα οποία έχουν μέγεθος 1500 bytes[58] . Διαιρώντας την ταχύτητα με το μέγεθος, προκύπτει πως ο χρόνος μετάδοσης ενός πακέτου είναι περίπου (το πακέτο Ethernet φέρει και κάποια επιπλέον μεταδεδομένα) 12μs. Ένας επεξεργαστής χρονισμένος στα 3GHz με μέσο αριθμό κύκλων ανά εντολή ίσο με 4 χρειάζεται περίπου 1,2 ns για να κάνει μία σύγκριση. Οι δύο παραπάνω τιμές έχουν χάσμα της τάξης του 10^4, οπότε η παραπάνω προσέγγιση δίνει επιτάχυνση στους υπολογισμούς μόνο στην περίπτωση όπου η συλλογή είναι τάξης 10^5 ή μεγαλύτερη, και μόνο αν έχουν εξαρχής όλοι οι κόμβοι αντίγραφο της λίστας. Στην προκειμένη περίπτωση, θα ήταν πιο συνετό να χρησιμοποιηθεί μία παράλληλη αρχιτεκτονική αντί κατανεμημένης.

Στον αντίποδα, υπάρχει σημαντική επιτάχυνση κατά την εκτέλεση ενός αλγόριθμου όταν το σύνολο των λογικών πράξεων που κατανέμεται στους κόμβους έχει υψηλή χρονική πολυπλοκότητα σε αναλογία με την καθυστέρηση λόγω δικτυακής εκπομπής μηνυμάτων. Ένα τέτοιο παράδειγμα είναι η ταξινόμηση μίας λίστας αριθμών. Η πολυπλοκότητα της απλούστερης μορφής ταξινόμησης είναι $O(n^2)$, καθώς κάθε αριθμός πρέπει να σαρωθεί και να συγκριθεί με τους υπόλοιπους. Μπορεί όμως, να χρησιμοποιηθεί διαχωρισμός της λίστας με την ίδια λογική που λειτουργεί ο αλγόριθμος Bucketsort, όπου τα στοιχεία της λίστας χωρίζονται σε μικρότερες λίστες (buckets) ανάλογα του εύρους των τιμών τους. Αυτές οι λίστες ταξινομούνται, και μετά ξανασυνενώνονται σε μία μεγάλη λίστα. Τα συγκεκριμένα buckets μπορούν να κατανεμηθούν σε ισάριθμο πλήθος υπολογιστικών μονάδων όπου θα γίνουν παράλληλα οι ταξινομήσεις. Τότε η χρονική πολυπλοκότητα γίνεται $O((n/p)^2)$ όπου p

το πλήθος των υπολογιστικών μονάδων, και η πολυπλοκότητα επικοινωνίας γίνεται O(n) (2n εκπομπές, για την αρχική αποστολή κατά το διαχωρισμό και την τελική αποστολή πριν την συνένωση). Σε πρακτικά νούμερα, λαμβάνοντας υπ όψιν βάσει των παραπάνω υπολογισμών πως η μετάδοση είναι περίπου 10^4 φορές πιο αργή από την λογική πράξη και πως σε κάθε αποστολή μεταδίδονται περίπου 1500 bytes, αν θεωρηθεί πως “α” είναι ο χρόνος εκτέλεσης μιας λογικής πράξης, ο συνολικός χρόνος εκτέλεσης κατά προσέγγιση είναι:

$$T(a)=\left(\frac{a}{p}\right)^2+2*10^4\frac{a}{1500}+a$$

όπου ο πρώτος όρος αντιστοιχεί στις λογικές πράξεις κάθε υπολογιστή, ο δεύτερος στην μετάδοση των δεδομένων και ο τρίτος στην αρχική σάρωση της λίστας ώστε κάθε αριθμός να τοποθετηθεί στην αντίστοιχη υπό-λίστα (bucket). Επειδή η παραπάνω συνάρτηση είναι παραβολικής μορφής, όσο αυξάνεται το α (κατ’ αντιστοιχία το μέγεθος της λίστας), μπορεί ο δεύτερος και ο τρίτος όρος να θεωρηθούν αμελητέοι (Σχήμα 5.2), οπότε προκύπτει επιτάχυνση υπολογισμών της τάξης του p^2.

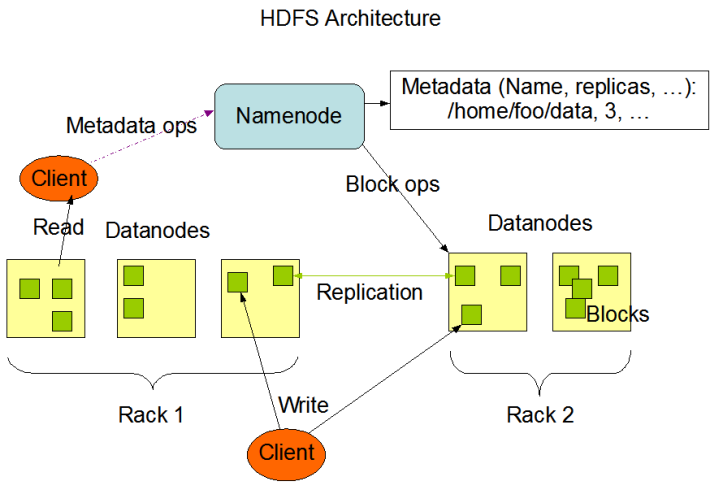


Σχήμα 5.2: ασυμπωτική σύγκλιση για μεγάλες τιμές του α των συναρτήσεων $T(\alpha) = \alpha^2/100$ (μπλε) και $T(\alpha) = \alpha^2/100+15\alpha$ (κόκκινο).

Όπως προαναφέρθηκε η κατανομή των εργασιών σε αντίθεση με την παραλληλία γίνεται σε επίπεδο λογισμικού και όχι μηχανής, οπότε έχουν δημιουργηθεί διάφορα εξειδικευμένα προγραμματιστικά πλαίσια που την εκμεταλλεύονται. Δύο από τα πιο γνωστά πλαίσια είναι το Hadoop MapReduce και το Spark.

5.3.1. MapReduce

Το MapReduce ως μοντέλο επεξεργασίας μεγάλων συλλογών δεδομένων σε κατανεμημένα συστήματα αναπτύχθηκε και χρησιμοποιήθηκε για πρώτη φορά από την Google[59]. Έκτοτε, έχουν δημιουργηθεί διάφορες υλοποιήσεις, με ίσως την πιο ευρέως διαδεδομένη να είναι αυτή της Apache, ως μέρος της συλλογής εφαρμογών κατανεμημένης επεξεργασίας Hadoop. Το συγκεκριμένο μοντέλο λειτουργεί σε συνέργεια με το κατανεμημένο σύστημα αρχείων της ιδίας συλλογής (Hadoop Distributed File System – HDFS)[60][61]. Το HDFS διατηρεί τις ιδιότητες των κατανεμημένων συστημάτων αρχείων όπως παρουσιάστηκαν στην παράγραφο 2.3.2 όπως την ανοχή σε σφάλματα, διαφάνεια, δημιουργία αντιγράφων ασφαλείας σε πολλαπλούς κόμβους. Η αρχιτεκτονική αυτού του συστήματος αρχείων ως προς την προοπτική του χρήστη είναι παρόμοια με αυτή των κλασικών συστημάτων, με δυνατότητες όπως δημιουργία φακέλων και υποφακέλων, μετονομασία αρχείων κλπ. Εσωτερικά, την οργάνωση αυτών των λειτουργιών καθώς και την προβολή όλου του συστήματος στον χρήστη την αναλαμβάνει ένας κύριος κόμβος, ο οποίος ονομάζεται “Namenode”. Τα αρχεία κατανέμονται από τον Namenode σε τμήματα στους υπόλοιπους κόμβους, οι οποίοι ονομάζονται “Datanodes”. Ο Namenode είναι υπεύθυνος επίσης για την δημιουργία, διαγραφή και παραγωγή αντιγράφων ασφαλείας τμημάτων των αρχείων, ανάλογα με τις απαιτήσεις ορθής λειτουργίας και σταθερότητας του συστήματος. Τόσο ο κύριος κόμβος όσο και οι δευτερεύοντες, δεν αντιστοιχούν σε πραγματικούς κόμβους υποδομής (υπολογιστές), αλλά σε κόμβους που ορίζονται σε επίπεδο λογισμικού. Δηλαδή κάθε υπολογιστής (πραγματικός κόμβος) μπορεί να διαθέτει παραπάνω από έναν κόμβους του συστήματος HDFS, όπως για παράδειγμα ένας εξυπηρετητής ο οποίος μπορεί να διαθέτει τον Namenode αλλά και έναν Datanode.



Σχήμα 5.3: Αρχιτεκτονική του HDFS[60]

Οι υπολογισμοί στο πλαίσιο του MapReduce γίνονται πρακτικά σε δύο μέρη. Στο μέρος του Map και στο μέρος του Reduce. Στο πρώτο μέρος, εκτελούνται βάσει των αντίστοιχων προγραμματιστικών μεθόδων οι κατανεμημένες εργασίες που έχουν ανατεθεί, οι οποίες έχουν ως έξοδο ζεύγη κλειδιών-τιμών. Έπειτα το μέρος του Reduce λαμβάνει αυτά τα ζεύγη, τα επεξεργάζεται και καταλήγει στο ζητούμενο αποτέλεσμα. Ταυτόχρονα, το πλαίσιο αναλαμβάνει τη διαχείριση των προαναφερθέντων εργασιών (δρομολόγηση στους κόμβους), καθώς και την επανεκτέλεση των αποτυχημένων από αυτές. Κάθε φυσικός κόμβος της συστοιχίας διαθέτει ταυτόχρονα και κόμβο δεδομένων του κατανεμημένου συστήματος αρχείων (Datanode) αλλά και κόμβο υπολογισμών του πλαισίου (compute node) ώστε να γίνεται ευκολότερα η δρομολόγηση των εργασιών σε αυτούς. Για αυτόν τον λόγο υπάρχουν επίσης και εξειδικευμένοι μηχανισμοί παρακολούθησης (trackers) κατανεμημένοι στους κόμβους, συγκεκριμένα ένας JobTracker (παρακολούθηση της συνολικής εργασίας) και ένας TaskTracker (παρακολούθηση της κατανεμημένης υπο-εργασίας του κάθε κόμβου)[61].

Για να γίνει πιο κατανοητή η διαδικασία εκτέλεσης μίας εργασίας κατανεμημένα μέσω του πλαισίου MapRecude, παρουσιάζεται ένα από τα πιο τυπικά παραδείγματα χρήσης του, αυτό της καταμέτρησης λέξεων σε μία συλλογή αρχείων. Αρχικά, οι κλάσεις Map (μέσω της ομώνυμης μεθόδου) χωρίζουν τα αρχεία των κειμένων σε σειρές, και διατρέχουν την κάθε σειρά. Ανάλογα με το πως έχει χωριστεί η αρχική συλλογή των αρχείων από το σύστημα, ο mapper του κάθε κόμβου διατρέχει το αντίστοιχο αρχείο ή τμήμα αρχείου. Όσο η κάθε σειρά διαθέτει λέξεις, αυτές συλλέγονται και αποστέλλονται ως ζεύγος κλειδιού-τιμής της μορφής <λέξη, one>. Δηλαδή πρακτικά καταγράφεται και αποστέλλεται η πληροφορία πως εμφανίστηκε άλλη μία ύπαρξη της συγκεκριμένης λέξης. Έπειτα, η κλάση Reduce συλλέγει τα παραπάνω ζεύγη, και αθροίζει για κάθε ίδια λέξη τις εμφανίσεις της, παρουσιάζοντας έτσι το τελικό αποτέλεσμα. Επίσης μία άλλη σημαντική κλάση ενδιάμεσης χρήσης των δύο παραπάνω είναι η αποκαλούμενη Partitioner (διαχωρισμού). Όταν σε μία εκτέλεση του MapReduce υπάρχουν παραπάνω από ένας Reducer, τότε με τη

βοήθεια αυτής της κλάσης μπορεί ο χρήστης να επιλέξει σε ποιους Reducers θα καταλήξουν συγκεκριμένα ζεύγη από τους Mappers. Στο προηγούμενο παράδειγμα φέρ' ειπείν, αν γνωρίζει ο χρήστης πως υπάρχουν αναλογικά πολλές λέξεις σε μια συλλογή που αρχίζουν με "α", μπορεί να χρησιμοποιήσει Reducers αποκλειστικά για αυτές, διαμοιράζοντας τις υπόλοιπες λέξεις στους υπόλοιπους Reducers.

5.3.2. Spark

Το Spark είναι ένα σύγχρονο προγραμματιστικό πλαίσιο κατανεμημένης επεξεργασίας, το οποίο μπορεί να κάνει υπολογισμούς με χρήση δεδομένων απευθείας από την μνήμη RAM των υπολογιστών μίας συστοιχίας, δίνοντας του ισχυρό πλεονέκτημα ταχύτητας[62]. Λειτουργεί σε ανώτερο προγραμματιστικό επίπεδο, καθώς διαθέτει μεγάλη βιβλιοθήκη εντολών για διάφορες γλώσσες προγραμματισμού και μπορεί να εκτελεστεί από διάφορα και ποικίλα είδη συστοιχιών και συστήματα αρχείων. Η παραλληλία των υπολογισμών μπορεί να γίνει με απλή κλήση των αντίστοιχων μεθόδων, δίνοντας έτσι τη δυνατότητα κάθε εντολή στον κώδικα μιας εφαρμογής να κατανεμηθεί σε διάφορους κόμβους άμεσα και χωρίς τη γνώση της λειτουργίας τους σε χαμηλότερο επίπεδο από τον χρήστη. Συγκεκριμένα, στο παράδειγμα του υπολογισμού του αριθμού εμφάνισης συγκεκριμένων λέξεων μιας συλλογής κειμένων που παρουσιάστηκε στην παράγραφο 5.3.1 , στο πλαίσιο του MapReduce είναι απαραίτητο να οριστεί η κάθε κλάση και μέθοδος των Map και Reduce τμημάτων. Με τη χρήση του πλαισίου Spark αρκεί η κλήση των αντίστοιχων εντολών της βιβλιοθήκης και η εφαρμογή τους στο αρχείο εισόδου[63].

Το Spark διαθέτει βιβλιοθήκη η οποία ονομάζεται Mllib[73] και το όνομά της προέρχεται από το "Machine Learning Library". Μπορεί να δεχθεί ως είσοδο απευθείας αρχεία τύπου JSON, CSV και εικόνων, και όπως υποδηλώνει το όνομά της διαθέτει φύσει κλάσεις / μεθόδους οι οποίες σχετίζονται με την εξόρυξη δεδομένων. Παραδείγματα αυτών των μεθόδων είναι:

- TF – IDF (εξόρυξη κειμένων)
- Word2Vec (συσχέτιση και σύγκριση λέξεων)
- Naive Bayes (ταξινόμηση)
- K – Means (συσταδοποίηση)
- Power Iteration Clustering (συσταδοποίηση)
- FP Growth (εξόρυξη συχνών μοτίβων)
- MapReduce

Εικόνα 5.1: καταμέτρηση λέξεων στο πλαίσιο Spark[63], χρησιμοποιώντας γλώσσα προγραμματισμού Java

ΚΕΦΑΛΑΙΟ 6

ΑΛΓΟΡΙΘΜΟΙ ΟΜΑΔΟΠΟΙΗΣΗΣ

6.1 K – Means

Ο αλγόριθμος K-Means, βάσει των ορισμών που δόθηκαν στο κεφάλαιο 4, είναι αλγόριθμος μη εποπτευόμενης συσταδοποίησης και ανήκει στις μεθόδους περιγραφικής εξόρυξης δεδομένων. Ο σκοπός του αλγόριθμου είναι να δημιουργήσει συστάδες δεδομένων από μία αρχική συλλογή. Η διαδικασία που ακολουθείται είναι η εξής[64]: Αρχικά, επιλέγονται κάποια κέντρα ισάριθμα με τον αναμενόμενο αριθμό των συστάδων. Αυτά τα κέντρα μπορεί να αντιστοιχούν σε στοιχεία της συλλογής ή να είναι πλήρως τυχαία (στην ορολογία η πρώτη περίπτωση ονομάζεται “μεσοειδές” ενώ η δεύτερη “κεντροειδές”). Η επιλογή μπορεί να γίνει τυχαία ή μέσω συγκεκριμένων κριτηρίων, όπως για παράδειγμα να βρεθούν τα ακραία στοιχεία της συλλογής, να χωριστεί ο χώρος σε ισάριθμα των συστάδων και ισομεγεθή διαστήματα και να επιλεγούν οι μέσες τιμές. Έπειτα υπολογίζονται για όλα τα στοιχεία οι ομοιότητες από το κάθε κέντρο, και το κάθε στοιχείο τοποθετείται σημασιολογικά στη συστάδα με την μέγιστη ομοιότητα στοιχείου – κέντρου. Όπως είναι αναμενόμενο με την κάθε προσθήκη (ή αφαίρεση στοιχείου) σε μία συστάδα, αλλάζει και το κέντρο της. Έτσι, η παραπάνω διαδικασία επαναλαμβάνεται έως ότου υπάρξει η απαραίτητη σύγκλιση, όπερ σε κάθε επανάληψη του αλγόριθμου να μην αλλάζει το κέντρο. Άλλα κριτήρια τερματισμού του αλγορίθμου μπορεί να είναι ένας προκαθορισμένος αριθμός μέγιστων επαναλήψεων, ή να μην αλλάζει αισθητά η διακύμανση των στοιχείων εντός μίας συστάδας. Για τον υπολογισμό της ομοιότητας μπορούν να χρησιμοποιηθούν διάφορες μετρικές αποστάσεων, όπως[65]:

Ευκλείδεια απόσταση:

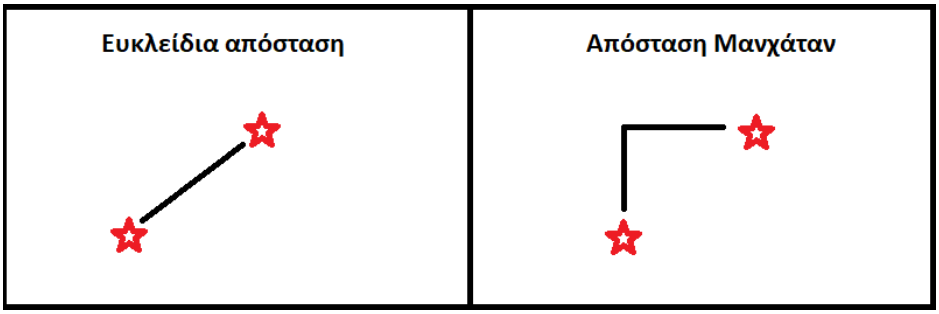
$$d^2 = \sum_i^k (c_i - x_i)^2$$

όπου c_i είναι το κέντρο της εκάστοτε συστάδας, x_i το προς μελέτη στοιχείο της συλλογής και k ο αριθμός των διαστάσεων του γεωμετρικού χώρου.

Απόσταση Μανχάταν:

$$d = \sum_i^k |c_i - x_i|$$

όπου διατηρείται η αντιστοίχηση μεταβλητών και εννοιών της προηγούμενης περίπτωσης. Η απόσταση Μανχάταν, αν και μικρότερης χρονικής πολυπλοκότητας, φέρει επίσης μικρότερη ακρίβεια, καθώς ενώ στην Ευκλείδεια απόσταση υπολογίζεται το μέτρο του k-διαστάσεων διανύσματος με άκρα το κέντρο και το προς μελέτη στοιχείο, στην απόσταση Μανχάταν υπολογίζεται απλά το άθροισμα των αποστάσεων των διαστάσεων. Έτσι, η απόσταση Μανχάταν είναι ακριβής μόνο για χώρο πυκνών δεδομένων ή για δεδομένα τοποθετημένα σε πλέγμα. Οι δύο αποστάσεις γίνονται ταυτόσημες για μονοδιάστατα στοιχεία.



Σχήμα 6.1: Ευκλείδεια, Μανχάταν απόσταση για στοιχεία δισδιάστατου χώρου

Συνημίτονο διανυσμάτων:

Εάν θεωρηθεί πως ο χώρος των στοιχείων είναι διανυσματικός (με κάθε διάνυσμα να έχει αρχή την αρχή των αξόνων του χώρου και τέλος το εκάστοτε στοιχείο) μπορεί να χρησιμοποιηθεί η αναλυτική έκφραση του συνημιτόνου για την εύρεση της ομοιότητας τους. Το συνημίτονο της γωνίας των διανυσμάτων, όταν τα διανύσματα είναι κανονικοποιημένα (οι τιμές των διαστάσεων κυμαίνονται στο διάστημα [0,1]) πέραν την γωνίας δείχνει την καθολική ομοιότητά τους, καθώς η κανονικοποίηση τους μετατρέπει το μέτρο τους ίσο με τη μονάδα. Διανύσματα με συνημίτονο ίσο με 1

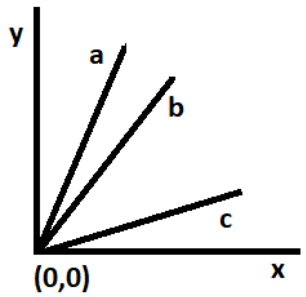
έχουν τη μέγιστη ομοιότητα ενώ είναι πλήρως ανόμοια αν είναι ίσο με 0. Η έκφραση του συνημιτόνου των διανυσμάτων c, x βάσει του εσωτερικού γινομένου είναι:

$$\cos(x,c)=\frac{cx}{|c||x|}$$

όπου οι όροι στον παρανομαστή είναι τα μέτρα των διανυσμάτων:

$$|x|^2=\sum_i^k |x_i^2|$$

Η χρονική πολυπλοκότητα του υπολογισμού του συνημιτόνου είναι και αυτή υψηλή, αλλά έχει ένα μεγάλο πλεονέκτημα. Στην περίπτωση που τα δεδομένα είναι κανονικοποιημένα, το μέτρο τους ισούται πάντα με τη μονάδα, οπότε η πολυπλοκότητα ισούται με την πολυπλοκότητα του αριθμητή. Έτσι η χρήση αυτής της μετρικής είναι η βέλτιστη σε περιπτώσεις όπως ο υπολογισμός ομοιότητας στατιστικών δεδομένων.



Σχήμα 6.2: ομοιότητα κανονικοποιημένων διανυσμάτων - το διάνυσμα b παρουσιάζει μεγαλύτερη ομοιότητα με το a απ ότι με το c

Επιλογή των νέων κέντρων

Τα νέα κέντρα υπολογίζονται με τη χρήση του μαθηματικού μέσου εφαρμοσμένο σε κάθε συστάδα ξεχωριστά. Αν C είναι η συστάδα, c' το νέο κέντρο και x οι τιμές που ανήκουν στο παρόν βήμα εντός της συστάδας τότε για την συστάδα i και τα στοιχεία αυτής j:

$$c' = \frac{1}{|C_i|} \sum_{x_j \in C_i} x_j$$

όπου στην περίπτωση των διανυσμάτων μπορεί να υπολογιστεί με ποικίλους τρόπους, όπως την εύρεση του μέσου της κάθε διάστασης ξεχωριστά ή της πρόσθεσης διανυσμάτων και διαίρεσης του αθροίσματος με τον αριθμό αυτών.

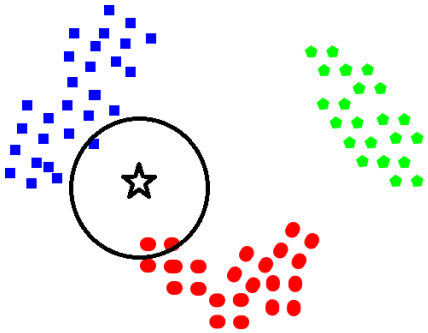
Πολυπλοκότητα

Η χρονική πολυπλοκότητα του αλγόριθμου k-means δεν είναι προβλέψιμη, καθώς εξαρτάται από τον αριθμό των επαναλήψεων τις οποίες θα χρειαστεί να εκτελέσει προκειμένου να επιτευχθεί η σύγκλιση και να τερματιστεί (εκτός και αν έχει οριστεί ως κριτήριο τερματισμού ορισμένος αριθμός επαναλήψεων).Στην απλούστερη μορφή του αλγόριθμου η πολυπλοκότητα είναι O(CDNI) όπου C είναι ο αριθμός των συστάδων (Clusters), D ο αριθμός των διαστάσεων των στοιχείων στον χώρο (Dimensions), N ο αριθμός των στοιχείων και I ο αριθμός των επαναλήψεων έως ότου τερματιστεί ο αλγόριθμος (Iterations). Αυτή η μορφή δεν αναφέρεται στην πολυπλοκότητα των ίδιων των πράξεων (μετρικές αποστάσεων, υπολογισμός μαθηματικού μέσου), αλλά στον αριθμό των στοιχείων και των υποστοιχείων που απαιτείται να προσπελαστούν. Βελτιστοποίηση του αλγορίθμου μπορεί να προκύψει είτε μειώνοντας τη μαθηματική πολυπλοκότητα (επιλογή συνημιτόνου ως μετρική ομοιότητας σε κανονικοποιημένα διανύσματα) είτε την χρονική (χρήση κατανεμημένης επεξεργασίας σε P αριθμό κόμβων επεξεργασίας).

6.2. K – Nearest Neighbors

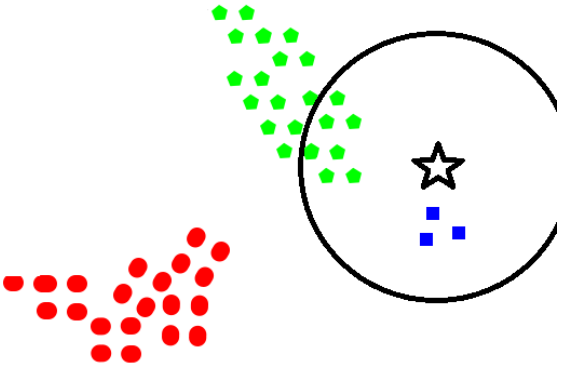
Ο αλγόριθμος K- Nearest Neighbors (K – Πλησιέστεροι Γείτονες) ή εν συντομία KNN είναι ένας αλγόριθμος περιγραφικής κατάταξης και ανήκει στην κατηγορία της εποπτευόμενης εξόρυξης δεδομένων. Τα στοιχεία της συλλογής τοποθετούνται σε "τάξεις" (παρόμοια σημασιολογία με τις συστάδες, η διαφορετική ονομασία χρησιμοποιείται για αποφυγή σύγχυσης με τις μεθόδους συσταδοποίησης), βάσει προκαθορισμένου χαρακτηρισμού αυτών των ομάδων[66]. Συγκεκριμένα, επιλέγονται

στοιχεία με εκ των προτέρων γνώριμες ιδιότητες και χαρακτηρίζονται βάσει αυτών. Ένα νέο στοιχείο που θα ερευνηθεί, θα κατηγοριοποιηθεί βάσει του αριθμού των πλησιέστερων γειτόνων του. Ο αριθμός K των γειτόνων που θα ερευνηθούν ορίζεται εκ των προτέρων από τον δημιουργό ή χρήστη του αλγορίθμου. Η απόσταση από τους γείτονες, όπως και στην περίπτωση του k-means μπορεί να βασίζεται σε διάφορες μετρικές, όπως ευκλείδεια απόσταση, απόσταση Μανχάταν ή ομοιότητα συνημιτόνου. Επειδή στην περίπτωση του k-nn δεν γίνονται επιπλέον υπολογισμοί εντός των τάξεων (όπως αντιθέτως γίνεται στην περίπτωση του k-means για τον υπολογισμό των νέων κέντρων), η πολυπλοκότητά του είναι O(KDN), όπου K είναι ο αριθμός των πλησιέστερων γειτόνων, D το μέγεθος των διαστάσεων του δείγματος και N ο αριθμός των στοιχείων της συλλογής. Η παραπάνω χρονική πολυπλοκότητα αντιστοιχεί μόνο στην περίπτωση της κατηγοριοποίησης ενός στοιχείου εντός μιας ομάδας, και προφανώς αυξάνεται γραμμικά με τον αριθμό των αγνώστων στοιχείων (O(KDNX) όπου X ο αριθμός των άγνωστων στοιχείων).



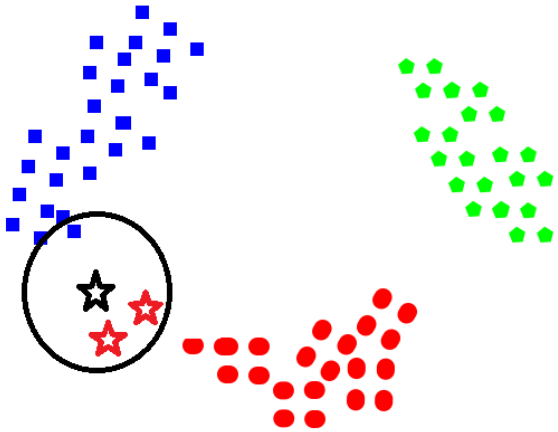
Σχήμα 6.3: Για K = 3, το νέο στοιχείο τοποθετείται στην κατηγορία "κόκκινο" (2 πλησιέστεροι γείτονες έναντι του 1 μπλε)

Ο αλγόριθμος K-NN υπερτερεί του K-Means στην περίπτωση αναζήτησης κατηγορίας για λίγα στοιχεία, καθώς απαιτούνται πολύ λιγότεροι υπολογισμοί. Στον αντίποδα, απαιτείται η εκ των προτέρων ύπαρξη κατηγοριών (training set) ώστε να γίνει η καταχώρηση του νέου στοιχείου. Επίσης πρέπει να γίνει σωστή επιλογή του αριθμού K, καθώς μικρή τιμή του μπορεί να έχει ως αποτέλεσμα τη μικρή ακρίβεια (ύπαρξη ακραίων τιμών) των αποτελεσμάτων, αλλά πολύ μεγάλη τιμή μπορεί να αγνοήσει κάποιο μικρότερο μοτίβο εντός του χώρου μελέτης (Σχήμα 6.4).



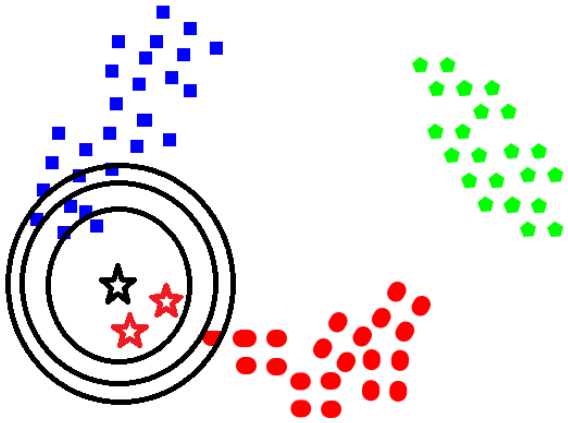
Σχήμα 6.4: Λόγω μεγάλου K (10) το νέο στοιχείο κατηγοριοποιείται λανθασμένα ως "πράσινο"

Ένα άλλο μειονέκτημα του αλγορίθμου KNN έναντι του K-Means έγκειται στο γεγονός πως πρακτικά ο πρώτος αλγόριθμος κατηγοριοποιεί τα δεδομένα βάσει των ακραίων τιμών όπου τα νέα στοιχεία δύναται να γίνουν οι νέες ακραίες τιμές, ενώ ο δεύτερος βάσει των μέσων οι οποίες υπολογίζονται δυναμικά σε κάθε επανάληψη. Έτσι μία τάξη μπορεί να “επεκταθεί” και νέες τιμές προς διερεύνηση να τοποθετηθούν σε αυτήν λανθασμένα. Αυτό το μειονέκτημα γίνεται αντιληπτό αν ληφθεί υπόψιν το γεγονός πως ο KNN υπολογίζει την ομάδα που αντιστοιχεί ένα νέο στοιχείο σε ήδη έτοιμο σετ ομάδων, ενώ ο K-Means κατά τη διάρκεια δημιουργίας αυτών. Στο παρακάτω παράδειγμα (Σχήμα 6.5), το νέο στοιχείο θα κατηγοριοποιηθεί ως "κόκκινο", ενώ αν δεν είχαν προηγουμένως κατηγοριοποιηθεί ως "κόκκινα" τα άλλα δύο νέα στοιχεία (κόκκινοι αστερίσκοι) θα κατηγοριοποιούταν ως “μπλε”.



Σχήμα 6.5

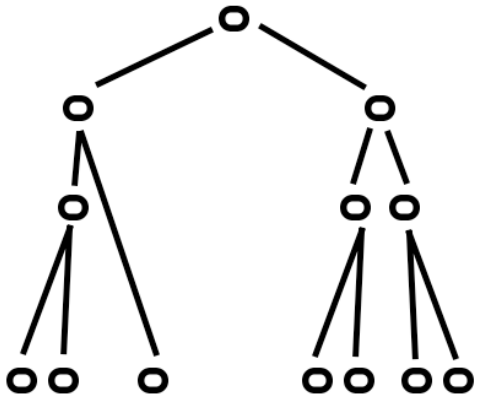
Μία λύση που έχει προταθεί για τα στατιστικά προβλήματα του KNN είναι ο υπολογισμός παραπάνω από ενός συνόλου πλησιέστερων γειτόνων, για διαφορετικά K. Έτσι αν το πόρισμα εξαχθεί από τα αποτελέσματα όλων των συνόλων, είναι πιο πιθανό να αντιστοιχεί στην πραγματικότητα[67]. Στην παρακάτω εκτέλεση (Σχήμα 6.6) γίνεται αναζήτηση πλησιέστερων γειτόνων του προηγούμενου παραδείγματος για K=3, K=8 και K= 10. Ενώ στην πρώτη περίπτωση το νέο στοιχείο τοποθετείται στην "κόκκινη" ομάδα, στις άλλες δύο τοποθετείται στην "μπλε", οπότε εν τέλει χαρακτηρίζεται ως "μπλε".



Σχήμα 6.6

6.3 Συσσωρευτικός Αλγόριθμος Ιεραρχικής Συσταδοποίησης

Όπως αναφέρθηκε στην παράγραφο 2.3.4 , ένας αλγόριθμος ιεραρχικής συσταδοποίησης μπορεί να έχει λογική "από κάτω προς τα πάνω" ή αντίθετα. Ο συσσωρευτικός αλγόριθμος ιεραρχικής συσταδοποίησης (Hierarchical Agglomerative Clustering - HAC) ανήκει στην πρώτη κατηγορία. Κάθε αντικείμενο της συλλογής δεδομένων θεωρείται αρχικά ως συστάδα μοναδιαίου μεγέθους. Σε κάθε βήμα της συσταδοποίησης ο αλγόριθμος συνενώνει τα δύο κοντινότερα στοιχεία, κάνοντας χρήση μετρικών ομοιότητας όπως η ευκλείδεια απόσταση ή η ομοιότητα συνημιτόνου[68]. Έπειτα τα αρχικά δημιουργημένες συστάδες συνενώνονται μεταξύ τους ή με στοιχεία που δεν ανήκουν ακόμη σε κάποια από αυτές συνεχίζοντας τη διαδικασία έως ότου δημιουργηθεί μία καθολική συστάδα που εμπεριέχει ολόκληρη τη συλλογή. Έτσι, σε αντίθεση με τον K-Means ή τον KNN δεν δημιουργείται ένα επίπεδο συστάδων, αλλά ένα δενδροδιάγραμμα πολλαπλών επιπέδων.



Σχήμα 6.7: Δενδροδιάγραμμα ιεραρχικής συσταδοποίησης

Ενώ στο πρώτο επίπεδο συσταδοποίησης (αρχικές συστάδες) η συνένωση των στοιχείων είναι μονοσήμαντη. Αρκεί η εφαρμογή μιας μετρικής ομοιότητας κάθε στοιχείου με τα υπόλοιπα, και ένα κατώφλι της τιμής αυτής ώστε να θεωρηθεί ότι είναι αρκετά όμοια ώστε να τοποθετηθούν στην ίδια συστάδα. Στα επόμενα επίπεδα όμως, όπου απαιτείται η συνένωση των συστάδων, πρέπει να επιλεγεί ο ιδανικός κατά περίπτωση τρόπος υπολογισμού της ομοιότητας, καθώς αυτές αποτελούνται από διάφορα στοιχεία τα οποία διαθέτουν διαφορετικές ιδιότητες[69]. Χαρακτηριστικά παραδείγματα μεθόδων είναι η χρήση πλησιέστερου γείτονα, μέσου όρου ή κεντροειδούς των ήδη υπαρχόντων συστάδων. Οι Lance – Williams έχουν προτείνει μία γενική εξίσωση υπολογισμού της απόστασης η οποία παραμετροποιείται ανά περίπτωση. Αν τα στοιχεία i,j (μοναδιαία ή συστάδες) σχηματίζουν μία συστάδα της οποίας πρέπει να υπολογιστεί η απόσταση d από ένα σημείο k, τότε:

$$d(i \cup j, k) = \alpha_i d(i, k) + \alpha_j d(j, k) + \beta d(i, j) + \gamma |d(i, k) - d(j, k)|$$

όπου οι μεταβλητές α, β, γ ορίζονται ανάλογα την περίσταση.

Για παράδειγμα, στην περίπτωση του πλησιέστερου γείτονα οι τιμές αντιστοιχούν σε α = 0.5, β = 0 και γ = -0.5. Έτσι η παραπάνω σχέση γίνεται:

$$d(i \cup j, k) = 0.5d(i, k) + 0.5d(j, k) - 0.5|d(i, k) - d(j, k)|$$

που αντιστοιχεί στην ελάχιστη απόσταση από τις d(i, k) και d(j, k) (πλησιέστερος γείτονας), καθώς μαθηματικά το ελάχιστο μεταξύ ν στοιχείων είναι το άθροισμα αυτών, αφαιρώντας τις απόλυτες διαφορές των ανά ζεύγη συνδυασμό τους και διαιρώντας με το πλήθος τους. Ομοίως το προαναφερθέν ανάγεται για περισσότερα από 2 στοιχεία. Παρόμοια, για:

$$a_i = \frac{|i|}{|i|+|j|}$$

$$\beta = -\frac{|i||j|}{(|i|+|j|)^2}$$

$$\gamma = 0$$

όπου οι τιμές εντός των απολύτων αντιστοιχούν στον αριθμό των στοιχείων των συστάδων, υπολογίζεται η απόσταση ενός στοιχείου k από το κεντροειδές των i,j. Η παραπάνω σχέση είναι μία πολύ καλή προσέγγιση της απόστασης από το κέντρο των i,j και βασίζεται στην εύρεση αριθμητικού μέσου.

Πολυπλοκότητα

Η πολυπλοκότητα του ιεραρχικού αλγόριθμου συσταδοποίησης είναι της τάξης του $O(n^3)$, καθώς σε κάθε επίπεδο γίνονται n^2 συγκρίσεις (όπου n ο αριθμός των στοιχείων του επιπέδου), σε συνδυασμό με n μέγιστο αριθμό επαναλήψεων ώστε να δημιουργηθεί η καθολική συστάδα. Η χρονική αυτή πολυπλοκότητα γίνεται ακόμα πιο σύνθετη, αν ληφθεί υπ' όψιν το γεγονός πως σε κάθε επίπεδο του αλγορίθμου εκτελούνται αυτόνομες μέθοδοι ομαδοποίησης, όπως ο υπολογισμός κεντροειδών ή η εύρεση πλησιέστερου γείτονα. Το πλεονέκτημα που φέρει η εκτέλεση ενός αλγορίθμου τέτοιας πολυπλοκότητας, είναι πως μπορούν να αξιοποιηθούν και να χρησιμοποιηθούν σε δεύτερο χρόνο όλα τα επίπεδα της συσταδοποίησης, ανάλογα με τις απαιτήσεις του χρήστη / προγραμματιστή.

ΚΕΦΑΛΑΙΟ 7

ΕΦΑΡΜΟΓΗ ΕΞΟΡΥΞΗΣ ΔΕΔΟΜΕΝΩΝ ΣΕ ΚΕΙΜΕΝΑ ΚΑΙ ΕΙΚΟΝΕΣ

Στην παράγραφο 4.1 παρουσιάστηκαν τα βήματα της εξόρυξης δεδομένων:

- Εξερεύνηση
- Εύρεση / ταυτοποίηση μοτίβου
- Ανάπτυξη

Στο παρόν κεφάλαιο θα αναπτυχθούν οι μέθοδοι που χρησιμοποιούνται σε κάθε βήμα στην περίπτωση της κατηγοριοποίησης κειμένων και εικόνων βάσει των σημασιολογικών χαρακτηριστικών τους.

7.1 Κείμενα

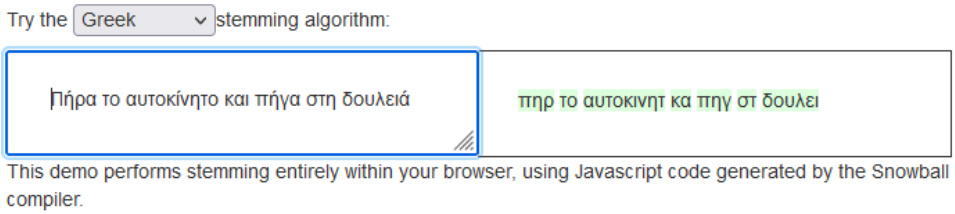
Η επεξεργασία κειμένων με σκοπό την εξαγωγή χρήσιμων πληροφοριών ή την πρόβλεψη μοτίβων, όπως παρουσιάστηκε και στην παράγραφο 4.2.1, εμφανίζεται σε διάφορες μορφές. Παρακάτω θα αναλυθούν βηματικά οι μέθοδοι αυτής της επεξεργασίας και θα προταθούν βέλτιστες τεχνικές για κάθε ένα από τα βήματα, χρησιμοποιώντας τεχνολογίες που αναπτύχθηκαν στα όρια της παρούσας εργασίας.

Εξερεύνηση

Κατά τη διάρκεια της εξερεύνησης, ένα κείμενο πρέπει να μετατραπεί από πλήρως αδόμητο σε ημιδομημένη ή πλήρως δομημένη μορφή. Η συνηθέστερη μορφή στην οποία μετατρέπεται είναι η διανυσματική[70]. Ο λόγος της επιλογής αυτής, πέραν της ευκολίας επεξεργασίας και διαχείρισης των διανυσμάτων, είναι η μορφολογία τους και η ευκολία εκτέλεσης πράξεων σε αυτά, όπως η εύρεση ευκλείδειων αποστάσεων (αν θεωρηθούν τα άκρα τους ως σημεία στο χώρο) ή η εφαρμογή εσωτερικού γινομένου και υπολογισμός του συνημιτόνου τους.

Πριν από την μετατροπή του κειμένου σε διανύσματα, πρέπει πρώτα να γίνει η εκκαθάρισή του. Σημασιολογικά, λέξεις όπως άρθρα, σύνδεσμοι και μονοσύλλαβες άκλιτες λέξεις όπως τα μόρια καθώς και τα σημεία στίξης δεν έχουν κάποια αξία. Σε ένα κείμενο, σε επίπεδο κατηγοριοποίησης του περιεχομένου του, μπορεί να εξαχθεί η ίδια πληροφορία από τη φράση “θα πάω για ένα περίπατο σήμερα” και από την

φράση “πάω περίπατο σήμερα”. Γίνεται λοιπόν αξιολόγηση των χρήσιμων λέξεων και βάσει αυτής αποφασίζεται ποιες λέξεις θα διατηρηθούν και ποιες θα καταργηθούν. Ένα άλλο βήμα της εκκαθάρισης, είναι η μετατροπή της κάθε λέξης στην βασική της μορφή, ώστε να επιτευχθεί μία ομαδοποίηση των σημασιολογικά όμοιων στοιχείων. Για παράδειγμα, οι λέξεις “αυτοκίνητο”, “αυτοκίνητα” και “αυτοκινήτων” έχουν την ίδια σημασιολογία, με την βασική μορφή να είναι η πρώτη. Μία εύκολη μέθοδος για την ομαδοποίηση των παραπάνω τριών λέξεων είναι η αφαίρεση των τόνων και η αποκοπή των καταλήξεων (stemming) οπότε και οι τρεις λέξεις μετατρέπονται σε “αυτοκινητ”.



Εικόνα 7.1: Stemming σε φράση γραμμένη στα ελληνικά[71]

Στην περίπτωση της μετατροπής μίας συλλογής κειμένων σε διανύσματα, οι διαστάσεις αυτών είναι ισάριθμες με το πλήθος των μοναδικών όρων που εμφανίζονται εντός των κειμένων, και το μέγεθος των διαστάσεων υποδηλώνει τη συχνότητα εμφάνισης τους. Για παράδειγμα, έστω τα παρακάτω κείμενα:

Κείμενο Α:

“Η Μαρία παρήγγειλε και έφαγε φαγητό. Η Γεωργία έφαγε γλυκό”

Κείμενο Β:

“Παραγγείλετε τώρα από το κατάστημα μας για να φάτε γλυκό ή φαγητό”

Κείμενο Γ:

“Η παρουσίαση του νέου βιβλίου του πετυχημένου συγγραφέα είναι το απόγευμα”

Εφόσον γίνει η αφαίρεση των σημασιολογικά μικρής αξίας λέξεων και η αναγωγή των υπολοίπων στη ριζική μορφή τους, δημιουργούνται τα παρακάτω διανύσματα:

Πίνακας 7.1

	Κείμενο Α	Κείμενο Β	Κείμενο Γ
μαρ	1	0	0
παραγγ	1	1	0
τρω	3	2	0
γλυκ	1	1	0
τωρ	0	1	0
καταστημ	0	1	0
γεωργ	1	0	0
παρουσι	0	0	1
νε	0	0	1
βιβλ	0	0	1
πετυχημ	0	0	1
συγγραφ	0	0	1
απογευμ	0	0	1

Οι ευκλείδειες αποστάσεις των διανυσμάτων είναι κατά προσέγγιση:

ΑΒ: 2.23

ΑΓ: 4.35

ΒΓ: 3.7

Αν τα παραπάνω διανύσματα που εξάγονται από τα κείμενα κανονικοποιηθούν (αναγωγή σε διανύσματα με μέτρο ίσο με τη μονάδα) τότε γίνονται:

$$A:(\frac{1}{4},\frac{1}{2},\frac{3}{4},\frac{1}{4},0,0,\frac{1}{4},0,0,0,0,0,0)$$

$$B:(0,\frac{1}{2\sqrt{2}},\frac{1}{\sqrt{2}},\frac{1}{2\sqrt{2}},\frac{1}{2\sqrt{2}},\frac{1}{2\sqrt{2}},0,0,0,0,0,0,0)$$

$$\Gamma:(0,0,0,0,0,0,0,\frac{1}{\sqrt{6}},\frac{1}{\sqrt{6}},\frac{1}{\sqrt{6}},\frac{1}{\sqrt{6}},\frac{1}{\sqrt{6}})$$

Με ομοιότητα συνημιτόνου:

- cos(ΑΒ) = 0.78
- cos(ΑΓ) = 0
- cos(ΒΓ) = 0

Και στις δύο περιπτώσεις μετρικών ομοιότητας / ανομοιότητας, τα αποτελέσματα ποιοτικά είναι αναμενόμενα, βάσει του περιεχομένου των κειμένων. Όμως, ενώ οι ευκλείδειες αποστάσεις διακρίνονται από τυχαίες αριθμητικές αποστάσεις, η ομοιότητα συνημιτόνου παρουσιάζει ξεκάθαρα αποτελέσματα. Συγκεκριμένα, με τη χρήση της ευκλείδειας απόστασης, είναι δυσκολότερο για τον ερευνητή να υπολογίσει πόσο κοντινότερο είναι το 2.23 σε σχέση με το 4.35 ή το 3.7. Αντίθετα, το συνημίτονο δύο διανυσμάτων θετικών διαστάσεων μπορεί να πάρει τιμές στο διάστημα [0,-1] , με 0 να αντιστοιχεί στην πλήρη ανομοιότητα και το 0 στην πλήρη ομοιότητα (όμοια διανύσματα).

Μία μέθοδος βελτιστοποίησης της ομοιότητας συνημιτόνου είναι η χρήση συντελεστών βαρύτητας για κάθε λέξη που εμφανίζεται στο κείμενο. Λέξεις που εμφανίζονται συχνά σε ένα κείμενο, αλλά σπανιότερα στο σύνολο των κειμένων, αναμένεται να έχουν μεγαλύτερο βάρος στο σημασιολογικό περιεχόμενο του εν λόγω κειμένου. Αντίθετα, λέξεις που μεν εμφανίζονται συχνά σε ένα κείμενο, αλλά εμφανίζονται επίσης συχνά σε ολόκληρη τη συλλογή του κειμένου, πρέπει να θεωρηθούν σημασιολογικά μικρότερης αξίας. Η αξία αυτή των εκάστοτε λέξεων μπορεί να υπολογιστεί με τη χρήση της μεθόδου “συχνότητα όρου – αντίστροφη συχνότητα όρου στα έγγραφα” (αγγλιστί “term frequency–inverse document frequency” / TF-IDF). Ο συντελεστής βαρύτητας w_j μιας λέξης j σε ένα κείμενο d_i υπολογίζεται από τη σχέση:

$$w_j = \frac{n_{ij}}{|d_i|} \log \frac{N}{n_j}$$

όπου το πρώτο κλάσμα αντιστοιχεί στο σύνολο των εμφανίσεων της λέξης j στο κείμενο d_i προς το μέγεθος του κειμένου d_i, και το κλάσμα εντός του λογαρίθμου αντιστοιχεί στο σύνολο των κειμένων προς τα κείμενα που διαθέτουν έστω και μία φορά την λέξη j.

Έτσι, τα παραπάνω διανύσματα μετατρέπονται σε:

$$A: (0.067, 0.024, 0.072, 0.024, 0, 0, 0.067, 0, 0, 0, 0, 0, 0)$$

$$B: (0, 0.028, 0.057, 0.028, 0.078, 0.078, 0, 0, 0, 0, 0, 0, 0)$$

$$\Gamma: (0, 0, 0, 0, 0, 0, 0, 0.078, 0.078, 0.078, 0.078, 0.078, 0.078)$$

με το εσωτερικό γινόμενό τους:

- cos(ΑΒ) = 0.34
- cos(ΑΓ) = 0

- cos(ΒΓ) = 0

Τα αποτελέσματα και σε αυτή τη περίπτωση είναι (ποιοτικά) αναμενόμενα. Παρατηρείται όμως μικρότερη ομοιότητα μεταξύ των κειμένων Α και Β. Αυτό είναι αναμενόμενο, καθώς ο συντελεστής TF-IDF μείωσε τη βαρύτητα των όρων που είναι κοινοί στα δύο κείμενα, και επίσης ο συνολικός αριθμός των κειμένων ήταν αρκετά μικρός (N = 3), οπότε το κομμάτι IDF του συντελεστή TF-IDF (λογάριθμος) αντιστοιχούσε σε μικρό κλασματικό λόγο. Σε μεγαλύτερη συλλογή κειμένων, αναμένεται όροι που εμφανίζονται σε λίγα από τα κείμενα αυτά, να αποκτούν αρκετά μεγαλύτερο βάρος (μεγιστοποίηση του λόγου N/n_j).

Εύρεση / ταυτοποίηση μοτίβου

Τα διανύσματα που δημιουργήθηκαν κατά τη διάρκεια της εξερεύνησης θα πρέπει να ομαδοποιηθούν για να εξαχθεί η κατηγορία στην οποία ανήκει το κάθε κείμενο. Πρέπει να επιλεχθεί ο βέλτιστος αλγόριθμος για αυτή τη κατηγοριοποίηση, καθώς και η μέθοδος εκτέλεσης του. Αρχικά, θα πρέπει να ληφθεί υπ όψιν εάν υπάρχουν ήδη κατηγορίες στις οποίες καλούνται να τοποθετηθούν τα νέα στοιχεία (κείμενα) ή αν θα αναλυθούν εκ του μηδενός. Στην πρώτη περίπτωση, θα πρέπει να επιλεχθεί μορφή εποπτευόμενης ομαδοποίησης ενώ στη δεύτερη μη εποπτευόμενη. Επίσης, λόγω της υψηλής πολυπλοκότητας των αλγορίθμων, θα πρέπει να επιλεγεί κάποιο μη σειριακό σύστημα επεξεργασίας, κατά προτίμηση κατανεμημένο. Τέλος, προτείνεται κάποια από τα στάδια της εξερεύνησης να εκτελεστούν σε κατανεμημένο περιβάλλον, καθώς επιτυγχάνεται σημαντική επιτάχυνση.

Συγκεκριμένα, προτείνεται η χρήση του πλαισίου MapReduce (ή η αντίστοιχη μέθοδος του πλαισίου Spark) για την καταμέτρηση των λέξεων των κειμένων (χρησιμοποιώντας τον αλγόριθμο word count), με τη χρήση combiner ώστε να συναθροίζονται οι εμφανίσεις της κάθε λέξης πριν την αποστολή τους μέσω δικτύου στους Reducers. Επίσης, κάθε μονάδα της συστοιχίας μπορεί να επεξεργαστεί διαφορετικά από τα διανύσματα που προκύπτουν ώστε να επιτευχθεί παράλληλα η κανονικοποίησή τους. Έπειτα, πρέπει να γίνει η αντίστοιχη ομαδοποίηση.

Εποπτευόμενη κατάταξη

Προτείνεται να γίνει χρήση του αλγόριθμου KNN, με πολλαπλή εκτέλεσή του για διάφορες τιμές του K (ιδανικά 5 διαφορετικές τιμές). Ανάλογα με το πλήθος των επεξεργαστικών μονάδων, μπορεί να γίνει ο διαχωρισμός είτε βάσει του K (κάθε υπολογιστής εκτελεί τον αλγόριθμο για συγκεκριμένη τιμή) είτε βάσει των υπολογισμών κάθε αυτών. Επειδή στον αλγόριθμο KNN δεν υπάρχει κάποιος επαναϋπολογισμός των ομάδων (πέραν των ακραίων τιμών που προστίθενται), μπορεί να γίνει χωρίζοντας τα προς διερεύνηση στοιχεία σε διαφορετικές μονάδες επεξεργασίας. Αν και αυτή η επιλογή ενέχει τον κίνδυνο της ταυτόχρονης εγγραφής στοιχείων και ως εκ τούτου αλλαγή του αριθμού εν δυνάμει πλησιέστερων γειτόνων, αυτός ο κίνδυνος εξαλείφεται από την χρήση πολλαπλών K. Η επιτάχυνση στην

πρώτη περίπτωση είναι της τάξης του πλήθους αριθμών K, ενώ στην δεύτερη περίπτωση της τάξης του πλήθους των υπολογιστικών μονάδων εντός της συστοιχίας.

Μη εποπτευόμενη συσταδοποίηση

Στην περίπτωση που απαιτείται η εκ του μηδενός ομαδοποίηση των κειμένων, προτείνεται η χρήση του αλγόριθμου K-Means, μέσω του πλασίου MapReduce, σε συστοιχία κατανεμημένης επεξεργασίας. Το Map τμήμα του αλγορίθμου τοποθετεί τις εγγραφές (διανύσματα) στις αντίστοιχες συστάδες, ενώ το Reduce τμήμα υπολογίζει τα νέα κέντρα αυτών. Η επιτάχυνση του αλγορίθμου επιτυγχάνεται στο κομμάτι του Map, καθώς χρονική πολυπλοκότητα γίνεται O(KNDI/p) όπου p ο αριθμός των υπολογιστικών μονάδων. Με την χρήση combiners η πολυπλοκότητα επικοινωνίας γίνεται O(KD), καθώς αποστέλλονται ομαδικά οι τιμές που αντιστοιχούν στις ίδιες συστάδες (χωρίς την ομαδοποίηση ο φόρτος δικτύου θα ήταν αρκετά μεγάλος, της τάξης του O(ND).

Ανάπτυξη

Οι ομάδες των κειμένων μπορούν να χρησιμοποιηθούν με ποικίλους τρόπους, όπως η κατηγοριοποίηση ειδησιογραφικών άρθρων, η εύρεση του περιεχομένου μίας ανάρτησης σε κοινωνικά δίκτυα, ή η εμφάνιση αποτελεσμάτων κατά προσέγγιση και βάσει περιεχομένου μιας μηχανής αναζήτησης.

7.2 Εικόνες

Ένας από τους πιο διαδεδομένους αλγόριθμους για την εξόρυξη πληροφοριών από εικόνες είναι ο SIFT (scale-invariant feature transform) ο οποίος παρουσιάστηκε το 1999 από τον David Lowe[72][74][75]. Στην παρούσα παράγραφο γίνεται παρουσίαση του αλγορίθμου αυτού καθώς και των μαθηματικών μεθόδων που τον διέπουν. Έπειτα θα προταθεί η χρήση των δεδομένων εξόδου του αλγορίθμου (διανύσματα παραγόμενα από εικόνες) με σκοπό την εξαγωγή χρήσιμων πληροφοριών.

Εξερεύνηση

Επιλογή των περιγραφικών στοιχείων

Αρχικά θα πρέπει να οριστεί ποια στοιχεία μιας εικόνας θα την περιέγραφαν μοναδικά. Για παράδειγμα, αν ζητηθεί από κάποιον άνθρωπο να περιγράψει την παρακάτω εικόνα (εικόνα 7.2), θα απαντούσε πως βλέπει έναν άνθρωπο να φωτογραφίζεται μπροστά από τον πύργο του Άιφελ. Αντίστοιχα λοιπόν, αν ανατεθεί

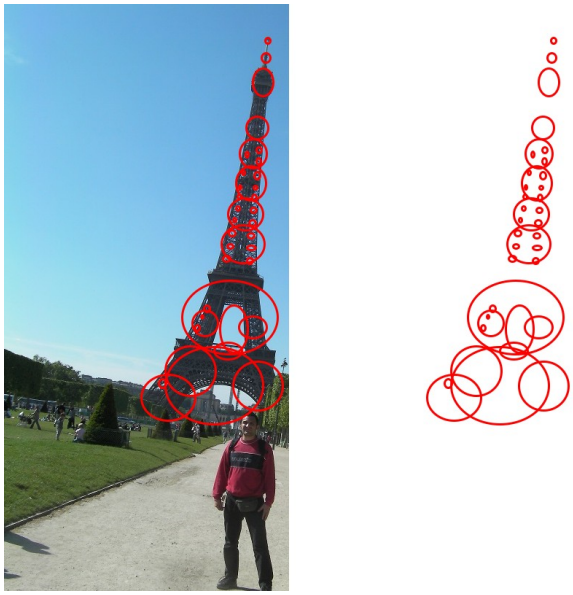
σε κάποιον αλγόριθμο να περιγράψει την ίδια εικόνα, αναμένεται να δοθούν ως έξοδος του αλγορίθμου οι φράσεις-κλειδιά “άνθρωπος” , “πύργος του Άιφελ”.

Εικόνα 7.2

Όπως και ο ανθρώπινος εγκέφαλος λοιπόν, ο αλγόριθμος καλείται να αγνοήσει περιοχές της εικόνας με μικρό ενδιαφέρον, όπως ο γαλάζιος ουρανός ή το γκρίζο έδαφος. Στατιστικά, στις περισσότερες εικόνες, τα σημεία ενδιαφέροντος βρίσκονται στις χρωματικές εναλλαγές της εικόνας (έγχρωμη) ή στην αλλαγή της φωτεινότητας (έγχρωμη, ασπρόμαυρη).

Βάσει αυτού λοιπόν, θα πρέπει αρχικά να αναζητηθούν τα σημεία της εικόνας τα οποία (συνδυαστικά) μπορούν να δώσουν μια εννοιολογική απεικόνιση των αντικειμένων που βρίσκονται σε αυτή. Μία φαινομενικά καλή επιλογή θα ήταν να επιλεχθούν οι άκρες των αντικειμένων, καθώς χαρακτηρίζονται από μεγάλη εναλλαγή της φωτεινότητας και αντίθεσης των χρωμάτων, αλλά και είναι χαρακτηριστικές των αντικειμένων (αν στην παραπάνω εικόνα ανιχνευθούν οι άκρες και οι αποστάσεις αυτών στο αντικείμενο “πύργος του Άιφελ” τότε μπορεί να αναγνωριστεί και το ίδιο το αντικείμενο). Η επιλογή των ακρών όμως, δημιουργεί έδαφος για ενός τύπου επικίνδυνο σφάλμα: Διαφορετικά αντικείμενα, μπορεί να δημιουργήσουν ίδιες άκρες, ειδικά στην περίπτωση όμοιου φόντου.

Η λύση στο παραπάνω πρόβλημα είναι η επιλογή κλειστών περιοχών εναλλαγής χρώματος / φωτεινότητας, που στην αγγλική βιβλιογραφία αναφέρονται ως “blobs” (άμορφη μάζα ή σταγόνα). Η επιλογή αυτών των περιοχών μειώνει κατά πολύ την εμφάνιση του παραπάνω σφάλματος, καθώς ο ίδιος συνδυασμός τέτοιων “σταγόνων” σε διαφορετικά αντικείμενα είναι μηδενικός. Όπως φαίνεται και στην παρακάτω εικόνα (εικόνα 7.3) αν σε ένα αντικείμενο ανιχνευθούν σωστά οι σταγόνες αυτές, οι αποστάσεις τους, καθώς και η κλίμακά τους, τότε υπάρχει ξεκάθαρη μοναδικότητα του αντικειμένου.



Εικόνα 7.3: Ανίχνευση blobs διαφόρων μεγεθών στο αντικείμενο “πύργος του Άιφελ”

Ανίχνευση των περιγραφικών στοιχείων

Έστω η περιοχή μιας εικόνας που διαθέτει ένα τέτοιο στοιχείο (blob) το οποίο θα πρέπει να ανιχνευθεί. Για χάριν απλοποίησης θα γίνει μελέτη στη μία διάσταση και μονοχρωματικά. Οι ίδιες αρχές που θα εφαρμοστούν στο παρακάτω παράδειγμα μπορούν να εφαρμοστούν και στις δύο διαστάσεις και για πολυχρωματική κλίμακα (RGB). Η απεικόνιση ενός τέτοιου blob φαίνεται στο παρακάτω διάγραμμα (σχήμα 7.1), όπου ο άξονας x συμβολίζει τον άξονα x (εικονοστοιχεία σε μία διάσταση) μιας εικόνας και ο άξονας του y την ένταση του σήματος (εδώ την χρωματική ένταση στην μονοχρωματική εικόνα).

Σχήμα 7.1: blob σε μία διάσταση

Στην παρούσα μορφή του, αυτό το στοιχείο είναι δύσκολο να ανιχνευθεί, καθώς στην περίπτωση αναζήτησης σε κάθε εικονοστοιχείο (pixel) ξεχωριστά δημιουργείται μεγάλη υπολογιστική πολυπλοκότητα και για την εύρεση του αλλά και για τον υπολογισμό του μεγέθους του.

Μία μέθοδος για την εξαγωγή παλμών όπως ο παραπάνω είναι η συνέλιξη του με τις απαραίτητες συναρτήσεις. Η συνέλιξη είναι η μέθοδος που μία συνάρτηση λειτουργεί ως τανυστής σε μία άλλη, ή πιο απλά, η μέθοδος όπου μία συνάρτηση λειτουργεί ως φίλτρο μίας άλλης.

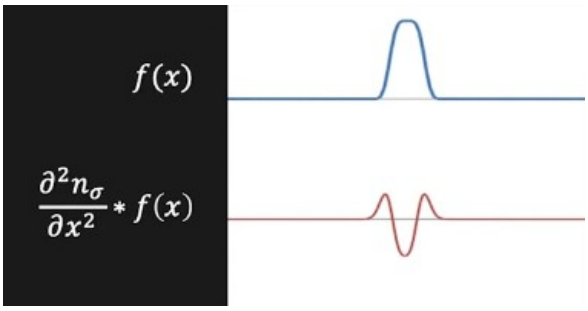
Η μαθηματική διατύπωση της συνέλιξης δυο συναρτήσεων f,g είναι η εξής:

$$(f * g) := \int_{-\infty}^{\infty} f(\tau) g(t - \tau) \, d\tau$$

ενώ για διακριτές συναρτήσεις γίνεται:

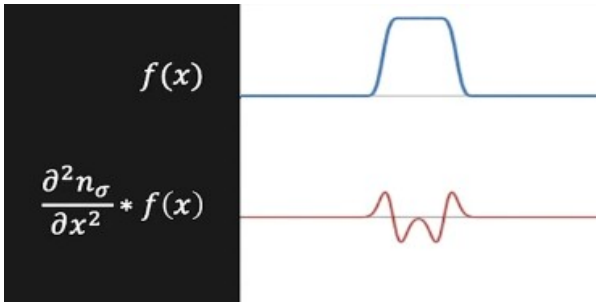
$$(f * g)[n] = \sum_{m=-\infty}^{\infty} f[n - m] g[m]$$

Στο παρόν πρόβλημα ο υπολογισμός γίνεται χρησιμοποιώντας τη μορφή που αναφέρεται στις διακριτές συναρτήσεις, καθώς υπάρχει διακριτότητα ανά εικονοστοιχείο στον παλμό (εξ' ορισμού η συνεχής μορφή μετατρέπεται σε διακριτή). Η συνάρτηση η οποία χρησιμοποιείται είναι η δεύτερη παράγωγος της κανονικής κατανομής. Το αποτέλεσμα αυτής της συνέλιξης φαίνεται στο παρακάτω σχήμα (σχήμα 7.2).



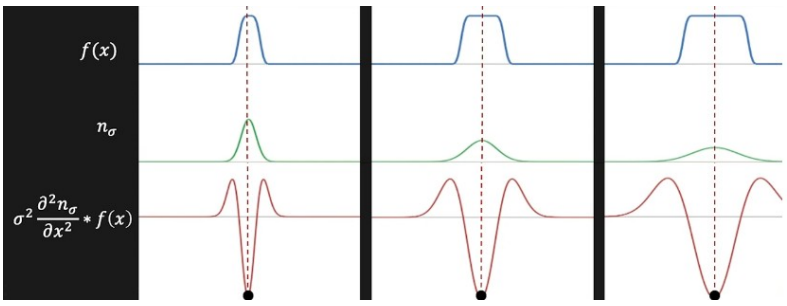
Σχήμα 7.2: συνέλιξη της δεύτερης παραγώγου κατανομής Gauss με παλμό [74]

Σε αυτή τη περίπτωση το στοιχείο (blob) είναι πολύ πιο εύκολο να ανιχνευθεί, καθώς πλέον αναζητείται σημείο με τοπικό μέγιστο, με σημεία τομής της συνάρτησης με το μηδέν εκατέρωθεν του μεγίστου. Δημιουργείται όμως μία επιπλέον πρόκληση, καθώς ανάλογα με το πλάτος του παλμού (blob) αλλάζει και η μορφή του αποτελέσματος της συνέλιξης, χωρίς να δημιουργείται η επιθυμητή παραπάνω μορφή.



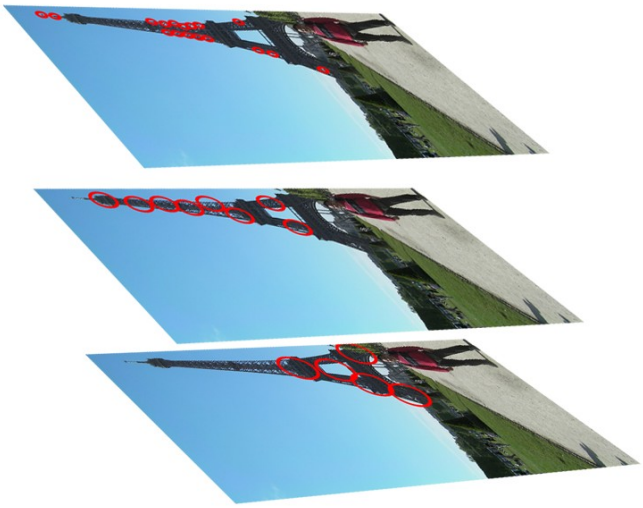
Σχήμα 7.3: μη επιθυμητό αποτέλεσμα συνέλιξης[74]

Για την επίλυση του παραπάνω προβλήματος, μπορεί να γίνει κανονικοποίηση της δεύτερης παραγώγου της κανονικής κατανομής, πολλαπλασιάζοντας με το τετράγωνο της διασποράς (διακύμανση) αυτής, και δοκιμάζοντας διαφορετικές διακυμάνσεις. Με αυτόν τον τρόπο επιπλέον γίνεται και κατηγοριοποίηση των στοιχείων (blobs) σε μεγέθη, χρησιμοποιώντας τα διάφορα σ (διασπορές) για τα οποία βρέθηκαν τα στοιχεία, ως κλίμακες αυτών (σχήμα 7.4). Η μεγαλύτερη διασπορά στην κανονική κατανομή βοηθάει στην σωστή συνέλιξη με μεγαλύτερους παλμούς καθώς μεγαλώνει το εύρος της. Αυτό όμως έχει ως αποτέλεσμα την μικρότερη τιμή για τον μέσο της κορυφής (σταθερό εμβαδόν κατανομής Gauss) , οπότε γίνεται και η κανονικοποίηση, πολλαπλασιάζοντας το αποτέλεσμα της συνέλιξης με τη διακύμανση.



Σχήμα 7.4: εύρεση διαφόρων μεγεθών blobs χρησιμοποιώντας κανονικοποιημένες παραγώγους κανονικών κατανομών [74]

Μετά τη συλλογή των σημείων μπορεί να δημιουργηθεί μία πυραμίδα διαφόρων κλιμάκων στις οποίες ανήκουν αυτά. Το κάθε επίπεδο της πυραμίδας αντιστοιχεί σε διαφορετική κλίμακα εύρεσης στοιχείων (blobs).



Εικόνα 7.4: Ανίχνευση διαφόρων κλιμάκων στοιχείων

Όπως και στην περίπτωση της εξόρυξης δεδομένων από κείμενο, η μορφή που επιλέγεται για τα περιγραφικά στοιχεία μιας εικόνας ώστε να μπορούν να επεξεργαστούν και να ομαδοποιηθούν είναι η διανυσματική. Δημιουργείται ένα πλέγμα γύρω από το κέντρο του στοιχείου (blob), με μέγεθος ανάλογο με το μέγεθος (κλίμακα) του στοιχείου. Κάθε στοιχείο αυτού του πλέγματος περιγράφει την κλίση (gradient) των εικονοστοιχείων στο αντίστοιχο σημείο. Η κλίση αυτή υπολογίζεται από την παρακάτω παραγώγιση:

$$\nabla f(x,y)=\begin{bmatrix} g_x \\ g_z \end{bmatrix}=\begin{bmatrix} \frac{df}{dx} \\ \frac{df}{df} \end{bmatrix}=\begin{bmatrix} f(x+1,y)-f(x-1,y) \\ f(x,y+1)-f(x,y-1) \end{bmatrix}$$

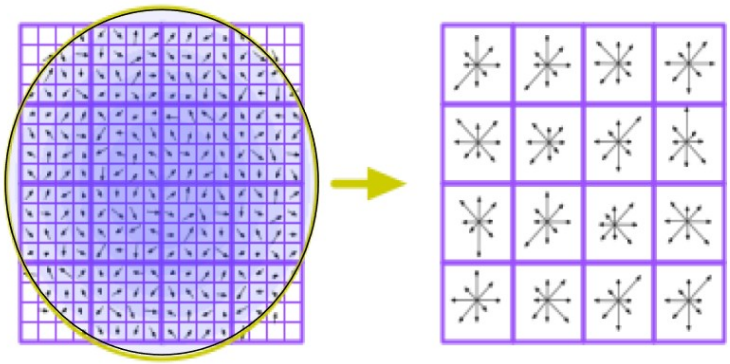
όπου f(x,y) είναι η χρωματική ένταση του εικονοστοιχείου στο σημείο x,y. Πρακτικά δηλαδή υπολογίζεται ο ρυθμός αύξησης ή μείωσης της χρωματικής διαφοράς γειτονικών εικονοστοιχείων. Από την παραπάνω διαφόριση, προκύπτει η κλίση του στοιχείου του πλέγματος (με τον υπολογισμό του τόξου εφαπτομένης):

$$\theta = \arctan\left(g_y / g_x\right)$$

καθώς και το μέγεθος του κάθε διανύσματος της κλίσης

$$\gamma = \sqrt{g_x^2 + g_y^2}$$

Οι κλίσεις αυτές είναι διακριτές και δεν ανήκουν σε συνεχές φάσμα. Αντιθέτως, επιλέγονται 8 αντιπροσωπευτικές κλίσεις με διαφορά 45° και αν η υπολογισμένη κλίση δεν συμπίπτει με κάποια αντιπροσωπευτική, μετατρέπεται στην κοντινότερη αυτής. Από τον υπολογισμό των κλίσεων στο πλέγμα, δημιουργείται ένα νέο πλέγμα συνήθως διαστάσεων 2X2 ή 4X4, το οποίο έχει προκύψει από την συνένωση (άθροιση) των επιμέρους σημείων που ανήκουν στα τεταρτημόρια του, λαμβάνοντας υπόψιν την κλίση και το μέγεθος αυτών. Το διάνυσμα που προκύπτει από το νέο πλέγμα διαθέτει εν τέλει ως πληροφορία τα στοιχεία του πλέγματος, και το μέγεθος των οκτώ κλίσεων που ανήκουν σε αυτό (οπότε και προκύπτει διάνυσμα 4x4x8 = 128 διαστάσεων).



Σχήμα 7.5: αρχικό και τελικό πλέγμα [72]

Μία συλλογή εικόνων, μπορεί να εμπεριέχει το ίδιο αντικείμενο απεικόνισης με ποικίλους τρόπους. Οι δυνατές παραλλαγές με τις οποίες μπορεί να εμφανίζεται ένα αντικείμενο από εικόνα σε εικόνα είναι οι εξής:

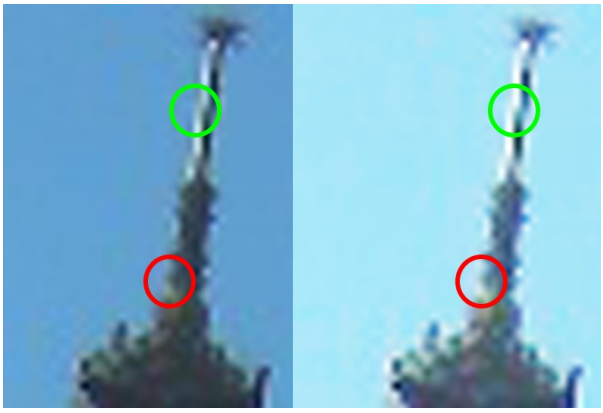
1. διαφορά φωτεινότητας / αντίθεσης
2. περιστροφή

3. κλίμακα

Οι παραπάνω παραλλαγές είναι φυσικό και αναμενόμενο να εμφανίζονται, καθώς ένα αντικείμενο μπορεί να έχει αποτυπωθεί από διαφορετικές γωνίες λήψης, υπό διαφορετικό φωτισμό και διαφορετικές αποστάσεις. Όταν δοθούν στον υπό μελέτη αλγόριθμο οι εικόνες ώστε να ομαδοποιηθούν βάσει των υπό απεικόνιση αντικειμένων θα πρέπει να έχουν αρθεί αυτές οι παραλλαγές και να μην λαμβάνονται υπ όψιν.

Διαφορά φωτεινότητας / αντίθεσης

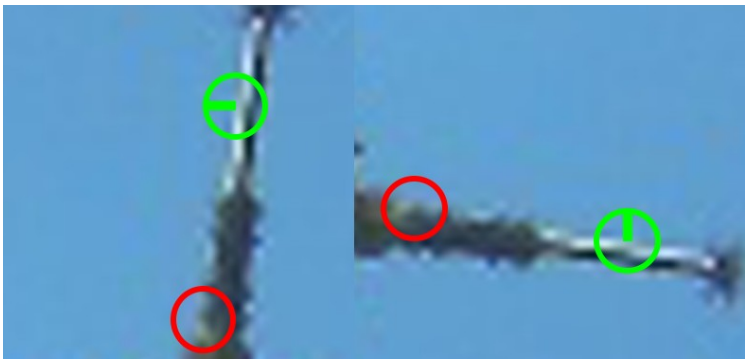
Τα διανύσματα που υπολογίστηκαν με τη βοήθεια της παραπάνω διαδικασίας, κανονικοποιούνται βάσει του αθροίσματος τους (κανονικοποίηση στη μονάδα). Έτσι η τελική μορφή τους δεν αντιστοιχεί στο πραγματικό τους μέγεθος, αλλά στο ποσοστιαίο μέγεθος τους σε σχέση με το σύνολο. Αυτό έχει ως αποτέλεσμα την ανεξαρτησία των υπολογισμών σε σχέση με την αντίθεση και τη φωτεινότητα της εικόνας εισόδου, καθώς τα εικονοστοιχεία μίας εικόνας που έχει υποστεί την αντίστοιχη επεξεργασία αλλάζουν τα χρωματικά τους μεγέθη, αλλά όχι την αναλογία αυτών.



Εικόνα 7.5: τα σημειωμένα με κόκκινο και πράσινο εικονοστοιχεία αλλάζουν το χρωματικό μέγεθος τους από εικόνα σε εικόνα, αλλά όχι την χρωματική αναλογία τους

Περιστροφή και κλίμακα

Από το πλέγμα των κλίσεων που παρουσιάστηκε παραπάνω υπολογίζεται η κυρίαρχη κλίση για κάθε περιγραφικό στοιχείο της εικόνας. Έτσι πριν τον οποιονδήποτε περαιτέρω υπολογισμό, υπολογίζεται η αντιπροσωπευτική κλίση του αντικειμένου (μέγιστη κλίση των αθροισμάτων των οχτώ κλίσεων του πλέγματος) και κανονικοποιείται ως προς αυτήν (περιστροφή στοιχείου). Στην περίπτωση πολλαπλών μέγιστων κλίσεων, λαμβάνονται παραπάνω από μία υπόψιν, εφόσον αυτές ξεπερνούν το 80% της κυρίαρχης. Ταυτόχρονα, υπολογίζεται ο λόγος των κλιμάκων των περιγραφικών στοιχείων του αντικειμένου και διατηρείται αυτός ως ταυτότητα αυτών, έναντι του αυτούσιου μεγέθους τους (κανονικοποίηση βάσει αναλογίας των μεγεθών και των αποστάσεων των περιγραφικών στοιχείων).



Εικόνα 7.6: περιστροφή περιγραφικού στοιχείου βάσει την επικρατούσας κλίσης

Εύρεση / ταυτοποίηση μοτίβου

Κατά τη διάρκεια της εξόρυξης πληροφορίας από εικόνες, μπορεί να απαιτηθεί να γίνει κατηγοριοποίηση ενός συνόλου εικόνων σε ομάδες, να τοποθετηθούν εικόνες σε ήδη υπάρχουσες ομάδες ή να ανιχνευθούν αντικείμενα εντός αυτών. Όπως και στην εξόρυξη πληροφορίας από κείμενα, έτσι και στην περίπτωση της εικόνας μπορεί να ακολουθηθεί παρόμοια κατανεμημένη λογική. Ανάλογα με το είδος της ομαδοποίησης που αναμένεται, επιλέγονται και οι αντίστοιχοι αλγόριθμοι. Αντίθετα όμως με την περίπτωση των κειμένων, ένα διάνυσμα αντιστοιχεί σε ένα τμήμα / αντικείμενο εντός της εικόνας, και όχι στο σύνολο αυτής.

Στην περίπτωση της ομαδοποίησης σε νέες ή ήδη υπάρχουσες ομάδες των αντικειμένων εντός μίας εικόνας προτείνεται η μη εποπτευόμενη και εποπτευόμενη εξόρυξη αντίστοιχα, όπως αυτές αναπτύχθηκαν στην προηγούμενη παράγραφο (ταυτοποίηση μοτίβου σε κείμενα), με την ίδια βελτιστοποίηση στους τελικούς υπολογισμούς και αναμονή παρόμοιων αποτελεσμάτων. Όταν αναζητείται η ανάλυση ολόκληρης της εικόνας στο σύνολό της, προτείνεται να υπολογιστούν τα blobs αυτής

τοπικά σε συγκεκριμένη επεξεργαστική μονάδα και έπειτα ως αυτούσια διανύσματα πλέον να αναλυθούν από το υπόλοιπο κατανεμημένο σύστημα, με τη χρήση των μεθόδων που έχουν ήδη παρουσιαστεί. Τέλος, επιστρέφονται τα αποτελέσματα (κατηγορίες στις οποίες εμπίπτουν τα διανύσματα) στην αρχική επεξεργαστική μονάδα, γίνεται η συνάθροιση αυτών και η καθολική κατηγοριοποίηση της εικόνας.

Ανάπτυξη

Τα κατηγοριοποιημένα πλέον διανύσματα εντός μίας εικόνας μπορούν να χρησιμοποιηθούν για την ανίχνευση αντικειμένων εντός αυτής, με χαρακτηριστικά παραδείγματα την εύρεση ενός σημείου αναφοράς σε μία ακτινογραφία, την αναγνώριση ενός προσώπου σε ένα σύστημα κλειστής παρακολούθησης ή την ανάλυση των στοιχείων μιας εικόνας σε ένα μέσο κοινωνικής δικτύωσης. Επίσης, η μελέτη μιας εικόνας στο σύνολό της μπορεί να δείξει αν αυτή εμπίπτει σε συγκεκριμένη κατηγορία κατοχυρωμένων πνευματικών δικαιωμάτων ή αν αποτελεί προϊόν υποκλοπής από κάποιον χρήστη.

ΚΕΦΑΛΑΙΟ 8

ΣΥΜΠΕΡΑΣΜΑΤΑ

Στη σημερινή εποχή, με τον τεράστιο όγκο πληροφορίας που απαρτίζει και διακινείται μέσω του Παγκόσμιου Ιστού, είναι απαραίτητη η ύπαρξη αυτοματοποιημένων μεθόδων εξαγωγής πληροφορίας. Τα κείμενα και οι εικόνες, φύσει όντας οι συνηθέστερες μορφές δεδομένων που εμφανίζονται στο διαδίκτυο αλλά και στην καθημερινότητα, είναι και οι συνηθέστερες μορφές από τις οποίες απαιτείται πληροφοριακή ανάλυση. Έχουν αναπτυχθεί διάφορες τεχνολογίες σε επίπεδο υλικού (συστοιχίες υπολογιστών) και λογισμικού (αλγόριθμοι, προγραμματιστικά πλαίσια) για να επιτευχθεί αυτή η εξαγωγή. Σε κάθε αίτημα ανάλυσης πληροφορίας, πρέπει να επιλέγεται ο κατάλληλος συνδυασμός τεχνολογιών με σκοπό την βελτιστοποίηση (χρονικά και εκ του αποτελέσματος) της επίλυσης του προβλήματος, καθώς και να αναζητείται αν κρίνεται απαραίτητη η χρήση των τεχνολογιών αυτών. Προτείνεται η αποφυγή καθολικής χρήσης ενός συνδυασμού μεθόδων εξόρυξης, αλλά η μαθηματική διερεύνηση της εν δυνάμει επιτάχυνσης της εκάστοτε επίλυσης, προκειμένου να υπάρξει βελτιστοποίηση αυτής και να αποφευχθούν περιπτώσεις επιβράδυνσης ή άσκοπης χρήσης υπολογιστικών πόρων.

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] United Nations Statistics https://www.itu.int/en/ITU-D/Statistics/Pages/stat/default.aspx

[2] Gantz, Reinsel. (2012). THE DIGITAL UNIVERSE IN 2020 : BigData, Bigger Digital Shadows,and Biggest Growth in the Far East

[3] The Zettabyte Era Officially Begins (How Much is That?) Thomas Barnett, Jr. Cisco Systems, Inc

[4] Diebold, Francis. (2012). On the Origin(s) and Development of the Term 'Big Data'. SSRN Electronic Journal. 10.2139/ssrn.2152421.

[5] De Mauro, Andrea & Greco, Marco & Grimaldi, Michele. (2014). What is Big Data? A Consensual Definition and a Review of Key Research Topics. 10.13140/2.1.2341.5048.

[6] De Mauro, Andrea & Greco, Marco & Grimaldi, Michele. (2016). A formal definition of Big Data based on its essential features. Library Review. 65. 122-135. 10.1108/LR-06-2015-0061.

[7] Kalbandi, Ishwarappa & Anuradha, J.. (2015). A Brief Introduction on Big Data 5Vs Characteristics and Hadoop Technology. Procedia Computer Science. 48. 319-324. 10.1016/j.procs.2015.04.188.

[8] Zerhari, Btissam & Ait Lahcen, Ayoub & Mouline, Salma. (2015). Big Data Clustering: Algorithms and Challenges.

[9] Fisher, DeLine, Czerwinski & Drucker. (2012). Interactions with Big Data Analytics.

[10] Cuquet, Martí & Vega-gorgojo, Guillermo & Lammerant, Hans & Finn, Rachel & ul Hassan, Umair. (2017). Societal impacts of big data: challenges and opportunities in Europe.

[11] Bakri, Mohamed. (2020). Big data healthcare paper.

[12] China launches ‘close contact detector app’ for coronavirus risk https://www.pharmaceutical-technology.com/news/china-coronavirus-contact-detector-app/

[13] How Coursera uses Data Visualization and Clustering to Categorize Content https://www.analyticsvidhya.com/blog/2019/01/coursera-data-driven-content-categorization-algorithm/

[14] Hewage, Thulara & Halgamuge, Malka & Syed, Ali & Ekici, Gullu. (2018). Review: Big Data Techniques of Google, Amazon, Facebook and Twitter. Journal of Communications. 13. 94-100. 10.12720/jcm.13.2.94-100.

[15] Skroutz Analytics https://engineering.skroutz.gr/blog/skroutz-analytics/

[16] Pourghomi, Pardis & Dordevic, Milan & Safieddine, Fadi. (2020). Facebook Fake Profile Identification: Technical and Ethical Considerations. International Journal of Pervasive Computing and Communications.

[17] Menon, Aravind. (2012). Big Data @ Facebook. 10.1145/2378356.2378364.

[18] Curran, Kevin & Graham, Sarah & Temple, Christopher. (2011). Advertising on Facebook. International Journal of E-Business Development. 1.

[19] Masri, Hela & Tekaya, Balkiss & Feki, Sirine & Tekaya, Tasnim. (2020). Recent applications of big data in finance. 10.1145/3423603.3424056.

[20] Al-Khasawneh, Mahmoud. (2020). Big Data Applications and Tools.

[21] Salas, Liliana. (2020). Analysis of YouTube's Content ID System Through Two Different Perspectives. 10.1007/978-3-030-43687-2_18.

[22] Yan, Fei. (2019). Music Recognition Algorithm based on T-S Cognitive Neural Network. Translational Neuroscience. 10. 135-140. 10.1515/tnsci-2019-0023.

[23] How Reface works https://hey.reface.ai/howitworks/

[24] Makinist, Semiha & Ay, Betul & Aydin, Galip. (2020). Average Neural Face Embeddings for Gender Recognition. European Journal of Science and Technology. 522-527. 10.31590/ejosat.araconf67.

[25] Ahsan, Umar & Bais, Abdul. (2016). A Review on Big Data Analysis and Internet of Things. 325-330. 10.1109/MASS.2016.048.

[26] Cho, Wonhee & Choi, Eunmi. (2017). Spatial Big Data Analysis System for Vehicle-Driving GPS Trajectory. 296-302. 10.1007/978-981-10-5041-1_50.

[27] OASA - Τηλεματική App https://www.oasa.gr/%CE%B5%CE%BE%CF%85%CF%80%CE%B7%CF%81%CE%AD%CF%84%CE%B7%CF%83%CE%B7-%CE%B5%CF%80%CE%B9%CE%B2%CE%B1%CF%84%CF%8E%CE%BD/%CE%B5%CF%81%CE%B3%CE%B1%CE%BB%CE%B5%CE%AF%CE%B1/%CF%84%CE%B7%CE%BB%CE%B5%CE%BC%CE%B1%CF%84%CE%B9%CE%BA%CE%AE-app/

[28] Ji, Changqing & LI, YU & Qiu, Daowen & JIN, YINGWEI & XU, YUJIE & Awada, Uchechukwu & Li, Keqiu & QU, WENYU. (2013). Big data processing: Big challenges. Journal of Interconnection Networks. 13. 10.1142/S0219265912500090.

[29] Ünver, Mahmut & Erguzen, Atilla. (2016). A STUDY ON DISTRIBUTED FILE SYSTEMS: An example of NFS.

[30] IBM - Hadoop vs. Spark: What's the Difference? https://www.ibm.com/cloud/blog/hadoop-vs-spark

[31] Graham, Richard & Woodall, Timothy & Squyres, Jeffrey. (2005). Open MPI: A flexible high performance MPI. 228-239. 10.1007/11752578_29.

[32] Araújo Silva, Vitor & Barros, Graiciany & Campagnole dos Santos, André & Campolina, Daniel. (2019). THE LTHN COMPUTER CLUSTER.

[33] Microsoft - What is IaaS? https://azure.microsoft.com/en-us/overview/what-is-iaas/#overview

[34] Elavarasi, kilandeswari & Sathiyabhama. (2011). A SURVEY ON PARTITION CLUSTERING ALGORITHMS

[35] Theofilou. (2019). Ανάλυση Κόμβων και Σύγκριση Τεχνικών Συσταδοποίησης μέσω Εξόρυξης Δεδομένων Κοινωνικών Δικτύων

[36] Sint, Rolf & Stroka, Stephanie & Schaffert, Sebastian & Ferstl, Roland. (2009). Combining Unstructured, Fully Structured and Semi-Structured Information in Semantic Wikis..

[37] Bharati, M. & Ramageri, Bharati. (2010). Data mining techniques and applications. Indian Journal of Computer Science and Engineering. 1.

[38] Coenen, Frans. (2011). Data mining: Past, present and future. Knowledge Eng. Review. 26. 25-29. 10.1017/S0269888910000378.

[39] Sumiran, Keerthi. (2018). An Overview of Data Mining Techniques and Their Application in Industrial Engineering.

[40] Gupta, Vishal & Lehal, Gurpreet. (2009). A Survey of Text Mining Techniques and Applications. Journal of Emerging Technologies in Web Intelligence. 1. 10.4304/jetwi.1.1.60-76.

[41] Google - How Google autocomplete works in Search https://blog.google/products/search/how-google-autocomplete-works-search/

[42] Li, Yang & Yang, Tao. (2017). Word Embedding for Understanding Natural Language: A Survey. 10.1007/978-3-319-53817-4.

[43] Turku University - Word Embedding Demo http://bionlp-www.utu.fi/wv_demo/

[44] Zahradníková, Barbora & Duchovičová, Soňa & Schreiber, Peter. (2015). Image Mining: Review and New Challenges. International Journal of Advanced Computer Science and Applications. 6. 10.14569/IJACSA.2015.060732.

[45] Vijayalatha. (2017). A SURVEY OF IMAGE MINING TECHNIQUES AND APPLICATIONS

[46] Qizhi Xiao, Kun Qin, Zequn Guan and Tao Wu, "Image mining for robot vision based on concept analysis," 2007 IEEE International Conference on Robotics and Biomimetics (ROBIO), 2007, pp. 207-212, doi: 10.1109/ROBIO.2007.4522161.

[47] Brahmaih, Kala. (2021). Graph Mining and Exploration Techniques.

[48] Leong, Mihalcea & Hassan. (2010). Text Mining for Automatic Image Tagging

[49] Mehta.(2012). Online Matching and Ad Allocation

[50] Nitin, Jain. (2010). CHAPTER 2 Pipelining Pipelining: Basic and Intermediate Concepts.

[51] Etiemble, Daniel. (2018). 45-year CPU evolution: one law and two equations.

[52] Intel - Intel® Core™ i5-9600K Processor https://ark.intel.com/content/www/us/en/ark/products/134896/intel-core-i59600k-processor-9m-cache-up-to-4-60-ghz.html

[53] Sanchez, Luis & Fernández, Javier & Sotomayor, Rafael & Escolar, Soledad & García, José. (2013). A Comparative Study and Evaluation of Parallel Programming Models for Shared-Memory Parallel Architectures. New Generation Computing. 31. 139-161. 10.1007/s00354-013-0301-5.

[54] Jogalekar, Prasad & Woodside, Murray. (2000). Evaluating the scalability of distributed systems. Parallel and Distributed Systems, IEEE Transactions on. 11. 589 - 603. 10.1109/71.862209.

[55] Amazon - Types of Cloud Computing https://aws.amazon.com/types-of-cloud-computing/

[56] Rehman, Khawaja Ubaid & Rehman, Kh & Ashraf, Muhammad. (2019). A Comparative Analysis of Distributed and Parallel Computing. 13. 60-67. 10.21015/vtse.v13i2.507.

[57] Kshemkalyani & Singhal. (2008). Distributed Computing Principles, Algorithms, and Systems

[58] N. M. Garcia, M. M. Freire and P. P. Monteiro, "The Ethernet Frame Payload Size and Its Effect on IPv4 and IPv6 Traffic," 2008 International Conference on Information Networking, 2008, pp. 1-5, doi: 10.1109/ICOIN.2008.4472813.

[59] Dean & Ghemawat. (2004). MapReduce: Simplified Data Processing on Large Clusters

[60] Apache - HDFS Architecture Guide https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html

[61] Apache - MapReduce Tutorial https://hadoop.apache.org/docs/r1.2.1/mapred_tutorial.html

[62] Apache - Apache Spark™ is a unified analytics engine for large-scale data processing. https://spark.apache.org/

[63] Apache - Apache Spark Examples https://spark.apache.org/examples.html

[64] Salman, Raied & Kecman, Vojislav & Li, Qi & Strack, Robert & Test, Erick. (2011). Two-Stage Clustering with k-Means Algorithm. 10.1007/978-3-642-21937-5_11.

[65] Morissette, Laurence & Chartier, Sylvain. (2013). The k-means clustering technique: General considerations and implementation in Mathematica. Tutorials in Quantitative Methods for Psychology. 9. 15-24. 10.20982/tqmp.09.1.p015.

[66] Zhang, Zhongheng. (2016). Introduction to machine learning: K-nearest neighbors. Annals of Translational Medicine. 4. 218-218. 10.21037/atm.2016.03.37.

[67] Guo, Gongde & Wang, Hui & Bell, David & Bi, Yaxin. (2004). KNN Model-Based Approach in Classification.

[68] Nielsen, Frank. (2016). Hierarchical Clustering. 10.1007/978-3-319-21903-5_8.

[69] Murtagh, Fionn & Contreras, Pedro. (2011). Methods of Hierarchical Clustering. Computing Research Repository - CORR. 10.1007/978-3-642-04898-2_288.

[70] Dan, MUNTEANU. (2007). Vector space model for document representation in information retrieval. Annals of Dunarea de Jos. 2007.

[71] Snowball Steming - Stemming Demo https://snowballstem.org/demo.html

[72] Jason Clemons. SIFT: SCALE INVARIANT FEATURE TRANSFORM BY DAVID LOWE

[73] Apache – Machine Learning Library (MLlib) Guide https://spark.apache.org/docs/latest/ml-guide.html

[74] Shree Nayar - Computer Science Department, School of Engineering and Applied Sciences, Columbia University - First Principles of Computer Vision Lecture Series https://fpcv.cs.columbia.edu/

[75] Lowe. (2004).Distinctive Image Features from Scale-Invariant Keypoints.