Αναφορά Πρακτικής Άσκησης

Κανακάρης Νικόλαος it21113@hua.gr

Σεπτέμβριος 2015

Περιεχόμενα

1 Γενικά

Η πρακτική άσκηση εκπονήθηκε στην εταιρεία **Software Competitiveness International**, κατά το χρονικό διάστημα 24/07/2015 - 24/09/2015. Επιβλέπων της πρακτικής άσκησης ήταν η Κα **Κατερίνα Ρουκουνάκη**. Η συγκεκριμένη αναφορά περιορίζεται από τη συμφωνία τήρησης απορρήτου (**Non-disclosure agreement (NDA)**) που ισχύει για το έργο, στα πλαίσια του οποίου εκπονήθηκε.

Το αντικείμενο της άσκησης ήταν η εναλλακτική υλοποίηση ενός αλγορίθμου ιεραρχικής συσταδοποίησης (**hierarchical cluster**).

2 Περιγραφή

2.1 Σκοπός

Ο σκοπός της πρακτικής άσκησης ήταν η υλοποίηση ενός χρονικά αποδοτικού αλγορίθμου για συσσωρευτική ιεραρχική συσταδοποίηση.

2.2 Έρευνα

Αρχικά, μελετήθηκε η υπάρχουσα υλοποίηση ενός αλγορίθμου συσσωρευτικής ιεραρχικής συσταδοποίησης (**Average Linkage Hierarchical Clustering**), με χρήση του κριτηρίου μέσης σύνδεσης (**Average Linkage**). Η συγκεκριμένη υλοποίηση είχε υπερβολικά μεγάλο χρόνο εκτέλεσης για μέσου μεγέθους σύνολα δεδομένων εισόδου.

Γι αυτό το λόγο ξεκίνησε η αναζήτηση κάποιου άλλου αλγορίθμου ο οποίος θα απαιτούσε μικρότερο χρόνο εκτέλεσης. Η αναζήτηση έγινε σε διάφορες δημοσιεύσεις. Συγκεκριμένα, οι δημοσιεύσεις που μελετήθηκαν ήταν:

- **Distributed Hierarchical Document Clustering** [1]
- **Parallel Hierarchical Clustering on Shared Memory Platforms** [2]
- **RACHET: An Efficient Cover-Based Merging of Clustering Hierarchies from Distributed Datasets**
- **A Novel Parallel Algorithm for Clustering Documents Based on the Hierarchical Agglomerative Approach** [3]
- **Minimum Spanning Trees and Single Linkage Cluster Analysis (MiST-SLiCA)** [4]
- **DiSC: A Distributed Single-Linkage Hierarchical Clustering Algorithm using MapReduce (DiSC)** [5]

Αφού συγκεντρώθηκαν όλες οι δημοσιεύσεις, μελετήθηκαν, με σκοπό να βρεθεί η κατάλληλη. Τελικά, η δημοσίευση στην οποία βασίστηκε η πρακτική άσκηση, ήταν η δημοσίευση στην οποία περιγράφεται ο αλγόριθμος DiSC.

3 Σχεδίαση

Ο αλγόριθμος βασίζεται στο γεγονός, ότι το πρόβλημα της συσσωρευτικής ιεραρχικής συσταδοποίησης ενός συνόλου δεδομένων με χρήση του κριτηρίου μονής σύνδεσης (**Single Linkage Criterion**) είναι ισοδύναμο με το πρόβλημα της εύρεσης του ελάχιστου επικαλυπτικού δέντρου (**minimum spanning tree**) στο γράφο που προκύπτει με βάση το ίδιο σύνολο δεδομένων.

Υπάρχουν αρκετοί αλγόριθμοι που βρίσκουν με αποδοτικό και γρήγορο τρόπο το ελάχιστο επικαλυπτικό δέντρο ενός γράφου, μεταξύ των οποίων είναι:

- **ο αλγόριθμος του Kruskal**
- **ο αλγόριθμος του Prim**
- **ο αλγόριθμος του Boruvka**

Ο αλγόριθμος DiSC κατασκευάζει αρχικά έναν μη κατευθυνόμενο πλήρη γράφο με βάση τα δεδομένα εισόδου. Στη συνέχεια διαχωρίζει το σύνολο των κόμβων που αποτελούν το γράφο αυτό σε **s** ξένα μεταξύ τους υποσύνολα, και συνδυάζει τα υποσύνολα αυτά ανά δύο προκειμένου να κατασκευάσει $\binom{s}{2}$ υπογράφους.

Σε κάθε υπογράφο εφαρμόζεται ο αλγόριθμος του Prim με σκοπό να υπολογιστεί το ελάχιστο επικαλυπτικό δέντρο του. Η επιλογή του αλγορίθμου του Prim βασίστηκε στο ότι δεν απαιτεί κάθε φορά την εύρεση της μικρότερης ακμής που δε δημιουργεί κύκλο, στο γράφο, σε αντίθεση με τον αλγόριθμο του Kruskal.

Τα ελάχιστα επικαλυπτικά δέντρα που προκύπτουν από το παραπάνω βήμα συγχωνεύονται ανά K με χρήση του αλγορίθμου του Kruskal μέχρι να προκύψει τελικά ένα ελάχιστο επικαλυπτικό δέντρο που να περιέχει όλους τους κόμβους του αρχικού γράφου.

Το τελευταίο βήμα του αλγορίθμου περιλαμβάνει τη μετατροπή της λύσης στο πρόβλημα της εύρεσης του ελάχιστου επικαλυπτικού δέντρου στη λύση του ισοδύναμου προβλήματος συσσωρευτικής ιεραρχικής συσταδοποίησης.

3.1 Ο αλγόριθμος DiSC

Ο ψευδοκώδικας του αλγορίθμου DiSC στον οποίο βασίστηκε η υλοποίηση φαίνεται παρακάτω. Οι παράμετροι που δέχεται ως είσοδο ο αλγόριθμος είναι το σύνολο **D** των δεδομένων που θέλουμε να ομοδοποιήσουμε και το πλήθος **K** των **MSTs** που συγχωνεύονται σε κάθε βήμα.

Algorithm 1 Outline of DiSC, a distributed SHC algorithm

INPUT: a dataset D, K

OUTPUT: a MST for D

1: Divide D into s roughly equal-sized splits: $D_1, D_2, ..., D_s$
2: Form C_s^2 subgraphs containing the complete subgraph for every pair in $\{(D_i, D_j)|i < j \text{ and } i, j \in [1, s]\}$
3: Use Prim's algorithm to compute the local MST for each subgraph in parallel, and output the MST's edge list in increasing order of edge weight
4: **repeat**
5: Merge the intermediate MSTs for every K subgraphs using the idea of Kruskal's algorithm
6: **until** all vertices belong to the same MST
7: **return** the final MST

3.2 Μετατροπή ελάχιστου επικαλυπτικού δέντρου σε δενδρόγραμμα

Η μεταροπή ενός ελάχιστου επικαλυπτικού δέντρου στο αντίστοιχο δενδρόγραμμα βασίστηκε στον παρακάτω ψευδοκώδικα:

Algorithm 2 Μετατροπή MST σε Dendrogram

INPUT: οι ακμές του MST E
OUTPUT: το dendrogram D

1: Δημιούργησε μία ταξινομημένη ουρά Q από τις ακμές E
2: Έστω ο πίνακας C
3: Δημιούργησε clusters του ενός κόμβου και αποθήκευσέ τα στον πίνακα C
4: **while** όσο δε βρίσκονται όλοι οι κόμβοι σε ένα cluster **do**
5: $e \leftarrow Poll(Q)$
6: $v \leftarrow e[1]$
7: $w \leftarrow e[2]$
8: Συγχώνευσε το cluster στο οποίο βρίσκεται ο κόμβος v με το cluster στο οποίο βρίσκεται ο κόμβος w
9: Πρόσθεσε το στιγμιότυπο του C στη λίστα D
10: **end while**
11: **return** D

4 Υλοποίηση

Η υλοποίηση του DiSC έγινε στη γλώσσα προγραμματισμού Java. Παρόλο που η συγκεκριμένη υλοποίηση είναι παράλληλη ο αλγόριθμος DiSC είναι σχεδιασμένος για κατανεμημένα περιβάλλοντα.

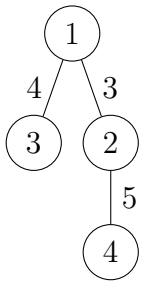
Για την υλοποίηση του αλγορίθμου χρησιμοποιήθηκαν οι υλοποιήσεις των αλγορίθμων **Prim, Kruskal** που βρίσκονται στην ιστοσελίδα http://algs4.cs.princeton.edu/code/. Ο κάτοχός της διανέμει τον κώδικα με βάση το **GNU General Public License, version 3 (GPLv3)** [6].

5 Παράδειγμα εκτέλεσης

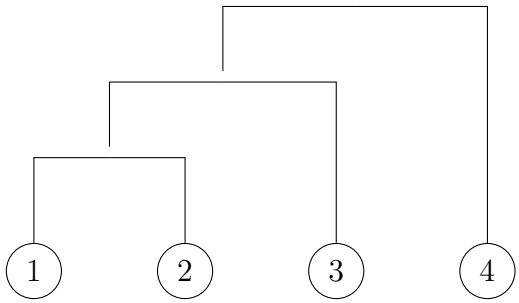
Έστω ότι έχουμε ένα σύνολο σημείων που οι αποστάσεις τους περιγράφονται από τον παρακάτω πίνακα:

	1	**2**	**3**	**4**
1	0	3	4	10
2	3	0	20	5
3	4	20	0	10
4	10	5	10	0

Με βάση τον παραπάνω πίνακα η υλοποίηση κατασκευάζει το παράκατω ελάχιστο επικαλυπτικό δέντρο και μετά το δενδρόγραμμα που αντιστοιχεί σε αυτό, εμφανίζεται παρακάτω:



Η έξοδος θα είναι το παρακάτω **dendrogram**:



6 Αποτελέσματα

Η αρχική υλοποίηση για συσσωρευτική ιεραρχική συσταδοποίηση βασιζόταν στο κριτήριο μέσης σύνδεσης και χρειαζόταν περίπου 48 λεπτά για την κατασκευή του δενδρογράμματος από ένα σύνολο περίπου 2.000 σημείων. Αυτό οφείλεται στο ότι με τη χρήση του συγκεκριμένου κριτηρίου για την κατασκευή κάθε επιπέδου του δενδρογράμματος απαιτείται ο υπολογισμός όλων των τιμών του πίνακα αποστάσεων από την αρχή.

Η υλοποίηση που έγινε στα πλαίσια της πρακτικής άσκησης και βασιζόταν εν μέρει στον αλγόριθμο DiSC κατάφερε να παράγει το δενδρόγραμμα για το ίδιο σύνολο δεδομένων σε λιγότερο από 10 δευτερόλεπτα.

7 Αναφορές

[1] http://citeseerx.ist.psu.edu/viewdoc/download?rep=rep1&type=pdf&doi=10.1.1.128.2309

[2] http://www.computer.org/csdl/proceedings/hipc/2012/2372/00/06507511.pdf

[3] http://airccse.org/journal/jcsit/0411csit11.pdf

[4] http://www.cs.ucsb.edu/~veronika/MAE/mstSingleLinkage_GowerRoss_1969.pdf

[5] http://sc13.supercomputing.org/sites/default/files/WorkshopsArchive/pdfs/wp106s1.pdf

[6] http://www.gnu.org/copyleft/gpl.html