ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ
ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ
ΚΑΤΕΥΘΥΝΣΗ: ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΗΡΕΣΙΩΝ

ΣΥΛΛΟΓΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΕΞΟΡΥΞΗ ΓΝΩΣΗΣ ΑΠΟ ΚΟΙΝΩΝΙΚΑ ΔΙΚΤΥΑ

ΕΦΑΡΜΟΓΗ DATA ANALYSIS ΤΕΧΝΙΚΩΝ ΣΕ ΣΥΝΟΛΑ ΔΕΔΟΜΕΝΩΝ ΑΠΟ ΤΟ ΚΟΙΝΩΝΙΚΟ ΔΙΚΤΥΟ TWITTER

Συγγραφέας: Τσούμας Ηλίας | Ε 10169
Επιβλέπων Καθηγητής: Δουλκερίδης Χρήστος

Πτυχιακή Εργασία υποβληθείσα στο Τμήμα Ψηφιακών Συστημάτων του Πανεπιστημίου Πειραιώς

Πειραιάς, Σεπτέμβριος 2016

UNIVERSITY OF PIRAEUS
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGIES
DEPARTMENT OF DIGITAL SYSTEMS

BACHELOR GRADUATE THESIS
AREA OF STUDY: E-SERVICES

DATA MINING AND KNOWLEDGE DISCOVERY FROM SOCIAL MEDIA

IMPLEMENTATION DATA ANALYSIS METHODS ON DATA COLLECTION FROM TWITTER

Author: Tsoumas Ilias | E 10169
Supervising Professor: Doulkeridis Christos

Bachelor Thesis submitted to the Department of Digital Systems of the University of Piraeus

Piraeus, Greece, September 2016

Ευχαριστίες

Ευχαριστώ αρχικά όλους τους φίλους μου, που ανέχονται τις παραξενιές μου, και πιο συγκεκριμένα σε σχέση με την παρούσα εργασία τους σχετικούς με το αντικείμενο Σταύρο, Τιμολέοντα και Χρήστο που ήταν πάντα εκεί ώστε να μοιραστώ τους προβληματισμούς μου, να συζητήσουμε σχετικά και να εμπνευστώ από την δική τους πορεία. Εν συνεχεία ευχαριστώ τους γονείς μου Κωνσταντίνο, Μαριγούλα και την αδερφή μου Γεωργία που μου στάθηκαν διακριτικά και συγχρόνως αδιάλειπτα όπως σε κάθε έκφανση της ζωής μου. Τέλος ευχαριστώ θερμά για την πολύτιμη συμβολή του στη διεξαγωγή και την ολοκλήρωση αυτής της εργασίας, τον επιβλέποντα καθηγητή μου, Δουλκερίδη Χρήστο για την καθοδήγηση και την προσφορά των γνώσεων του με άμεσο και συνεπή τρόπο, πράγμα που αποδεικνύουν και τα 103 mails που ανταλλάξαμε σε αυτό το διάστημα, συμπεριφορά που κρίνω αναγκαίο να αναφέρω σαν άλλο ένα παράδειγμα θετικής λειτουργίας του Ελληνικού Πανεπιστημίου, στις δύσκολες συνθήκες της σημερινής ελληνικής κοινωνίας, όταν περισσεύουν οι θετικές προθέσεις.

Συλλογή Δεδομένων και Εξόρυξη Γνώσης από Κοινωνικά Δίκτυα

Εφαρμογή data analysis τεχνικών σε σύνολα δεδομένων από το κοινωνικό δίκτυο twitter

Περίληψη

Η πτυχιακή αυτή εργασία πραγματοποιήθηκε στο πλαίσιο του προπτυχιακού προγράμματος σπουδών του τμήματος Ψηφιακών Συστημάτων του Πανεπιστημίου Πειραιώς. Καθότι ένα τμήμα με πρόγραμμα σπουδών κυρίως προσανατολισμένο στους κλάδους των Δικτυοκεντρικών Συστημάτων και Υπηρεσιών και των Τηλεπικοινωνιακών Συστημάτων και Δικτύων και με σκοπό την ανάδειξη εξειδικευμένων επιστημόνων ικανών να συμβάλλουν στην ανάπτυξη, υλοποίηση και διαχείριση συστημάτων σύγχρονης ψηφιακής τεχνολογίας, επιλέχθηκε για την ολοκλήρωση του ένα θέμα γύρω από το σχετικά νεοσύστατο κλάδο των μεγάλων δεδομένων, της διαχείρισης αυτών και κυρίως της εξόρυξης γνώσης από τον παγκόσμιο ιστό και ειδικότερα τα κοινωνικά δίκτυα.

Ζούμε σε μια εποχή που οι άνθρωποι αφιερώνουν σημαντικότατο μέγεθος του χρόνου τους στα κοινωνικά δίκτυα, όπου καταναλώνουν αλλά και παράγουν ασύλληπτα, για παλαιότερες εποχές, μεγέθη πληροφορίας. Η διαχείριση όλης αυτής της πληροφορίας έχει πολύπλευρα ωφέλη. Με την κατάλληλη επεξεργασία μπορούμε να εξάγουμε πολύτιμη γνώση και συμπεράσματα σχεδόν για τις περισσότερες εκφάνσεις της ανθρώπινης δραστηριότητας μιας και έχουμε να κάνουμε με πληροφορίες που γεννιούνται από ένα τεράστιο και πολύμορφο πληθυσμό ατόμων σε ένα περιβάλλον που ομοιάζει αρκετά ως προς αυτό της πραγματικής κοινωνίας.

Την λύση σε αυτό το πρόβλημα εξόρυξης των δεδομένων και εξαγωγής γνώσης από αυτά έρχονται να δώσουν οι κλάδοι της πληροφορικής “data mining”, “data analysis”. Στην παρούσα εργασία θα ασχοληθούμε αρχικά με την εξαγωγή δεδομένων από το κοινωνικό δίκτυο twitter και έπειτα με την απαιτούμενη επεξεργασία αυτών ώστε με αυτά να τροφοδοτήσουμε αλγορίθμους machine learning ώστε να μπορέσουμε να έχουμε μια αυτόματη ομαδοποίηση των δεδομένων βάσει του περιεχομένου τους. Τέλος θα ακουμπήσουμε λίγο τον τομέα του “topic detection” ώστε με τα εργαλεία που δίνει να βγάλουμε στην επιφάνεια τις κρυμμένες ενότητες που ενυπάρχουν στις συλλογές δεδομένων μας.

Σημαντικοί όροι: K-means, Ward, Παραγοντοποίηση Πίνακα (NMF), ανάλυση δεδομένων, εξόρυξη γνώσης, twitter, συσταδοποίηση, ιεραρχική ανάλυση, κοινωνικά δίκτυα

Data Mining and Knowledge Discovery from Social Media

Implementation Data Analysis methods on Data Collection from Twitter

Abstract

This thesis was carried out as part of the undergraduate degree program Digital Systems, University of Piraeus, a curriculum mainly oriented in the sectors of Network-Oriented and Telecommunication Systems and Services aiming to develop future scientists capable of contributing to the development, implementation and management of modern digital systems. To this end, subject of the thesis is related to the newly developed domain of Big Data, their management and knowledge extraction from the web and especially social networks.

We live in an age where people devote an important amount of their time on social networks, where they consume and produce unimaginable for earlier times, information sizes. The management of all this information has multifaceted benefits. With proper treatment of the data, we can extract valuable knowledge and conclusions almost for most aspects of human activity, as the disclosed information comes from a huge and diverse population of individuals in an environment that is similar enough to the real society.

The solution to the problem of knowledge extraction from data comes from the IT industry and more specifically with the technologies of “data mining” and “data analysis”. In this document we will first present how we can export data from the social network Twitter, followed by processing them in order to able to “feed” machine learning algorithms and cluster the data according to their content. In the end we will deal with “topic detection”, i.e. a number of tools provided in order to discover hidden themes and concepts from out data collections.

Keywords: K-means, Ward, matrix factorization (NFM), data analysis, knowledge mining, twitter, clustering, agglomerative clustering, social networks

Περιεχόμενα

ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΙΚΕΣ ΕΝΝΟΙΕΣ

1.1 Αντικείμενο Εργασίας

Είναι γεγονός ότι ζούμε στην εποχή της πληροφορίας και σε αυτό έχει συμβάλει τα τελευταία χρόνια η ραγδαία ανάπτυξη και χρήση του διαδικτύου. Το τελευταίο εμπλουτίζεται καθημερινά με κείμενο που παράγεται από τους ίδιους τους χρήστες του και η πληροφορία είναι πλέον διαθέσιμη πιο εύκολα από ποτέ. Παρόλα αυτά, η τεράστια αυτή εξάπλωση δημιουργεί το πρόβλημα ότι η πληροφορία που διακινείται είναι αχανής με αποτέλεσμα η ανεύρεση πληροφορίας από τους χρήστες να καταντά μια εργασία χρονοβόρα και επίπονη. Έτσι, η ανάπτυξη συστημάτων αυτόματης ομαδοποίησης των δεδομένων και ακόμα περισσότερο η αναγνώριση του σημασιολογικού περιεχομένου των ομάδων αυτών είναι πιο αναγκαία από ποτέ. Η παρούσα εργασία ασχολείται με την συλλογή κειμένων και την εξόρυξη γνώσης από αυτά και ειδικότερα με την συλλογή μεγάλων ομάδων δεδομένων (tweets) από το κοινωνικό δίκτυο twitter και την κατάλληλη επεξεργασία τους για την εξαγωγή γνώσης. Στο πλαίσιο εκπόνησης της, χρησιμοποιήθηκαν κάποιες τεχνικές συλλογής δεδομένων από τον ιστό και κάποιοι βασικοί αλγόριθμοι εξόρυξης γνώσης.

Οι αλγόριθμοι που εφαρμόστηκαν είναι οι:

1. K-means

2. Hierarchical document clustering (Ward algorithm)

3. Non-negative Matrix Factorization - NMF

Παράλληλα με τη χρήση των αλγορίθμων, η οποία γινόταν με την βοήθεια της γλώσσας Python, έγινε και μία προσπάθεια σύνδεσης όλων των σταδίων (επικοινωνία με κοινωνικό δίκτυο, συλλογή δεδομένων, προεπεξεργασία δεδομένων, δημιουργία συλλογής κειμένων, ευρετηριοποίηση της συλλογής, κατασκευή πινάκων term-document, εφαρμογή αλγορίθμου, αποθήκευση αποτελεσμάτων) σε ένα ενιαίο command line tool εργαλείο για linux περιβάλλοντα.

1.2 Δομή Εργασίας

Στο Κεφάλαιο 1 γίνεται μια συνοπτική αναφορά στο περιεχόμενο της εργασίας. Στο Κεφάλαιο 2 κάνουμε μια θεωρητική παρουσίαση των απαραίτητων “εργαλείων” για την παρούσα εργασία στα θέματα “data mining”, “data analysis”, “topic detection”. Στο Κεφάλαιο 3 γίνεται η παρουσίαση των αλγορίθμων που χρησιμοποιούνται. Στο Κεφάλαιο 4 περιγράφονται εκτενώς οι φάσεις σχεδίασης, ανάπτυξης, και τα θέματα που αντιμετωπίστηκαν. Στο Κεφάλαιο 5 παρουσιάζονται τα αποτελέσματα των πειραμάτων που διεξήχθησαν και γίνεται σχολιασμός και σύγκριση. Τέλος, στο Κεφάλαια 6 γίνεται μια παρουσίαση κάποιων παρατηρήσεων και συμπερασμάτων και τέλος προτείνονται κάποια σενάρια μελλοντικών επεκτάσεων.

ΚΕΦΑΛΑΙΟ 2

ΑΝΑΣΚΟΠΗΣΗ

2.1 Εισαγωγή

Για την καλύτερη κατανόηση όλων όσων θα ακολουθήσουν στα επόμενα κεφάλαια, προηγείται σε αυτό το κεφάλαιο μια αναφορά σε βασικές έννοιες της ανάκτησης πληροφορίας και εξόρυξης γνώσης.

2.2 Λέξεις, κείμενα και συλλογές κειμένων

Η θεμελιώδης μονάδα σε ένα οποιοδήποτε κείμενο είναι η λέξη (term). Κάθε λέξη αποτελείται από χαρακτήρες και όλες μαζί αποτελούν τις βασικές μονάδες εκείνες, από τις οποίες προκύπτουν οι έννοιες και η σημασία σε ένα κείμενο. Ο συνδυασμός των λέξεων με τους κανόνες που ορίζει η γραμματική και το συντακτικό σε μια γλώσσα παράγει αυτό που ονομάζουμε προτάσεις, οι οποίες κρύβουν μέσα τους πληροφορία για το θέμα του κειμένου. Με τη σειρά τους οι προτάσεις σχηματίζουν τις παραγράφους, οι οποίες αποτελούν μία πολύ σημαντική δομή ενός κειμένου αφού περιέχουν μία σειρά από ιδέες που έχουν σχέση μεταξύ τους. Όσο το μέγεθος του κειμένου μεγαλώνει, προστίθενται κι άλλες δομές και έτσι μιλούμε για ενότητες (sections), κεφάλαια (chapters), ολόκληρα έγγραφα (documents) και τέλος για συλλογή εγγράφων (corpus, collection of documents).

Τα έγγραφα στην εξόρυξη γνώσης θεωρούνται βασικές μονάδες της ανάλυσης που θα εκτελεστεί στη συλλογή, διότι αυτά είναι συνήθως γραμμένα από ένα συγκεκριμένο συγγραφέα και περιέχουν έννοιες για συγκεκριμένο θέμα. Έγγραφα μπορούν να θεωρηθούν οι επιστημονικές δημοσιεύσεις (papers), οι εκθέσεις, τα άρθρα εφημερίδων και περιοδικών, τα βιβλία. Ανάλογα όμως με την ανάλυση η οποία πρόκειται να γίνει ή ανάλογα με τους στόχους του αναλυτή "έγγραφο" μπορεί να αποτελέσει ένα ξεχωριστό κεφάλαιο, μία παράγραφος ή και μία πρόταση ακόμα. Στην παρούσα εργασία θα θεωρήσουμε ως έγγραφο κάθε tweet.

2.3 Μοντέλο "Bag of Words" *[24]*

Αν και όσα αναφέρθηκαν προηγουμένως είναι απόλυτα κατανοητά και λογικά στη διαίσθηση μας για τον τρόπο που έχουμε οι άνθρωποι να κατανοούμε ένα κείμενο όταν το διαβάζουμε, παρ' όλα αυτά στην εξόρυξη γνώσης η συντακτική δομή των προτάσεων και των παραγράφων αγνοείται. Αυτό γίνεται με σκοπό την αποτελεσματικότερη διαχείριση του όγκου των κειμένων. Σαν αποτέλεσμα, οι προτάσεις και ένα ολόκληρο κείμενο μπορούν να θεωρηθούν απλώς σαν ένα σετ από λέξεις ή ως ένα "bag of words" όπως συχνά λέγεται στο χώρο της εξόρυξης γνώσης.

Παράδειγμα Bag of Word μοντέλου:

Ας υποθέσουμε ότι έχουμε δύο κείμενα:

The sky is blue and all the leaves are green.

The sun is bright.

Από τη μικρή αυτή συλλογή προκύπτει το παρακάτω λεξιλόγιο με 10 όρους:

$\{"the" : 1,$

$"sky" : 2,$

$"is" : 3,$

$"blue" : 4,$

$"and" : 5,$

$"leaves" : 6,$

$"are" : 7,$

$"green" : 8,$

$"sun" : 9,$

$"bright" : 10,\}$

Και έτσι το κάθε κείμενο αναπαριστάται με ένα διάνυσμα που περιέχει 10 στοιχεία:

$[2,1, 1, 1, 1, 1, 1, 1, 0, 0]$

$[1,0, 1, 0, 0, 0, 0, 0, 1, 1]$

με το $i - στο$ στοιχείο να περιέχει τη συχνότητα εμφάνισης του όρου i στο κείμενο.

2.3.1 Πίνακας Term-Document

Ο πίνακας term-document είναι άμεσα συνυφασμένος με το μοντέλο bag of words και είναι ένας πίνακας που περιέχει τις συχνότητες εμφάνισης των όρων μιας συλλογής στα κείμενα. Συνήθως οι γραμμές αντιπροσωπεύουν τους όρους και οι στήλες τα κείμενα αλλά μπορεί ανάλογα με την υλοποίηση να συμβαίνει και το αντίστροφο. Σε κάθε περίπτωση η ουσία δεν αλλάζει. Ένας τέτοιος πίνακας για τα 2 κείμενα που είχαμε προηγουμένως είναι ο παρακάτω:

$terms/documents$	$document_1$	$document_2$
the	*2*	*1*
sky	*1*	*0*
is	*1*	*1*
$blue$	*1*	*0*
and	*1*	*0*
$leaves$	*1*	*0*
are	*1*	*0*
$green$	*1*	*0*
sun	*0*	*1*
$bright$	*0*	*1*

Πίνακας 1

Ένας term-document πίνακας έχει το χαρακτηριστικό ότι είναι πολύ αραιός, δηλαδή έχει πολλά μηδενικά στοιχεία. Σε μια παρουσίασή του για το PLSA ο Thomas Hofmann αναφέρει χαρακτηριστικά ότι ένας τέτοιος πίνακας μοιάζει με έναν "ωκεανό" από μηδενικά μέσα στον οποίο ζουν μερικά μικρά "νησιά" από μη αρνητικές τιμές. Για αυτό το λόγο είναι πολύ πρακτικό ο πίνακας term-document να χρησιμοποιείται σε spare μορφή.

2.4 Προεπεξεργασία κειμένων *[7]*

2.4.1 Αφαίρεση διγραμμάτων, συμβόλων, σημείων στίξης, αριθμών, emails και urls

Για να μειώσουμε κατά πολύ τη διάσταση των μοναδικών όρων συχνά αφαιρούμε τα σημεία στίξης, τους αριθμούς, τα emails και τα urls από τα κείμενα με το σκεπτικό ότι δεν προσφέρουν κάτι στην εξόρυξη θέματος. Επίσης στην παρούσα εργασία αφαιρέθηκαν και τα emoticons των οποία εκτεταμένη χρήση γίνεται στα κοινωνικά δίκτυα, και χρειάζεται ιδιαίτερη μεταχείριση ώστε να παραχθούν συμπεράσματα από αυτά.

2.4.2 Αφαίρεση τετριμμένων λέξεων

Η αφαίρεση τετριμμένων λέξεων (stopwords) είναι μία κοινή τακτική κατά την οποία δεν λαμβάνονται υπόψη λέξεις που θεωρούνται ότι δεν προσθέτουν ιδιαίτερη σημασία στο νόημα του κειμένου. Για παράδειγμα τέτοιες λέξεις στα αγγλικά μπορεί να είναι οι: "the", "this", "he", "somebody","is" ενώ για τα ελληνικά μπορεί να είναι οι: "το", "κάποιος", "κάθε", "άλλος". Στις περιπτώσεις που χρειάζεται να γίνει αναζήτηση ολόκληρων φράσεων πχ "The Second World War", "Take That" μέσα στις οποίες υπάρχουν τετριμμένες λέξεις προτείνεται η διατήρησή και όχι η αφαίρεσή τους. Παρόλα αυτά επειδή στη δική μας περίπτωση δεν παρέχεται η δυνατότητα αναζήτησης, θεωρήθηκε σκόπιμο να αφαιρεθούν.

2.4.3 Αποκοπή καταλήξεων

Η αποκοπή καταλήξεων (stemming) είναι μια διαδικασία κατά την οποία αφαιρείται η κατάληξη μιας λέξης. Επειδή η κάθε γλώσσα έχει τους δικούς της κανόνες και καταλήξεις δεν μπορεί να χρησιμοποιηθεί ο ίδιος αλγόριθμος stemming για όλες τις γλώσσες. Έτσι στην εργασία χρησιμοποιήθηκε Snowball(Porter2) Stemmer για την αγγλική. Το πλεονέκτημα αυτής της μεθόδου είναι ότι μας βοηθά να μειώσουμε τη διάσταση των όρων, αφού διαφορετικοί (ορθογραφικά) όροι όπως "άνθρωπος", "ανθρώπου", "ανθρώπους", "άνθρωποι" γίνονται ο ίδιος όρος "ανθρωπ" για την ευρετηρίαση.

2.5 Μοντέλο διανυσματικού χώρου(Vector Space Model) - Συχνότητα Όρου και Αντίστροφη Συχνότητα Όρου (TF-IDF)

Το μοντέλο διανυσματικού χώρου (Vector Space Model - VSM) είναι ένα αλγεβρικό μοντέλο αναπαράστασης κειμένων(και οποιουδήποτε αντικειμένου, γενικά) ως διανύσματα. Τα επιμέρους διανύσματα αυτού μπορεί να μας παρουσιάζουν την σημαντικότητα ενός όρου-term (tf-idf) ή την παρουσία ή απουσία ενός όρου-term (Bag of Words) σε ένα κείμενο-document. Ευρέως το VSM ερμηνεύεται ως ένας χώρος όπου όροι και κείμενα αναπαρίστανται ως διανύσματα αριθμών αντί της αρχικής συμβολοσειράς που τα αναπαριστούσε. Έτσι ο VSM παρουσιάζει τα χαρακτηριστικά που εξάγονται από ένα κείμενο.

Παρακάτω θα προσπαθήσουμε να ορίσουμε μαθηματικά το VSM και μαζί το tf-idf(term frequency–inverse document frequency που είναι ο στατιστικός τρόπος που πρότεινε ο Salton μαζί με το κλασικό VSM για να υπολογίζουμε την σημαντικότητα ενός όρου σε μια συλλογή κειμένων) με ένα παράδειγμα.

2.5.1 Διανυσματική Αναπαράσταση και μια αναλυτική κατασκευή πίνακα Συχνότητας Όρου(TF) *[19]*

Το πρώτο βήμα είναι να δημιουργήσουμε ένα λεξικό(Bag of Words) όπως παρουσιάστηκε στο 2.3. Εδώ απλοποιούμε ακόμα περισσότερο το δοθέντα κείμενα χάριν ευκολίας.

$document_1$: The sky is blue.

$document_2$: The sun is bright.

$$E(t) = \begin{cases} 1, t = "blue" \\ 2, t = "sun" \\ 3, t = "bright" \\ 4, t = "sky" \end{cases}$$

Όμως όπως βλέπουμε το λεξικό είναι λίγο διαφορετικό απ’ ότι αυτό στο 2.2 γιατί έχουμε λάβει υπόψιν κάποια κομμάτια από την απαραίτητη προεπεξεργασία των δεδομένων που ορίσαμε στο κεφ. 2.3. Έχουν αφαιρεθεί τα σημεία στίξης, οι τετριμμένες λέξεις (stopwords), ώστε να μικρύνουμε το τελικό πίνακα μας, να αφαιρέσουμε λέξεις που δεν περιέχουν σημαντικό σημασιολογικό περιεχόμενο και

εμφανίζονται στο μεγαλύτερο αριθμό των κειμένων μας και δρουν, τελικά, ως θόρυβος για τον σκοπό της εργασίας μας.

Το επόμενο βήμα είναι να χρησιμοποιήσουμε την term-frequency ώστε να αναπαραστήσουμε κάθε όρο στο διανυσματικό χώρο. Αυτό θα το κάνουμε με την βοήθεια της σχέσης 1, η οποία δεν κάνει τίποτα παραπάνω από το να μετράει πόσες φορές κάθε όρος του λεξικού μας $E(t)$ παρουσιάζεται σε κάθε έγγραφο.

$$tf(t,d) = \sum_{x \in d} fr(x,t)$$

Σχέση 1

όπου $fr(x,t)$ ορίζεται ως εξής:

$$fr(x,t) = \begin{cases} 1, & x = t \\ 0, & \acute{\alpha}\lambda\lambda o \end{cases}$$

Σχέση 2

Οπότε η $tf(t,d)$ κάθε φορά επιστρέφει τον αριθμό των εμφανίσεων του όρου t στο κείμενο d. Ένα παράδειγμα αυτού μπορεί να είναι $tf(\text{"sun"}, document_2) = 1$, δηλαδή έχουμε 1 εμφάνιση του όρου "sun" στο κείμενο νούμερο 2. Στην συνέχεια αναπαριστούμε την γενική μορφή του διανύσματος ενός εγγράφου.

$$\overrightarrow{v_{d_n}} = \left(tf(t_1,d_n),\ \ tf(t_2,d_n), \dots, tf(t_n,d_n)\right)$$

και πιο συγκεκριμένα στο παράδειγμα μας

$$\overrightarrow{v_{d_1}} = \left(tf(t_1,d_1),\ \ tf(t_2,d_1), \dots, tf(t_4,d_1)\right) = (1,0,0,1)$$

$$\overrightarrow{v_{d_2}} = \left(tf(t_1,d_2),\ \ tf(t_2,d_2), \dots, tf(t_4,d_2)\right) = (0,1,1,0)$$

Πλέον με τις πληροφορίες που έχουμε μπορούμε να δημιουργήσουμε έναν term document matrix ή αλλιώς έναν term-frequency matrix (απαντάται και με τους δύο

τίτλους, αναφέρονται στην ίδια δομή) $|D| \times F$, όπου $|D|$η πληθικότητα χώρου των εγγράφων ή πιο απλά το πλήθος των εγγράφων και F ο αριθμός των χαρακτηριστικών, στην περίπτωση μας το μέγεθος του λεξικού.

Ο πίνακας του παραδείγματος μας είναι ο εξής:

$$M_{|D|\times F} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

Οι πίνακες tf που θα δημιουργούμε, όπως και αυτός του παραδείγματος θα είναι αρκετά αραιοί(sparse) όπως αναφέραμε και στην ενότητα 2.3. Αυτό το χαρακτηριστικό θα το χρησιμοποιήσουμε στην συνέχεια, ώστε να επιλύσουμε προβλήματα που δημιουργούνται με τους μεγάλου μεγέθους πίνακες.

2.5.2 Συχνότητα Όρου και Αντίστροφη Συχνότητα Όρου(TF-IDF) *[18], [7]*

Παραπάνω είδαμε αρκετά αναλυτικά πως δημιουργείται ο πίνακας συχνότητας όρου TF για την αναπαράσταση ενός κειμένου. Ωστόσο το βασικό πρόβλημα με την συγκεκριμένη προσέγγιση έγκειται στο ότι δίνει περισσότερη σημασία στους όρους που εμφανίζονται συχνότερα και υποβαθμίζει αυτούς που είναι πιο σπάνιοι, ο οποίοι διαισθητικά αλλά και εμπειρικά αντιλαμβανόμαστε πως είναι αυτοί που φέρουν την περισσότερη πληροφορία.

Η μέθοδος συχνότητας όρου και αντίστροφης συχνότητας όρου (TF-IDF) έρχεται να λύσει το παραπάνω πρόβλημα. Ο TF-IDF υπολογίζει το πόσο σημαντικός είναι ένας όρος ενός εγγράφου μέσα σε μια συλλογή, γι' αυτό και χρησιμοποιεί τοπικές αλλά και καθολικές παραμέτρους, διότι λαμβάνει υπόψη του όχι τον κάθε όρο αποκομμένο, αλλά έως μέρος του συνόλου-συλλογής των εγγράφων. Αυτό που γίνεται στην συνέχεια, είναι πως ο TF-IDF περιορίζει την ένταση των συχνών όρων, ενώ αυξάνει αυτή την σπάνιων. Ουσιαστικά με αυτό το τρόπο, συνυπολογίζει, την παραπάνω διαίσθηση μας πως ένας συχνότερος όρος που εμφανίζεται π.χ. 100 φορές περισσότερο από ό, τι ένας άλλος δεν είναι και 100 φορές πιο σημαντικός. Γι' αυτό στους αντίστοιχους υπολογισμούς του ο TF-IDF χρησιμοποιεί τον λογάριθμο.

Το να χρησιμοποιήσουμε απλά τον TF ως έχει για τον TF-IDF μπορεί να μας οδηγήσει σε προβλήματα όπως το spamming keywords, το οποίο πρόβλημα δημιουργείται γιατί μπορεί σε κάποιο έγγραφο να έχουμε επανάληψη όρων για βελτίωση της κατάταξης του σε συστήματα IR(Information Retrieval). Ο πιο συχνός τρόπος διαχείρισης αυτού του προβλήματος είναι να εφαρμόσουμε κανονικοποίηση στο TF των όρων του διανυσματικού μας χώρου.

Ο χώρος των εγγράφων ορίζεται ως $D = \{d_1, d_2, \dots, d_n\}$ όπου n ο αριθμός των εγγράφων της συλλογής.

Ο πίνακας αντίστροφης συχνότητας όρου(idf) ορίζεται από την σχέση 3, παρακάτω.

$$idf(t) = \log \frac{|D|}{1 + |\{d \in D : t \in D\}|}$$

Σχέση 3

πιο απλά,

$$idf(t) = \log \frac{|\pi\lambda\acute{\eta}\theta o\varsigma\ \kappa\epsilon\iota\mu\acute{\epsilon}\nu\omega\nu\ \sigma\tau\eta\ \sigma\upsilon\lambda\lambda o\gamma\acute{\eta}|}{1 + |\pi\lambda\acute{\eta}\theta o\varsigma\ \kappa\epsilon\iota\mu\acute{\epsilon}\nu\omega\nu\ \pi o\upsilon\ \pi\epsilon\rho\iota\acute{\epsilon}\chi o\upsilon\nu\ \tau o\nu\ \acute{o}\rho o\ t|}$$

και έπειτα ο TF-IDF από την σχέση 4.

$$tf - idf(t) = tf(t, d) \times idf(t)$$

Σχέση 4

Θεωρούμε τέσσερα νέα κείμενα ώστε να υλοποιήσουμε τον TF-IDF.

doc_1: Welcome to Wikipedia, the free encyclopedia that anyone can edit. Free free free anyone edit edit.

doc_2: The Halifax Gibbet was an early guillotine, or decapitating machine. free anyone edit.

doc_3: Ask questions about using Wikipedia. Free anyone edit.

doc_4: Serving as virtual librarians, Wikipedia volunteers tackle your questions on a wide range of subjects. Free.

Επιλέγουμε τους όρους με $DF >= 2$ και έχουμε τον εξής πίνακα TF:

TF	doc_1	doc_2	doc_3	doc_4
anyone	2	1	1	0
edit	3	1	1	0
free	4	1	1	1
questions	0	0	1	1
wikipedia	1	0	1	1

Πίνακας 2

$$M_{tf} = \begin{bmatrix} 2 & 1 & 1 & 0 \\ 3 & 1 & 1 & 0 \\ 4 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix}$$

Πριν υπολογίσουμε τον TF-IDF πίνακα υπολογίζουμε το διάνυσμα "doclength" και τις στήλες "DF", "iDF":

TF	doc_1	doc_2	doc_3	doc_4	DF	iDF
anyone	2	1	1	0	3	0
edit	3	1	1	0	3	0
free	4	1	1	1	4	-0.22314
questions	0	0	1	1	2	0.287682
wikipedia	1	0	1	1	3	0
doclength	4	3	5	3		

Πίνακας 3

δημιουργούμε έναν διαγώνιο πίνακα με τις idf τιμές κάθε όρου που θα τις υπολογίσουμε με την σχέση 3.

$$M_{idf} = \begin{bmatrix} idf(t_1) & 0 & 0 & 0 & 0 \\ 0 & idf(t_2) & 0 & 0 & 0 \\ 0 & 0 & idf(t_3) & 0 & 0 \\ 0 & 0 & 0 & idf(t_4) & 0 \\ 0 & 0 & 0 & 0 & idf(t_5) \end{bmatrix}$$

$$M_{tf-idf} = M_{tf} \times M_{idf}$$

Σχέση 5

Το διάνυσμα doclength περιέχει τα μη-μηδενικά στοιχεία κάθε στήλης, δηλαδή δηλώνει το πλήθος των διαφορετικών όρων που υπάρχουν σε ένα κείμενο. Το διάνυσμα DF αναγράφει το πλήθος των κειμένων που περιέχουν τον όρο στην αντίστοιχη γραμμή (πχ ο όρος "questions" υπάρχει στα κείμενα 3,4 άρα έχει DF=2). Το διάνυσμα iDF υπολογίζεται με τον τύπο που δόθηκε παραπάνω.

Τέλος παράγουμε τον πίνακα TF-IDF με την εφαρμογή του τύπου παραπάνω και τον κανονικοποιούμε με την Ευκλείδια νόρμα L2-norm.

$TF - IDF$	doc_1	doc_2	doc_3	doc_4
anyone	0	0	0	0
edit	0	0	0	0
free	-0.44629	-0.12883	-0.09979	-0.12883
questions	0	0	0.128655	0.166093
wikipedia	0	0	0	0

Πίνακας 4

οπότε το κάθε στοιχείο i, j του πίνακα προκύπτει από την σχέση 6.

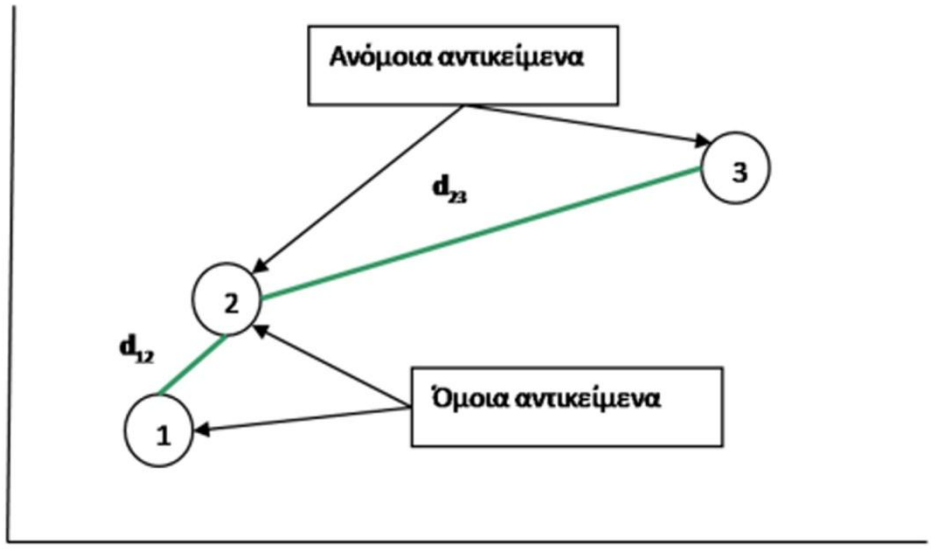
$$M_{tf-idf} = \frac{M_{tf-idf}}{\left\| M_{tf-idf} \right\|_2}$$

Σχέση 6

2.6 Ομοιότητα και απόσταση *[29]*

Οι αλγόριθμοι συσταδοποίησης στους οποίους θα αναφερθούμε στο επόμενο κεφάλαιο ομαδοποιούν τις παρατηρήσεις σύμφωνα με την ομοιότητα τους. Έτσι

είναι φανερό ότι ένα από τα βασικότερα ζητήματα είναι ο καθορισμός μέτρων ομοιότητας. Ένας τρόπος καθορισμού του βαθμού ομοιότητας δύο παρατηρήσεων είναι με τη χρήση της απόστασης τους. Ας θεωρήσουμε αρχικά μια απλή περίπτωση, όπου οι παρατηρήσεις αποτελούνται από δύο μόνο γνωρίσματα Χ και Υ και ότι και τα δύο γνωρίσματα παίρνουν αριθμητικές τιμές. Κάθε παρατήρηση μπορεί να αναπαρασταθεί στον δισδιάστατο χώρο Χ,Υ ως ένα σημείο. Δύο σημεία, τα οποία βρίσκονται κοντά στον δισδιάστατο χώρο, θεωρούνται όμοια, ενώ δύο σημεία, τα οποία βρίσκονται μακριά στον δισδιάστατο χώρο, θεωρούνται ανόμοια. Στο Σχήμα 2.1 απεικονίζονται τρία σημεία στον δισδιάστατο χώρο. Τα σημεία 1 και 2 θεωρούνται όμοια, ενώ τα σημεία 2 και 3 θεωρούνται ανόμοια.



Διάγραμμα 1

Εάν οι παρατηρήσεις έχουν n γνωρίσματα, τότε θεωρούνται σημεία στο χώρο των n διαστάσεων, και η ομοιότητα τους υπολογίζεται από την απόσταση τους σε αυτόν τον χώρο. Για τον υπολογισμό της ομοιότητας όμως δεν υπάρχουν μόνο οι διάφορες μετρικές υπολογισμού της απόστασης τους. Παρακάτω παρουσιάζονται οι πιο δημοφιλείς μέθοδοι υπολογισμού της ομοιότητας δύο στοιχείων.

2.6.1 Μέτρα ομοιότητας ως προς την απόσταση

Αρχικά το πιο δημοφιλές, ίσως, μέτρο ομοιότητας δύο παρατηρήσεων x_a και x_b που χρησιμοποιείται ευρέως είναι η **Ευκλείδεια** απόσταση. Θεωρούμε ότι οι παρατηρήσεις έχουν n γνωρίσματα. Η απόσταση μεταξύ των σημείων x_a και x_b συμβολίζεται ως $d(x_a, x_b)$. Η Ευκλείδεια απόσταση των σημείων x_a και x_b δίνεται από την σχέση 7, παρακάτω.

$$d(x_a, x_b) = \sqrt{\sum_{j=1}^{n} (x_{aj} - x_{bj})^2}$$

Σχέση 7

όπου x_{aj} είναι η τιμή της μεταβλητής j της παρατήρησης x_a. Η Ευκλείδεια απόσταση είναι η πιο διαδεδομένη, ωστόσο δεν είναι η μοναδική. Μια παραλλαγή της, η οποία χρησιμοποιείται επίσης συχνά, είναι η απόσταση **Manhattan**. Η απόσταση Manhattan ορίζεται από την σχέση 8.

$$d(x_a, x_b) = \sum_{j=1}^{n} |x_{aj} - x_{bj}|$$

Σχέση 8

Τέλος γενίκευση της Ευκλείδειας απόστασης και της απόστασης Manhattan είναι η απόσταση **Minkowski**, η οποία ορίζεται ως στην σχέση 9.

$$d(x_a, x_b) = \sqrt[p]{\sum_{j=1}^{n} (x_{aj} - x_{bj})^q}$$

Σχέση 9

2.6.2 Μέτρα ομοιότητας ανεξάρτητα της απόστασης

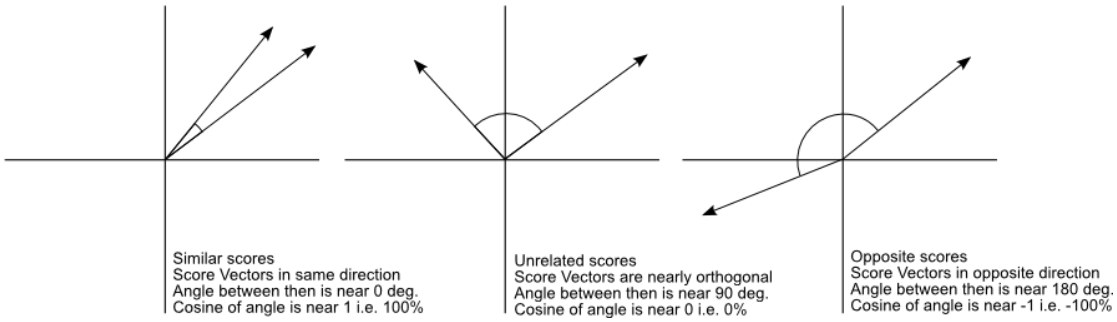
Όπως προαναφέρθηκε υπάρχουν πολλά μέτρα για τον καθορισμό της ομοιότητας δύο παρατηρήσεων. Παρακάτω θα σας παρουσιάσουμε μερικές που δεν σχετίζονται με την μεταξύ απόσταση των στοιχείων.

i. Cosine Similarity *[21]*

Η **ομοιότητα συνημιτόνου (cosine similarity)** μεταξύ δύο διανυσμάτων (δύο αντικειμένων ή πιο στοχευόμενα στην εργασία μας δύο εγγράφων του διανυσματικού χώρου που έχουμε κατασκευάσει σύμφωνα με τη συλλογή εγγράφων μας) είναι ένα μέτρο υπολογισμού του συνημιτόνου της γωνίας που σχηματίζεται μεταξύ των διανυσμάτων. Αυτή η μετρική λαμβάνει υπόψη μόνο την διεύθυνση και όχι το μέγεθος, και συγκρίνει πόσο σχετικά είναι τα δύο διανύσματα κοιτάζοντας μόνο με τη γωνία.

$$\cos\theta = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \, \|\vec{b}\|}$$

Σχέση 10



Διάγραμμα 2

Σημειώστε ακόμα πως δύο διανύσματα που το πρώτο δείχνει σε ένα σημείο μακριά από το δεύτερο διάνυσμα μπορούν να έχουν μια μικρή γωνία και αυτό είναι το κεντρικό σημείο για τη χρήση της ομοιότητας συνημιτόνου. Αν υποθέσουμε ότι έχουμε ένα έγγραφο με τη λέξη "ουρανό" να εμφανίζεται 200 φορές και ένα άλλο έγγραφο με τη λέξη "ουρανό" να εμφανίζεται 50, η Ευκλείδεια απόσταση μεταξύ τους θα είναι μεγάλη, αλλά η γωνία τους θα εξακολουθεί να είναι μικρή, επειδή δείχνουν προς την ίδια κατεύθυνση, το οποίο είναι αυτό που έχει σημασία όταν συγκρίνουμε έγγραφα.

ii. Jaccard similarity

Το μέτρο ομοιότητας που πρότεινε ο Paul Jaccard είναι μάλλον το απλούστερο από τα μέτρα ομοιότητας παραπάνω. Δεν είναι τίποτα περισσότερο από το μέτρο των κοινών χαρακτηριστικών τους(τομή) ως προς όλα τα χαρακτηριστικά και των δύο μαζί(ένωση). Μαθηματικά διατυπώνεται από την σχέση 11.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

Σχέση 11

ΚΕΦΑΛΑΙΟ 3

ΑΛΓΟΡΙΘΜΟΙ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ

3.1 Εισαγωγή

Η επιστημονική βιβλιογραφία παρουσιάζει ένα μεγάλο αριθμό διαφορετικών μεθόδων ομαδοποίησης και εξαγωγής λανθάνουσας γνώσης και ειδικότερα κρυμμένου περιεχομένου από μεγάλες συλλογές κειμένων.

Αναλυτικότερη αναφορά στην παρούσα εργασία θα γίνει σε δύο αλγορίθμους συσταδοποίησης, τον διαχωριστικό k-means clustering algorithm και τον ιεραρχικό Ward clustering algorithm, και τέλος στον εντελώς διαφορετικής προσέγγισης αλγόριθμο, που ανήκει στην ομάδα αλγορίθμων παραγοντοποίησης πίνακα, τον Non-negative Matrix Factorization - NMF.

3.2 Συσταδοποίηση - Διαχωριστική Ανάλυση Συστάδων

Οι διαχωριστικές μέθοδοι θεωρούν ένα πλήθος N σημείων και ένα πλήθος k συστάδων, και διαμερίζουν τα σημεία στις συστάδες. Τυπικά, το πλήθος των συστάδων k προκαθορίζεται από το χρήστη. Ξεκινώντας από έναν αρχικό διαχωρισμό, με μια επαναληπτική διαδικασία, τα σημεία μετακινούνται από μια συστάδα σε μιαν άλλη. Ο σχηματισμός των συστάδων γίνεται με τρόπο τέτοιο, ώστε να βελτιστοποιείται ένα κριτήριο διαχωρισμού. Στόχος είναι να δημιουργηθούν συστάδες, οι οποίες να περιέχουν όμοια αντικείμενα, ενώ τα αντικείμενα διαφορετικών συστάδων να είναι ανόμοια. Οι διαχωριστικές μέθοδοι παρουσιάζουν ευαισθησία στις αρχικές τους συνθήκες. Ένα σημαντικό πρόβλημα είναι το πλήθος των συστάδων k. Η εργασία του Dubes (1987) παρέχει καθοδήγηση για τον καθορισμό του πλήθους των συστάδων. Επίσης, για την εύρεση της καθολικά βέλτιστης λύσης θα έπρεπε να δοκιμαστούν όλοι οι δυνατοί διαχωρισμοί. Ωστόσο, λόγω υπολογιστικού κόστους, αυτό δεν είναι εφικτό. Στην πράξη εφαρμόζεται μια διαδικασία αρχικοποίησης του διαχωρισμού, και στη συνέχεια, μετακίνησης των σημείων. Οι διαχωριστικές μέθοδοι δημιουργούν ένα σύνολο συστάδων, σε αντίθεση με τις ιεραρχικές μεθόδους, οι οποίες δημιουργούν μια ιεραρχική δομή

διαδοχικών επιπέδων, όπου κάθε επίπεδο ορίζει ένα σύνολο συστάδων. Επίσης, είναι υπολογιστικά λιγότερο ακριβές από τις ιεραρχικές μεθόδους, και για τον λόγο αυτό μπορούν να εφαρμοστούν σε μεγαλύτερα σύνολα δεδομένων. Η πιο γνωστή μέθοδος διαχωριστικής ανάλυσης συστάδων είναι ο αλγόριθμός k-Means.

3.2.1 k-Means *[8], [1]*

Ο διαμεριστικός αλγόριθμος k-means είναι ένας από τους πιο απλούς και δημοφιλέστερους αλγορίθμους ομαδοποίησης που ανήκουν στην ευρύτερη κατηγορία των τεχνικών μάθησης χωρίς επίβλεψη. Ο αλγόριθμος αυτός είναι δημοφιλής εξαιτίας της απλότητας της υλοποίησης του και της γραμμικής πολυπλοκότητας του η οποία είναι της τάξης n (O(n)), όπου n το σύνολο των στοιχείων. Η διαδικασία της ομαδοποίησης ενός συνόλου δεδομένων με βάση τον k-means είναι εύκολη, αρκεί να είναι εκ των προτέρων καθορισμένος ο αριθμός (k) των clusters (ομάδων) που θα προκύψουν. Η κύρια ιδέα είναι να προσδιοριστούν αρχικά k centroids (κεντροειδή), ένα για κάθε cluster. Αυτά τα αρχικά centroids πρέπει να επιλεγούν με επιδέξιο τρόπο, γιατί διαφορετικές αρχικές θέσεις για τα centroids δίνουν διαφορετικά αποτελέσματα. Δηλαδή, η αρχική θέση των centroids επηρεάζει το αποτέλεσμα που θα δώσει ο αλγόριθμος. Έτσι, συχνά θεωρείται καλύτερη η επιλογή εκείνων των centroids ώστε να απέχουν μεταξύ τους όσο περισσότερο γίνεται. Το επόμενο βήμα είναι επιλογή κάθε στοιχείου από το σύνολο δεδομένων και συσχέτιση του με το κοντινότερο σε αυτό centroid. Όταν αυτό γίνει για όλα τα στοιχεία του συνόλου δεδομένων, το πρώτο βήμα έχει ολοκληρωθεί και μία πρώτη και «πρόχειρη» ομαδοποίηση έχει ήδη προκύψει. Στη συνέχεια, απαιτείται να υπολογιστούν ξανά k νέα centroids, τα οποία θα αποτελούν το κέντρο βάρους για κάθε ένα cluster που προέκυψε από το προηγούμενο βήμα. Αφού λοιπόν οριστούν τα νέα k centroids, ακολουθεί και πάλι η ίδια διαδικασία ανάθεσης καθενός από τα στοιχεία του συνόλου δεδομένων στο κοντινότερο με αυτό, νέο πλέον, centroid. Έτσι, γίνεται μια επανάληψη της ίδιας διαδικασίας. Αποτέλεσμα αυτής της επανάληψης είναι ότι σε κάθε βήμα τα centroids αλλάζουν θέση (ορίζονται νέα) και τα στοιχεία ανατίθενται στο κατάλληλο cluster κάθε φορά με βάση το κοντινότερο centroid. Όταν σε κάποια επανάληψη δεν σημειωθούν αντιμεταθέσεις στοιχείων, τότε τερματίζει η εκτέλεση του αλγορίθμου. Το αποτέλεσμα που προκύπτει είναι η ομαδοποίηση του συνόλου δεδομένων σε k clusters.

Ο αλγόριθμος στοχεύει να ελαχιστοποιήσει μία αντικειμενική συνάρτηση, την λεγόμενη συνάρτηση τετραγωνικού λάθους που ορίζεται από την σχέση 12.

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2$$

Σχέση 12

όπου

$$\left\| x_i^{(j)} - c_j \right\|^2$$

Σχέση 13

είναι ένα μέτρο απόστασης που χρησιμοποιείται για να μετρά την απόσταση κάθε στοιχείου $x_i^{(j)}$ από το centroid c_j του κάθε cluster. Όπου n το σύνολο των στοιχείων του συνόλου δεδομένων.

Ο παρακάτω πίνακας δείχνει συνοπτικά τα βήματα του αλγορίθμου k-means:

Είσοδος:

$D = \{x_1, x_2, \dots, x_n\}$// Σύνολο στοιχείων
k // Αριθμός επιθυμητών clusters

Έξοδος:

k // Σύνολο clusters

k-Means αλγόριθμος:

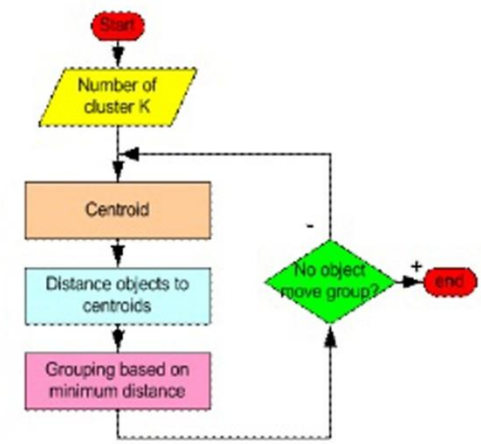
Ανέθεσε τιμές στα αρχικά centroids $C_1, C_2, \dots, C_n$
Επανέλαβε
    Ανέθεσε κάθε i x στο cluster με του οποίου το centroid η απόσταση είναι η μικρότερη
    Υπολόγισε νέα centroids για κάθε cluster
Μέχρι να συναντηθεί το κριτήριο σύγκλισης

Πίνακας 5

Αν και μπορεί να αποδειχθεί ότι ο αλγόριθμος πάντα τερματίζει, αξίζει να τονιστεί ότι δεν καταφέρνει πάντα να βρίσκει τη βέλτιστη λύση. Ο αλγόριθμος επηρεάζεται σημαντικά από τα αρχικά centroids. Για αυτό πολλές φορές συνίσταται η εκτέλεση του πολλές φορές μέχρι να μειωθεί η επίδραση αυτή.

Έστω ότι υπάρχουν n διανύσματα τα $x_1, x_2, \dots, x_n$ και όλα είναι της ίδιας διάστασης. Ακόμη είναι γνωστό ότι όλα εμπίπτουν σε k συμπαγή clusters, για $k < n$. Έστω m_i είναι το μέσο διάνυσμα του i cluster. Εφόσον τα clusters είναι σαφώς διαχωρισμένα μεταξύ τους, μπορεί να χρησιμοποιηθεί σαν μέτρο απόστασης μεταξύ των στοιχείων η Ευκλείδεια απόσταση ή και άλλα δημοφιλή μέτρα απόστασης, που έχουν αναλυθεί σε προηγούμενο κεφάλαιο. Αυτό σημαίνει ότι σε

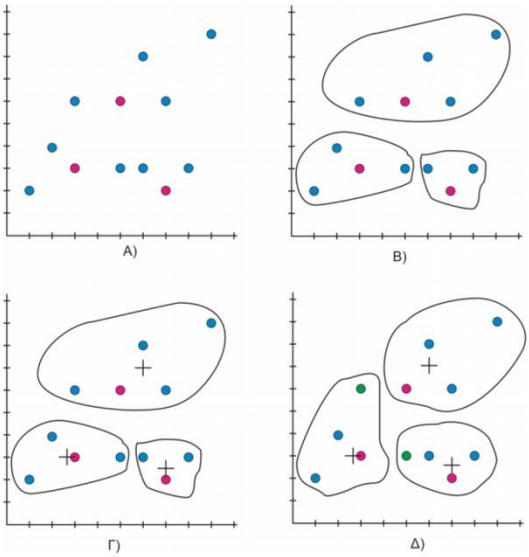
κάθε βήμα θα λέγεται: το στοιχείο x ανήκει στο cluster i, εάν η Ευκλείδεια απόσταση του από το centroid του i cluster είναι η μικρότερη σε σχέση με όλες τις άλλες αποστάσεις του από τα centroids των άλλων clusters. Έτσι βρίσκονται οι Ευκλείδειες αποστάσεις για όλα τα στοιχεία και κάθε ένα από αυτά ανατίθεται στο cluster από του οποίου το centroid απέχει λιγότερο (δηλαδή η Ευκλείδεια απόσταση είναι η μικρότερη). Στην συνέχεια υπολογίζονται τα νέα centroids και μετά πάλι οι Ευκλείδειες αποστάσεις όλων των στοιχείων για τα νέα centroids. Γίνονται οι κατάλληλες μετακινήσεις στοιχείων και η ίδια διαδικασία επαναλαμβάνεται μέχρι κανένα στοιχείο να μην μετακινείται σε άλλο cluster, δηλαδή τα clusters να μένουν αμετάβλητα.



Διάγραμμα 3

Παράδειγμα:

- Δίνονται: $\{2,4,10,12,3,20,30,11,25\}, k = 2$
- Τυχαία ανάθεσε μέσους όρους: $m_1 = 3, m_2 = 4$
- $k_1 = \{2,3\},\ k_2 = \{4,10,12,20,30,11,25\},\ m_1 = 2.5,\ m_2 = 16$
- $k_1 = \{2,3,4\},\ k_2 = \{10,12,20,30,11,25\},\ m_1 = 3,\ m_2 = 18$
- $k_1 = \{2,3,4,10\},\ k_2 = \{12,20,30,11,25\},\ m_1 = 4.75,\ m_2 = 19.6$
- $k_1 = \{2,3,4,10,11,12\},\ k_2 = \{20,30,25\},\ m_1 = 7,\ m_2 = 25$
- Σταμάτησε όταν τα clusters με αυτούς τους μέσους παραμένουν αμετάβλητα.



Διάγραμμα 4

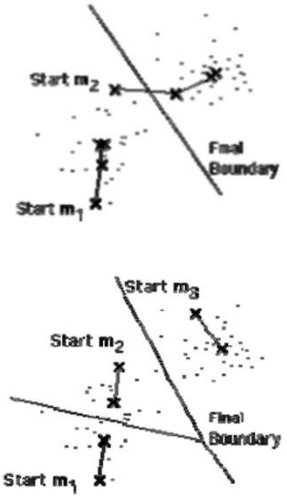
Αδυναμίες:

- Ο αλγόριθμος συγκλίνει σε τοπικό βέλτιστο και όχι σε καθολικό βέλτιστο.
- Ο τρόπος με τον οποίο ορίζονται τα αρχικά centroids δεν είναι σαφώς καθορισμένος. Ένας αρκετά δημοφιλής τρόπος επιλογής των αρχικών centroids, είναι να επιλεχθούν με τυχαίο τρόπο. Αυτή η μέθοδος εφαρμόζεται και στην παρούσα διπλωματική. Το αποτέλεσμα που προκύπτει εξαρτάται από τα αρχικά centroids. Συχνά προκύπτει μη βέλτιστη λύση λόγω της «κακής» αρχικής επιλογής των centroids. Για αυτό τον λόγο συνίσταται να γίνουν πολλές δοκιμές εκτέλεσης με διαφορετικά αρχικά centroids κάθε φορά.
- Μπορεί ακόμη ένα cluster να μείνει χωρίς μέλη και έτσι να μην ανανεωθεί κάποιο centroid. Πρόκειται για το γνωστό πρόβλημα των απόμακρων στοιχείων που πολλές φορές δεν συμπεριλαμβάνονται στη διαδικασία.
- Τα αποτελέσματα εξαρτώνται και από το μέτρο απόστασης που χρησιμοποιείται. Πολλές φορές χρειάζεται να γίνει κανονικοποίηση των στοιχείων του συνόλου δεδομένων προκειμένου να εφαρμοστεί κάποιο μέτρο απόστασης. Έτσι για παράδειγμα όταν ο αλγόριθμος εκτελεί clustering βάσει της Ευκλείδειας απόστασης, προϋποθέτει ότι τα δεδομένα των clusters έχουν όλα σφαιρικό σχηματισμό. Διάφορες έρευνες έχουν καταφέρει να επεκτείνουν τον k-means, ώστε να μπορεί να δουλέψει όχι

μόνο σε δεδομένα με σφαιρικό σχηματισμό αλλά και σε δεδομένα με ελλειπτικό σχηματισμό.

- Ακόμη μια αδυναμία του αλγορίθμου, είναι ότι δυσκολεύεται να αναγνωρίσει ομάδες με διαφορετικό σχηματισμό και μέγεθος. Το πρόβλημα εντείνεται κυρίως σε πολύ μεγάλα σύνολα δεδομένων. Συνήθως το επίπεδο δυσκολίας που αντιμετωπίζει ο αλγόριθμος σε τέτοια μεγάλα σύνολα δεδομένων έχει να κάνει και με την πυκνότητα που εμφανίζουν τα στοιχεία, που μπορεί αλλού να είναι μεγάλη αλλού μικρή και γενικά να ποικίλει.

- Τα αποτελέσματα εξαρτώνται από την τιμή του k, η οποία αποτελεί στοιχείο εισόδου για τον αλγόριθμο. Αν και υπάρχουν πολλοί τρόποι εκτίμησης του k και έχουν γίνει πολλές προσπάθειες προς αυτή την κατεύθυνση, δυστυχώς το πρόβλημα παραμένει ακόμη άλυτο. Ο αλγόριθμος δεν καταφέρνει να βρει το βέλτιστο k από μόνος του και να γίνει αυτό ευρέως αποδεκτό. Φυσικά με το βέλτιστο k, εννοείται εκείνο το k που αποδίδει με τον καλύτερο τρόπο τον διαχωρισμό του εκάστοτε συνόλου δεδομένων έτσι ώστε οι ομάδες που προκύπτουν να έχουν νόημα. Οι διάφοροι τρόποι εκτίμησης του k που υπάρχουν σήμερα στη βιβλιογραφία, έχουν αναφερθεί στο προηγούμενο κεφάλαιο. Αυτή η τελευταία αδυναμία του k-means μπορεί να γίνει πιο εύκολα κατανοητή μέσα από ένα παράδειγμα. Συχνά αποτελεί ενοχλητικό παράγοντα μιας και δεν είναι πάντα δυνατόν να υπάρχει γνώση για το πόσα clusters υπάρχουν, όταν πρέπει να εφαρμοστεί ομαδοποίηση σε δεδομένα του πραγματικού κόσμου. Στο Διάγραμμα 5 φαίνεται ότι όταν ο k-means εφαρμόστηκε στο ίδιο σύνολο δεδομένων την μία φορά έδωσε δύο clusters και την άλλη τρία (φαίνονται και οι μετακινήσεις των centroids κατά την εκτέλεση). Η απόφαση για το ποιο από τα δύο clusterings είναι καλύτερο και ποιο χειρότερο, δεν είναι απλή και γενικά δεν υπάρχει κάποιο γενικό κριτήριο αποδοχής της μίας λύσης έναντι της άλλης. Όπως ειπώθηκε, σε προηγούμενο κεφάλαιο για την εκτίμηση του αριθμού των clusters (k) έχουν προταθεί οι κατάλληλοι δείκτες, μέσω των οποίων μπορεί να εκτιμηθεί το βέλτιστο k. Ωστόσο χρειάζεται προσοχή γιατί όταν αυξάνεται πολύ το k μπορεί να συμβεί επικάλυψη.



Διάγραμμα 5

Παραλλαγές:

Υπάρχει ένας μεγάλος αριθμός παραλλαγών που έχουν προταθεί για αυτόν τον αλγόριθμο, έτσι για να αποφασίσει κανείς ποια έκδοση θα χρησιμοποιήσει πρέπει να εστιάσει στους σκοπούς της εφαρμογής. Μια πολύ γνωστή παραλλαγή του k-means είναι ο αλγόριθμος Lloyd's. Αυτός βασίζεται στην απλή παρατήρηση ότι η βέλτιστη θέση για ένα κέντρο είναι στο centroid του σχετικού cluster. Εξαιτίας της απλότητας και της ευελιξίας του, ο αλγόριθμος Lloyd's είναι πολύ δημοφιλής στην στατιστική ανάλυση. Πιο συγκεκριμένα, δεδομένου οποιουδήποτε αλγορίθμου clustering, ο Lloyd's μπορεί να εφαρμοστεί σαν μια φάση προ-επεξεργασίας για να βελτιωθεί η τελική παραμόρφωση. Αν και επιτυγχάνονται σημαντικές βελτιώσεις συνήθως η υλοποίησή του είναι αρκετά αργή, εξαιτίας του κόστους υπολογισμού των κοντινότερων γειτόνων. Μια απλή και συνάμα αποτελεσματική υλοποίηση του Lloyd's, ονομάζεται filtering αλγόριθμος. Αυτός ο αλγόριθμος, ξεκινά αποθηκεύοντας τα δεδομένα σε ένα kd-tree. Σε κάθε στάδιο του Lloyd's, υπολογίζεται το κοντινότερο κέντρο κάθε σημείου και κάθε κέντρο μετακινείται στο centroid του σχετικού γείτονα. Η ιδέα είναι να διατηρείται, για κάθε κόμβο του δέντρου ένα υποσύνολο υποψήφιων κέντρων. Αυτά τα υποψήφια κέντρα, για κάθε κόμβο κλαδεύονται, ή φιλτράρονται, καθώς μεταβιβάζονται στους κόμβους παιδιά. Επειδή το kd-tree υπολογίζεται για τα στοιχεία αντί για τα κέντρα, δεν υπάρχει η ανάγκη ενημέρωσης αυτής της δομής σε κάθε στάδιο του Lloyd's αλγορίθμου. Ο αλγόριθμος ISODATA ο οποίος περιλαμβάνει μία διαδικασία για αναζήτηση του καλύτερου αριθμού ομάδων με βάση κάποιο κόστος εκτέλεσης. Ο Fuzzy C-Means ο οποίος επεκτείνει τον κλασικό αλγόριθμο k-means χρησιμοποιώντας την θεωρία της ασαφής λογικής. Ο SAS PROC FASTCLUS, ο οποίος ελέγχει την διαδικασία ομαδοποίησης υιοθετώντας δύο ακόμα παραμέτρους, την min_size και max_rad. Η

πρώτη παράμετρος ελέγχει τον ελάχιστο αριθμό στοιχείων που μπορεί να έχει κάθε ομάδα ενώ η δεύτερη καθορίζει ότι η απόσταση κάθε στοιχείου μίας ομάδας από το κέντρο της ομάδας δεν πρέπει να είναι μεγαλύτερη του max_rad.

3.3 Συσταδοποίηση - Ιεραρχική Ανάλυση *[29]*

Η Ιεραρχική ΑΣ συνίσταται σε μια διαδικασία διαδοχικών συγχωνεύσεων ή διασπάσεων συστάδων. Οι σχετικές τεχνικές αντιστοίχως χωρίζονται σε συσσωρευτικές και διαιρετικές. Οι συσσωρευτικές (agglomerative) μέθοδοι αρχικά θεωρούν κάθε ξεχωριστό αντικείμενο ως μια συστάδα. Τα πιο όμοια αντικείμενα επιλέγονται και συγχωνεύονται, δημιουργώντας μια νέα συστάδα. Από τις συστάδες που προκύπτουν, επιλέγονται οι πιο όμοιες και συγχωνεύονται. Η διαδικασία επαναλαμβάνεται μέχρι να ενταχθούν όλα τα αντικείμενα σε μια ενιαία συστάδα. Οι συσσωρευτικές μέθοδοι έχουν ως αφετηριακό σημείο το κατώτερο επίπεδο της ιεραρχίας των διαδοχικών συγχωνεύσεων, και σταδιακά ανέρχονται τα επίπεδα. Υιοθετούν δηλαδή μια προσέγγιση «από κάτω προς τα επάνω» (bottom up). Οι διαιρετικές (divisive) μέθοδοι αρχικά θεωρούν όλα τα αντικείμενα ως μέλη μιας ενιαίας συστάδας. Η αρχική αυτή συστάδα διαιρείται σε δύο υποομάδες. Η διάσπαση γίνεται με τέτοιον τρόπο, ώστε οι υποομάδες οι οποίες θα προκύψουν θα έχουν τη μεγαλύτερη ανομοιότητα. Η διαδικασία των διαδοχικών διασπάσεων επαναλαμβάνεται μέχρι κάθε αντικείμενο να αποτελεί μια ξεχωριστή υποομάδα. Οι διαιρετικές μέθοδοι έχουν αφετηριακό σημείο το ανώτατο επίπεδο της ιεραρχίας και ακολουθούν μια προσέγγιση «από επάνω προς τα κάτω» (top down). Για την επιλογή των συστάδων δημιουργείται ένας πίνακας ανομοιότητας. Εάν τα δεδομένα περιέχουν N σημεία, τότε ο πίνακας είναι διαστάσεων NXN. Κάθε εγγραφή του πίνακα είναι ένα μέτρο ανομοιότητας ή απόστασης μεταξύ δύο σημείων. Ο πίνακας ανομοιότητας έχει την ακόλουθη μορφή:

$$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & \dots & 0 & & \\ \dots & \dots & \dots & 0 & \\ d(N,1) & \dots & \dots & d(N,N-1) & 0 \end{bmatrix}$$

όπου $d(x_1, x_2)$ είναι η απόσταση μεταξύ των σημείων x1 και x2. Εφόσον η απόσταση κάθε σημείου από τον εαυτό του είναι μηδενική $(d(xi, xi) = 0)$, οι εγγραφές της διαγωνίου από επάνω και αριστερά προς κάτω και δεξιά έχουν

μηδενικές τιμές. Επειδή η απόσταση μεταξύ δύο σημείων είναι συμμετρική $(d(xi, xj) = d(xj, xi))$, η διαγώνιος χωρίζει τον πίνακα σε δύο κατοπτρικά μέρη, οπότε διατηρούνται μόνο οι εγγραφές οι οποίες βρίσκονται κάτω από τη διαγώνιο.

Στην Ιεραρχική ΑΣ δημιουργείται μια ιεραρχία, η οποία περιλαμβάνει ένα σύνολο από δυνατές συστάδες. Κάθε επίπεδο της ιεραρχίας περιγράφει ένα συγκεκριμένο τρόπο διαμοιρασμού των αντικειμένων σε συστάδες. Αποτελεί αρμοδιότητα του χρήστη να αποφασίσει ποιο είναι το κατάλληλο επίπεδο, το οποίο περιγράφει έναν φυσικό τρόπο διαμοιρασμού των αντικειμένων, δηλαδή ποιες είναι οι συστάδες, οι οποίες είναι επαρκώς όμοιες μεταξύ τους. Εάν στα δεδομένα μας υπάρχουν N σημεία, τότε και στις δύο κατηγορίες μεθόδων υπάρχουν N-1 επίπεδα.

Τα βασικά **πλεονεκτήματα** των Ιεραρχικών Μεθόδων είναι τα ακόλουθα:

- Οι ιεραρχικές μέθοδοι παρουσιάζουν καλή προσαρμοστικότητα. Μπορούν να εντοπίσουν καλά διαχωρισμένες, επιμήκεις και ομόκεντρες συστάδες.
- Δημιουργούν πολλαπλά επίπεδα φωλιασμένων συστάδων και επιτρέπουν στον χρήστη να επιλέξει το επίπεδο που αυτός επιθυμεί.

Μειονεκτήματα των Ιεραρχικών μεθόδων είναι τα εξής:

- Κάθε ενέργεια, η οποία πραγματοποιείται σε ένα στάδιο, δεν είναι αντιστρέψιμη. Από τη στιγμή που δύο αντικείμενα ενταχθούν στην ίδια ομάδα, θα παραμείνουν στην ίδια ομάδα, και δεν υπάρχει δυνατότητα να διαχωριστούν αργότερα και να ενταχθούν σε διαφορετικές ομάδες.
- Οι ιεραρχικές μέθοδοι χρειάζεται να ελέγξουν πολλές αποστάσεις, και για τον λόγο αυτό καθυστερούν όταν χρειάζεται να επεξεργαστούν μεγάλο αριθμό αντικειμένων. Το υπολογιστικό κόστος είναι τουλάχιστον $O(N^2)$ όπου N το πλήθος των αντικειμένων.

3.3.1 Ιεραρχική Συσσωρευτική Ανάλυση Συστάδων

Οι συσσωρευτικές μέθοδοι εκτελούν διαδοχικές συγχωνεύσεις συστάδων. Σε κάθε επανάληψη οι δύο πλησιέστερες συστάδες συνενώνονται. Ο γενικός αλγόριθμος της Ιεραρχικής Συσσωρευτικής ΑΣ έχει ως ακολούθως:

- Αρχικά, κάθε ένα από τα N σημεία θεωρείται ως μια ξεχωριστή συστάδα. Στον πίνακα αποστάσεων καταγράφονται οι αποστάσεις μεταξύ των σημείων.

- Εντοπίζεται στον πίνακα αποστάσεων η μικρότερη τιμή. Η τιμή αυτή είναι η απόσταση των δύο πιο όμοιων συστάδων U και V (d(U,V)).
- Οι συστάδες U και V συνενώνονται σε μια ενιαία συστάδα UV. Στον πίνακα αποστάσεων, διαγράφονται οι γραμμές και οι στήλες που αντιστοιχούν στις συστάδες U και V, και προστίθεται μια γραμμή και μια στήλη για τη νέα συστάδα UV. Επαναϋπολογίζονται οι αποστάσεις μεταξύ των συστάδων.
- Επαναλαμβάνονται τα βήματα 2 και 3, N-1 φορές. Σε κάθε επανάληψη καταγράφονται οι συστάδες που συγχωνεύονται καθώς και οι αποστάσεις τους.

Για τον υπολογισμό της εγγύτητας των συστάδων είναι απαραίτητο ένα μέτρο. Έχουν προταθεί διάφοροι τρόποι μέτρησης της απόστασης μεταξύ των συστάδων. Εναλλακτικές μέθοδοι συσωρευτικής ΑΣ διαφοροποιούνται μεταξύ τους, ανάλογα με το μέτρο απόστασης το οποίο εφαρμόζουν. Στην παρούσα εργασία χρησιμοποιήθηκε η στατιστική μέθοδος Ward.

3.3.2 Μέθοδος Ward

Η μέθοδος του Ward (1963) διαφέρει σημαντικά από τις προηγούμενες μεθόδους, καθώς δεν υπολογίζει κάποια «απόσταση» μεταξύ των συστάδων. Κριτήριο για τη δημιουργία συστάδων είναι η μεγιστοποίηση της ομοιογένειας στο εσωτερικό των συστάδων. Το μέτρο που εφαρμόζεται είναι το άθροισμα του τετραγωνικού σφάλματος, και επιδίωξη της μεθόδου είναι η ελαχιστοποίηση του. Το ίδιο κριτήριο χρησιμοποιείται και από τον αλγόριθμο k-Means, οπότε η μέθοδος Ward μπορεί να θεωρηθεί το ιεραρχικό ανάλογο του k-Means.

$$E = \sum_{\mathrm{x} \in \mathrm{C_i}} (\mathrm{x} - \mathrm{m_i})^2$$

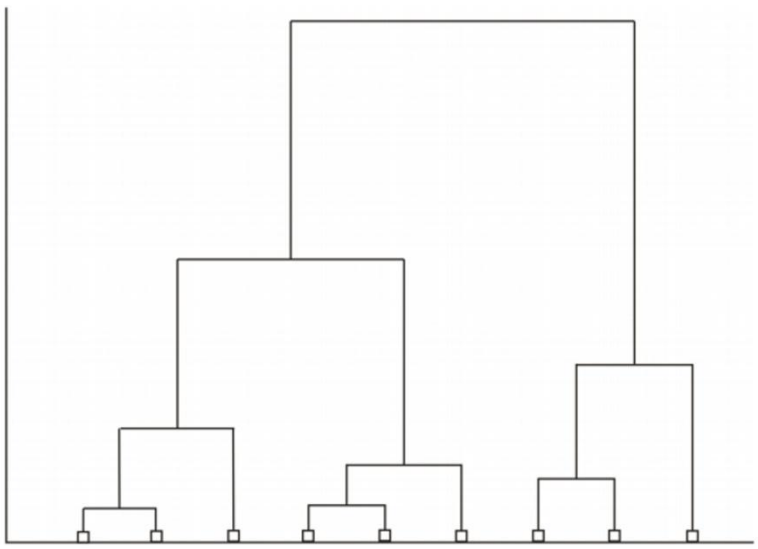
Σχέση 14

όπου Ci, στη σχέση 14 είναι μια κλάση και mi είναι το μέσο σημείο της. Η μέθοδος, για να συνενώσει δύο συστάδες από συνολικό πλήθος k συστάδων, ελέγχει τα δυνατά k(k-1)/2 ζεύγη συστάδων τα οποία μπορούν να δημιουργηθούν και επιλέγει το ζεύγος, το οποίο όταν ενωθεί θα μας δώσει τη συστάδα με το ελάχιστο τετραγωνικό σφάλμα. Η μέθοδος του Ward έχει την τάση να παράγει ισοπληθείς ομάδες.

3.3.3 Δενδρογράμματα

Τα Δενδρογράμματα είναι ένας γραφικός τρόπος αναπαράστασης της διαδικασίας των διαδοχικών συγχωνεύσεων ή διασπάσεων και ο οποίος χρησιμοποιείται για την οπτικοποίηση των αποτελεσμάτων των ιεραρχικών αλγορίθμων. Το Δενδρόγραμμα έχει τη μορφή ανεστραμμένου δένδρου. Στα φύλλα του δένδρου, δηλαδή στο κατώτερο επίπεδο, βρίσκονται τα μεμονωμένα αντικείμενα. Κάθε κόμβος του δένδρου αντιπροσωπεύει μια συστάδα. Επίσης, κάθε κόμβος αποτελεί αφετηρία δύο κλάδων. Στη συσσωρευτική ομαδοποίηση, ένας κόμβος με τους κλάδους και τα τέκνα του συμβολίζει τη συγχώνευση των συστάδων-τέκνων, και τη δημιουργία της συστάδας-γονέα. Στη διαιρετική ομαδοποίηση, ένας κόμβος με τους κλάδους και τα τέκνα του συμβολίζει τη διάσπαση του κόμβου-γονέα, και τη δημιουργία των συστάδων-τέκνων.

Σε όλες τις συσσωρευτικές μεθόδους και ορισμένες διαιρετικές μεθόδους, ο βαθμός ανομοιότητας αυξάνεται μονότονα με το επίπεδο. Ο σχεδιασμός του δένδρου γίνεται με τέτοιον τρόπο, ώστε η διαφορά ύψους των επιπέδων να αποτυπώνει την αύξηση της ανομοιότητας. Ο χρήστης μπορεί να χρησιμοποιήσει το Δενδρόγραμμα για να επιλέξει ένα επίπεδο και να αποφασίσει ένα συγκεκριμένο τρόπο διαμοιρασμού των αντικειμένων σε συστάδες. Ωστόσο, ο χρήστης πρέπει να γνωρίζει ότι διαφορετικές μέθοδοι ιεραρχικής ομαδοποίησης ή και μικρές αλλαγές στα δεδομένα μπορούν να δημιουργήσουν σημαντικά διαφορετικά Δενδρογράμματα.



Διάγραμμα 6

3.4 Μέτρα Εγκυρότητας Αλγορίθμων Συσταδοποίησης

Τα δύο μέτρα εγκυρότητας που θα παρουσιάσουμε ενδείκνυνται για τον έλεγχο εγκυρότητας unsupervised αλγορίθμων συσταδοποίησης, μιας και χρησιμοποιούν μόνο το ίδιο το μοντέλο που παράγεται από τους αλγορίθμους ώστε να κρίνουν το πόσο καλή ομαδοποίηση έχει γίνει και δεν χρειάζεται κάποια πρότερη γνώση για τη συλλογή.

3.4.1 Συντελεστής Silhouette *[18]*

Ο συγκεκριμένος συντελεστής ορίζεται για κάθε δείγμα της συλλογής και αποτελείται από δύο βαθμολογίες:

α: Η μέση απόσταση ενός δείγματος από τα υπόλοιπα της ίδιας συστάδας.

β: Η μέση απόσταση ενός δείγματος από τα δείγματα τις κοντινότερης συστάδας.

Ο συντελεστής s λοιπόν για ένα δείγμα δίνεται από την σχέση 15.

$$s = \frac{\beta - \alpha}{\max(\alpha, \beta)}$$

Σχέση 15

και για να υπολογίσουμε τον s για μια συλλογή δειγμάτων αρκεί να υπολογίσουμε το μέσο όρο των επιμέρους συντελεστών κάθε δείγματος του.

3.4.2 Δείκτης Calinski-Harabaz *[18]*

Ο δείκτης αυτός s είναι ο λόγος της διασποράς των διαφορετικών clusters με την εσωτερική-διασπορά του cluster. Η σχέση 16 τον περιγράφει.

$$s(k) = \frac{Tr(B_k)}{Tr(W_k)} \times \frac{N - k}{k - 1}$$

Σχέση 16

3.5 Μη-αρνητική Παραγοντοποίηση Πίνακα (NMF) *[7]*

Στον αλγόριθμο Μη-αρνητικής Παραγοντοποίησης Πίνακα (Non-negative Matrix Factorization - NMF) ένας πίνακας X μετασχηματίζεται σε γινόμενο δύο άλλων πινάκων W και H, όπως φαίνεται στην σχέση 17.

$$nmf(X, k) = W \times H$$

Σχέση 17

Ο αλγόριθμος παίρνει σαν είσοδο, εκτός από τον πίνακα X, και ένα θετικό αριθμό k. Ο πίνακας W που παράγεται έχει k στήλες, ενώ ο H έχει k γραμμές. Γενικά, η παραγοντοποίηση πινάκων δε γίνεται με ένα μοναδικό τρόπο, συνεπώς έχει αναπτυχθεί ένα πλήθος διαφορετικών μεθόδων (πχ principal component analysis, singular value decomposition) με την κάθε μία να έχει τους δικούς της περιορισμούς. Στο NMF υπάρχει ο περιορισμός ότι ο πίνακας X περιέχει μόνο στοιχεία μεγαλύτερα ή ίσα του μηδενός. Οι πίνακες W, H υπόκεινται στον ίδιο περιορισμό και παράγονται από τον αλγόριθμο έτσι ώστε να μην έχουν αρνητικά στοιχεία.

3.5.1 Ιστορία

Οι πρώτες εργασίες πάνω στην παραγοντοποίηση πίνακα με θετικές τιμές γίνεται από μία ομάδα Φιλανδών ερευνητών στα μέσα της δεκαετίας του '90 και έχει το όνομα "positive matrix factorization". Αργότερα γίνεται ευρέως γνωστό ως "non-negative matrix factorization" με τους Lee και Seung να ερευνούν τις ιδιότητες του και να δημοσιεύουν μερικούς απλούς και χρήσιμους αλγορίθμους για δύο τύπους παραγοντοποίησης.

3.5.2 Υπόβαθρο

Έστω V το γινόμενο των πινάκων W,H τέτοιο ώστε:

$$W \times H = V$$

Σχέση 18

Ο πολλαπλασιασμός των πινάκων μπορεί να γίνει με το γραμμικό συνδυασμό των διανυσμάτων στηλών του W με τις τιμές του H. Έτσι κάθε στήλη του πίνακα V υπολογίζεται με τον την σχέση 19.

$$v_i = \sum_{j=1}^{N} H_{ij} W_J$$

Σχέση 19

όπου:

- N, ο αριθμός στηλών του W
- V_i , το i-στο διάνυσμα στήλης του παραγόμενου πίνακα V
- H_{ij} , η τιμή του πίνακα H στην j γραμμή και i στήλη
- W_J , η j-στη στήλη του πίνακα W

Στον πολλαπλασιασμό πινάκων, οι πίνακες που αποτελούν τους παράγοντες του γινομένου μπορούν να είναι σημαντικά μικρότερου βαθμού (rank) από τον παραγόμενο πίνακα και είναι αυτή η ιδιότητα που αποτελεί τη βάση του NMF. Αν μπορούμε να παραγοντοποιήσουμε ένα πίνακα σε παράγοντες μικρότερου βαθμού του αρχικού τότε τα διανύσματα στήλες του πρώτου παράγοντα είναι διανύσματα γεννήτορες (spanning vectors) στο διανυσματικό χώρο που ορίζεται από τον αρχικό πίνακα.

3.5.3 Εφαρμογή στην εξόρυξη κειμένων

Ας υποθέσουμε ότι ένας πίνακας V που πρόκειται να παραγοντοποιηθεί έχει 10.000 γραμμές και 500 στήλες. Οι γραμμές αντιστοιχούν σε λέξεις και οι στήλες σε κείμενα. Δηλαδή έχουμε καταγράψει σε μία συλλογή 500 κειμένων 10.000 διαφορετικούς όρους και έχουμε κρατήσει τις συχνότητες εμφάνισής τους. Σε ένα τέτοιο πίνακα, το διάνυσμα στήλης αντιστοιχεί σε κείμενο της συλλογής.

Δίνουμε ως είσοδο στον αλγόριθμο NMF τον παραπάνω πίνακα και K=10, δηλαδή του ζητούμε να ανακαλύψει 10 θέματα. Θα παραχθούν οι πίνακες W και H με διαστάσεις (10.000 γραμμές x 10 στήλες) και (10 γραμμές x 500 στήλες) αντίστοιχα. Δεν ξεχνάμε ότι πολλαπλασιάζοντας τους δύο αυτούς πίνακες παίρνουμε πάλι έναν πίνακα με τις ίδιες διαστάσεις του αρχικού V (10.000 γραμμές x 500 στήλες) και αν η παραγοντοποίηση δούλεψε καλά, έναν πίνακα που αποτελεί πολύ καλή προσέγγιση του αρχικού.

Τελικώς, κάθε στήλη του πίνακα που προκύπτει αν πολλαπλασιάσουμε τους W, H είναι γραμμικός συνδυασμός των 10 διανυσμάτων στηλών του πίνακα W με τους συντελεστές από τον πίνακα H.

Αυτό το τελευταίο σημείο είναι η βάση του NMF αφού μπορούμε να θεωρήσουμε ότι κάθε αρχικός πίνακας (σαν αυτόν του παραδείγματος) δημιουργείται από μια μικρή ομάδα από κρυμμένες μεταβλητές. Αυτές οι κρυμμένες μεταβλητές είναι τα θέματα και ο αλγόριθμος NMF τις παράγει ανάλογα με το k που του δίνεται ως είσοδος.

Συνεπώς, σκεφτόμαστε πλέον τον πίνακα W ως τον πίνακα εκείνο τα διανύσματα στηλών του οποίου περιέχουν τα θέματα, με την κάθε τιμή του διανύσματος να δείχνει το βάρος που έχει στο θέμα ο αντίστοιχος όρος. Ανάλογα, μπορούμε να πούμε ότι η κάθε γραμμή αντιστοιχεί σε κάποιο όρο και το διάνυσμα γραμμής δείχνει το βάρος του όρου σε καθένα από τα K θέματα. Επίσης, σκεφτόμαστε τον πίνακα H ως τον πίνακα εκείνο τα διανύσματα στηλών του οποίου περιέχουν τα κείμενα με τις τιμές να δείχνουν το βαθμό στον οποίο ένα κείμενο ανήκει σε κάποιο από τα k θέματα.

3.5.4 Επαναληπτικές μέθοδοι

Υπάρχουν διάφοροι τρόποι με τους οποίους οι πίνακες W και H παράγονται. Οι Lee και Seung πρότειναν μία μέθοδο με πολλαπλασιαστικό κανόνα για την ανανέωση των πινάκων σε κάθε επανάληψη η οποία είναι ιδιαίτερα δημοφιλής εξαιτίας της απλότητας υλοποίησής της. Από τότε έχουν προταθεί και άλλες προσεγγίσεις περισσότερο αποδοτικές.

Οι αλγόριθμοι που είναι διαθέσιμοι αυτή τη στιγμή είναι sub-optimal καθώς εγγυώνται να βρουν ένα τοπικό ελάχιστο της συνάρτησης κόστους πάρα ένα ολικό ελάχιστο. Μάλιστα, η εύρεση και απόδειξη ενός βέλτιστου αλγορίθμου NMF είναι απίθανη στο κοντινό μέλλον καθώς το πρόβλημα έχει δειχθεί ότι είναι γενική μορφή του προβλήματος συσταδοποίησης K-means (K-means clustering problem), το οποίο είναι γνωστό ότι αποτελεί NP-complete πρόβλημα. Παρόλα αυτά, όπως και σε άλλες περιπτώσεις εφαρμογών εξόρυξης γνώσης, ένα τοπικό ελάχιστο μπορεί να φανεί χρήσιμο. Πρόσφατα, οι Sanjeev Arora, Rong Ge, Ravi Kannan and Ankur Moitra (2012) στην εργασία τους με θέμα "Computing a Nonnegative Matrix Factorization Provably" [9] δίνουν έναν αλγόριθμο NMF που τρέχει σε πολυωνυμικό χρόνο αν ένας από τους παράγοντες W ικανοποιεί τη συνθήκη διαχωριστικότητας (separability condition).

ΚΕΦΑΛΑΙΟ 4

ΣΧΕΔΙΑΣΜΟΣ & ΥΛΟΠΟΙΗΣΗ ΣΥΣΤΗΜΑΤΟΣ

4.1 Εισαγωγή

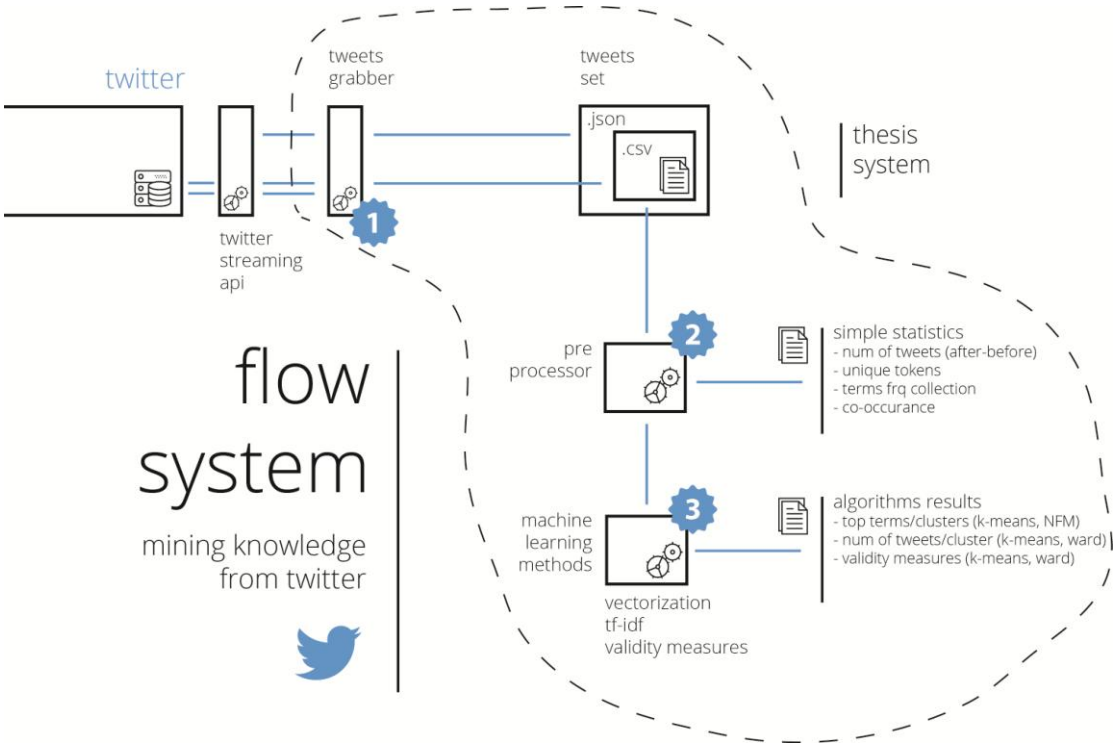
Όπως έχει προαναφερθεί στην εισαγωγή της εργασίας, σκοπός μας είναι:

α. η συλλογή δεδομένων από το κοινωνικό δίκτυο twitter, και ειδικότερα tweets και

β. ο αυτόματος διαχωρισμός αυτών των συλλογών από tweets σε θεματικές ομάδες σύμφωνα με το περιεχόμενο τους.

Για να υλοποιήσουμε τον σκοπό μας αυτό άλλοτε χρησιμοποιήσαμε έτοιμα εργαλεία, και άλλοτε κληθήκαμε να υλοποιήσουμε καινούρια, σύμφωνα με τις απαιτήσεις αυτών που επιδιώκαμε. Σε όλη αυτή την διαδικασία χρησιμοποιήσαμε εξ' ολοκλήρου την προγραμματιστική γλώσσα Python.

Παρακάτω ακολουθεί ένα σχεδιάγραμμα ροής του συστήματος.



Διάγραμμα 7

4.2 Επικοινωνία με το Twitter

Το κοινωνικό δίκτυο twitter προσφέρει ένα αρκετά εξελιγμένο api με αρκετές ανοιχτές δυνατότητες και στον απλό χρήστη-προγραμματιστή. Ειδικότερα παρέχει δύο είδη, το rest api και το streaming api. Από το πρώτο μπορείς να πάρεις περισσότερες πληροφορίες και να κάνεις διάφορες αναζητήσεις στα tweets της τελευταίας εβδομάδας, επίσης μπορείς να διαδράσεις και να χρησιμοποιήσεις σχεδόν όλες τις δυνατότητες ενός twitter profile, όπως να κάνεις re-tweet, να γράψεις καινούριο tweet κ.α.

Για τον σκοπό της εργασίας, όμως, μας φάνηκε καταλληλότερο το streaming api. Μέσω του οποίου μπορείς να μαζεύεις tweets μόνο πραγματικού χρόνου, χωρίς να έχεις κανένα όριο ως προς το χρόνο και το πλήθος των tweets που μπορείς να μαζεύεις όπως στο rest api.

4.2.1 Streaming Api

Η σύνδεση με το streaming api γίνεται μέσο του γνωστού OAuth 2.0 πρωτοκόλλου, το οποίο επιτρέπει την ασφαλή και περιορισμένη επικοινωνία τρίτων εφαρμογών (η εφαρμογή μας) με κάποιον πάροχο - εφαρμογή, στην προκειμένη περίπτωση το twitter.

Όπως προαναφέρθηκε, έπειτα από μια ορθή αυθεντικοποίηση, συνδέεσαι σε κάποιο endpoint που ορίζει το twitter και εκεί δημιουργείται μια μακροχρόνια http σύνδεση, μέσω της οποίας τροφοδοτείσαι με μια συνεχής ροή από tweets, πραγματικού χρόνου, σύμφωνα με τα tracking φίλτρα που όρισες μέσο τις εφαρμογής σου. Κάθε λογαριασμός μπορεί να έχει ανοιχτή μόνο μια τέτοια σύνδεση και αυτή να είναι σταθερή.

4.2.2 Παραμετροποίηση αιτημάτων

Το streamig api δίνει κάποιες συγκεκριμένες παραμέτρους ώστε να ορίσεις στο twitter ακριβέστερα τα tweets που θες να λαμβάνεις. Πιο συγκεκριμένα:

1. delimited
2. stall_warnings
3. filter_level
4. language
5. follow

6. track
7. locations
8. count
9. with
10. replies
11. stringify_friend_id

Δεν θα αναλωθούμε εδώ στην επεξήγηση της κάθε παραμέτρου, αυτό μπορεί καθένας να το βρει στο επίσημο documentation του twitter. Θα αναφέρουμε τις δύο παραμέτρους που χρησιμοποιήσαμε για να μαζεύουμε δεδομένα με το σύστημα μας και θεωρούμε πως είναι σημαντικό να εξηγηθούν περισσότερο ώστε να είναι πιο διακριτά τα χαρακτηριστικά των δεδομένων που συλλέγει το σύστημα.

language

Μέσω αυτής της παραμέτρου μπορείς να δηλώσεις αν θες να λαμβάνεις tweets μόνο κάποιας ή κάποιων συγκεκριμένων γλωσσών ως προς το tweet text. Συγκεκριμένα εμείς ορίσαμε την παράμετρο '*language=en*' γιατί επιλέξαμε να λαμβάνουμε και να ασχοληθούμε έπειτα με text documents-tweets που είναι μόνο στην αγγλική γλώσσα.

track

Με την παράμετρο αυτή ορίζουμε ποιά tweets θέλουμε να παρακολουθούμε βάσει κάποιων λέξεων ή και φράσεων. Σε μια φράση οι λέξεις πρέπει να χωρίζονται μεταξύ τους με κενά. Τα ξεχωριστά keywords πρέπει να χωρίζονται με κόμμα. Παρακάτω ακολουθεί ένα κατατοπιστικός πίνακας με το τι επιστρέφει το twitter βάσει του τρόπου που έχουμε δηλώσει τους επιθυμητούς όρους στο track.

Parameter value	**Will match...**	**Will not match...**
Twitter	**TWITTER** twitter “Twitter” twitter. #twitter @twitter http://twitter.com	**TwitterTracker** #newtwitter

Twitter’s	I like Twitter’s new design	Someday I’d like to visit @Twitter’s office
twitter api,twitter streaming	**The Twitter API is awesome** The twitter streaming service is fast Twitter has a streaming API	I’m new to Twitter
example.com	Someday I will visit example.com	There is no example.com/foobarbaz
example.com/foobarbaz	**example.com/foobarbaz** www.example.com/foobarbaz	example.com
www.example.com/foobarbaz		www.example.com/foobarbaz
example com	**example.com** www.example.com foo.example.com foo.example.com/bar I hope my startup isn’t merely another example of a dot com boom!	

Πίνακας 6

4.2.3 Η ανατομία ενός Tweet

Το κάθε tweet λαμβάνεται σε json μορφή. Σε κάθε json εμπεριέχεται αρκετή πληροφορία γι' αυτό. Ακολουθεί ένα τυχαίο tweet στην json μορφή που συλλέχτηκε

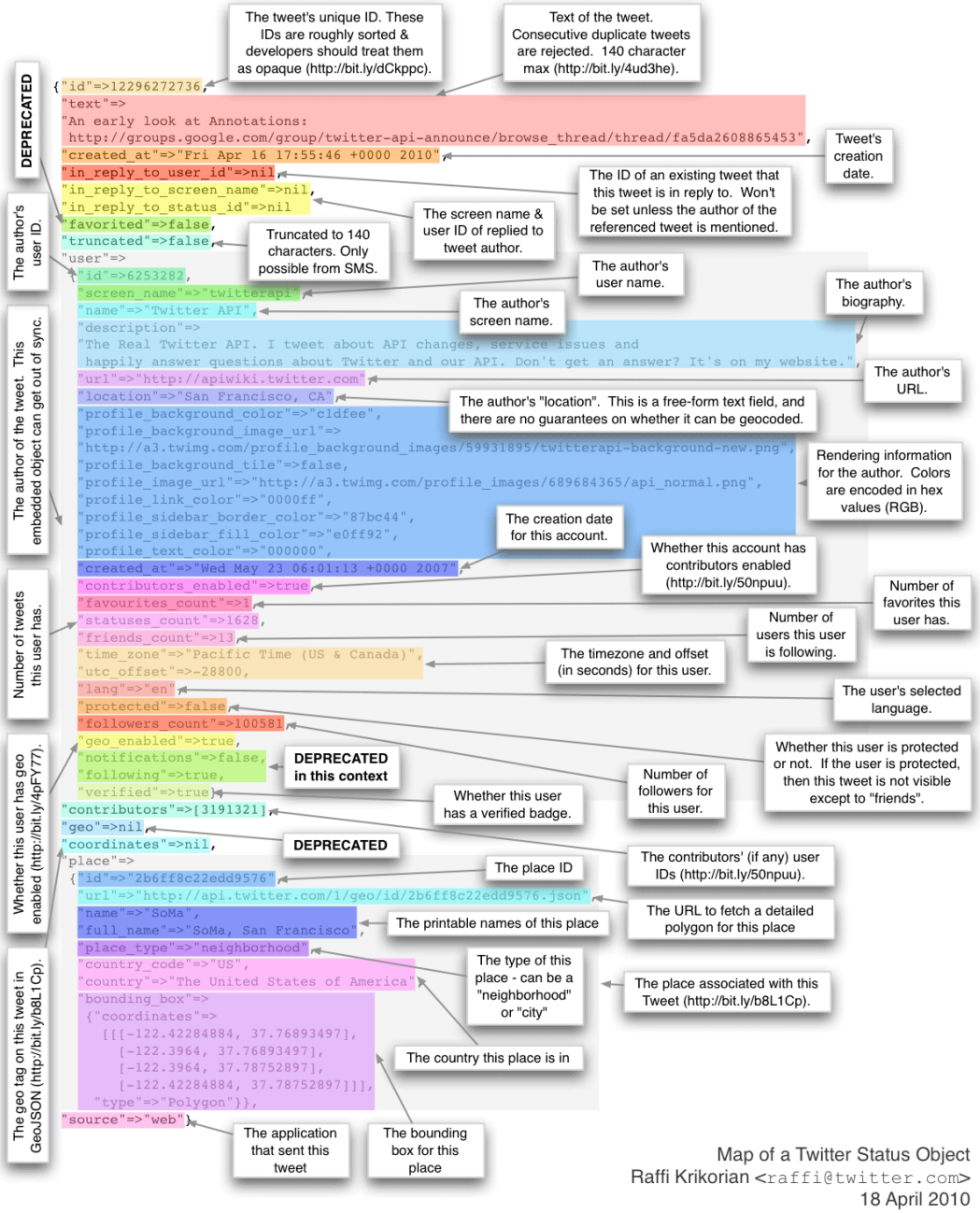

```
{"favorited": false, "contributors": null, "truncated": false, "text": "RT @UEFAEURO:
Nani's strike for Portugal was the 600th goal in EURO tournament history! #EURO2016
https://t.co/Ek7qydJpmT", "possibly_sensitive": false, "is_quote_status": false,
"in_reply_to_status_id": null, "user": {"follow_request_sent": null,
"profile_use_background_image": true, "default_profile_image": false, "id":
1948700964, "verified": false, "profile image url https":
```

```
"https://pbs.twimg.com/profile_images/688480969033003008/40PRH7y9_normal.jpg",
"profile_sidebar_fill_color": "DDEEF6", "profile_text_color": "333333",
"followers_count": 53, "profile_sidebar_border_color": "C0DEED", "id_str":
"1948700964", "profile_background_color": "C0DEED", "listed_count": 0,
"profile_background_image_url_https":
"https://abs.twimg.com/images/themes/theme1/bg.png", "utc_offset": -25200,
"statuses_count": 130, "description": "The Silicon Valley Vipers are a Community
Quidditch team based out of the Bay Area in Northern California. Message us if you
want to try Quidditch!", "friends_count": 140, "location": "San Jose, CA",
"profile_link_color": "0084B4", "profile_image_url":
"http://pbs.twimg.com/profile_images/688480969033003008/40PRH7y9_normal.jpg",
"following": null, "geo_enabled": false, "profile_banner_url":
"https://pbs.twimg.com/profile_banners/1948700964/1465626157",
"profile_background_image_url": "http://abs.twimg.com/images/themes/theme1/bg.png",
"name": "SV Vipers Quidditch", "lang": "en", "profile_background_tile": false,
"favourites_count": 327, "screen_name": "VipersQuidditch", "notifications": null,
"url": null, "created_at": "Wed Oct 09 06:38:51 +0000 2013", "contributors_enabled":
false, "time_zone": "Pacific Time (US & Canada)", "protected": false,
"default_profile": true, "is_translator": false}, "filter_level": "low", "geo": null,
"id": 742813800257060864, "favorite_count": 0, "lang": "en", "retweeted_status":
{"contributors": null, "truncated": false, "text": "Nani's strike for Portugal was the
600th goal in EURO tournament history! #EURO2016 https://t.co/Ek7qydJpmT",
"is_quote_status": false, "in_reply_to_status_id": null, "id": 742803523394473984,
"favorite_count": 785, "source": "<a href=\"http://twitter.com\"
rel=\"nofollow\">Twitter Web Client</a>", "retweeted": false, "coordinates": null,
"entities": {"user_mentions": [], "symbols": [], "hashtags": [{"indices": [74, 83],
"text": "EURO2016"}], "urls": [], "media": [{"expanded_url":
"http://twitter.com/UEFAEURO/status/742803523394473984/photo/1", "display_url":
"pic.twitter.com/Ek7qydJpmT", "url": "https://t.co/Ek7qydJpmT", "media_url_https":
"https://pbs.twimg.com/tweet_video_thumb/Ck731AOWsAAmTtL.jpg", "id_str":
"742803478528045056", "sizes": {"large": {"h": 556, "resize": "fit", "w": 456},
"small": {"h": 415, "resize": "fit", "w": 340}, "medium": {"h": 556, "resize": "fit",
"w": 456}, "thumb": {"h": 150, "resize": "crop", "w": 150}}, "indices": [84, 107],
"type": "photo", "id": 742803478528045056, "media_url":
"http://pbs.twimg.com/tweet_video_thumb/Ck731AOWsAAmTtL.jpg"}]},
"in_reply_to_screen_name": null, "id_str": "742803523394473984", "retweet_count": 828,
"in_reply_to_user_id": null, "favorited": false, "user": {"follow_request_sent": null,
"profile_use_background_image": true, "default_profile_image": false, "id":
1469402426, "verified": true, "profile_image_url_https":
"https://pbs.twimg.com/profile_images/710860107677081601/qwmglOC8_normal.jpg",
"profile_sidebar_fill_color": "DDEEF6", "profile_text_color": "333333",
"followers_count": 816622, "profile_sidebar_border_color": "FFFFFF", "id_str":
"1469402426", "profile_background_color": "C0DEED", "listed_count": 2777,
"profile_background_image_url_https":
"https://pbs.twimg.com/profile_background_images/512520964530130944/iPhsIvFw.jpeg",
"utc_offset": 7200, "statuses_count": 14471, "description": "The official home of
#EURO2016 on Twitter. (@EURO2016: official French language account)", "friends_count":
497, "location": null, "profile_link_color": "0084B4", "profile_image_url":
"http://pbs.twimg.com/profile_images/710860107677081601/qwmglOC8_normal.jpg",
"following": null, "geo_enabled": true, "profile_banner_url":
"https://pbs.twimg.com/profile_banners/1469402426/1458317336",
"profile_background_image_url":
"http://pbs.twimg.com/profile_background_images/512520964530130944/iPhsIvFw.jpeg",
"name": "UEFA EURO 2016", "lang": "en", "profile_background_tile": false,
"favourites_count": 355, "screen_name": "UEFAEURO", "notifications": null, "url":
"http://euro2016.tickets.uefa.com", "created_at": "Thu May 30 10:08:05 +0000 2013",
"contributors_enabled": false, "time_zone": "Amsterdam", "protected": false,
"default_profile": false, "is_translator": false}, "geo": null,
"in_reply_to_user_id_str": null, "possibly_sensitive": false, "lang": "en",
"created_at": "Tue Jun 14 19:38:55 +0000 2016", "filter_level": "low",
"in_reply_to_status_id_str": null, "place": {"full_name": "Paris, France", "url":
"https://api.twitter.com/1.1/geo/id/09f6a7707f18e0b1.json", "country": "France",
"place_type": "city", "bounding_box": {"type": "Polygon", "coordinates": [[[2.224101,
48.815521], [2.224101, 48.902146], [2.469905, 48.902146], [2.469905, 48.815521]]]},
"country_code": "FR", "attributes": {}, "id": "09f6a7707f18e0b1", "name": "Paris"},
"extended_entities": {"media": [{"expanded_url":
"http://twitter.com/UEFAEURO/status/742803523394473984/photo/1", "display_url":
"pic.twitter.com/Ek7qydJpmT", "url": "https://t.co/Ek7qydJpmT", "media_url_https":
"https://pbs.twimg.com/tweet_video_thumb/Ck731AOWsAAmTtL.jpg", "video_info":
{"aspect_ratio": [114, 139], "variants": [{"url":
"https://pbs.twimg.com/tweet_video/Ck731AOWsAAmTtL.mp4", "bitrate": 0, "content_type":
"video/mp4"}]}, "id_str": "742803478528045056", "sizes": {"large": {"h": 556,
"resize": "fit", "w": 456}, "small": {"h": 415, "resize": "fit", "w": 340}, "medium":
{"h": 556, "resize": "fit", "w": 456}, "thumb": {"h": 150, "resize": "crop", "w":
150}}, "indices": [84, 107], "type": "animated_gif", "id": 742803478528045056,
"media_url": "http://pbs.twimg.com/tweet_video_thumb/Ck731AOWsAAmTtL.jpg"}]}},
```

```
"entities": {"user_mentions": [{"id": 1469402426, "indices": [3, 12], "id_str":
"1469402426", "screen_name": "UEFAEURO", "name": "UEFA EURO 2016"}], "symbols": [],
"hashtags": [{"indices": [88, 97], "text": "EURO2016"}], "urls": [], "media":
[{"source_user_id": 1469402426, "source_status_id_str": "742803523394473984",
"expanded_url": "http://twitter.com/UEFAEURO/status/742803523394473984/photo/1",
"display_url": "pic.twitter.com/Ek7qydJpmT", "url": "https://t.co/Ek7qydJpmT",
"media_url_https": "https://pbs.twimg.com/tweet_video_thumb/Ck731AOWsAAmTtL.jpg",
"source_user_id_str": "1469402426", "source_status_id": 742803523394473984, "id_str":
"742803478528045056", "sizes": {"large": {"h": 556, "resize": "fit", "w": 456},
"small": {"h": 415, "resize": "fit", "w": 340}, "medium": {"h": 556, "resize": "fit",
"w": 456}, "thumb": {"h": 150, "resize": "crop", "w": 150}}, "indices": [98, 121],
"type": "photo", "id": 742803478528045056, "media_url":
"http://pbs.twimg.com/tweet_video_thumb/Ck731AOWsAAmTtL.jpg"}]},
"in_reply_to_user_id_str": null, "retweeted": false, "coordinates": null,
"timestamp_ms": "1465935585451", "source": "<a
href=\"http://twitter.com/download/android\" rel=\"nofollow\">Twitter for
Android</a>", "in_reply_to_status_id_str": null, "in_reply_to_screen_name": null,
"id_str": "742813800257060864", "extended_entities": {"media": [{"source_user_id":
1469402426, "source_status_id_str": "742803523394473984", "expanded_url":
"http://twitter.com/UEFAEURO/status/742803523394473984/photo/1", "display_url":
"pic.twitter.com/Ek7qydJpmT", "url": "https://t.co/Ek7qydJpmT", "media_url_https":
"https://pbs.twimg.com/tweet_video_thumb/Ck731AOWsAAmTtL.jpg", "source_user_id_str":
"1469402426", "source_status_id": 742803523394473984, "video_info": {"aspect_ratio":
[114, 139], "variants": [{"url":
"https://pbs.twimg.com/tweet_video/Ck731AOWsAAmTtL.mp4", "bitrate": 0, "content_type":
"video/mp4"}]}, "id_str": "742803478528045056", "sizes": {"large": {"h": 556,
"resize": "fit", "w": 456}, "small": {"h": 415, "resize": "fit", "w": 340}, "medium":
{"h": 556, "resize": "fit", "w": 456}, "thumb": {"h": 150, "resize": "crop", "w":
150}}, "indices": [98, 121], "type": "animated_gif", "id": 742803478528045056,
"media_url": "http://pbs.twimg.com/tweet_video_thumb/Ck731AOWsAAmTtL.jpg"}]}, "place":
null, "retweet_count": 0, "created_at": "Tue Jun 14 20:19:45 +0000 2016",
"in_reply_to_user_id": null}
```


Πίνακας 7

Είναι προφανές πως χρειάζεται μια κάποιου είδους "αποκωδικοποίηση" της παραπάνω πληροφορίας ώστε να μπορέσουμε να συνεχίσουμε. Ευτυχώς για εμάς στο Internet υπάρχουν αρκετές πληροφορίες. Ακολουθεί μια χαρτογράφηση, αν και λίγο παλιά(2010), ενός json tweet του Raffi Krikorian, η οποία όμως είναι αρκετά αντιπροσωπευτική, αρκετά πιο αναλυτικές και σύγχρονες πληροφορίες μπορεί να βρει κάποιος ανατρέχοντας στο documentation που προσφέρει η ίδια η twitter.

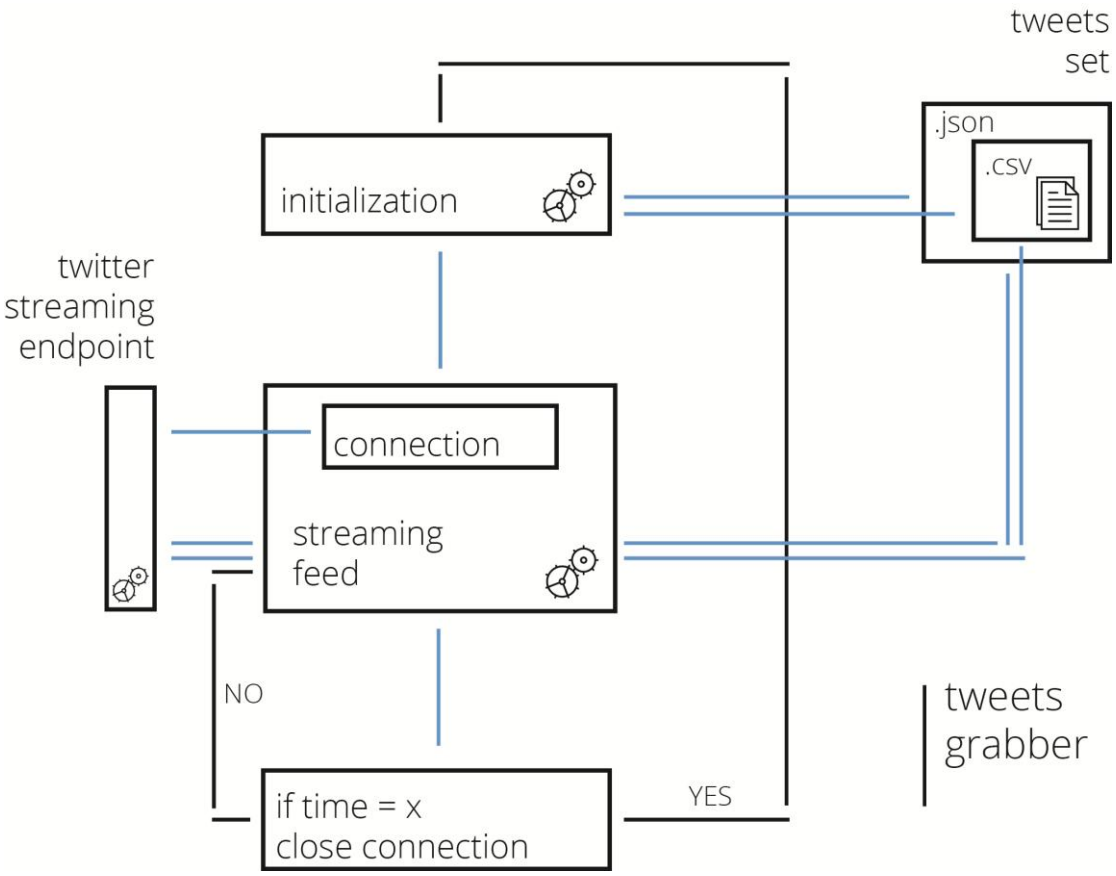


Διάγραμμα 8

4.3 Αναλυτική Περιγραφή Συστήματος

Παρακάτω θα προσπαθήσουμε μια αναλυτική παρουσίαση του συστήματος που υλοποιήσαμε και του οποίου μια γενική εικόνα δόθηκε στο διάγραμμα ροής 4.1.

4.3.1 Tweets Grabber



Διάγραμμα 9

Το πρώτο module του συστήματος ονομάζεται tweets grabber. Ο κεντρικός σκοπός του είναι η συγκομιδή και αποθήκευση ενός συνόλου tweets. Στο διάγραμμα 4.2 φαίνονται σχηματικά οι εργασίες που κάνει, παρακάτω θα δούμε πιο συγκεκριμένα και αναλυτικά τα βήματα που εκτελεί.

Βήματα

Πρώτο. Γίνεται η απαραίτητη αρχικοποίηση. Ορίζονται οι όροι βάση των οποίων θα γίνεται η παρακολούθηση και η συγκομιδή των tweets. Δημιουργούνται ένα αρχείο .csv(ορίζοντας και την πρώτη γραμμή με τους τίτλους κάθε στήλης) και .json(κενό) με τίτλο την τρέχουσα ημερομηνία και τελικό επίθεμα τους όρους για τους οποίους θα γίνεται παρακολούθηση των tweets. Τέλος τραβιούνται από ένα κατάλληλα ενημερωμένο αρχείο τα κλειδιά πρόσβασης που έχουμε προμηθευτεί από το twitter για την επίτευξη επικοινωνίας και συνδιαλλαγής με το API του.

Δεύτερο. Γίνεται αυθεντικοποίηση και σύνδεση με το twitter streaming API. Έπειτα επιστρέφει σε real time τα tweets που παράγονται και εμπεριέχουν κάποιον από τους όρους που ορίσαμε στο **Πρώτο** βήμα. Για την επικοινωνία

με το streaming API και την συγκομιδή των tweets από το endpoint της twitter χρησιμοποιήθηκε η "βιβλιοθήκη" Python Twitter Tools (PTT).

Τρίτο. Μέσα σε μια δομή επανάληψης γράφουμε το κάθε tweet στην json μορφή που δίνεται από την twitter στο αντίστοιχο .json αρχείο που δημιουργήθηκε στο **Πρώτο** βήμα. Έπειτα κρατούμε μόνο τις πληροφορίες χρήστης, κείμενο(στο οποίο γίνεται μετατροπή σε UTF-8 encoding ώστε να μην υπάρχει πρόβλημα με διάφορους περίεργους χαρακτήρες), τοποθεσία(αν έχει οριστεί από το χρήστη, όχι συντεταγμένες μέσο gps) και την ημ/νια δημιουργίας του tweet και τις γράφουμε στο αντίστοιχο .csv. Επίσης οι τελευταίες πληροφορίες τυπώνονται και στο terminal με χρωματισμένα μάλιστα τα σημαντικά στοιχεία.

Τέταρτο. Ελέγχεται ο χρόνος. Όταν ο χρόνος είναι αυτός που έχουμε ορίσει γίνεται τερματισμός του scipt και δημιουργείται ένα καινούριο στιγμιότυπο αυτού το οποίο ξεκινά από το **Πρώτο** βήμα και πάλι και έτσι δημιουργεί νέα αρχεία με νέα ονομασία.

Το πρόβλημα του μεγέθους των αρχείων

Όπως αναφέρθηκε παραπάνω το σύστημα αποθηκεύει συγχρόνως τα tweets σε δύο διαφορετικές μορφές. Δημιουργεί ένα αρχείο .json όπου μεταφέρει τα tweets στην json μορφή που τα παραλαμβάνει από το endpoint του streaming api. Επίσης δημιουργεί παράλληλα ένα αρχείο .csv στο οποίο αποθηκεύει μόνο την πληροφορία που είναι χρήσιμη στην παρούσα εργασία, αναλυτικότερα, το χρήστη που έκανε το tweet, το κείμενο του tweet, την τοποθεσία και την ημ/νια που δημιουργήθηκε.

Αυτή η τεχνική έχει ως αποτέλεσμα να κρατάμε όλη την πληροφορία για πιθανές μελλοντικές χρήσης σε json, αλλά και να αποθηκεύουμε την απαραίτητη και μονό πληροφορία στα αρχεία .csv μειώνοντας έτσι σημαντικά το μέγεθος του αρχείου το οποίο θα χρησιμοποιήσει μετά η εφαρμογή μας. Ενδεικτικά παραθέτονται τα αρχεία που δημιουργήθηκαν από το tweets grabber script που δημιουργήσαμε, την ημέρα του τελικού του euro 2016, με τις τιμές του μεγέθους τους δίπλα.

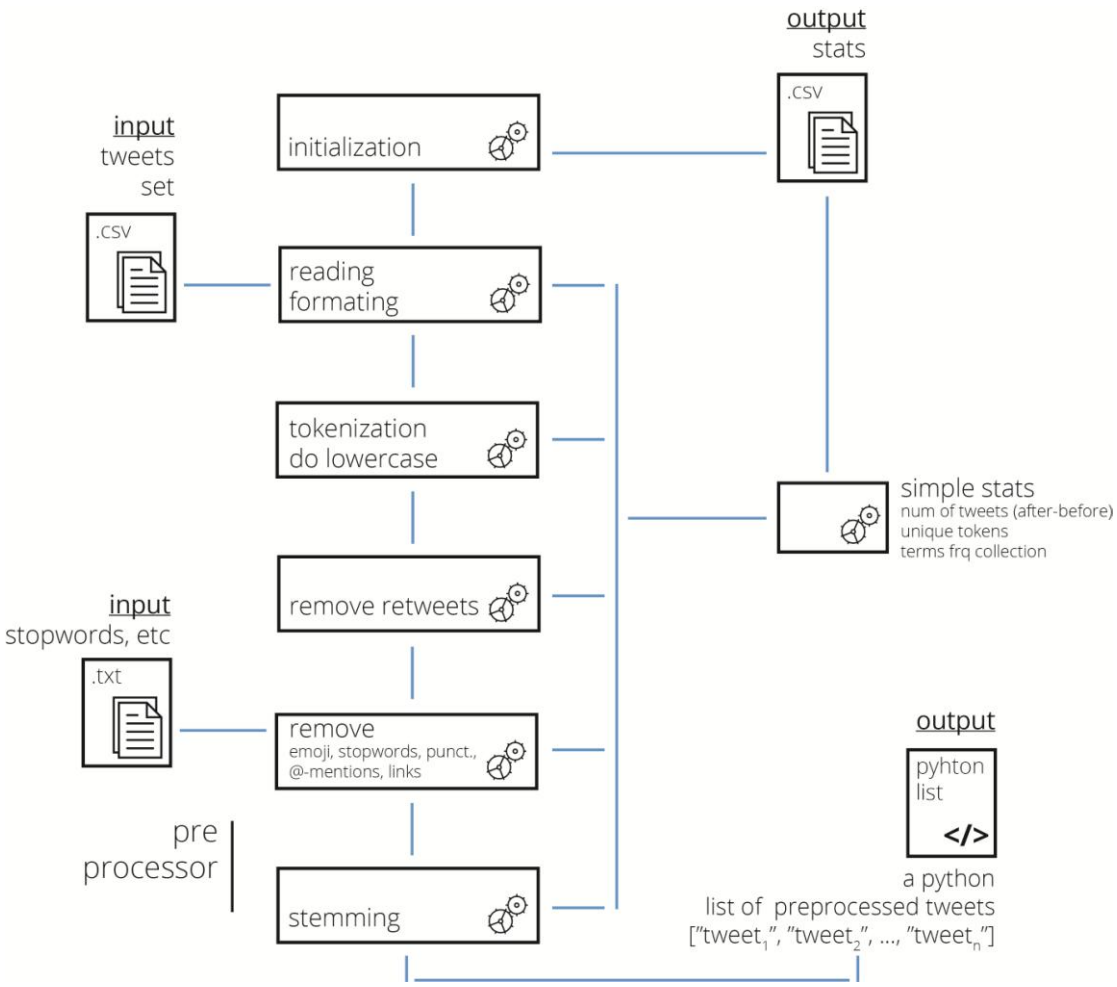
10072016-040101_euro2016-euro.csv, μέγεθος: 186,5Mb

10072016-040101_euro2016-euro.json, μέγεθος: 6,5Gb

Όπως γίνεται εμφανές από τους παραπάνω τίτλους των αρχείων ενσωματώσαμε άλλη μια λειτουργία στον tweets grabber ώστε τα αρχεία να γίνουν ποιο διαχειρίσιμα ως προς το μέγεθος τους. Ειδικότερα, ορίστηκε κάθε βράδυ να κλείνει η συλλογή των tweets και έπειτα να ξεκινάει μια καινούρια συλλογή για την άλλη

μέρα σε δύο καινούρια αρχεία .json, .csv. Ο χρόνος που θα γίνεται αυτό μπορεί να οριστεί από εμάς σύμφωνα με τον ρυθμό παραγωγής των tweets που μαζεύουμε.

4.3.2 Pre-Processor



Διάγραμμα 10

Το δεύτερο module του συστήματος ονομάζεται preprocessor. Ο κεντρικός σκοπός του είναι επεξεργαστεί κατάλληλα τα tweets ώστε να αφαιρέσει οποιαδήποτε περιττή πληροφορία και θόρυβο και να ετοιμάσει κατάλληλα την συλλογή ώστε να χρησιμοποιηθούν στο επόμενο module ως είσοδος. Ακόμα δημιουργεί κάποια απλά στατιστικά που θα μας βοηθήσουν στην καλύτερη παραμετροποίηση των clustering αλγορίθμων στο επόμενο module. Στο διάγραμμα 4.3 φαίνονται σχηματικά οι εργασίες που κάνει αλλά πάμε να δούμε πιο συγκεκριμένα και αναλυτικά τα βήματα που εκτελεί.

Βήματα

Πρώτο. Γίνεται η απαραίτητη αρχικοποίηση. Δημιουργούνται δύο αρχεία .csv με τίτλο την τρέχουσα ημερομηνία στα οποία θα αποθηκευτούν κάποια απλά στατιστικά στην συνέχεια. Τέλος ορίζονται οι κανονικές εκφράσεις οι οποίες θα χρησιμοποιηθούν για την ομαλή μορφοποίηση του κειμένου των tweets.

Δεύτερο. Ανοίγεται τα αρχείο με την συλλογή των tweets. Διαβάζεται κάθε tweet και μορφοποιείται το καθένα με την χρήση των κανονικών εκφράσεων που έχουν οριστεί στο **Πρώτο** βήμα ώστε να είναι σε μορφή αναγνώσιμη αλλά και να γίνει σωστή επεξεργασία στα επόμενα βήματα.

Τρίτο. Κάθε tweet μετατρέπεται σε μια λίστα από τους επιμέρους όρους (λέξεις, emojis, links κλπ.) οι οποίο διαχωρίζονται αν υπάρχει κενό μεταξύ τους. Για παράδειγμα ένα tweet με κείμενο "Hello World" θα γίνει μια λίστα με δύο όρους ["Hello","World"] αλλά αν το κείμενο στο tweet ήταν "HelloWorld" το tweet θα γινόταν μια λίστα με ένα όρο ["HelloWorld"]. Έπειτα όλοι οι όροι μετατρέπονται σε πεζά.

Τέταρτο. Σε αυτό το βήμα εντοπίζονται τα retweets και τελικά αφαιρούνται τα (n-1) tweets με το ίδιο περιεχόμενο, η στρατηγική που ακολουθήσαμε περιγράφεται στο τέλος του 4.3.2 υποκεφαλαίου.

Πέμπτο. Αφαιρούνται από κάθε tweet της συλλογής οι τετριμμένες λέξεις, τα emoticons, τα σημεία στίξης. Τα urls έχουν αφαιρεθεί στο παραπάνω βήμα.

Έκτο. Εφαρμόζεται stemming. Δηλαδή σε κάθε tweet οι όροι που το αποτελούν αντικαθιστούνται από τον αντίστοιχο όρο ρίζα τους. Η τεχνική παρουσιάζεται αναλυτικότερα στο κεφάλαιο 2.

Με το πέρας και του **Έκτου** βήματος έχουμε έτοιμη την συλλογή με τα tweets ώστε να την χρησιμοποιήσουμε στο τρίτο και τελευταίο module του συστήματος ως είσοδο.

Έβδομο. Τέλος όμως έχουμε ένα επιπλέον βήμα το οποίο παράγει κάποια απλά στατιστικά παίρνοντας διάφορες πληροφορίες απ' όλα τα προηγούμενα βήματα. Ενημερώνει το ένα από τα δύο αρχεία του **Πρώτου** βήματος με έναν πίνακα με όλους τους όρους της συλλογής όπου δίπλα τους δίνονται οι εξής πληροφορίες, αριθμός εμφάνισης κάθε όρου και συχνότητα εμφάνισης στην συλλογή. Αυτός ο πίνακας είναι χρήσιμος για να ορίσουμε στο επόμενο module άνω-κάτω όρια για το ποιους όρους θα αγνοήσουμε στην δημιουργία του πίνακα tf-idf. Το άλλο αρχείο το ενημερώνει με το πλήθος των tweets που λάβαμε από το twitter, και με το πως αυτό αλλάζει στα διάφορα στάδια επεξεργασίας για αφαίρεση των retweets, τον αριθμό των μοναδικών όρων πριν και μετά την επεξεργασία. Τέλος εδώ παράγεται και προστίθεται στο δεύτερο αρχείο ένας πίνακας co-occurance το οποίο απλά γίνεται χάριν συμπλήρωσης των στατιστικών και όχι γιατί έχει να προσθέσει κάποια απαραίτητη πληροφορία για την ροή της εργασίας μας.

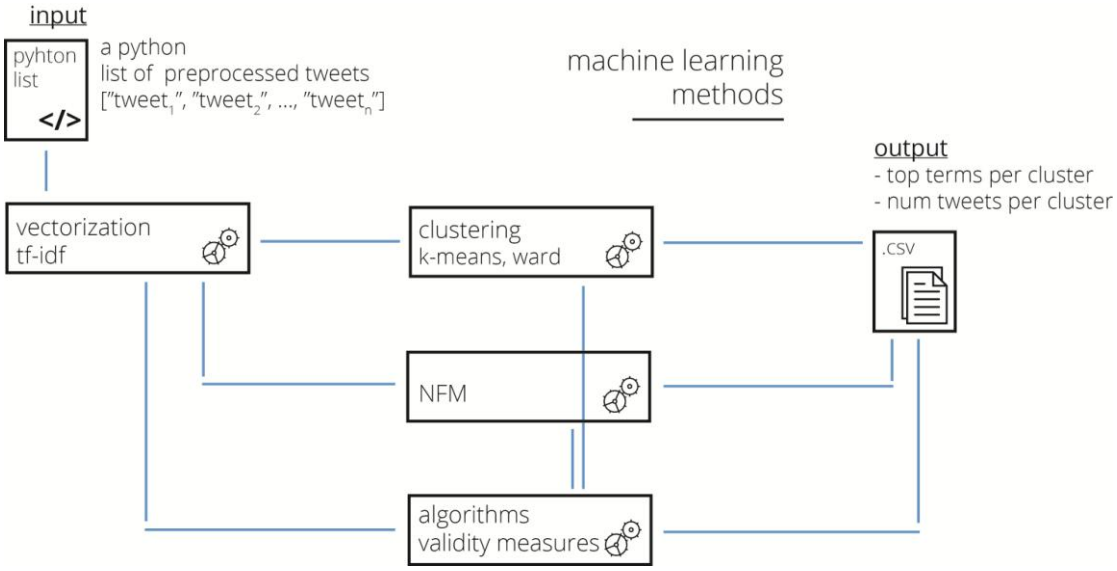
Το πρόβλημα με τα retweets

Κάτι που από την μια μας δημιούργησε πρόβλημα μα από την άλλη ήρθε να βοηθήσει στο πρόβλημα του 4.3 ήταν το περίεργο είδος των δεδομένων των συλλογών που δημιουργούσαμε. Πολλά από τα tweets που παίρναμε ήταν retweets, δηλαδή δεδομένα με το ίδιο περιεχόμενο, έτσι έπρεπε να απαλλαγούμε από αυτά. Το api του twitter δεν δίνει καμία επίσημη λύση για αναγνώριση των retweets, πέραν την ανεπίσημης σύστασης πως πολλά retweets φέρουν το διακριτικό 'RT' στην αρχή του κειμένου, πράγμα όμως που γίνεται κατά το δοκούν από τους χρήστες που κάνουν retweet.

Για την αντιμετώπιση του προβλήματος

1. κρατάγαμε μόνο ένα από τα tweets που έμοιαζαν να είναι ακριβώς ίδια ως προς το text τους,
2. έπειτα αφαιρούσαμε από την λίστα όσα είχαν στην αρχή τους το διακριτικό "RT" και
3. παρατηρήσαμε πως έμεναν κάποια retweet με ίδιο κείμενο αλλά με διαφορετικά links έτσι αποφασίσαμε να αφαιρέσουμε όλα τα links και να ελέγξουμε και πάλι ποια tweets έχουν ίδιο text και να κρατήσουμε μόνο ένα κάθε φορά.

4.3.3 Machine Learning Methods



Διάγραμμα 11

Το τρίτο module του συστήματος ονομάζεται machine learnging methods. Ο κεντρικός σκοπός του είναι να χρησιμοποιήσει unsupervised αλγορίθμους, δηλαδή αλγορίθμους που δεν χρειάζονται επίβλεψη, έτσι ώστε να ομαδοποίηση αυτόματα τα tweets της συλλογής σε ομάδες βάση του περιεχομένου τους. Στο διάγραμμα 4.4 φαίνονται σχηματικά οι εργασίες που κάνει αλλά πάμε να δούμε πιο συγκεκριμένα και αναλυτικά τα βήματα που εκτελεί.

Πριν ξεκινήσουμε να επισημάνουμε πως για την υλοποίηση όλων το παρακάτω βημάτων χρησιμοποιήθηκε η γνωστή βιβλιοθήκη SciKit Learn της python η οποία περιέχει πολλά εργαλεία για data mining και data analysis.

Βήματα

Πρώτο. Λαμβάνει ως είσοδο τη λίστα με τα επεξεργασμένα tweets από το module pre-processor και δημιουργεί ένα διανυσματικό χώρο όπου τα παραπάνω tweets που έλαβε και οι όροι από τους οποίους αποτελούνται αναπαρίστανται πλέον με διανύσματα. Ειδικότερα δημιουργεί έναν tf-idf πίνακα, ο οποίος αναλύσαμε πως δημιουργείται και πως ερμηνεύεται στο υποκεφάλαιο 2.4 όπως και για την αναπαράσταση διανυσματικού χώρου. Μάλιστα σε αυτό το βήμα επιλέγουμε ποιοι όροι θα αγνοηθούν στην δημιουργία του πίνακα. Οι επιλογές που πρέπει να γίνουν είναι τρεις: οι όροι που εμφανίζονται με πάνω από ένα συγκεκριμένο ποσοστό στη συλλογή, οι όροι που εμφανίζονται λιγότερο από έναν συγκεκριμένο αριθμό στη συλλογή και τέλος οι n-grams, για $n \leq 3$, όροι να αφαιρεθούν.

Δεύτερο. Γίνεται εφαρμογή των k-means, ward αλγορίθμων, με είσοδο τον tf-idf πίνακα(ανατρέξτε στο κεφάλαιο 3, για το πως αυτός λειτουργεί). Πριν την εφαρμογή ορίζονται ο αριθμός k των συστάδων που επιθυμούμε και ο μέγιστος αριθμός επαναλήψεων για τον kmeans. Παράγονται οι συστάδες με ομαδοποιημένα τα tweets και γράφονται σε ένα .csv αρχείο οι κορυφαίοι όροι ανά συστάδα, οι οποίοι λαμβάνουν και τον ρόλο ετικέτας της κάθε συστάδας και ο αριθμός των tweets ανά συστάδα.

Τρίτο. Γίνεται εφαρμογή του NMF αλγορίθμου, με είσοδο τον tf-idf πίνακα(ανατρέξτε στο κεφάλαιο 3, για το πως αυτός λειτουργεί). Παράγεται ο αριθμός των ενοτήτων που του έχουμε ζητήσει να "ανακαλύψει" και γράφονται σε ένα .csv αρχείο οι κορυφαίοι όροι ανά θέμα, οι οποίοι λαμβάνουν και τον ρόλο ετικέτας της κάθε θεματικής.

Τέταρτο. Τέλος γίνεται έλεγχος εγκυρότητας των συστάδων που δημιουργήθηκαν από τους δύο clustering αλγορίθμους και γράφονται επίσης στο .csv αρχείο. Ειδικότερα γίνεται χρήση του Silhouette συντελεστή, και του δείκτη Calinski-Harabaz, οι οποίοι είναι μέτρα εγκυρότητας που ενδείκνυνται για unsupervised clustering αλγορίθμους. Το πως αυτοί λειτουργούν αναφέρεται αναλυτικά στο κεφάλαιο 3.

ΚΕΦΑΛΑΙΟ 5

ΠΕΙΡΑΜΑΤΑ & ΑΠΟΤΕΛΕΣΜΑΤΑ

5.1 Εισαγωγή

Στο παρόν κεφάλαιο θα προσπαθήσουμε να παρουσιάσουμε τα αποτελέσματα από μερικά πειράματα που επιχειρήσαμε χρησιμοποιώντας το σύστημα που σας περιγράψαμε αναλυτικά στο κεφάλαιο 4.

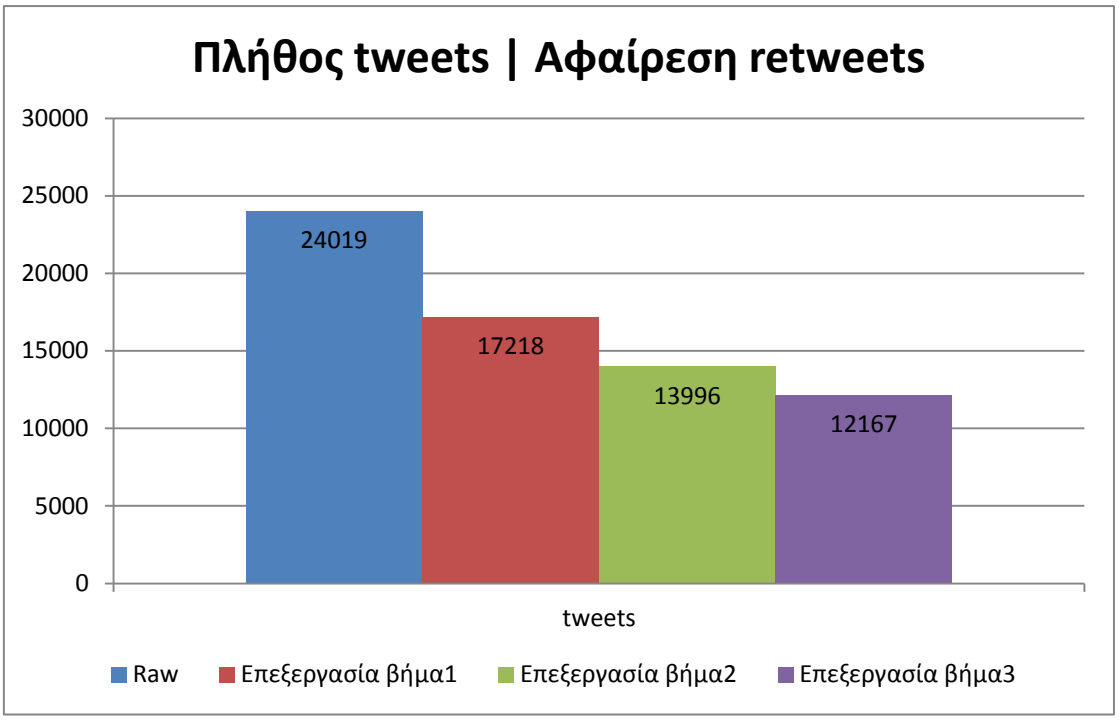
5.2 Συλλογές

Με τη δημιουργία του tweets grabber μας ήταν πολύ εύκολο να συλλέγουμε διάφορα tweets, παρακολουθώντας ποικίλους όρους, όμως για την οικονομία της παρούσας εργασίας αποφασίσαμε να σας παρουσιάσουμε κάποια πειράματα και αποτελέσματα πάνω σε μια συγκεκριμένη συλλογή. Ενδεικτικά κάποιοι όροι για τους οποίους μαζέψαμε συλλογές tweets παρουσιάζονται στο παράρτημα και οι συλλογές είναι διαθέσιμες στην ηλεκτρονική μορφή τους στα αρχεία .json, .csv που παράγει ο tweets grabber.

Πιο συγκεκριμένα τα tweets συλλέχθηκαν στο χρονικό διάστημα 28/10/2016 13:21:20 έως 29/10/2016 04:00:00. Τα αρχεία που δημιουργήθηκαν ήταν:
28102016-132120_gr.csv, μέγεθος: 4.7Mb
28102016-132120_gr.json, μέγεθος: 139.2Mb
Οι όροι βάση των οποίον παρακολουθούσαμε το stream του twitter ήταν οι εξής:
greece, greek, gr.

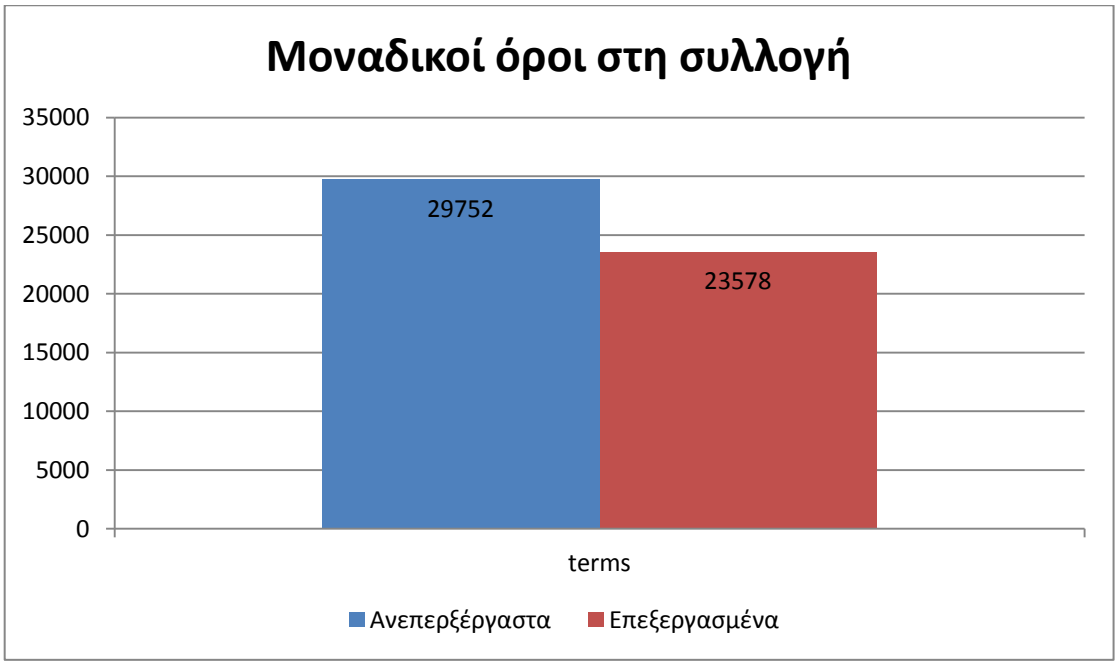
5.3 Αποτελέσματα Pre-Processor

Ακολουθούν τα αποτελέσματα από την προεπεξεργασία της συλλογής. Εκτός από την Python λίστα με τα κατάλληλα επεξεργασμένα tweets που δημιούργησε το module preprocessor για να τροφοδοτήσει στη συνέχεια τους αλγορίθμους machine learning δημιούργησε και δύο αρχεία με διάφορα βοηθητικά στατιστικά στοιχεία όπως είχε αναφερθεί στο 4.3.2 κεφάλαιο, που περιγράφουν τη συλλογή.



Διάγραμμα 12

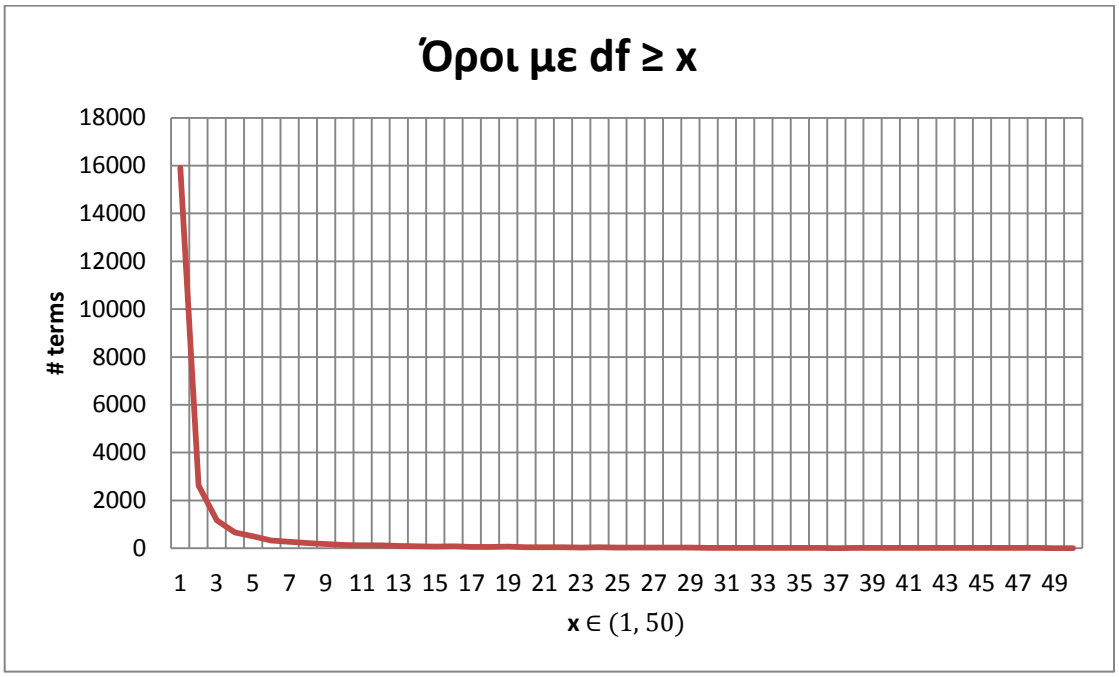
Στο διάγραμμα 12 βλέπουμε τη μείωση των tweets ανά βήμα, με την αφαίρεση των retweets, εφαρμόζοντας τα βήματα επεξεργασίας όπως αυτά παρουσιάστηκαν στο τέλος του κεφαλαίου 4.3.2.



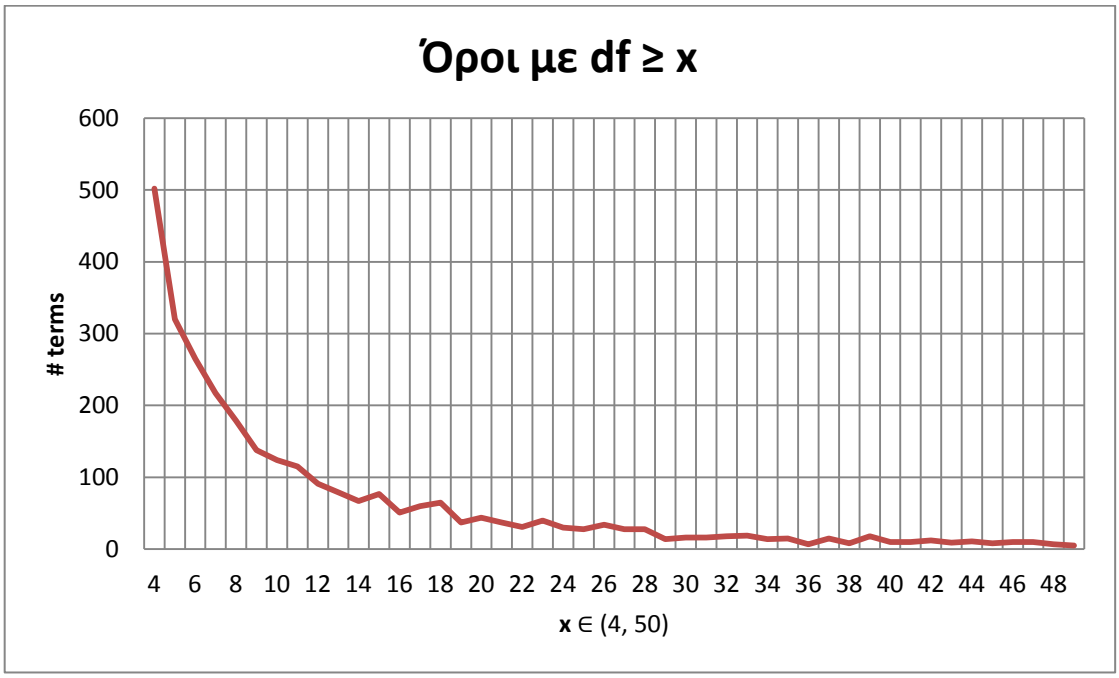
Διάγραμμα 13

Στο διάγραμμα 13 μας δίνεται μια εικόνα με τους μοναδικούς όρους που υπάρχουν στη συλλογή πριν την επεξεργασία και μετά, ειδικότερα μετά από την αφαίρεση των emoticons, των σημείων στίξεως, των συμβόλων, των τετριμμένων λέξεων και των

υπολοίπων που αναφέρθηκαν στο σχετικό υποκεφάλαιο προεπεξεργασία του κεφαλαίου 2.3 και που στο κεφαίλαιο 4.3.2 φαίνεται πως εφαρμόζονται.



Διάγραμμα 14



Διάγραμμα 15

Στα διαγράμματα 14, 15 εμφανίζονται οι αριθμοί των όρων που έχουν document frequency από 1 και 4 (αντίστοιχα) έως 50. Δηλαδή λαμβάνουμε μια σχηματική εικόνα του πλήθους των λέξεων που βρίσκονται σε 1 και περισσότερα κείμενα, 2 και περισσότερα κείμενα κ.ο.κ. Στα πειράματά μας παρακάτω χρησιμοποιήσαμε $DF \geq 2$ όπου ο αριθμός των όρων ήταν 13975 και $DF \geq 4$ όπου ο αριθμός των

όρων ήταν 5453. Υπενθυμίζουμε πως ο αριθμός των μοναδικών όρων μετά την επεξεργασία ήταν 23578. Τα παραπάνω στοιχεία βοήθησαν στον προσδιορισμό των απαραίτητων ορισμάτων, στο τρίτο module machine learning methods, για την δημιουργία του VSM και του tf-idf πίνακα, με τον οποία έπειτα τροφοδοτούμε τους αλγορίθμους συσταδοποίησης.

Ακολουθεί ενδεικτικά ένα μικρό μέρος του πίνακα, από το ένα .csv αρχείο που παράγει ο preprocessor, με τους πρώτους 60 όρους σε εμφάνιση στη συλλογή.

α/α	word	occurrences	frequency	α/α	word	occurrences	frequency
1	greek	4380	36.00	31	free	237	1.95
2	greec	3001	24.67	32	ancient	232	1.91
3	gr	1987	16.33	33	come	229	1.88
4	amp	670	5.51	34	great	228	1.87
5	big	667	5.48	35	god	225	1.85
6	read	666	5.47	36	fuck	224	1.84
7	stock	565	4.64	37	nake	224	1.84
8	girl	564	4.64	38	28	217	1.78
9	profit	542	4.45	39	th	210	1.73
10	weed	541	4.45	40	time	209	1.72
11	microcap	539	4.43	41	play	204	1.68
12	now	467	3.84	42	can	196	1.61
13	day	422	3.47	43	peopl	184	1.51
14	sex	419	3.44	44	island	180	1.48
15	via	385	3.16	45	look	177	1.45
16	s	379	3.11	46	best	176	1.45
17	like	374	3.07	47	yogurt	176	1.45
18	love	347	2.85	48	make	176	1.45
19	new	332	2.73	49	women	176	1.45
20	athen	327	2.69	50	us	173	1.42
21	today	303	2.49	51	see	170	1.40
22	just	296	2.43	52	oxi	168	1.38
23	porn	292	2.40	53	say	168	1.38
24	video	288	2.37	54	know	168	1.38
25	get	276	2.27	55	thank	167	1.37
26	will	254	2.09	56	sexi	165	1.36
27	nude	252	2.07	57	travel	164	1.35
28	im	245	2.01	58	life	163	1.34
29	go	244	2.01	59	good	159	1.31
30	one	242	1.99	60	turkey	156	1.28

Πίνακας 8

Ο πίνακας 5.5 αντίστοιχα με τα γραφήματα 5.3, 5.4 τα οποία και αυτά δημιουργήθηκαν από αυτόν και τις υπόλοιπες εγγραφές του(ολόκληρος όπως παράγεται αυτόματα από το preprocessor) μας βοηθάει στο να ορίσουμε έπειτα ποιοι όροι βάσει της μεγάλης συχνότητας εμφάνισης τους θέλουμε να αγνοηθούν επίσης.

5.4 Αποτελέσματα Machine Learning Methods

Ακολουθούν τα αποτελέσματα από την εφαρμογή των αλγορίθμων που παρουσιάστηκαν στο κεφάλαιο 3 πάνω στα επεξεργασμένα δεδομένα της συλλογής.

5.4.1 Αποτελέσματα K-Means

Παράμετροι πειράματος Α: $2 \leq df \leq 2\%$ έδωσε ένα tf-idf (12167, 13975), 6 θέματα, 10 επαναλήψεις με διαφορετικά κέντρα.

A	Clusters					
Clusters	0	1	2	3	4	5
top 10 terms per cluster	god	nake	help	flower	time	big weed
	great	fuck	debt	flower anytim	oxi	stock profit
	ancient	greek girl	greec debt	anytim	celebr	weed stock
	play	sexi	greek help	flower special	28	microcap read
	best	teen	problem	flower impact	octob	paa
	island	women	esm	flower expert	nation	sil
	good	amateur	visit	expert	happi	read stock
	yogurt	nake greek	student	impact	1940	ome
	life	free	decad	power flower	oxiday	ew
	travel	girl sex	obama	special	greek oxi	brk
tweets	10392	856	124	64	542	189
Silhouette Coefficient	0,011234322					
Calinski-Harabaz Index	26,5735733973					

Πίνακας 9

Παρατηρείται μια μάλλον ορθή ομαδοποίηση από τους κορυφαίους όρους που βλέπουμε όμως από τα δύο μέτρα εγκυρότητας πληροφορούμαστε πως έχουμε μια μέτρια ομαδοποίηση. Ο Silhouette συντελεστής μπορεί να πάρει τιμές γύρω από το διάστημα $(-1,1)$ και γνωρίζουμε πως στην περίπτωση που η τιμή είναι κοντά στο μηδέν, όπως εδώ, τότε ο συντελεστής υποδεικνύει πως υπάρχουν επικαλυπτόμενες ομάδες. Ο Calinski-Harabaz δείκτης θα μας δώσει πληροφορία συγκρινόμενος με τα επόμενα πειράματα, ποιο ήταν το καλύτερο. Περισσότερες πληροφορίες ως προς την ερμηνεία των κριτηρίων εγκυρότητας υπάρχουν στο κεφάλαιο 3.4.

Παράμετροι πειράματος Β: $4 \leq df \leq 3\%$ έδωσε ένα tf-idf (12167, 5030), 6 θέματα, 10 επαναλήψεις με διαφορετικά κέντρα.

B	Clusters					
Clusters	0	1	2	3	4	5
top 10 terms per cluster	oxi	love	ancient	80s	profit microcap	video
	28	god	ancient greek	90s	big weed	amateur
	celebr	athen	coin	greatest	stock profit	porn
	today	great	ancient greec	hit	weed stock	greek video
	octob	porn	greec ancient	radio	microcap read	nake
	1940	nude	greek ancient	play	microcap big	amateur video
	nation	fuck	bc	80s	son	youtub
	oxiday	time	word	play	vip	video greek
	happi	10	histori	greatesT	big read	video porn
	greek oxi	yogurt	god	90s	key	greek amateur
tweets	420	10755	226	108	337	321
Silhouette Coefficient			0,020006272			
Calinski-Harabaz Index			55,3103846078			

Πίνακας 10

Και σε αυτό το πείραμα παρατηρείται μια μάλλον ορθή ομαδοποίηση από τους κορυφαίους όρους που βλέπουμε όμως επίσης από τα δύο μέτρα εγκυρότητας πληροφορούμαστε πως έχουμε μια γενικά μέτρια ομαδοποίηση. Σε σύγκριση όμως με τα αποτελέσματα του πειράματος Α στον πίνακα 10 έχουμε μια καλύτερη ομαδοποίηση, μιας και οι δύο δείκτες εγκυρότητας έχουν διπλασιαστεί.

Παράμετροι πειράματος Γ: $4 \leq df \leq 2\%$ έδωσε ένα tf-idf (12167, 5022), 6 θέματα, 100 επαναλήψεις με διαφορετικά κέντρα.

Γ	Clusters					
Clusters	0	1	2	3	4	5
top 10 terms per cluster	play	god	nake	stock profit	great	big weed
	radio	ancient	fuck	microcap read	job	stock profit
	90s	time	greek girl	read microcap	great greek	weed stock
	rock	best	sexi	weed stock	great gr	microcap read
	80s	yogurt	teen	weed big	celebr great	bio
	greatest	island	women	bbw	indoor	rick
	hit	good	nake	por	celebr	cur
	play	peopl	greek	profit weed	time	tw
	80s	life	hot	lo	place	exp
	webradio	live	girl	ne	design	tis
tweets	211	10737	735	71	227	186
Silhouette Coefficient			0,022707919			
Calinski-Harabaz Index			54,4405064366			

Πίνακας 11

Το μόνο που άλλαξε στο πείραμα Γ από το Β είναι πως στο τελευταίο ρυθμίσαμε ο K-Means να δουλέψει για 100 επαναλήψεις και κατεβάσαμε το πάνω όριο στο 2%.

Το δεύτερο είχε ως αποτέλεσμα να κοπούν επιπλέον μόνο 8 λέξεις πράγμα αναμενόμενο, ανατρέχοντας στο πίνακα 9. Η πρώτη αλλαγή, μιας και φαινομενικά πιο έντονη μοιάζει να μην είχε καμία επίδραση στα μέτρα εγκυρότητας της ομαδοποίησης, όμως από την άλλη διαβάζοντας του κορυφαίους όρους συνειδητοποιούμε πως έχει "εξαφανιστεί" η συστάδα με κορυφαίους όρους για το 1940, και στην θέση της έχει εμφανιστεί μια καινούρια, η συστάδα νούμερο 4 με κάπως πιο γενική και απροσδιόριστη θεματολογία.

Παράμετροι πειράματος Δ: $2 \leq df \leq 80\%$ έδωσε ένα tf-idf (12167, 13997), 6 θέματα, 10 επαναλήψεις με διαφορετικά κέντρα.

Δ	Clusters					
Clusters	0	1	2	3	4	5
top 10 terms per cluster	greek	greek	gr	webradio	greec	microcap
	god	girl	amp	metal	athen	weed
	amp	sex	10	rock	day	profit
	love	porn	student	nowplay	travel	stock
	yogurt	nude	11	eat	turkey	read
	great	nake	today	amp	love	big
	day	greek girl	12	amp eat	oxi	read big
	play	video	class	webradio nowpl	itali	profit microcap
	best	sexi	ut	eat rock	airport	big weed
	mythолog	fuck	ascend	metal amp	28	stock profit
tweets	5704	1018	1973	95	2838	539
Silhouette Coefficient			0,020239221			
Calinski-Harabaz Index			96,8111761334			

Πίνακας 12

Τέλος επιχειρήσαμε πολλά ακόμα πειράματα αλλάζοντας όλες τις πιθανές παραμέτρους του k-means, ειδικότερα τον αριθμό των clusters και των επαναλήψεων και παίξαμε με διάφορους περιορισμούς ως προς το ποιοί όροι θα αγνοηθούν στην δημιουργία του tf-idf. Αυτά που παρατηρήσαμε ήταν πως καθώς ανεβάζαμε τον αριθμό των clusters και επίσης μειώναμε τις λέξεις που λαμβάναμε υπόψη οι μετρικές εγκυρότητας γινόντουσαν όλο και καλύτερες, μιας και ήταν πιο εύκολα τα περισσότερα clusters μας να μην αλληλεπικαλύπτονται, παρ' όλα αυτά χωρίς κάποια ιδιαίτερη αλλαγή και χωρίς κάποιο νέο θέμα να φανερώνεται από τα περισσότερα clusters.

5.4.2 Αποτελέσματα Ward

Για τον αλγόριθμο ward επιχειρήσαμε μόνο ένα πείραμα για το παρόν σύνολο. Ο λόγος ήταν πως παρ' ότι χρησιμοποιήσαμε ένα από τα μικρότερα σύνολα μας, ο συγκεκριμένος αλγόριθμος φάνηκε αρκετά χρονοβόρος. Η πολυπλοκότητα του

συγκεκριμένου είναι στην καλύτερη περίπτωση $O(n^2 \log n)$ και στην χειρότερη $O(2^n)$. Επίσης η υλοποίηση που θα μας επέστρεφε και τους top όρους αδυνατούσε να τερματίσει.

Παράμετροι πειράματος: $4 \leq df \leq 2\%$ έδωσε ένα tf-idf (12167, 5022), 6 θέματα, μετρική η απόσταση Manhattan

	Clusters					
Clusters	0	1	2	3	4	5
tweets	11508	114	135	121	142	147
Silhouette Coefficient			-0,282887521			
Calinski-Harabaz Index			12,8351347436			

Πίνακας 13

Από τα μέτρα εγκυρότητας φαίνεται να έχουμε μια κακή ομαδοποίηση.

5.4.3 Αποτελέσματα Non-negative Matrix Factorization - NMF

Παράμετροι πειράματος Α: $2 \leq df \leq 30\%$ έδωσε ένα tf-idf (12167, 13996), 6 θέματα

A	Clusters					
Clusters	0	1	2	3	4	5
top 20 terms per cluster	microcap	greec	gr	girl	play	amp
	weed	athen	10	sex	radio	rock
	profit	day	amp	greek girl	90s	eat
	stock	travel	student	porn	80s	nowplay
	read	love	today	nude	greatest	webradio
	big	turkey	love	nake	hit	metal
	read big	arianagrand	great	girl sex	play 80s	amp eat
	profit microcap	today	day	video	greatest 90s	webradio nowpl
	big weed	oxi	11	fuck	90s play	eat rock
	stock profit	island	class	sexi	radio hit	metal amp
	weed stock	athen greec	st	teen	hit greatest	webradio amp
	microcap read	ancient	love gr	nake greek	radio greatest	eat nowplay
	read microcap	28	12	free	love	rock webradio
	microcap weed	oxiday	work	hot	version	amp rock
	microcap big	itali	friend	women	version greatest	rock metal
	big read	nation	good	sex greek	radio amp	eat webradio
	microcap stock	airport	halloween	pic	love radio	rock eat
	weed big	refuge	game	amateur	edit	greec amp
	big profit	arianagrand gr	pretti	nude girl	amp hit	metal eat
	profit weed	octob	year	greek sex	hit version	metal webradio

Πίνακας 14

Επιχειρήσαμε και άλλα πειράματα με περισσότερες θεματικές και με διαφορετικά όρια στο ποιες λέξεις θα αγνοηθούν μα δεν παρατηρήθηκε κάποια σημαντική διαφοροποίηση των αποτελεσμάτων και γι' αυτό δεν τα παραθέτουμε.

5.5 Co-occurance Πίνακας

Τελευταίο αφήσαμε προς παρουσίαση τον co-occurance πίνακα, μιας και δεν είναι απαραίτητος στην ροή της εργασίας. Όπως προαναφέρθηκε στο κεφάλαιο 4 παράγεται έως ένα επιπλέον στατιστικό στο module preprocessor και αποθηκεύεται στο .csv με τα υπόλοιπα στατιστικά που εξάγει το συγκεκριμένο module.

α/α	couple	occurrences	α/α	couple	occurrences
1	(u'big', u'read')	540	20	(u'athen', u'greec')	211
2	(u'microcap', u'profit')	539	21	(u'greek', u'nake')	206
3	(u'microcap', u'weed')	539	22	(u'god', u'greek')	202
4	(u'microcap', u'read')	539	23	(u'greek', u'like')	198
5	(u'microcap', u'stock')	539	24	(u'day', u'greek')	193
6	(u'read', u'weed')	539	25	(u'day', u'greec')	187
7	(u'read', u'stock')	539	26	(u'amp', u'greek')	185
8	(u'big', u'microcap')	539	27	(u'girl', u'sex')	177
9	(u'big', u'profit')	539	28	(u'greec', u'greek')	176
10	(u'big', u'weed')	539	29	(u'amp', 'gr')	175
11	(u'big', u'stock')	539	30	('2', 'gr')	173
12	(u'stock', u'weed')	539	31	(u'fuck', u'greek')	160
13	(u'profit', u'weed')	539	32	(u'ancient', u'greek')	152
14	(u'profit', u'read')	539	33	(u'free', u'greek')	149
15	(u'profit', u'stock')	539	34	('1', 'gr')	142
16	(u'girl', u'greek')	443	35	('gr', u'now')	140
17	(u'greek', u'sex')	351	36	(u'greec', u'travel')	137
18	(u'greek', u'porn')	249	37	(u'day', u'oxi')	131
19	(u'greek', u'nude')	213	38		

Πίνακας 15

ΚΕΦΑΛΑΙΟ 6

Συμπεράσματα Εργασίας - Μελλοντικές Επεκτάσεις

Με την ολοκλήρωση αυτής της εργασίας, αποκτήσαμε θεωρητική και πρακτική εμπειρία πάνω σε διάφορους από τους τρόπους που προτείνονται για την ομαδοποίηση και την εξαγωγή θεματικών από συλλογές κειμένων.

Παρακάτω παρουσιάζονται συνοπτικά κάποιες σημαντικές παρατηρήσεις – συμπεράσματα που προέκυψαν κατόπιν της εκπόνησης της εργασίας και ιδιαίτερα των κεφαλαίων 4-5, δηλαδή της σχεδίασης, της υλοποίησης και την ανάλυσης των αποτελεσμάτων.

- Όσον αφορά εργασίες που ασχολούνται με την εξόρυξη γνώσης από το κοινωνικό δίκτυο Twitter πρέπει να δίνεται ιδιαίτερη προσοχή στο ιδιόμορφο των documents-tweets. Ειδικότερα ένα σημαντικό πρόβλημα το οποίο και λύσαμε είναι τα retweets, έπειτα σημαντικό πρόβλημα δημιουργούν και τα bots που κατακλύζουν οποιαδήποτε συλλογή με ένα σημαντικό αριθμό tweets-διαφημίσεων.

- Ο αλγόριθμος kmeans δίνει ανεκτές ομαδοποιήσεις για συλλογές tweets όμως πρέπει να δοκιμαστούν πιο εξειδικευμένες τεχνικές που να προσπερνούν κάποια αρνητικά του. Συγκεκριμένα το πρόβλημα με τον ορισμό των κέντρων αλλά και με την επιλογή των αρχικών κέντρων, όταν αυτό θέλουμε να γίνεται αυτόματα, γιατί κακή επιλογή των αρχικών κέντρων μπορεί να φέρει λανθασμένες ομαδοποιήσεις.

- Ενδελεχέστερη εξερεύνηση των ιεραρχικών αλγορίθμων και πως αυτοί μπορούν να υλοποιηθούν πιο αποδοτικά.

- Ενασχόληση και εμβάθυνση με πιο εξειδικευμένους και αποδοτικούς αλγορίθμους ως προς το topic model όπως οι LDA, PLSA, EM.

- Ενοποίηση των εργαλείων που δημιουργήσαμε και ελεύθερη παροχή τους με μια διαδικτυακή εφαρμογή.

Παράρτημα

Τετριμμένες Λέξεις για την Αγγλική γλώσσα

πηγή: http://www.ranks.nl/stopwords

a	keeps	t
able	kept	take
about	kg	taken
above	km	taking
abst	know	tell
accordance	known	tends
according	knows	th
accordingly	l	than
across	largely	thank
act	last	thanks
actually	lately	thanx
added	later	that
adj	latter	that'll
affected	latterly	thats
affecting	least	that've
affects	less	the
after	lest	their
afterwards	let	theirs
again	lets	them
against	like	themselves
ah	liked	then
all	likely	thence
almost	line	there
alone	little	thereafter
along	'll	thereby
already	look	thered
also	looking	therefore
although	looks	therein
always	ltd	there'll
am	m	thereof
among	made	therere
amongst	mainly	theres
an	make	thereto
and	makes	thereupon
announce	many	there've
another	may	these
any	maybe	they
anybody	me	theyd
anyhow	mean	they'll
anymore	means	theyre

anyone	meantime	they've
anything	meanwhile	think
anyway	merely	this
anyways	mg	those
anywhere	might	thou
apparently	million	though
approximately	miss	thoughh
are	ml	thousand
aren	more	throug
arent	moreover	through
arise	most	throughout
around	mostly	thru
as	mr	thus
aside	mrs	til
ask	much	tip
asking	mug	to
at	must	together
auth	my	too
available	myself	took
away	n	toward
awfully	na	towards
b	name	tried
back	namely	tries
be	nay	truly
became	nd	try
because	near	trying
become	nearly	ts
becomes	necessarily	twice
becoming	necessary	two
been	need	u
before	needs	un
beforehand	neither	under
begin	never	unfortunately
beginning	nevertheless	unless
beginnings	new	unlike
begins	next	unlikely
behind	nine	until
being	ninety	unto
believe	no	up
below	nobody	upon
beside	non	ups
besides	none	us
between	nonetheless	use
beyond	noone	used
biol	nor	useful
both	normally	usefully
brief	nos	usefulness

briefly	not	uses
but	noted	using
by	nothing	usually
c	now	v
ca	nowhere	value
came	o	various
can	obtain	've
cannot	obtained	very
can't	obviously	via
cause	of	viz
causes	off	vol
certain	often	vols
certainly	oh	vs
co	ok	w
com	okay	want
come	old	wants
comes	omitted	was
contain	on	wasnt
containing	once	way
contains	one	we
could	ones	wed
couldnt	only	welcome
d	onto	we'll
date	or	went
did	ord	were
didn't	other	werent
different	others	we've
do	otherwise	what
does	ought	whatever
doesn't	our	what'll
doing	ours	whats
done	ourselves	when
don't	out	whence
down	outside	whenever
downwards	over	where
due	overall	whereafter
during	owing	whereas
e	own	whereby
each	p	wherein
ed	page	wheres
edu	pages	whereupon
effect	part	wherever
eg	particular	whether
eight	particularly	which
eighty	past	while
either	per	whim
else	perhaps	whither

elsewhere	placed	who
end	please	whod
ending	plus	whoever
enough	poorly	whole
especially	possible	who'll
et	possibly	whom
et-al	potentially	whomever
etc	pp	whos
even	predominantly	whose
ever	present	why
every	previously	widely
everybody	primarily	willing
everyone	probably	wish
everything	promptly	with
everywhere	proud	within
ex	provides	without
except	put	wont
f	q	words
far	que	world
few	quickly	would
ff	quite	wouldnt
fifth	qv	www
first	r	x
five	ran	y
fix	rather	yes
followed	rd	yet
following	re	you
follows	readily	youd
for	really	you'll
former	recent	your
formerly	recently	youre
forth	ref	yours
found	refs	yourself
four	regarding	yourselves
from	regardless	you've
further	regards	z
furthermore	related	zero
g	relatively	
gave	research	
get	respectively	
gets	resulted	
getting	resulting	
give	results	
given	right	
gives	run	
giving	s	
go	said	

goes	same
gone	saw
got	say
gotten	saying
h	says
had	sec
happens	section
hardly	see
has	seeing
hasn't	seem
have	seemed
haven't	seeming
having	seems
he	seen
hed	self
hence	selves
her	sent
here	seven
hereafter	several
hereby	shall
herein	she
heres	shed
hereupon	she'll
hers	shes
herself	should
hes	shouldn't
hi	show
hid	showed
him	shown
himself	showns
his	shows
hither	significant
home	significantly
how	similar
howbeit	similarly
however	since
hundred	six
i	slightly
id	so
ie	some
if	somebody
i'll	somehow
im	someone
immediate	somethan
immediately	something
importance	sometime
important	sometimes

in	somewhat
inc	somewhere
indeed	soon
index	sorry
information	specifically
instead	specified
into	specify
invention	specifying
inward	still
is	stop
isn't	strongly
it	sub
itd	substantially
it'll	successfully
its	such
itself	sufficiently
i've	suggest
j	sup
just	sure
k	
keep	

Διαθέσιμες Συλλογές από tweets

Για ένα μεγάλο χρονικό διάστημα είχαμε ενεργό τον tweets grabber και μαζεύαμε συλλογές από tweets, παρακολουθώντας διάφορους όρους. Ειδικότερα είχαμε εγκαταστήσει τον tweets grabber σε ένα virtual machine στο σύστημα Okeanos και έτσι είχαμε την δυνατότητα να το αφήσουμε να δουλεύει ανενόχλητα. Ενδεικτικά αναφέρουμε τις πιο σημαντικές συλλογές από tweets και τους όρους παρακολούθησης.

Όροι Παρακολούθησης	Χρονική Περίδος	Μέγεθος
euro2016, euro	22/06/2016 έως 17/08/2016	88,2Gb
turkey, coup	βράδυ 15/07 και 16/06/2016	10,9Gb

Κώδικας

Παρακάτω ακολουθούν ενδεικτικά κάποια κομμάτια κώδικα από το machine learning module.

```
#vectorize the text, convert the strings to numeric features
vectorizer = TfidfVectorizer(max_df=0.3, min_df=2, stop_words='english',
use_idf=True, ngram_range=(1,2))
tfidf_matrix = vectorizer.fit_transform(tweets)
```

```
print tfidf_matrix.shape

#cluster documents k-means
true_k = 6
model = KMeans(n_clusters=true_k, init='k-means++', max_iter=300, n_init=10)
model.fit(tfidf_matrix)

#print top terms per cluster clusters
print("Top terms per cluster:")
order_centroids = model.cluster_centers_.argsort()[:, ::-1]
terms = vectorizer.get_feature_names()
for i in range(true_k):
   print "Cluster %d:" % i,
   for ind in order_centroids[i, :10]:
      print ' %s' % terms[ind],
   print

from sklearn import metrics
from sklearn.metrics import pairwise_distances

# calc and print silhouette and calinski&harabaz validity measures
klabels = model.labels_
silh_coef = metrics.silhouette_score(tfidf_matrix, klabels, metric='euclidean')
print silh_coef
X = tfidf_matrix.toarray()
calhar = metrics.calinski_harabaz_score(X, klabels)
print calhar

# calc and print tweets per cluster
nums = Counter(klabels)
print nums

# extract topics with NMF
from sklearn.decomposition import NMF
nmf = NMF(n_components=6, random_state=1).fit(tfidf_matrix)

feature_names = vectorizer.get_feature_names()

#print top terms per topic
for topic_idx, topic in enumerate(nmf.components_):
   print("Topic #%d:" % topic_idx)
   print("  ".join([feature_names[i]
      for i in topic.argsort()[:-20-1:-1]]))
   print
```

Βιβλιογραφία - Αναφορές

[1] - Εξόρυξη Γνώσης από Βάσεις Δεδομένων και τον Παγκόσμιο Ιστό, Μ.Χαλκίδη και Μ.Βαζιργιάννης, 2005, Τυπωθήτω

[2] - Data Mining: Εισαγωγικά και Προηγμένα Θέματα Εξόρυξης Γνώσης από Δεδομένα, Dunham M.H., 2004, Εκδόσεις Νέων Τεχνολογιών.

[3] - Γραμμική Άλγεβρα και Εφαρμογές, Gilbert Strang, 2010, ΠΕΚ

[4] - Principles of Data Mining, David Hand, Heikki Mannila, and Padhraic Smyth , 2001, MIT Press

[5] - Algorithms for Non-negative Matrix Factorization,Daniel D. Lee and H. Sebastian Seung, NIPS 2000, MIT Press

[6] - Python for Data Analysis, Wes McKinney, 2013, O'Reilly

Φοιτητικές εργασίες

[7] - Μια πειραματική διερεύνηση στην εξόρυξη θεματικών κατηγοριών από κοινωνικά δίκτυα, Διπλωματική εργασία, Γώγος Αναστάσιος, Θεόδωρος Καλαμπούκης, 2012, Οικονομικό Πανεπιστήμιο Αθηνών ,ΠΜΣ: Επιστήμη Υπολογιστών

[8] - Μελέτη του αλγορίθμου ομαδοποίησης k-means σε δεδομένα του παγκόσμιου ιστού, Διπλωματική εργασία, Ακακιάδου Γεωργία, Παπαδημητρίου Γεώργιος, 2007, Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης Τμήμα Πληροφορικής

[9] - Ανάλυση Συστάδων, Διπλωματική εργασία, Καράγεωργα Ισμήνη, Φίλιππος Αλεβίζος, 2012, Πανεπιστήμιο Πατρών, ΠΜΣ: Μαθηματικά των Υπολογιστών και των Αποφάσεων

Διαδικτυακοί Τόποι

[10] - http://mike.verdone.ca/twitter/#about

[11] - http://socialmedia-class.org/index.html

[12] - http://scikit-learn.org/stable/modules/clustering.html

[13] - https://datasciencelab.wordpress.com/2013/12/12/clustering-with-k-means-in-python/

[14] - https://iksinc.wordpress.com/2015/06/23/how-to-use-words-co-occurrence-statistics-to-map-words-to-vectors/

[15] - http://brandonrose.org/clustering

[16] - http://scikit-learn.org/stable/auto_examples/text/document_clustering.html

[17] - https://dzone.com/articles/machine-learning-text-feature

[18] - http://scikit-learn.org/stable/documentation.html

[19] - http://blog.christianperone.com/2011/09/machine-learning-text-feature-extraction-tf-idf-part-i/

[20] - http://blog.christianperone.com/2011/10/machine-learning-text-feature-extraction-tf-idf-part-ii/

[21] - http://blog.christianperone.com/2013/09/machine-learning-cosine-similarity-for-vector-space-models-part-iii/

[22] - http://scikit-learn.org/stable/auto_examples/applications/topics_extraction_with_nmf_lda.html#sphx-glr-auto-examples-applications-topics-extraction-with-nmf-lda-py

[23] - https://en.wikipedia.org/wiki/Tf%E2%80%93idf

[24] - https://en.wikipedia.org/wiki/Bag-of-words_model

[25] - https://en.wikipedia.org/wiki/Jaccard_index

[26] - https://en.wikipedia.org/wiki/K-means_clustering

[27] - https://en.wikipedia.org/wiki/Hierarchical_clustering

[28] - https://en.wikipedia.org/wiki/Non-negative_matrix_factorization

[29] - https://repository.kallipos.gr/bitstream/11419/1238/2/Kef._11.pdf