

ΕΛΛΗΝΙΚΟ ΜΕΣΟΓΕΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

Τμήμα Ηλεκτρονικών Μηχανικών

Πρόγραμμα Μεταπτυχιακών Σπουδών:
«Ηλεκτρονικά Συστήματα Τηλεπικοινωνιών & Αυτοματισμών (ΗΣΤΑ)»

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του σπουδαστή

Παπαβασιλείου Βλάσιου του Βασιλείου

ΑΕΜ:ΜΤ10

Θέμα:
“Αύξηση της εισροής τουριστών δια της στοχευμένης προώθησης δραστηριοτήτων των τοπικών κοινωνιών σε σελίδες κοινωνικής δικτύωσης βάσει των εξατομικευμένων ενδιαφερόντων των χρηστών”

Subject:
“Increase of tourist influx through the promotion of local community activities on social media pages based on personalized user interests”

Τριμελής εξεταστική επιτροπή:

Επιβλέπων καθηγητής: Κωσταντάρας Αντώνιος
Μέλη: Αντωνιδάκης Εμμανουήλ, Κόκκινος Ευάγγελος.

ΧΑΝΙΑ

Σεπτέμβρης 2024

Copyright © Βλάσιος Παπαβασιλείου, 2024 Με επιφύλαξη παντός δικαιώματος.
All rights reserved.

Η έγκριση της πτυχιακής εργασίας από το πρόγραμμα δεν υποδηλώνει απαραιτήτως και αποδοχή των απόψεων του συγγραφέα εκ μέρους του Τμήματος.

Περίληψη

Η εργασία αυτή εξετάζει τις δυνατότητες που παρέχουν οι νέες τεχνολογίες στην έρευνα διά της συλλογής και επεξεργασίας δεδομένων τα οποία εκουσίως αναρτούν οι χρήστες των Μέσων Κοινωνικής Δικτύωσης σε μορφή γραπτού λόγου. Εξετάζεται η περίπτωση του X (πρώην Twitter), μιας εφαρμογής αναρτήσεων η οποία επικεντρώνεται κυρίως στην γραπτή έκφραση των χρηστών ως προς τη στάση τους και τη διάθεσή τους προς διάφορα θέματα, πρόσωπα και καταστάσεις. Σκοπός της εργασίας είναι να εξετάσει τρόπους στοχευμένης διάδοσης τουριστικών προορισμών σε χρήστες των ΜΚΔ βάσει των προτιμήσεών τους.

Η συλλογή δεδομένων έγινε με τη χρήση δύο μεθόδων:

Γεωγραφικά, χρησιμοποιώντας την επισήμανση της γεωγραφικής περιοχής βάσει συντεταγμένων και με χρήση ενός ή περισσοτέρων hashtags και λέξεων κλειδιών.

Για την αξιοποίηση των δεδομένων των χρηστών που προκύπτουν από τις αναρτήσεις τους, χρησιμοποιήθηκε η δυνατότητα που μας δίνει η διασύνδεση του προγράμματός, γραμμένο σε Python, με το REST API του X. Τα γλωσσικά εργαλεία που χρησιμοποιήθηκαν στον κώδικα του προγράμματος είναι το NLTK, μία πλατφόρμα εργαλείων επεξεργασίας φυσικής γλώσσας και η βιβλιοθήκη λέξεων Wordnet. Για τον καθαρισμό από περιττά μέρη του λόγου χρησιμοποιήθηκαν εκτενώς κανονικές εκφράσεις ενώ απομονώθηκαν τα ουσιαστικά τα οποία συνοδεύονται από θετικού νοήματος επίθετα.

Τα αποτελέσματα καταχωρούνται σε σχεσιακή βάση δεδομένων. Σε δεύτερο χρόνο αντλούνται από τη βάση δεδομένων και συγκρίνονται με θεματικά λεξιλόγια. Η συχνότητα των λέξεων κλειδιών που χρησιμοποιούνται από το χρήστη και ανήκουν σε συγκεκριμένη κατηγορία εξάγει το τελικό αποτέλεσμα με το όνομα χρήστη και τις 5 δημοφιλέστερες κατηγορίες που ταιριάζουν στο προφίλ του.

Λέξεις κλειδιά: Χρήστες, αναρτήσεις, γλώσσα, επεξεργασία, κώδικας, λεξιλόγια.

Abstract

This paper deals with the possibilities provided by new technologies in research through the collection and processing of data which are voluntarily posted by Social Media users in the form of written speech. The case of X is examined, a microblogging application that focuses mainly on the written expression of users in terms of their attitude and mood towards various topics, people and situations. The purpose of this paper is to examine ways of targeted dissemination of tourist destinations to users of SM.

Those data were collected using two methods:

Geographically, using geotagging based on coordinates and by reference, using one or more hashtags and keywords.

To take advantage of user data resulting from their posts, we used the ability provided by the interface of the program, written in Python, with the X API. The language tools used in the program code are NLTK, a natural language editing platform, and the Wordnet database which consists of words and terms. Normal expressions were used extensively to purify unnecessary parts of speech, while nouns accompanied by positive adjectives were isolated for further use.

The results have been recorded to a relational database. In the next phase, they are extracted from the database and compared with thematic vocabularies. The frequency of the keywords used by the user which are part of a specific category form the result consisted by the users' username and the 5 most popular categories that fits their profile.

Keywords: Users, posts, language, processing, code, vocabularies.

Περιεχόμενα

Περιεχόμενα

Περίληψη 3
Περιεχόμενα 5
Συντομεύσεις: 6
Εισαγωγή 7
1. ΜΕΣΑ ΚΟΙΝΩΝΙΚΗΣ ΔΙΚΤΥΩΣΗΣ 8
Σύντομη ιστορική αναδρομή 8
Από το USENET στο TikTok 9
Τα μέσα κοινωνικής δικτύωσης ως πηγές δεδομένων 10
Η εμπορική χρήση των ΜΚΔ 10
Επεξεργασία Φυσικής Γλώσσας (ΕΦΓ) 11
Ανάλυση του κειμένου: 12
2. ΕΡΓΑΛΕΙΑ ΚΑΙ ΜΕΘΟΔΟΙ ΕΠΕΞΕΡΓΑΣΙΑΣ ΦΥΣΙΚΗΣ ΓΛΩΣΣΑΣ 17
Η γλώσσα προγραμματισμού Python 17
Το X ως πηγή δεδομένων 18
Οντότητες και αντικείμενα 18
Τα πεδία και μεταδεδομένα των αναρτήσεων 21
X API και η βιβλιοθήκη tweepy 24
Στάδια λειτουργίας 26
Εργαλειοθήκη φυσικής γλώσσας (NLTK) 32
Σήμανση μερών του λόγου (pos tagging) 34
Wordnet 36
3ο ΚΕΦΑΛΑΙΟ 38
Ανάπτυξη της εργασίας 38
Διασύνδεση με τις λειτουργίες προγραμματιστή του X 38
Συλλογή δεδομένων από το X 41
4ο ΚΕΦΑΛΑΙΟ 48
Αποτελέσματα - Συμπεράσματα 48
Βιβλιογραφία 53
ΠΑΡΑΡΤΗΜΑ 55
Κώδικας: 55
Θεματικά λεξικά 66

Συντομεύσεις:

Ε.Φ.Γ. = Επεξεργασία Φυσικής Γλώσσας

ΜΚΔ = Μέσα Κοινωνικής Δικτύωσης

API = Application Programming Interface

SM=Social Media

Εισαγωγή

Η τέταρτη βιομηχανική επανάσταση έφερε τεράστιες καινοτομίες και ευκαιρίες σε όλους τους τομείς της οικονομικής αλλά και της προσωπικής ζωής με αιχμή του δόρατος την επικράτηση του διαδικτύου ως μέσο επικοινωνίας. Η έλευση των εφαρμογών κοινωνικής δικτύωσης αποτελούν μία πλούσια πηγή δεδομένων, μια πρώτη ύλη προς επεξεργασία και αξιοποίησή για τη βελτίωση τομέων της οικονομίας, όπως είναι η παραγωγή, ο τουρισμός, το εμπόριο κ.α. Τα δεδομένα αυτά, με τη χρήση κατάλληλων προγραμματιστικών εργαλείων, μπορούν να πάρουν τη μορφή μιας αξιοποιήσιμης πληροφορίας.

Το αντικείμενο της παρούσας εργασίας είναι η εξέταση της δυνατότητα; ανάπτυξης του τουριστικού κλάδου με την συνδρομή σύγχρονων προγραμματιστικών εργαλείων και τεχνολογιών τα οποία αξιοποιούν τα Μέσα Κοινωνικής Δικτύωσης ως πηγές δεδομένων προς εξαγωγή και επεξεργασία προσωποποιημένων πληροφοριών. Έτσι επιτυγχάνεται η στοχευμένη προσέγγιση για προορισμούς και δραστηριότητες που άπτονται του τουριστικού κλάδου

Το 1ο κεφάλαιο αναφέρεται στην κοινωνική δικτύωση ως φαινόμενο και μέσο προόδου διαφόρων τομέων της οικονομίας και της κοινωνίας. Αναλύεται το θεωρητικό υπόβαθρο της Επεξεργασίας Φυσικής Γλώσσας και οι δυνατότητές της.

Στο 2ο κεφάλαιο γίνεται μια εκτενής ανάλυση της γλωσσικής επεξεργασίας, των εργαλείων και των μεθόδων για την ανάλυση και αξιοποίηση των αναρτήσεων σε ένα μέσο με σκοπό τη δημιουργία ενός προτιμησιακού προφίλ του χρήστη. Επικεντρώνεται στο μέσο το οποίο έχει επιλεγεί για την εργασία, το Χ, στον τρόπο λειτουργίας του, στα αντικείμενα, τα δεδομένα, τα μεταδεδομένα και τις οντότητες τα οποία χρειάζεται να κατανοηθούν για την ανάπτυξη της εργασίας.

Στο 3ο κεφάλαιο αναλύονται η μέθοδος και τα βήματα της εργασίας μαζί με κομμάτια του κώδικα.

Το 4ο κεφάλαιο περιέχει τα αποτελέσματα και τα συμπεράσματα που προκύπτουν από την εργασία

1. ΜΕΣΑ ΚΟΙΝΩΝΙΚΗΣ ΔΙΚΤΥΩΣΗΣ

Σύντομη ιστορική αναδρομή

"...Ἐκ τούτων οὖν φανερὸν ὅτι τῶν φύσει ἡ πόλις ἐστί, καὶ ὅτι ὁ ἄνθρωπος φύσει πολιτικὸν ζῷον…"* (Μοσκόβης,1939) Με αυτό το απόσπασμα στο έργο του "Τα πολιτικά" ο Αριστοτέλης κατέδειξε την ανάγκη του ανθρώπου να ανήκει σε μια κοινωνία και ότι η σύναψη δεσμών με τους ομοίους του είναι φυσική επιταγή. Πάνω σε αυτή την αποδεδειγμένη αρχή βασίστηκε η εξέλιξη των κοινωνιών και η αλλαγές των κοινωνικοοικονομικών συστημάτων. Η τηλεπικοινωνία ήταν πάντα αναπόσπαστο κομμάτι και δείκτης της κοινωνικής εξέλιξης. Από την φρυκτωρία έως τον τηλέγραφο και από το πρώτο τηλέφωνο του Alex Graham Bell στα έξυπνα κινητά, καταδεικνύεται η ανάγκη που που υπάρχει για πιο άμεση και πιο μαζικής κλίμακας τηλεπικοινωνία και που εκφράζεται με τη συνεχώς αυξανόμενη τεχνολογική πρόοδο του τομέα αυτού. Αυτή η εξελικτική πορεία των τηλεπικοινωνιών ακολουθεί τις νέες δυνατότητες που δημιουργούνται στους τομείς της οικονομίας και της παραγωγής και μοιραία ακουμπάει και στην διαμόρφωση των κοινωνικών σχέσεων

Η εμφάνιση των ΜΚΔ έφερε αδιαμφισβήτητα μια επανάσταση στην επικοινωνία και αλληλεπίδραση των ανθρώπων σε όλο τον πλανήτη, εκμηδενίζοντας την επικοινωνιακή απόσταση μεταξύ τους. Η ραγδαία εξέλιξη σε τεχνολογικούς τομείς, όπως το διαδίκτυο και οι υπολογιστές διεύρυναν τις δυνατότητες για άμεση και μαζική αλληλεπίδραση μεταξύ των χρηστών. Τα ΜΚΔ είναι αυτή τη στιγμή τα πιο προσφιλή συστήματα επικοινωνίας στον κόσμο.

Το να ορίσει κάποιος κάτι που εξελίσσεται και μεταλλάσσεται συνεχώς είναι δύσκολο εγχείρημα. Ένας τελείως αφαιρετικός ορισμός των ΜΚΔ θα μπορούσε να είναι "Η άμεση διαδικτυακή αλληλεπίδραση χρηστών σε ευρεία κλίμακα". Ωστόσο, ένας τέτοιος ορισμός είναι ασαφής και δεν εμπεριέχει τις έννοιες της τεχνολογικής εξέλιξης που θα έπρεπε να το συνοδεύουν. Οι Kaplan & Henlein έδωσαν μια βασική διάσταση της οντότητας αυτών των μέσων που είναι η εξής: *"Τα Μέσα κοινωνικής Δικτύωσης είναι μια ομάδα διαδικτυακών εφαρμογών, αναπτυγμένες πάνω στα ιδεολογικα και τεχνολογικά θεμέλια του Web 2.0 και που επιτρέπουν την δημιουργία και ανταλλαγή περιεχομένου παραγόμενου από τους χρήστες" (Kaplan & Haenlein, 2009).* Αυτός ο ορισμός μπορεί να αποτυπώσει σωστά τα γενικά χαρακτηριστικά του

συνόλου αυτών των μέσων χωρίς να εμβαθύνει παραπέρα όπου και βλέπουμε μεγάλες διαφοροποιήσεις ως προς το περιεχόμενο και τον σκοπό. Με την εξέλιξη στις ταχύτητες σε επίπεδο εξοπλισμού και δικτύου, τα ΜΚΔ επέκτειναν τις δυνατότητές τους σε πεδία όπως διαδικτυακά παιχνίδια, επεξεργασία και ανάρτηση βίντεο, φωτογραφιών και μέσα κοινωνικής δικτύωση που υποστηρίζουν προγράμματα και δυνατότητες. Δεν είναι υπερβολή να ισχυριστεί κάποιος ότι σήμερα, ο κάθε χρήστης του διαδικτύου είναι μέλος τουλάχιστον ενός ΜΚΔ.

Από το USENET στο TikTok

Μια αρχέγονη μορφή ΜΚΔ ήταν το USENET (Users Network), μια ιδέα που υιοθετούσε τον τρόπο χρήσης ενός πίνακα ανακοινώσεων και υλοποιήθηκε από δύο απόφοιτους του πανεπιστημίου Duke της Νότιας Καρολίνας, Tom Truscott και Jim Ellis. Στο USENET δημιουργούνταν Newsgroups στα οποία μπορούσαν οι χρήστες να δημοσιεύσουν, να διαβάσουν και να προωθήσουν ειδήσεις. (Hauben, M., & Hauben, R., 1998)

Η εμφάνιση του “Open Diary”, το 1998 από τους Bruce και Susan Ableson ήταν η πρώτη εμφάνιση ενός υποτυπώδους ΜΚΔ στο οποίο οι χρηστες αφου έγραφαν σε κείμενο τι σκέψεις τους, μπορούσαν να δημοσιεύσουν συνδέσμους, φωτογραφιες, βίντεο ροής και να σχολιάσουν ο ένας στο περιεχόμενο του άλλου, δημιουργώντας έτσι την πρώτη μορφή διαδικτυακή κοινότητας. Τα weblogs - μετέπειτα blogs - συνέπεσαν χρονικά με τη δημιουργία του Open Diary και παρόλο που το τελευταίο έκλεισε επίσημα το 2014, αυτά υπάρχουν έως και σήμερα (Annamari Martinviita,2015).

Η αναβάθμιση των διαδικτυακών εφαρμογών σύμφωνα με τα πρότυπα του WEB 2.0 δημιούργησε τα μέσα κοινωνικής δικτύωσης με τη γνώριμη σύγχρονη μορφή. Το πρώτο ήταν το Myspace (2003), το οποίο αν και δημοφιλές τα πρώτα χρόνια λειτουργίας του, παραγκωνίστηκε από το Facebook(2004). Ακολούθησε η ίδρυση κι άλλων ΜΚΔ, με τα κυριότερα μεταξύ άλλων να είναι το YouTube (2005), το twitter (2006) το οποίο μετονομάστηκε σε X το 2022, το Pinterest (2010), το Instagram (2012) και το TikTok (2017).

Τα μέσα κοινωνικής δικτύωσης ως πηγές δεδομένων

Η εμπορική χρήση των ΜΚΔ

Το σύνολο του περιεχομένου στα ΜΚΔ προέρχεται από παραγόμενες και διαμοιραζόμενες από τους χρήστες δημοσιεύσεις. Οι χρήστες έχουν τη δυνατότητα να δημοσιεύσουν εμπειρίες, σκέψεις, μαρτυρίες, να αξιολογήσουν προϊόντα, να διαδώσουν ειδήσεις και γεγονότα. Συστατική λειτουργία στα μέσα κοινωνικής δικτύωσης είναι ο διαμοιρασμός του περιεχομένου, αντιγράφοντας τη κλασσική μέθοδο προτροπής “από στόμα σε στόμα”. Μαζί με την αξιολόγηση και τη δυνατότητα κριτικής, τα τρία αυτά στοιχεία καθιστούν το μέσο ένα ευρείας κλίμακας διαδικτυακό μηχανισμό αναφοράς και εκτίμησης (Dellarocas, C. 2003).

Με δεδομένα τα παραπάνω, τα ΜΚΔ αξιοποιήθηκαν και για την ανάπτυξη εμπορικών δραστηριοτήτων. Οι διαφημιστικές εταιρείες διαπίστωσαν ότι ο μέσος χρήστης είναι πιο πειστικός στο να προωθήσει ένα προϊόν που χρησιμοποιεί από οποιαδήποτε εμπορική διαφημιστική καμπάνια. Με τον καιρό, οι αξιολογήσεις προϊόντων έγιναν από ερασιτεχνικές ημι-επαγγελματικές ενώ αναδείχθηκε μια νέα μορφή αυτοαπασχολούμενων, οι influencers. Πάνω σε αυτό το μοντέλο εμπορίας και διαφήμισης χτίστηκε μια τεράστια οικονομία η οποία αριθμεί κέρδη δισεκατομμυρίων δολαρίων ετησίως. Τα ιστολόγια, κατά την περίοδο που είχαν κατακλύσει το διαδίκτυο ως μορφή θεματολογικής ενημέρωσης από απλούς χρήστες αλλά και εταιρείες, διαδραμάτισαν σημαντικό ρόλο στη διαμόρφωση εμπορικών τάσεων με προεκτάσεις σε κλάδους όπως ο τουρισμός (Akehurst G., 2009).

Πέρα όμως από την προφανή αξιοποίηση των ΜΚΔ για εμπορικούς σκοπούς, υπάρχει και η απεριόριστη δυνατότητα που δίνουν τα δισεκατομμύρια των δημοσιεύσεων των χρηστών με τη μορφή δεδομένων ή μεταδεδομένων. Αυτή η τεράστια δεξαμενή δεδομένων αξιοποιείται από εταιρείες προώθησης για την στοχευμένη προσέγγιση πελατών βάσει ενός προφίλ που διαμορφώνεται από τις προτιμήσεις τους, όπως αυτές εκφράζονται από τις δημόσιες τοποθετήσεις τους στα ΜΚΔ.

Επεξεργασία Φυσικής Γλώσσας (ΕΦΓ)

Η ανάγκη χρήσης της ΕΦΓ

Η επεξεργασία φυσικής γλώσσας χρησιμοποιεί υπολογιστικές τεχνικές για την ανάλυση, κατανόηση και την παραγωγή περιεχομένου της ανθρώπινης γλώσσας. Η αυτοματοποίηση της ανάλυσης της δομής της γλώσσας χρησιμοποιείται στην ανάπτυξη νέων τεχνολογιών όπως η αυτόματη μετάφραση, η αναγνώριση και η δημιουργία ομιλίας. Ενσωματώνεται με τη χρήση διαφόρων γλωσσικών εργαλείων σε πραγματικές εφαρμογές για την ανάλυση δεδομένων από πηγές όπως Μέσα Κοινωνικής Δικτύωσης και μέσα ενημέρωσης για την υγεία, την οικονομία και άλλους τομείς (Hirschberg,Manning, 2015). Ταυτόχρονα αξιοποιείται και στην ανάπτυξη καθημερινών λειτουργιών ευρείας χρήσης όπως η αυτόματη διόρθωση κειμένου, μετάφραση κειμένου, ανάπτυξη εικονικών βοηθών (π.χ. Apple Siri, Google Assistant). Με την εδραίωση της Τ.Ν στην καθημερινότητα οι δυνατότητες αυτές εκτοξεύτηκαν σε απεριόριστο βαθμό.

Η σημασία της επεξεργασίας φυσικής γλώσσας έγκειται στο γεγονός ότι είναι δυνατόν με αυτοματοποιημένο τρόπο να αναλυθούν και να “κατανοηθούν” από προγράμματα οι κανόνες της γλώσσας, η χρήση της και η σωστή εκφορά της.
Ωστόσο, κάτι τέτοιο είναι τόσο πολύπλοκο όσο και η πολυπλοκότητα της γλώσσας. Παρόλο που ο γραπτός λόγος, οποίος είναι και το αντικείμενο έρευνας στην παρούσα εργασία, έχει συγκεκριμένα χαρακτηριστικά που του προσδίδουν νόημα, περιεχόμενο και έμφαση μέσω του συντακτικού του, υπάρχουν σημεία ασάφειας που καθιστούν αυτό το εγχείρημα εξαιρετικά δύσκολο(Γεωργούλη, 2015).

Παράδειγμα (Ελληνικά):
Η δασκάλα διδάσκει από την έδρα της.
Η ομάδα έπαιζε στην έδρα της.
Ο κύβος έχει έξι έδρες.

Παράδειγμα (Αγγλικά):
He took a break to rest.
She tried to break the bonds with him.
Oh, please… give me a break!

Στον προφορικό λόγο η δυσκολία αυτή είναι ακόμα μεγαλύτερη εφόσον προστίθενται στο πρόβλημα οι ομώνυμες και παρώνυμες λέξεις, καθώς και η καθομιλουμένη που αποτελείται από πολλά νεωτεριστικά στοιχεία, εκφράσεις της καθημερινότητας και εκφράσεις από την αργκό.

Ανάλυση του κειμένου:

Στο προηγούμενο παράδειγμα διαφαίνεται η δυσκολία της εννοιολογικής ανάλυσης μιας πρότασης η οποία αποτελείται από λέξεις που για να αποδώσουν συγκεκριμένο νόημα πρέπει να τοποθετηθούν κατάλληλα μέσα σε μια πρόταση. Το περιεχόμενο συνολικά της πρότασης είναι αυτό που προσδίδει την έννοια σε μια λέξη ώστε η πρόταση να βγάζει λογικό νόημα. Αυτή είναι και η πρόκληση τη λεκτικής και εννοιολογικής ανάλυσης ενός κειμένου όταν γίνεται π.χ. μετάφραση μιας ολόκληρης περιόδου λόγου κι όχι μια μεμονωμένης λέξης.
Η ανάλυση ενός γραπτού κειμένου γίνεται με τη χρήση αναλυτών, οι οποίοι μπορεί να είναι από κομμάτια κώδικα ή και ολοκληρωμένα προγράμματα. Η ανάλυση του κειμένου γίνεται σε 4 φάσεις ανάλυσης (Γεωργούλη, 2015).

Μορφολογική ανάλυση

Η μορφολογική ανάλυση είναι η πρώτη φάση της γλωσσικής επεξεργασίας, κατά την οποία αναγνωρίζονται τα μορφήματα που απαρτίζουν μια λέξη. Το μόρφημα αντιπροσωπεύει το ελάχιστο γλωσσικό στοιχείο της λέξης που φέρει νόημα. Η συνηθισμένη πρακτική της μορφολογικής ανάλυσης είναι η χρήση λεξικών ευρετηρίων που περιέχουν τη ριζική μορφή των λέξεων και χρησιμοποιούνται για τον διαχωρισμό της λέξης σε λεξιλογικά και γραμματικά μορφήματα.

Τα λεξιλογικά μορφήματα διακρίνονται σε δύο κατηγορίες: ελεύθερα/δεσμευμένα και διαιρούνται περαιτέρω σε γραμματικά και λεξιλογικά.

Τα ελεύθερα μορφήματα είναι λέξεις που είναι αυτόνομες στο κείμενο, όπως "και", "που", "κάτι", "ένα", "δύο", και συνήθως δεν απαιτούν αναλυτική εξέταση κατά την γλωσσική ανάλυση της ελληνικής γλώσσας. Ωστόσο, στην ανάλυση της ελληνικής γλώσσας, η προσοχή επικεντρώνεται κυρίως στα δεσμευμένα μορφήματα λόγω της ευρείας χρήσης των πτώσεων και των κλιτικών στη λήμματα.

Στην γλωσσική ανάλυση της εργασίας στα αγγλικά χρησιμοποιείται ένα μέρος αυτών τα οποία φέρουν σημασία, όπως τα ουσιαστικά. Τα υπόλοιπα ελεύθερα αφαιρούνται ως άνευ σημασίας.

Το παράδειγμα της λέξης “ζωγράφος”, “ελαιοχρωματιστής”, “subordinate”.

Δεσμευμένο	Λεξικολογικό	Γραμματικό	ρίζα 1	ρίζα 2
ζωγράφος	ζωγραφ-	-ος	ζω-	γραφ-
Ελαιοχρωματιστής	Ελαιοχρωμα-	-ιστής	ελαιο	χρωμα-
subordinate	subordin-	-ate	-sub	-ordin

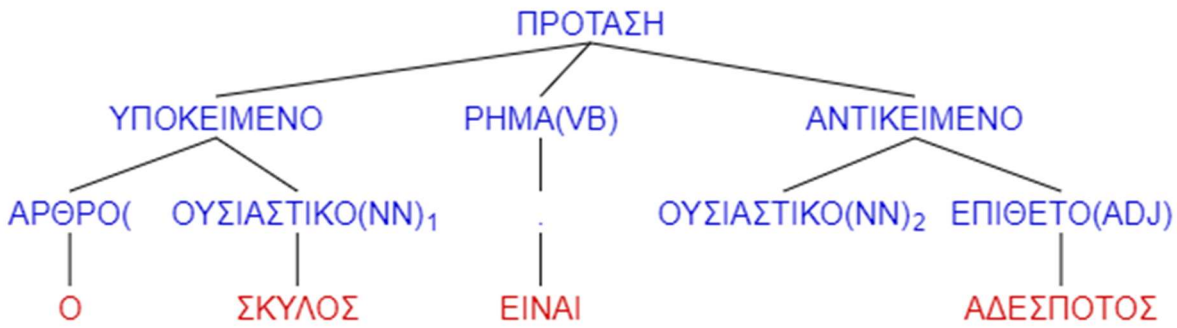
Πίνακας 1.1

Λεκτική ανάλυση

Η λεκτική ανάλυση αναλύει τις προτάσεις βάσει των λέξεων και της θέσης τους στο κείμενο. Για να απαιτείται μια προ-επεξεργασία κατά την οποία η πρόταση κατακερματίζεται σε μικρά κομμάτια που ονομάζονται κέρματα (tokens). Ο κατακερματισμός αυτός είναι απαραίτητος για να διακριθούν από μια πρόταση τα μέρη του λόγου και οι γραμματικές κατηγορίες έτσι ώστε να εξυπηρετηθεί η μορφολογία μιας πρότασης

Συντακτική ανάλυση

Η συντακτική ανάλυση πραγματοποιείται με τη χρήση της γραμματικής της γλώσσας κι ενός συντακτικού αναλυτή. Σε πρώτη φάση εξετάζεται η ορθή χρήση των λέξεων μέσα στην πρόταση. Κατόπιν εξετάζεται η εγκυρότητα της πρότασης βάσει των κανόνων της γραμματικής. Κατά τη συντακτική ανάλυση κατηγοριοποιούνται τα μέρη του λόγου σε ένα κείμενο και επισημαίνονται για περαιτέρω χρήση. Η ανάλυση ακολουθεί δενδρική μορφή από την πρόταση προς τα μέρη το λόγου.



Σχεδιάγραμμα 1.1

Σημασιολογική ανάλυση

Σημασιολογική είναι η ανάλυση που πραγματοποιείται στην πρόταση ώστε να αποδοθεί σωστά το νόημα των χρησιμοποιούμενων λέξεων μέσα στην πρόταση. Με αυτό τον τρόπο επιχειρείται να λυθεί το πρόβλημα που προκύπτει όταν η δομή της πρότασης με τα περιβάλλοντα μέρη του λόγου αποδίδουν σε μια λέξη διαφορετικό νόημα.

Η σημασιολογική ανάλυση μπορεί να είναι είτε λεκτική, όπου εξετάζεται το νόημα των λέξεων ξεχωριστά από την πρόταση είτε σύνθετη ανάλυση όπου εξετάζονται οι λέξεις ως μέρη μιας πρότασης. Με τη μέθοδο της σημασιολογικής αναπαράστασης της πρότασης εξάγεται το νόημά της αφού αναλυθεί η έννοια των λέξεων μέσα σε αυτή.

Παράδειγμα:
Η Μαρία είναι υπάλληλος του Γιώργου
Ο Γιώργος είναι υπάλληλος της Μαρίας

Στο παραπάνω παράδειγμα, ενώ περιέχονται οι ίδιες λέξεις με την ίδια συντακτική δομή, η διαφορετική θέση των ονομάτων στο κείμενο δίνει διαφορετική έννοια σε κάθε πρόταση.

Τα βασικά στοιχεία της σημασιολογικής ανάλυσης:

Υπώνυμο: Η λέξη που είναι επιμέρους στοιχείο μιας ευρύτερης εννοιολογικής κατηγορίας. Π.χ. ο σκύλος είναι ζώο, το κομοδίνο είναι έπιπλο. Δε συμβαίνει το αντίστροφο, το έπιπλο δεν είναι απαραίτητα κομοδίνο.

Ομώνυμο: Η λέξη που είναι ίδιας ορθογραφίας ή μορφής αλλά με διαφορετική έννοια. Π.χ. εργολάβος (γλυκό) με εργολάβος(επάγγελμα).

Πολύσημο: Είναι η λέξη που έχει διαφορετικές μεν, σχετικές μεταξύ τους δε έννοιες. Π.χ. Γραφείο είναι το έπιπλο αλλά γραφείο είναι και το μέρος στο οποίο εργάζεται ένας διοικητικός υπάλληλος.

Στη σημασιολογική ανάλυση χρησιμοποιείται η μέθοδος των αναπαραστάσεων. Τα δομικά στοιχεία ενός σημασιολογικού συστήματος που χρησιμοποιούνται στις αναπαραστάσεις είναι τα εξής:

Οι οντότητες: Αναπαριστούν μεμονωμένα πρόσωπα, τοποθεσίες, καταστάσεις όπως “Γιώργος”, “Αγγλία”, “αρρώστια”.

Το πλαίσιο των οντοτήτων: στο παράδειγμά μας μπορεί να είναι όνομα, χώρα, ασθένεια

Η συσχέτιση: Αφορά τις σχέσεις μεταξύ οντοτήτων και πλαισίου.

Κατηγορήματα: Αφορά τη δομή των ρημάτων ώστε να αποκτούν νόημα. Π.χ. στην περίπτωση των ρημάτων “είμαι” και “κάνω” ,συνοδεύονται πάντα από ένα αντικείμενο ή κατηγορούμενο το οποίο δίνει την έννοια του ρήματος. Μπορεί επίσης να αποτελούνται και από άλλα συντακτικά στοιχεία “Είμαι καλός”, “κάνω μάθημα”, “κάνω χειρωνακτική εργασία”.

Η σημασιολογική αναπαράσταση βοηθάει στο να συνδεθούν μεταξύ τους τα δομικά στοιχεία ενός σημασιολογικού συστήματος ώστε από αυτή τη σύνδεση να προκύπτει λογικό νόημα. (Herbrich, Graepel, 2010)

Πραγματολογική Ανάλυση

Σε αυτό το στάδιο ανάλυσης εξετάζεται το νόημα που αποκτά μια πρόταση όταν σχετίζεται με το γενικότερο πλαίσιο του κειμένου, τις περιγραφόμενες συνθήκες και τις υπόλοιπες προτάσεις (τα συμφραζόμενα).

Παραδείγματα:

1. Αναφορές: “το έφαγε”.
   Μία πρόταση η οποία προφανώς αναφέρεται στο επικείμενο κομμάτι που προηγήθηκε αυτής.
2. Διάλογοι:
   “- Πιστεύω ότι ο καιρός είναι ακόμα κρύος…
   - Επίτρεψέ μου να διαφωνήσω!”
   Σε αυτό το παράδειγμα αλληλουχίας προτάσεων από δύο υποκείμενα, η επόμενη πρόταση καθορίζεται από την προηγούμενη του συνομιλητή.
3. Αίτιο/Αποτέλεσμα: “Σήμερα έβρεχε όλη μέρα. Δεν βγήκα καθόλου έξω”
   Στην πρώτη περίοδο λόγου αποδίδεται το αίτιο της δεύτερης.
4. Ενέργειες: “Την τρίτη φύγαμε για διακοπές. Οδηγούσα όλη μέρα”
   Σε αυτό το παράδειγμα δίνεται το αίτιο και το αποτέλεσμα όπως στο προηγούμενο αλλά ως απόρροια κάποιας ενέργειας που συντελέστηκε.

2. ΕΡΓΑΛΕΙΑ ΚΑΙ ΜΕΘΟΔΟΙ ΕΠΕΞΕΡΓΑΣΙΑΣ ΦΥΣΙΚΗΣ ΓΛΩΣΣΑΣ

Η γλώσσα προγραμματισμού Python

Για την εξαγωγή και επεξεργασία των δεδομένων, χρησιμοποιήθηκε η γλώσσα προγραμματισμού Python.

Η Python είναι μια ευέλικτη, δυναμική και δημοφιλής αντικειμενοστραφής γλώσσα που μπορεί να χρησιμοποιηθεί για γλωσσική ανάλυση κειμένου. Ορισμένα από τα χαρακτηριστικά της που μπορούν να την καθιστούν κατάλληλη για αυτήν τη χρήση είναι τα εξής:

- Ευανάγνωστη σύνταξη: Η Python έχει μια απλή και ευανάγνωστη σύνταξη που καθιστά τον κώδικα ευκολοδιάβαστο, ιδίως όταν πρόκειται για επεξεργασία κειμένου και ανάλυση γλώσσας. Σε αυτό συμβάλλει η δυνατότητα δυναμικής εκχώρησης μεταβλητών που μειώνει σημαντικά τις γραμμές κώδικα.
- Ευρεία κοινότητα και βιβλιοθήκες: Υπάρχει μια πλούσια κοινότητα προγραμματιστών και εκατοντάδες διαθέσιμες βιβλιοθήκες που μπορούν να χρησιμοποιηθούν για γλωσσική ανάλυση κειμένου, όπως το Natural Language Toolkit (NLTK) ή για τη συλλογή, τον καθαρισμό και τον αναλυτικό έλεγχο δεδομένων του X, όπως η tweepy που χρησιμοποιούνται στην εργασία.
- Υποστήριξη Unicode: Έχει ενσωματωμένη υποστήριξη για Unicode, προσφέροντας δυνατότητες επεξεργασίας κειμένου σε διάφορους χαρακτήρες και γλώσσες.
- Ευελιξία και προσβασιμότητα: Είναι μια ευέλικτη γλώσσα που επιτρέπει την ανάπτυξη κώδικα με σαφείς και δομημένους τρόπους, καθιστώντας την κατάλληλη για γλωσσική ανάλυση και επεξεργασία κειμένου.
- Ευκολία στην εγκατάσταση και χρήση: Είναι εύκολη στην εγκατάσταση και χρήση, με πλούσια τεκμηρίωση και εκτεταμένη υποστήριξη από την κοινότητα των προγραμματιστών.

- Συμβατότητα: Είναι συμβατή με πολλές άλλες γλώσσες προγραμματισμού και προσφέρει δυνατότητες ενσωμάτωσης και επικοινωνίας με άλλες γλώσσες και εργαλεία, όπως η Java και η C++.

- Ευελιξία στην ανάπτυξη: Επιτρέπει την ανάπτυξη πρωτότυπων και προηγμένων αλγορίθμων γλωσσικής ανάλυσης κειμένου, χωρίς περιορισμούς στην αρχιτεκτονική ή στον τύπο του προβλήματος.

- Ευκολία στην αποσφαλμάτωση: Προσφέρει δυνατότητες ευκολίας στο debugging και την αντιμετώπιση λαθών, με ευανάγνωστα μηνύματα.(Bird, Klein, Loper, 2009).

Το Χ ως πηγή δεδομένων

Για την ανάπτυξη της παρούσας εργασίας εξετάστηκε η περίπτωση του Χ, μιας υπηρεσίας microblogging με δυνατότητα ανάρτησης σύντομων μηνυμάτων 280 χαρακτήρων* (* Το Νοέμβρη του 1917 η υπηρεσία αποφάσισε να αυξήσει το όριο των χαρακτήρων από 140 σε 280) που περιγράφουν τη στάση του χρήστη για ένα συγκεκριμένο θέμα, τις σκέψεις του ή τα συναισθήματά του.

Η δομή του Χ δίνει τη δυνατότητα επικοινωνίας μεταξύ των χρηστών με τρόπο έτσι ώστε σε σύντομο χρονικό διάστημα να ικανοποιείται η περιέργεια του χρήστη για κάποιο θέμα. Ο μικρός όγκος των μηνυμάτων παρέχει αμεσότητα και ευκολία στην επικοινωνία ενώ - σε αντίθεση με άλλα παραδοσιακά ΜΚΔ - η αλληλεπίδραση μπορεί να γίνεται και χωρίς την κοινή αποδοχή μεταξύ των χρηστών, όπως στην περίπτωση των ακολούθων όπου δεν είναι απαραίτητο κάποιος να εγκρίνει ή να αποδέχεται έναν ακόλουθο.

Οντότητες και αντικείμενα

Οι οντότητες των αναρτήσεων (tweets)

Κάθε ανάρτηση αποτελείται από πίνακες που περιέχουν αντικείμενα οντοτήτων. Υπάρχουν οι απλές οντότητες που αφορούν αντικείμενα τα οποία εμπεριέχονται μια φορά στην ανάρτηση και παρατίθενται παρακάτω με παραδείγματα. Υπάρχουν κι οι

επεκτάσιμες οντότητες, όταν έχουμε πολλαπλά ίδια αντικείμενα εικόνας, βίντεο και κινούμενης εικόνας σε μια ανάρτηση.(X API Documentation)

Τα αντικείμενα μια ανάρτησης και η δομή τους

Η ανάρτηση στο X είναι συστατικό στοιχείο του μέσου. Είναι γνωστή και ως “επικαιροποίηση κατάστασης”. Η ανάρτηση ως αντικείμενο έχει πληθώρα πρωταρχικών δομικών ιδιοτήτων, συμπεριλαμβανομένων και των βασικών όπως τα id, created_at και text. Ακολουθώντας το δενδρικό σύστημα, τα αντικείμενα μιας ανάρτησης μπορεί να αποτελούν τα γονικά αντικείμενα άλλων αντικειμένων τις οντότητες όπως user, entities.

Οι οντότητες είναι πίνακες με αντικείμενα που ανήκουν σε αυτές και εμπεριέχονται στην ανάρτηση. Οι απλές οντότητες αφορούν αντικείμενα που απαντώνται μια φορά στην ανάρτηση ενώ οι επεκτάσιμες οντότητες αφορούν ίδια αντικείμενα εικόνας, βίντεο και κινούμενων εικονιδίων που απαντώνται πολλές φορές στην ίδια ανάρτηση.

Τα συστατικά στοιχεία μια ανάρτησης

Hashtag #: Αποτελείται από το σύμβολο τη δίεσης ακολουθούμενο από μια λέξη κλειδί. Χρησιμοποιείται για δημιουργία ευρετηρίου για λέξεις ή θέματα που μπορούν να δημιουργηθούν με την εισαγωγή τους σε κάποια ανάρτηση από οποιονδήποτε χρήστη. Δεν υπάρχει περιορισμός για το πλήθος των hashtags που μπορεί να χρησιμοποιηθούν σε μια ανάρτηση, πέρα από εκείνον των 280 χαρακτήρων. Όταν επιλεχθεί με το ποντίκι ή αναζητηθεί στο πλαίσιο αναζήτησης, εμφανίζονται όλες οι αναρτήσεις στις οποίες έγινε χρήση του από τους χρήστες.

Πραγματοποιήθηκε η μετάβαση του ορειβατικού συλλόγου στον Παρνασσό #Παρνασσός #Ορειβασία

Σε περίπτωση που γίνεται ευρείας έκτασης χρήση μιας #κατηγορίας, τότε αυτή εμφανίζεται στις τάσεις με χρήση συγκεκριμένου αλγόριθμου δημοφιλίας,

Τάσεις: Σχηματίζονται από τον αλγόριθμο που λαμβάνει υπόψη τη συχνότητα χρήσης μιας κατηγορίας και εμφανίζονται σε έναν χρήστη σύμφωνα με τα ενδιαφέροντά του, το ποιους ακολουθεί και την τοποθεσία του. Ο αλγόριθμος επιλέγει να εμφανίζει στις τάσεις τις κατηγορίες που είναι δημοφιλείς στο άμεσο παρόν παρά αυτές που μπορεί να είναι ενδεχομένως πιο δημοφιλείς αριθμητικά αλλά για περισσότερο χρονικό διάστημα. Ο χρήστης μπορεί να καθορίσει από τις ρυθμίσεις να παρακολουθεί τις τάσεις που αφορούν συγκεκριμένη τοποθεσία.

Αναφορά χρήστη @: Με αυτό τον τρόπο Μπορεί κάποιος να αναφερθεί σε έναν συγκεκριμένο χρήστη του Χ και εκείνος να ενημερωθεί για αυτή την αναφορά. Μπορούν να γίνουν αναφορές σε πολλού χρήστες σε μια ανάρτηση ενώ μπορεί να συνδυαστεί και με τη χρήση hashtags.

Παράδειγμα: Πραγματοποιήθηκε η μετάβαση του @OrivatikosSyllogos @HionodromikoParnassou στον Παρνασσό #Παρνασσός #Ορειβασία

Tweet: Η ανάρτηση, το συστατικό στοιχείο του Χ είναι η βάση της αλληλεπίδρασης των χρηστών. Μπορούν να ενσωματωθούν σε αυτή όλα τα υπόλοιπα βασικά στοιχεία (πολυμέσα, hashtags, αναφορές χρηστών) και εμπλουτίζεται αναλόγως με δεδομένα, μεταδεδομένα και οντότητες.

Retweet: Είναι η διαμοίραση μιας ανάρτησης με όλους όσους έχουν επιλέξει να ακολουθούν το λογαριασμό που την πραγματοποιεί. Επίσης με τον ίδιο τρόπο είναι δυνατή η επισήμανση και επανανάρτηση παλαιότερων αναρτήσεων του ίδιου λογαριασμού. Η επιλογή γίνεται με ένα κουμπί που βρίσκεται κάτω από κάθε ανάρτηση και προσβάσιμο από όλους τους χρήστες

Like: Είναι η δήλωση αρεσκείας μιας ανάρτησης. Η διαφορά με το retweet είναι ότι το like είναι φαινομενικά ανώνυμο και πιστώνεται στην ανάρτηση στην οποία γίνεται ενώ δε φαίνεται καθόλου στο λογαριασμό του χρήστη που το πραγματοποιεί.

Κοινοποίηση: Είναι η δυνατότητα να μεταφέρεται ένα tweet εκτός δικτύου του Χ είτε με την αντιγραφή κι επικόλληση του συνδέσμου, είτε με e-mail, τηλεφώνου ή κάποιας άλλη συνεργαζόμενης εφαρμογής.

Ακολούθηση: Είναι η δυνατότητα κάποιος να ακολουθεί τη δραστηριότητα κάποιου άλλου χρήστη και να γίνει με αυτόν τρόπο «ακόλουθός» του. Αφορά τη

συσχέτιση μεταξύ χρηστών, δείχνει την προτίμηση και εμπιστοσύνη στην πηγή της πληροφορίας. Δεν απαιτείται έγκριση για να ακολουθήσει ένας χρηστης κάποιον άλλον. Με το που πραγματοποιηθεί, η χρονογραμμή του ακολουθούμενου χρήστη θα προβάλλεται στην αρχική σελίδα του ακόλουθου.

Χρονογραμμή: Είναι το ιστορικό αναρτήσεων , επαναναρτήσεων και retweets του κάθε χρήστη σε χρονική σειρά. Είναι πολύ σημαντικό στοιχείο καθώς περιέχει όλα τα στοιχεία τα οποία μπορούν να χρησιμοποιηθούν από εργαλεία ανάλυσης και επεξεργασίας. Στην παρούσα εργασία είναι η βασική πηγή δεδομένων για τη δημιουργία του προφίλ των προτιμήσεων των χρηστών και συμβάλλει στο χτίσιμο του προφίλ του χρήστη.

ID: Για την ταυτοποίηση των αντικειμένων του X χρησιμοποιείται ένας μη προσημασμένος ακέραιος αριθμός 64bit που είναι μοναδικός. Διαμορφώνεται από έναν αριθμό χρονοσήμανσης, έναν αριθμό εργασίας και έναν αύξοντα αριθμό.

Τα πεδία και μεταδεδομένα των αναρτήσεων

Κάθε ανάρτηση συνοδεύεται από έναν αριθμό μεταδεδομένων. Υπάρχουν μεταδεδομένα τα οποία στέλνονται αυτόματα με την ανάρτηση ενώ άλλα μπορούν να είναι ενεργοποιημένα ή όχι από της ρυθμίσεις στο προφίλ του χρήστη.
Το περιεχόμενο και η αλληλουχία των μεταδεδομένων αυτών μπορούν να προβληθούν σε ένα json αρχείο, το οποίο είναι ένα αρχείο το οποίο αποθηκεύει πληροφορία με συγκεκριμένη δομή, εξάγεται από το API του X και μπορεί να διαβαστεί από διαμορφωμένα γι αυτό το σκοπό προγράμματα. Παράδειγμα ενός τέτοιου αρχείου βρίσκεται παρακάτω.


```
[{
  "created_at":    "Thu    Jun    22    21:00:00    +0000    2019",
  "id":                                          877994604561387520,
  "id_str":                                    "877994604561387520",
  "text": "Displaying an example of tweet's data and metadata #Tweet",
  "user":                                                          {
    "id":                                                 772682568,
    "id_str":                                            "772682568",
    "name":                      "Tweet                   Extractor",
    "screen_name":                                          "mtt10",
    "location":                    "Chania,                 Greece",
```

```
    "description":    "Searching    the    potentials    of    tweeter",
    "url":               "https://www.researchgate.net/profile/mtt10",
    },
    "protected":                                                  false,
    "followers_count":                                             2145,
    "friends_count":                                                 18,
    "listed_count":                                                 328,
    "created_at":    "Wed    Aug    22    02:06:33    +0000    2012",
    "favourites_count":                                              57,
    "time_zone":                                                "Athens",
  },
}]
```


Στο παραπάνω παράδειγμα περιέχονται πληροφορίες για την ανάρτηση tweet και για τον ίδιο τον χρήστη. Σε μια ελεύθερη απόδοση οι πληροφορίες του tweet θα μπορούσαν να περιγραφούν ως εξής:

«Η ανάρτηση με μοναδικό αναγνωριστικό αριθμό 877994604561387520 δημιουργήθηκε την Πέμπτη, 22 Ιουνίου 2019 από τον χρήστη με όνομα mtt10 και περιέχει το κείμενο “Displaying an example of tweet’s data and metadata #Tweet"
Ο δημιουργός με μοναδικό αναγνωριστικό αριθμό 772682568 έχει περιγραφικό όνομα "Tweet Extractor", όνομα χρήστη "mtt10" και δημιούργησε το λογαριασμό την Τετάρτη, 22 Αυγούστου 2012. Έχει δηλώσει τόπο κατοικίας τα “Chania, Greece” με ζώνη ώρας “Αθήνα”, περιγραφή λογαριασμού το "Searching the potentials of tweeter" και σύνδεσμο το https://www.researchgate.net/profile/mtt10. Ο λογαριασμός του έχει 2145 ακολούθους, 18 φίλους, 57 αγαπημένες αναρτήσεις και φαίνεται σε 328 δημόσιες λίστες.»

Ανάκτηση δεδομένων και μεταδεδομένων μέσω API

Τα δεδομένα στην πληροφορική είναι ακατέργαστα κομμάτια πληροφορίας που μπορούν να γίνουν χρήσιμα και αξιοποιηθούν μετά την επεξεργασία τους. Συναντάμε τα δεδομένα αυτά σε πολλές μορφές, όπως τιμές σε αποδείξεις, ένδειξη σε μετρητές ρεύματος και βάρος προϊόντων.

Όπως ορίζεται και στο λεξικό Marriam-Webster τα μεταδεδομένα είναι δεδομένα που παρέχουν πληροφορίες για άλλα δεδομένα.

Τα μεταδεδομένα διακρίνονται στις εξής κύριες κατηγορίες (Kay-Lynn Saunders):

Περιγραφικά μεταδεδομένα

Τα περιγραφικά μεταδεδομένα παρέχουν στα ψηφιακά αντικείμενα πληροφορίες που μπορούν να χρησιμοποιηθούν για αναζήτηση, ταξινόμηση, περιήγηση και φιλτράρισμα, ή για να παράσχουν χρήσιμες πληροφορίες όταν ένα αντικείμενο επιλεγεί. Τα κύρια περιγραφικά μεταδεδομένα περιλαμβάνουν τίτλο, περιγραφή, δημιουργό, λέξεις-κλειδιά και μοναδικά αλφαριθμητικά αναγνώρισης. Αυτές οι πληροφορίες μπορούν να βοηθήσουν τους χρήστες να εντοπίσουν και να επιλέξουν τα αντικείμενα που ταιριάζουν καλύτερα στις ανάγκες και τα ενδιαφέροντά τους.

Δομικά μεταδεδομένα

Τα δομικά μεταδεδομένα αναφέρονται στα δεδομένα που παρέχουν πληροφορίες σχετικά με τον τρόπο δομής ενός ψηφιακού αντικειμένου. Αυτά τα μεταδεδομένα περιγράφουν τη δομή, την οργάνωση και την ολοκληρωμένη αναπαράσταση του αντικειμένου. Επίσης, παρέχουν πληροφορίες για το αν το αντικείμενο ανήκει σε ένα σύνολο και εάν είναι δυνατή η πλοήγηση μέσα σε αυτό με τη χρήση σημείων αναφοράς.

Ένα παράδειγμα δομικού μεταδεδομένου είναι μια σελίδα σε ένα ψηφιακό βιβλίο. Μπορούμε να αναζητήσουμε αυτήν τη σελίδα χρησιμοποιώντας διάφορα σημεία αναφοράς, όπως το κεφάλαιο, η παράγραφος, ο αριθμός σελίδας, το ευρετήριο ή το περιεχόμενο. Τα δομικά μεταδεδομένα μας βοηθούν να κατανοήσουμε το πώς οργανώνεται η πληροφορία εντός του βιβλίου και πώς μπορούμε να την ανακτήσουμε με ακρίβεια και αποτελεσματικότητα.

Διαχειριστικά δεδομένα

Τα διαχειριστικά δεδομένα αναφέρονται στα τεχνικά χαρακτηριστικά ενός ψηφιακού αντικειμένου και χωρίζονται σε τρεις κατηγορίες:

- Τεχνικά μεταδεδομένα: Αυτά τα μεταδεδομένα περιέχουν πληροφορίες σχετικά με τη χρησιμοποιούμενη τεχνολογία για το αντικείμενο. Μπορεί να περιλαμβάνουν τεχνικές προδιαγραφές, συμβατότητα με άλλες πλατφόρμες, έκδοση λογισμικού και άλλες τεχνικές λεπτομέρειες.
- Μεταδεδομένα συντήρησης: Αυτά τα μεταδεδομένα αφορούν τις ιδιότητες αρχειοθέτησης και τη διάρκεια ζωής του αντικειμένου. Μπορεί να περιλαμβάνουν πληροφορίες σχετικά με τη μορφή αρχείου, τη συμπίεση, τον

τρόπο αποθήκευσης, την ποιότητα και τη διατήρηση του αντικειμένου στον χρόνο.

- Δικαιωμάτων: Αυτά τα μεταδεδομένα αφορούν την ιδιοκτησία, την άδεια χρήσης και τα δικαιώματα πρόσβασης στο ψηφιακό αντικείμενο. Περιλαμβάνουν πληροφορίες σχετικά με τον κάτοχο του αντικειμένου, την άδεια που έχει παραχωρηθεί για τη χρήση του, τυχόν περιορισμούς στην πρόσβαση και άλλα σχετικά δικαιώματα και περιορισμούς που αφορούν το αντικείμενο.

Τα μεταδεδομένα προσφέρουν πληροφορίες σχετικά με τις ιδιότητες των δεδομένων και επιτρέπουν πιο εξειδικευμένη επεξεργασία, όπως ομαδοποίηση βάσει χρήστη, προσφέροντας ποιοτικά και ποσοτικά χαρακτηριστικά στις αντλούμενες πληροφορίες και χρησιμοποιούνται ευρέως στα κοινωνικά μέσα ενώ ενσωματώνονται στους αλγορίθμους προτεινόμενου περιεχομένου.

X API και η βιβλιοθήκη tweepy

Η Διεπαφή Προγραμματισμού Εφαρμογής (API) γενικά είναι ένας τρόπος διασύνδεσης με τις δυνατότητες και τις λειτουργίες ενός προγράμματος και αξιοποίησή του από εξωτερικού χρήστες με ασφαλή και ελεγχόμενο τρόπο. Μια εφαρμογή που παρέχει μιας τέτοιου είδους διασύνδεση σε μια άλλη εφαρμογή μέσω της διεπαφής, ελέγχει την πρόσβαση στο περιεχόμενό της με εφαρμογή περιορισμών και δικαιωμάτων ή μπορεί και να παρέχει περισσότερες δυνατότητες επί πληρωμή.

Η API του X παρέχει πρόσβαση στα περιεχόμενά του με συγκεκριμένες μεθόδους αυθεντικοποίησης και με επίπεδα πρόσβασης σε περιεχόμενο. Προγραμματιστικά, στη γλώσσα python αυτή η πρόσβαση επιτυγχάνεται με τη βιβλιοθήκη tweepy.

Η βιβλιοθήκη Tweepy είναι μια Python βιβλιοθήκη που παρέχει ευέλικτες και εύχρηστες λειτουργίες για την αλληλεπίδραση με το API του X. Προσφέρει πολλές δυνατότητες για ανάλυση, επεξεργασία και αλληλεπίδραση με τα δεδομένα του X, και επιτρέπει την εκπόνηση προηγμένων αναλύσεων και εφαρμογών πάνω στα tweets και τους χρήστες του.(Tweepy API V1.1 Reference)

Ορισμένες από τις βασικές δυνατότητες της βιβλιοθήκης Tweepy περιλαμβάνουν τα εξής:

- Αναζήτηση και ανάκτηση δεδομένων του X: Η Tweepy παρέχει ευέλικτες λειτουργίες για την αναζήτηση και ανάκτηση δεδομένων από το X με βάση κριτήρια όπως λέξεις-κλειδιά, χρονική περίοδο, χρήστες, τοποθεσία και άλλα φίλτρα.

- Αλληλεπίδραση με το API του X: Η Tweepy διευκολύνει την αποστολή, τον έλεγχο και τον χειρισμό των αιτημάτων προς το API του X για την ανάκτηση ή αλλαγή δεδομένων του X, όπως ανάρτηση tweets, ανάγνωση χρονοσειρών, τη διαχείριση φίλων και ακόλουθων, την αλλαγή προφίλ και άλλες δράσεις.

- Επεξεργασία και ανάλυση δεδομένων του X: Η Tweepy παρέχει λειτουργίες για την επεξεργασία και ανάλυση των δεδομένων του X. Μπορείς να εξάγεις το περιεχόμενο των tweets, τα hashtags, τις αναφορές, τα links και άλλες πληροφορίες προς επεξεργασία και άντληση πληροφοριών, στατιστικά και τάσεις από τα δεδομένα του X.

- Διαχείριση αδειών πρόσβασης: Η Tweepy επιτρέπει την εύκολη διαχείριση των αδειών πρόσβασης για το API του X, συμπεριλαμβανομένης της διαδικασίας εξουσιοδότησης και ανανέωσης των κλειδιών πρόσβασης.

- Ακολούθηση και παρακολούθηση: Η Tweepy επιτρέπει την παρακολούθηση των δραστηριοτήτων χρηστών του X, όπως των followers, των friends, των tweets και των αναφορών, και τη δυνατότητα ακολούθησης, ακύρωσης ακολούθησης και άλλων δράσεων.

- Εντοπισμός των τάσεων και των διαχρονικών αλλαγών: Η Tweepy επιτρέπει τον εντοπισμό των τάσεων και των διαχρονικών αλλαγών στο X με την ανάκτηση και τον χειρισμό των δεδομένων του X σε πραγματικό χρόνο.

- Συλλογή δεδομένων: Με τη βοήθεια της Tweepy, μπορείς να συλλέξεις δεδομένα από το X, όπως τα tweets ενός χρήστη, τα tweets που περιέχουν

συγκεκριμένα hashtags ή αναφορές, τα tweets που αναφέρονται σε συγκεκριμένες λέξεις-κλειδιά, τα tweets ενός συγκεκριμένου χρήστη ή ακόμα και τα tweets που εμφανίζονται σε ένα συγκεκριμένο γεωγραφικό τοποθεσία.

- Επεξεργασία και ανάλυση των δεδομένων:
  Με την Tweepy είναι δυνατή η επεξεργασία και ανάλυση των δεδομένων του X, όπως να εξάγονται τα hashtags, τα mentions, τα links και άλλες πληροφορίες από τα tweets. Είναι εφικτή επίσης η ανάλυση συναισθημάτων, η καταμέτρηση των αναρτήσεων ανά κατηγορία ή ανά χρονικό διάστημα, ο εντοπισμός των δημοφιλέστερων χρηστών και η παραγωγή αναφορών και γραφημάτων με τα αποτελέσματα των αναλύσεων.

- Αλληλεπίδραση με το X API:
  Ως βιβλιοθήκη προγραμματισμού για το X API, η Tweepy επιτρέπει την αλληλεπίδραση με το API του X, την αναζήτηση αναρτήσεων, την αποστολή τους, την ανάκτηση πληροφοριών του προφίλ των χρηστών, την διαχείριση λογαριασμών, την ανάλυση και επεξεργασία δεδομένων του X και πολλές άλλες λειτουργίες.

- Επέκταση των δυνατοτήτων του X:
  Η Tweepy επιτρέπει επίσης τη δημιουργία και διαχείριση χρηστών, την ακολούθηση χρηστών, την ανταπόκριση χρηστών σε αναφορές, την αποστολή μηνυμάτων άμεσης ανταλλαγής (direct messages) και άλλων δυνατοτήτων που επεκτείνουν τη λειτουργικότητα του X.

Στάδια λειτουργίας

Τα στάδια λειτουργίας της βιβλιοθήκης tweepy είναι αυθεντικοποίηση, διασύνδεση, συλλογή δεδομένων. (Tweepy API V1.1 Documentation)

Κατά την αυθεντικοποίηση γίνεται η πιστοποίηση της εφαρμογής στο σύστημα μέσω του API του tweet, κάνοντας χρήση συμβολοσειρών που αποτελούν μοναδικά κλειδιά. Κατόπιν καλούνται κλάσεις (Handlers) που η καθεμιά διαχειρίζεται τον κάθε τρόπο αυθεντικοποίησης.

Και στις 2 εκδόσεις του X API (1.1 & 2.0) η αυθεντικοποίηση γίνεται με 2 τρόπους.
Αυθεντικοποίηση σε επίπεδο εφαρμογής με χρήση OAuth 2.0 Bearer token, κατά την οποία χρησιμοποιούνται τα κλειδιά και συμβολοσειρές που είναι μοναδικά αναγνωριστικά της εφαρμογής. Σε αυτό το είδος αυθεντικοποίησης γίνεται κλήση της κλάσης OAuthBearerHandler ή τη κλάσης OAuthAppsHandler.
Αυθεντικοποίηση σε επίπεδο χρήστη με χρήση OAuth1.0a User Context γίνεται με κλήση της κλάσης OAuth1UserHandler στην οποία θα πρέπει να εισαχθούν 4 στοιχεία: Το κλειδί πελάτη, το μυστικό κλειδί πελάτη, τη συμβολοσειρά πρόσβασης και τη μυστική συμβολοσειρά πρόσβασης.
Κατά τη συλλογή δεδομένων, αφού έχει επιτευχθεί η αυθεντικοποίηση και η διασύνδεση με την πλατφόρμα, είναι δυνατή η συλλογή δεδομένων με είτε μέσω παρακολούθηση της ροής των αναρτήσεων είτε με αναζήτηση με χρήση hashtags και λέξεων κλειδιών.
Στην ανάπτυξη της εργασιας χρησιμοποιήθηκαν και οι 2 τρόποι συλλογής χρηστών βάσει κλειδιών αναζήτησης αλλά και ροής δεδομένων με κριτήριο τη γεωγραφική περιοχή.

Περιγραφή	κλειδί	Λειτουργία
Api key	oauth_consumer_key	Το μοναδικό αλφαριθμητικό αναγνώρισης της εφαρμογής
Secret key	oauth_consumer_secret	Το αλφαριθμητικό πιστοποίησης της εφαρμογής από το API
Access token	oauth_token	Το αλφαριθμητικό αναγνώρισης του χρήστη
oauth_token_secret		Το αλφαριθμητικό πιστοποίησης ου χρήστη από το API

Πίνακας 2.1

Αίτηση ανάκτησης περιεχομένου προς το API με OAuth2.0:
Η μέθοδος πιστοποίησης OAuth2.0 παρέχει πρόσβαση μόνο στις εφαρμογές και μόνο για αναγνώσιμο περιεχόμενο. Η χρήση ενός αλφαριθμητικού κλειδιού γίνεται με την ενσωμάτωσή του στην αίτηση προς το API στην παρακάτω μορφή

```
curl "https://api.X.com/2/tweets?ids=1261326399320715264,1278347468690915330" \
  -H "Authorization: Bearer
AAAAAAAAAAAAAAAAAAAAAFnp2wAAAAAAxFdQpb%2HIBDtThABbyNJon%2BA72K4%
3DeIaigY0QBrv6Rp8KZQQLOTpo9ubw5Jt?WRE8avbi"
```

API v2
Ενώ στην 1η έκδοση του API, γινόταν πρώτα ανάκτηση όλων των πεδίων και μετά γινόταν φιλτράρισμα με παραμετροποίηση, στην έκδοση 2 τα μόνα πεδία που ανακτώνται είναι τα απαραίτητα id και text. Η ανάκτηση επιπλέον πεδίων γίνεται με χρήση νέας αίτησης η οποία φέρνει τα πρόσθετα επιθυμητά πεδία ή αντικείμενα και τα τοποθετεί στην αρχική αίτηση. Επίσης στην έκδοση 2 εφαρμόζονται περιορισμοί ώς προς το πλήθος των ανακτώμενων αντικειμένων μέσα στο χρόνο για λόγους ομαλής λειτουργίας και αποφυγής επιβάρυνσης του δικτύου του X.

Μέσω του API γίνεται ανάκτηση αντικειμένων, πεδίων και οντοτήτων. Στην εργασία έχει γίνει ανάλυση των αντικειμένων “Tweet” και “User” μέσω των πεδίων τους. Τα πεδία είναι οι παράμετροι ενός αντικειμένου που περιέχουν λεπτομέρειες όπως χρονοσήμανση, αναγνωριστικό χρήστη, αναγνωριστικό ανάρτησης.
Στον παρακάτω πίνακα γίνεται η παρουσίαση των παραμέτρων των 2 προαναφερθέντων αντικειμένων που χρησιμοποιήθηκαν στην εργασία. Παρατηρείται ότι κάποια πεδία είναι κοινά, ωστόσο αφορούν διαφορετικά αντικείμενα οπότε έχουν διαφορετική χρήση και σημασία.

Πεδίο	Περιγραφή	Λειτουργία
id (αλφαριθμητικό)	Το μοναδικό αναγνωριστικό χρήστη του X	Ο μοναδικός αριθμός της ανάρτησης τύπου string, χρησιμοποιείται ώτσε να

Παράδειγμα	"id": "1105101162898921287"	ανακτηθεί προγραμματιστικά.
text (αλφαριθμητικό)	Το κείμενο μιας ανάρτησης σε κωδικοσελίδα utf-8	Το κείμενο της ανάρτησης που μπορεί να κερματιστεί σε λέξεις-κλειδία και να γίνει λεξιλογική και συναισθηματική ανάλυση και κατηγοριοποίηση.
Παράδειγμα	"text": "Ο Παρθενώνας στολίζει τον ιερό βράχο της ακρόπολης της Αθήνας για πάνω από 2000 χρόνια”	
author_id	Το αναγνωριστικό χρήστη	Ο μοναδικός αριθμός τύπου αλφαριθμητικό για αναγνώριση ενός χρήστη
Παράδειγμα	"author_id": "637564"	
created_at date (ISO 8601)	Χρονικό αποτύπωμα	Το χρονικό αποτύπωμα τη ανάρτησης περιέχει την ακριβή ώρα και ημερομηνία δημιουργίας για ανάλυση βάσει χρόνου
Παράδειγμα:	"created_at":"2022-03-04T23:12:08.000Z"	

geo	Το γεωγραφικό αποτύπωμα μας ανάρτησης.
	
	"geo": {
	"coordinates": {
	"type": "Point",
	"coordinates": [
	-73.99960455,
	40.74168819
		Εφόσον έχει επιλεγεί από τις ρυθμίσεις, το γεωγραφικό αποτύπωμα τις συντεταγμένες για τον ακριβή τόπο του χρήστη όταν πραγματοποίησε την ανάρτηση σε JSON μορφή.

Παράδειγμα		

	] }, "place_id": "01a9a39529b27f36" }	
lang	Η γλώσσα στην οποία γίνεται η ανάρτηση. "lang": "en"	Η σήμανση της γλώσσας της ανάρτησης, εφόσον αναγνωριστεί από το X, με το πρότυπο BCP47. Κατηγοριοποιεί τις αναρτήσεις βάσει γλώσσας.

Πίνακας 2.2 Πεδία αντικειμένου "Tweet"

Ιδιότητα	Περιγραφή	Λειτουργία
id	Αναγνωριστικός αριθμός χρήστη (Int64)	Το μοναδικό αναγνωριστικό χρήστη σε μορφή ακέραιου αριθμού. Το μεγάλο το μέγεθος, αν και λύνεται με καταχώρισή του σε 64b προσημασμένη ακέραια μεταβλητή, καθιστά την αλφαριθμητική μορφή του πιο φιλική για τον προγραμματιστή "id": 6253282
id_str	Αναγνωριστικός αριθμός	Το μοναδικό

	χρήστη (αλφαριθμητικός)	αναγνωριστικό χρήστη σε μορφή αλφαριθμητικού. Προτείνεται η χρήση του καθώς μπορεί να χρησιμοποιηθεί σε όλες τις εφαρμογές χωρίς προβλήματα σε θέματα συμβατότητας ή μνήμης. "id_str": "6253282"
name	Αναγνωριστικό όνομα χρήστη	Το όνομα που θέλει ο χρήστης να φαίνεται στο X. Example: "name": "X API"
screen_name	Όνομα χρήστη	Το μοναδικό όνομα χρήστη με όριο τους 15 χαρακτήρες. Μπορεί να αλλάξει σε άλλο, εφόσον διατηρείται η αρχή της μοναδικότητας. Example: "screen_name": "Xapi"
location	String	Η προεραιτικά δηλωμένη στο προφίλ τοποθεσία του χρήστη. Δεν αφορά την τρεέχουσα τοποθεσία του χρήστη κατά την

		δημιουργία της ανάρτησης. Example: "location": "San Francisco, CA"
created_at	String	Ημερομηνία και ώρα που ο χρήστης δημιούργησε το λογαριασμό του σε UTC μορφή. "created_at": "Mon Nov 29 21:18:15 +0000 2010"

Πίνακας 2.3 Πεδία αντικειμένου "user"

Εργαλειοθήκη φυσικής γλώσσας (NLTK)

Η NLTK δημιουργήθηκε το 2001 και είναι ένα σύνολο προγραμματιστικών εργαλείων τα οποία βασισμένα στους κανόνες της ΕΦΓ, δίνουν τη δυνατότητα στη γλώσσα Python να επεξεργαστεί το φυσικό λόγο. Παρέχει πρόσβαση σε μεγάλο αριθμό πηγών (λεξικά και σώματα κειμένου) ενώ στηρίζεται από μια ευρεία κοινότητα ανάπτυξης και υποστήριξης.(Jablonsky J.)

Οι βιβλιοθήκες επεξεργασίας κειμένου και τα δομοστοιχεία που περιέχει καθιστούν αξιοποιήσιμες τις τεχνικές επεξεργασίας φυσική γλώσσας σε προγραμματιστικό περιβάλλον:

Παραδείγματα προγραμματιστικής αξιοποίησης του NLTK

Εργασία επεξεργασίας Φ.Γ.	**NLTK δομοστοιχεία**	**Λειτουργικότητα**
Πρόσβαση σε σώμα κειμένου	corpus	Τυποποιημένη διασύνδεση σε σώματα κειμένου και λεξικά
Επεξεργασία κειμένου	tokenize, stem	κερματισμοί λέξεων, φράσεων και δημιουργία ετυμολογικής ρίζας
Ανεύρεση παράθεση	collocations	Στατιστική ανάλυση t-test, chi-squared, PMI
Σήμανση μερών του λόγου	tag	Μέθοδοι n-gram, backoff, Brill, HMM, TnT
Μηχανική μάθηση	classify, cluster, tbl	Αλγόριθμοι δέντρου αποφάσεων, μέγιστης εντροπίας, naive Bayes, EM, k-means
Τμηματοποίηση	chunk	κανονικές εκφράσεις, ακολουθία n-gram, ονοματική οντότητα
Ανάλυση	parse, ccg	χαρτογράφησης, chart, βάσει χαρακτηριστικών, ενοποίησης, πιθανοτική, εξάρτησης
Σημασιολογική ερμηνεία	sem, inference	lambda calculus, first-order logic, model checking
Μετρήσεις αξιολόγησης	metrics	ακρίβεια, ανάκληση, συντελεστες συμφωνίας
Πιθανότητα και εκτίμηση	probability	κατανομές συχνότητας, κατανομές ομαλής πιθανότητας

Εφαρμογές	app, chat	γραφικό συμφραστικός πίνακας, αναλυτές, περιηγητής WordNet , chatbots
Γλωσσολογία πεδίου	toolbox	Διαχείριση δεδομένων με την εργαλειοθήκη της S.I.L

Πίνακας 2.4 Η Προγραμματιστική αξιοποίηση της NLTK.

Σήμανση μερών του λόγου (pos tagging)

Για την καλύτερη οργάνωση και διαχείριση του περιεχομένου, το NLTK χρησιμοποιει κατηγοριοποίηση μερών του λόγου. Έτσι επιτυγχάνεται η μεταφορά τους και η γλωσσική επεξεργασία τους σε προγραμματιστικό περιβάλλον για την εξαγωγή του επιθυμητού αποτελέσματος.

Συντομογραφία	Χρήση
CC	συντονιστικός σύνδεσμος
CD	Απόλυτος αριθμός
DT	προσδιοριστής
EX	Υπαρξιακό εκεί (there)
FW	ξένη λέξη
IN	πρόθεση/δευτερεύων σύνδεσμος
JJ	επίθετο
JJR	επίθετο, συγκριτικός
JJS	επίθετο, υπερθετικός
LS	λίστα αγοράς
MD	βοηθητικό ρήμα

NN	κοινό ουσιαστικό, ενικός
NNS	κοινό ουσιαστικό, πληθυντικός
NNP	Κύριο ουσιαστικό, ενικός
NNPS	Κύριο ουσιαστικό, πληθυντικός
PDT	προσδιοριστικό επίθετο
POS	ουσιαστικό κτητικής
PRP	προσωπική αντωνυμία
PRP$	κτητική αντωνυμία (her, his, mine, my, our)
RB	επίρρημα (occasionally, swiftly)
RBR	επίρρημα συγκριτικό, (greater)
RBS	επίρρημα υπερθετικό, (biggest)
RP	Γραμματικό μόριο (about)
TO	infinite marker (to)
UH	επίκληση (goodbye)
VB	Ρήμα (eat)
VBG	Γερούνδιο ρήμα (eating)
VBD	ρήμα αορίστου (ate)
VBN	Μετοχή αορίστου (eaten)
VBP	Ρήμα ενεστώτα 1ου, 2ου ενικού
VBZ	Ρήμα ενεστώτα όλων των βαθμών

WDT	wh-οριστική αντωνυμία (that, what)
WP	wh- αναφορική αντωνυμία (who)
WRB	wh- επίρρημα (how)

Πίνακας 2.5 Συντομογραφίες μερών του λόγου

Wordnet

Το WordNet είναι ένα διαδικτυακό λεξικό της αγγλικής γλώσσας που παρέχει θεματικές ομάδες λέξεων (συνώνυμα, αντώνυμα, υπερώνυμα κλπ.) και τις συνδέσεις μεταξύ τους. Χρησιμοποιείται ευρέως στον τομέα της φυσικής γλώσσας και της επεξεργασίας του κειμένου για να προσφέρει σημασιολογικές πληροφορίες και να βελτιώσει την κατανόηση του κειμένου και την ανάλυση των λέξεων. Το WordNet περιλαμβάνει επίσης πληροφορίες για τη συνώνυμη και αντωνυμική σχέση μεταξύ των λέξεων, καθώς και τον τρόπο με τον οποίο συσχετίζονται με άλλες λέξεις μέσω των σημασιολογικών τους συνδέσεων.

Ο ρόλος του είναι σημαντικός σε πολλές εφαρμογές, όπως η μηχανική μετάφραση, η ανάλυση συναισθήματος στον τομέα της ανάλυσης συναισθήματος του κειμένου, η αναζήτηση πληροφοριών, η κατανόηση του κειμένου από μηχανές, η αποτελεσματική ανάλυση των λέξεων και η προετοιμασία τους για διάφορες επεξεργασίες. Επιπλέον, χρησιμοποιείται ευρέως στην επιστήμη της ψυχολογίας, της κοινωνιολογίας, της γλωσσολογίας και της κοινωνικής δικτύωσης για την ανάλυση και την κατανόηση των λεξικών συσχετίσεων και των σημασιολογικών σχέσεων μεταξύ των λέξεων. Αποτελεί ένα πολύτιμο εργαλείο για την ανάλυση και την κατανόηση της σημασίας των λέξεων στο πλαίσιο της φυσικής γλώσσας και της σημασιολογίας, και παρέχει πολλαπλές δυνατότητες για την ανάπτυξη προηγμένων εφαρμογών στον τομέα της επεξεργασίας του κειμένου.

Η ιδιαιτερότητα του WordNet σε σχέση με ένα παραδοσιακό λεξικό είναι ότι κατατάσσει τους όρους σε πέντε βασικές κατηγορίες: ουσιαστικά, ρήματα, επίθετα, επιρρήματα και λέξεις λειτουργίας. Στο αγγλικό λεξιλόγιο, ωστόσο, χρησιμοποιούνται κυρίως οι τέσσερις πρώτες κατηγορίες, καθώς οι λέξεις λειτουργίας είναι σπάνιες, αν όχι ανύπαρκτες. Αυτή η κατηγοριοποίηση έχει το μειονέκτημα ότι ορισμένες λέξεις μπορεί να ανήκουν σε δύο κατηγορίες (παράδειγμα η αγγλική λέξη "front", που

μπορεί να είναι επίρρημα "μπροστά" ή ουσιαστικό "μέτωπο"). Ωστόσο, το πλεονέκτημα είναι ότι οι κατηγορίες είναι δομημένες με μια σημασιολογική οργάνωση που τις καθιστά διακριτές, παρέχοντας έτσι τη δυνατότητα συστηματικής αξιοποίησής τους.
Με αυτόν τον τρόπο δημιουργείται μια αρχή σημασιολογικής οργάνωσης στην οποία αντανακλούν οι λεκτικές δομές με διάφορες μορφές. Τα ουσιαστικά ως θεματικές ιεραρχίες, τα ρήματα ως στοιχεία συνεπαγωγικών σχέσεων και τα επιθετα και επιρρήματα έχουν πολυδιάστατο χαρακτήρα. Βάσει της αρχής αυτής είναι δυνατή η ανάλυση των λεξικών σχέσεων μεταξύ των λέξεων, όπως τα συνώνυμα, τα αντώνυμα, τα υπερώνυμα και τα υπώνυμα, προσφέροντας έτσι ένα πλούσιο λεξιλόγιο για σημασιολογική ανάλυση και επεξεργασία του κειμένου.

Επιπλέον, το WordNet παρέχει πληροφορίες σχετικά με το πώς οι λέξεις συσχετίζονται μεταξύ τους σε ένα δίκτυο σημασιολογικών σχέσεων. Αυτό επιτρέπει την ανάλυση της σημασίας των λέξεων μέσω των συνδέσεών τους με άλλες λέξεις, και μπορεί να χρησιμοποιηθεί για την ανάπτυξη σημασιολογικών εφαρμογών όπως το σημασιολογικό ανάλυση κειμένου, το συναντήσεων και το ανάλυση συναισθηματικού περιεχομένου.

3ο ΚΕΦΑΛΑΙΟ

Ανάπτυξη της εργασίας

Στο κεφάλαιο αυτό αναλύονται τα βήματα της ανάπτυξης της εργασίας, τα οποία περιλαμβάνουν τη διασύνδεση με το API του X, τη δημιουργία και τον σχεδιασμό βάσης δεδομένων, την ανάπτυξη κώδικα για την ανάκτηση δεδομένων των αναρτήσεων χρηστών του X, καθώς και την ανάπτυξη κώδικα για την επεξεργασία των δεδομένων και την εξαγωγή αποτελεσμάτων βάσει κριτηρίων προτίμησης των χρηστών.

Διασύνδεση με τις λειτουργίες προγραμματιστή του X

Για τη διαδικασία διασύνδεσης με την πλατφόρμα έγινε χρήση του X API. Η διασύνδεση επιτρέπει στον προγραμματιστή την αξιοποίηση των εργαλείων που προσφέρει το X μέσω της βιβλιοθήκης tweepy

Ο παρακάτω κώδικας περιέχει τα διαπιστευτήρια πρόσβασης (API credentials) που απαιτούνται για την αυθεντικοποίηση και τη σύνδεση με το API του X, ώστε να είναι δυνατή η ανάκτηση των απαραίτητων δεδομένων από το X μέσω του αντίστοιχου κώδικα που θα αναπτυχθεί και περιέχει τα παρακάτω στοιχεία για τη διασύνδεση με το API του X:

Η κλάση "authentication" χρησιμοποιείται για τη διαχείριση των διαπιστευτηρίων (credentials) πρόσβασης στο API του X.

Οι μεταβλητές "consumer_key", "consumer_secret", "access_token", "access_token_secret" περιέχουν τα διαπιστευτήρια (credentials) πρόσβασης στο API του X, όπως το κλειδί του καταναλωτή (consumer key), το μυστικό κλειδί του καταναλωτή (consumer secret), το κλειδί πρόσβασης (access token) και το μυστικό κλειδί πρόσβασης (access token secret).

Οι μεθόδοι "getconsumer_key()", "getconsumer_secret()", "getaccess_token()", "getaccess_token_secret()" επιστρέφουν τις τιμές των διαπιστευτηρίων πρόσβασης στο API του X από τις αντίστοιχες μεταβλητές τους.

```
class                                                    authentication:

      def                                                      __init__(self):

            self.consumer_key                ="c532YnnqRwaQBMDd1vT7VWtwX"

      self.consumer_secret="ypxfs5PTyZxDavfn9kGFHAE6pN4CC1IiyrmJQLK8LJ
Hx49YOOA"

            self.access_token="3405775702-
XlxH1XnIdoPXykWqwvX8ODjaAmNOxa5LIprprRJ"

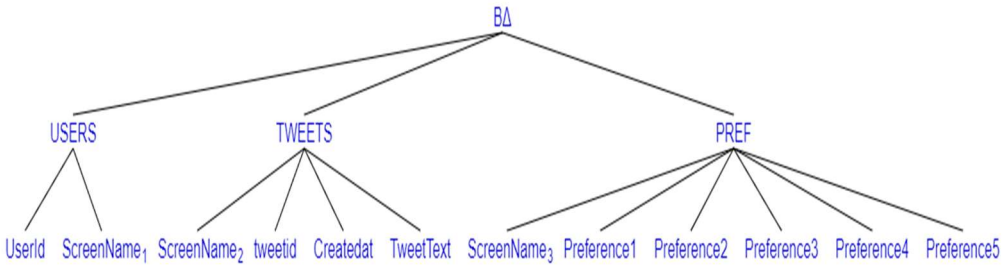
      self.access_token_secret="8IKa3Fb36VHPVrtWV0vGYliPXYnhZcoqn9YaMk
CAqZecY"

      def                                          getconsumer_key(self):
            return                                      self.consumer_key

      def                                       getconsumer_secret(self):
            return                                    self.consumer_secret
      def                                         getaccess_token(self):
            return                                      self.access_token
      def                                  getaccess_token_secret(self):
            return self.access_token_secret
```

Σχεδιασμός της Βάσης Δεδομένων

Σε όλη τη διαδικασία εκτέλεσης της εφαρμογής τα στοιχεία όπως ο χρήστες, τα αναγνωριστικά και τα tweets αποθηκεύτηκαν σε μια σχεσιακή βάση δεδομένων δύο πινάκων για την επεξεργασία Users και Tweets, και ενός πίνακα Pref για αποθήκευση των αποτελεσμάτων, όπως περιγράφονται από το παρακάτω σχεδιάγραμμα.



Ο ρόλος των πινάκων και των πεδίων τους περιγράφεται αναλυτικά παρακάτω με τις χρήσεις των πεδίων:

Πίνακας USERS: Ο πίνακας αυτός χρησιμοποιείται στην καταχώριση των στοιχείων του χρήστη, και αξιοποιείται στο τελευταίο στάδιο του προγράμματος αξιολόγησης και κατηγοριοποίησης των προτιμήσεων κάθε χρήστη.

Πεδία:

UserID: Ο μοναδικός αναγνωριστικός αριθμός χρήστη που δημιουργεί το Χ.

ScreenName: Το όνομα του χρήστη

Πίνακας Tweets: Σε αυτό τον πίνακα καταχωρούνται τα στοιχεία και το περιεχόμενο των tweet για συγκεκριμένο χρήστη.

Πεδία:

ScreenName: Το όνομα χρήστη στο οποίο θα πιστωθούν τα tweets που θα συγκεντρωθούν από τον κώδικα αναζήτησης και παρακολούθησης,

TweetId: Το μοναδικό αριθμητικό αναγνωριστικό κάθε ανάρτησης στο Χ

CreatedAt:Η ημερομηνία και ώρα ανάρτησης.

TweetText: Το περιεχόμενο της ανάρτησης όπως αυτό ορίζεται από τους εκάστοτε κανόνες της πλατφόρμας.

Πίνακας PREF: Είναι ο πίνακας των αποτελεσμάτων της επεξεργασμένης πληροφορίας από τα δεδομένα που έχουν συλλεχθεί για κάθε χρήστη.

Πεδία: ScreenName για το όνομα χρήστη

Preference 1 έως 5, τα πεδία των 5 επικρατέστερων προτιμήσεων του χρήστη.

Συλλογή δεδομένων από το X

Με τη χρήση του Tweepy είναι εφικτή η πρόσβαση και αλληλεπίδραση με το API του X. Μία από τις λειτουργίες που προσφέρει είναι η δυνατότητα αναζήτησης με χρήση μιας κλάσης listener για συγκεκριμένη γεωγραφική περιοχή.

Η λειτουργία αυτή είναι χρήσιμη όταν η συλλογή δεδομένων βασίζεται σε αναρτήσεις του X που σχετίζονται με συγκεκριμένη γεωγραφική περιοχή, όπως μια πόλη, μια χώρα ή μια καθορισμένη περιοχή.

Για να γίνει χρήση της λειτουργίας αυτής του Tweepy, θα χρειαστεί να οριστεί μια κλάση listener, η οποία θα κληρονομεί από την βασική κλάση tweepy.StreamListener και θα υλοποιεί τη μέθοδο on_status(), η οποία θα εκτελείται κάθε φορά που λαμβάνεται ένα νέο tweet από το X.

Μέσα στη μέθοδο *on_status()*, μπορεί να διαμορφωθεί ο listener ώστε να φιλτράρει τα tweets που ανακτώνται με βάση τη γεωγραφική τους τοποθεσία. Στο συγκεκριμένο παράδειγμα, η παράμετρος stream_loc που καθορίζεται από συντεταγμένες γεωγραφικής χωροθέτησης, χρησιμοποιείται κατά την εκκίνηση του αντικειμένου της κλάσης listener. Για τον καθορισμός της γεωγραφικής χωροθέτησης χρησιμοποιείται το διαδικτυακό εργαλείο bboxfinder.com

```
stream_loc = [-11.074218749999998, 49.781264058178344, 2.373046875, 59.62332522313024]
```

Τα αποτελέσματα της ροής περιλαμβάνουν τα στοιχεία χρήστη, της ανάρτησης και του περιεχομένου σε μορφή json. Μόνο τα στοιχεία χρήστη αποθηκεύονται επί του παρόντος σε βάση δεδομένων.

```
class OutListener(StreamListener):

    def on_status(self, status):
        print(status.text)
        name = status.user.screen_name
        print(name)

    def on_data(self, data):
        try:
            rawdata = json.loads(data)
            user_id = rawdata['user']['id']
            screen_name = rawdata['user']['screen_name']
            text = rawdata['text']
            created_at = parser.parse(rawdata['created_at'])
```

```
            location = rawdata['user']['location']
myListener = OutListener()
    stream = Stream(auth=api.auth, listener=myListener)

    while True:
        try:
            stream.filter(locations=stream_loc, is_async=True)
```

Αναζήτηση με λέξεις κλειδιά - hashtags

Η δυνατότητα αυτή του Tweepy είναι ένα ισχυρό εργαλείο που παρέχεται από το API του X, το οποίο επιτρέπει στους χρήστες να αναζητούν και να αποκτούν πρόσβαση σε δημοσιεύσεις και σχετικά δεδομένα που σχετίζονται με συγκεκριμένες λέξεις κλειδιά ή hashtags στο X.

Η διαδικασία αρχίζει με τη δημιουργία ενός αντικειμένου της κλάσης tweepy.API, το οποίο είναι η διεπαφή προγραμματισμού εφαρμογών (API) του X που χρησιμοποιεί το Tweepy για να επικοινωνήσει με το X. Στη συνέχεια, γίνεται κλήση της μεθόδου search του αντικειμένου API για να γίνει αναζήτηση με βάση τις λέξεις κλειδιά ή τα επιθυμητά hashtags. Το όνομα χρήστη αποθηκεύεται σε βάση δεδομένων.

```
api = tweepy.API(auth, wait_on_rate_limit=True,
                 wait_on_rate_limit_notify=True)

preferred_hashtags = ["#Chania", "Χανιά","#Crete"]
while 1 == 1:
    for hash in preferred_hashtags:
        search_hashtag(listoftweets=[],                keyword=hash,
numoftweets=10000)
    time.sleep(60*60*24)  # repeat every 24 hours
```

Επεξεργασία των αναρτήσεων κάθε χρήστη

Καθορισμός θετικών tweets

Για την ανάλυση των θετικών αναρτήσεων, επιλέχθηκε η χρήση θετικών επιρρημάτων και επιθέτων που συνοδεύουν τις αναρτήσεις. Με αυτόν τον τρόπο, απορρίπτονται αυτές που περιέχουν αρνητικό νόημα και ενδέχεται να είναι κακόβουλες, επικριτικές, δυσφημιστικές ή να προωθούν αρνητική διαφήμιση. Αυτή η μέθοδος είναι σχετικά απλή, προσφέρει άμεσα και αξιόπιστα αποτελέσματα για τον

σκοπό της εργασίας. Οι λίστες των θετικών μερών του λόγου αποθηκεύονται στα αρχεία "positive_adj.py", "positive_adv.py" και διαβάζονται τα περιεχόμενά τους.
Τα περιεχόμενα διαχωρίζονται σε λίστες θετικών επιθέτων και επιρρημάτων και στη συνέχεια, οι λίστες συνενώνονται και ταξινομούνται.

```
with open('positive_adj.py') as pos_adj_file:
    pos_adj = pos_adj_file.read().split(',')
    pos_adj_file.close()

with open('positive_adv.py') as pos_adv_file:
    pos_adv = pos_adv_file.read().splitlines()
    pos_adv_file.close()

positives = sorted(pos_adj + pos_adv)
```

Ορίζεται η συνάρτηση "tweet_process" με παραμέτρους το tweet και μια κενή λίστα "t_clean_en" που θα αποθηκεύει τα αποτελέσματα ως παραμέτρους.

```
def tweet_process(tweet, t_clean_en):
```

Μέσα στη συνάρτηση, πριν τον κερματισμό, με τη χρήση κανονικών εκφράσεων γίνεται καθαρισμός των αναρτήσεων από περιττά ή μη επιθυμητά στοιχεία όπως συνδέσμους, retweets, αναφορές σε χρήστες, σημειώσεις hashtags, αριθμούς και τελείες.

```
tweet = re.sub(r'https?:\/\/.*[\r\n]*', '', tweet)
# Αφαίρεση συνδέσμων (URLs) από το κείμενο του tweet
tweet = re.sub(r'^RT[\s]+', '', tweet)
# Αφαίρεση retweets
tweet = re.sub('@[^\s]+', '', tweet)
# Αφαίρεση αναφορών σε άλλους χρήστες στο X (mentions)
tweet = re.sub('@[^\w\s]', '', tweet)
# Αφαιρεση του συμβόλου "@" από hashtags και σύμβολα.
tweet = re.sub(r'\#\w*', '', tweet)
# Αφαίρεση των hashtags από το tweet. Τα hashtags ξεκινούν με το σύμβολο
"#" και ακολουθούνται από οποιοδήποτε αλφαριθμητικό.
tweet = re.sub(r'\d', '', tweet)
```

```
# Αφαίρεση αριθμών
tweet = re.sub(r'\.+', "", tweet)
# Αφαίρεση τελείων που εμφανίζονται σε συνεχόμενη ακολουθία.
```

Στο τέλος της εκτέλεσης αυτών των γραμμών κώδικα, το tweet θα έχει υποστεί τις απαραίτητες αλλαγές και θα έχουν αφαιρεθεί τα προαναφερθέντα στοιχεία. Ο κώδικας χρησιμοποιεί την συνάρτηση re.sub() από τη βιβλιοθήκη re της Python για να εκτελέσει τις αντικαταστάσεις κειμένου βάσει κανόνων που ορίζονται από τις κανονικές εκφράσεις (regular expressions).

```
pos_lem = WordNetLemmatizer()
tTok = TweetTokenizer()
```

Στη συνέχεια ο κώδικας διαχωρίζει την ανάρτηση σε λέξεις χρησιμοποιώντας έναν tokenizer που είναι αποθηκευμένος στην "tTok.tokenize()". Οι λέξεις που προκύπτουν αποθηκεύονται στη μεταβλητή "word_tok" στην οποία εφαρμόζεται η συνάρτηση "pos_tag" για να ετικετοποιηθούν οι λέξεις με τις ετικέτες μερών του λόγου και δημιουργείται μια νέα λίστα με λέξεις που ανήκουν σε κατηγορίες μερών του λόγου που αρχίζουν με "NN" (ουσιαστικά), "RB" (επιρρήματα) ή "JJ" (επίθετα).

```
word_tok = tTok.tokenize(tweet)

tags = pos_tag(word_tok)
pos_all = [word for word, pos in tags if (pos.startswith('NN') or
pos.startswith('RB') or pos.startswith('JJ'))]
```

Εξαγωγή θετικών λημμάτων

Γίνεται έλεγχος για τη γλώσσα στην οποία είναι γραμμένο κάθε tweet και αν αυτή είναι τα Αγγλικά, τότε συνεχίζεται η εκτέλεση του κώδικα. Κατόπιν γίνεται έλεγχος αν η ανάρτηση περιέχει θετικά επίθετα ή επιρρήματα και αν αυτό ισχύει τότε επιλέγονται τα ουσιαστικά που με τη σειρά τους ελέγχονται για stopwords, υφίστανται λημματοποίηση και επιστρέφεται η επεξεργασμένη λίστα λέξεων.

```
stop_words_en = set(stopwords.words("english"))
```

```
   try:
       if langdetect.detect(tweet) == "en":
           for pos_w in pos_all:
               wnl = WordNetLemmatizer()
               if pos_w in positives:

                   pos_noun = [pos_w for pos_w,pos in tags if
(pos.startswith('NN'))]
                   for w in pos_noun:
                       if w not in stop_words_en and w not in
string.punctuation:
t_clean_en.append(wnl.lemmatize(w))
```

Κατηγοριοποίηση και επιλογή των 5 δημοφιλέστερων προτιμήσεων των χρηστών

Για την κατηγοριοποίηση έχει προηγηθεί η δημιουργία κώδικα ο οποίος διαβάζει ένα αρχείο κειμένου που περιέχει κατηγορίες και λέξεις κλειδιά που αφορούν την κάθε κατηγορία και αποθηκεύει τα δεδομένα σε ένα λεξικό category_dict. Οι 9 κατηγορίες είναι οι εξής: Arts, astronomy, beach, castles, dance, music, minerals, sports, summer.

Κάθε γραμμή του αρχείου αντιστοιχεί σε ένα κλειδί και μια τιμή στο λεξικό, με τις κενές γραμμές να αποτελούν το σημείο τερματισμού της ανάγνωσης. Το αποτέλεσμα είναι ένα λεξικό που περιέχει τις κατηγορίες και τις αντίστοιχες τιμές από το αρχείο.

```
path = ".../main_data.txt"
category_dict = {}

with open(path) as cat_file:
   while True:
       category = cat_file.readline()
       elements = cat_file.readline()
       if category == '':
           break
```

```
        category_dict[category.strip()] =
[str(classification.strip()) for classification in
elements.strip().split(',')]
    print(category_dict)
```

Κατόπιν, η συνάρτηση tweet_selector() επεξεργάζεται τις αναρτήσεις κάθε χρήστη με τη βοήθεια ενός μετρητή που παίζει διπλό ρόλο: αφενός μετράει το πλήθος των αναρτήσεων του χρήστη, αφετέρου χρησιμοποιείται στον έλεγχο των αναρτήσεων με τέτοιο τρόπο έτσι ώστε ο κώδικας να προσπερνάει τους χρήστες που δεν εμπίπτουν στις κατηγορίες ενδιαφέροντος. Ο κώδικας στο τελικό στάδιο ανιχνεύει τις προτιμήσεις τους με βάση μια λεξική κατηγοριοποίηση (category_dict), και αποθηκεύει τις πέντε κορυφαίες προτιμήσεις του στη βάση δεδομένων.

```
def tweet_selector():

        counter = 0
        for tweet in fetchall_usertweets:
            tweet_process(tweet, t_clean_en)
            counter += 1

        usercats = {}
        for k in category_dict.keys():
            usercats[k] = 0
        for word in t_clean_en:
            for k in category_dict.keys():
                for v in category_dict[k]:
                    if v == word:
                        usercats[k] = 1 + usercats[k]
        my_keys = sorted(usercats, key=usercats.get,
reverse=True)[:5]
        if counter > 0:
            query_pref = "INSERT IGNORE INTO PREF (`ScreenName`,
`Preference1`, `Preference2`,
`Preference3`,`Preference4`,`Preference5`) Values (%s, %s, %s, %s,
%s, %s)"
            cursor1 = con.cursor()
```

```
            cursor1.execute(query_pref,       (user,       my_keys[0],
my_keys[1], my_keys[2], my_keys[3], my_keys[4]))
            cursor1.close()
            con.commit()
```

Η κατηγοριοποίηση γίνεται με τη χρήση αρχείων κειμένου τα οποία περιέχουν ουσιαστικά που αναφέρονται σε συγκεκριμένη κατηγορία. Ένα παράδειγμα, το αρχείο beach.txt για την κατηγορία των καλοκαιρινών διακοπών που περιέχει μέσα ουσιαστικά όπως “summer, sand, vacation, holidays”. Μετά τον καθαρισμό, ο κώδικας συγκρίνει τα προκυπτόμενα ουσιαστικά με τις λέξεις στα αρχεία κατηγοριών και για κάθε επιτυχία, αυξάνει την αναφορά κατηγορίας κατά 1.

4ο ΚΕΦΑΛΑΙΟ

Αποτελέσματα - Συμπεράσματα

Από την τελική επεξεργασία των δεδομένων από το X θα προκύψουν οι 5 δημοφιλέστερες προτιμήσεις του χρήστη, ενδεικτικά αποτελέσματα παρατίθενται παρακάτω από την επεξεργασία των tweets του χρήστη 0lympicker:

Processing user: 0lympicker

Tokenized tweet:
['RT', 'RT', 'Open', 'Port', 'A', 'Service', 'OPaaS', 'option', 'port', '…', 'RT', 'RT', 'Open', 'Port', 'A', 'Service', 'OPaaS', 'option', 'port', '…', 'Room', 'btw', 'Linux', 'Backdoors', 'room', 'review', 'New', 'Europol', 'threat', 'assessment', 'IS', '’', 'battle', 'relevance', 'Europol', '▪', 'Islamic', 'State', 'al-Qaeda', 'collision', 'course', '…', 'Set', 'Bacchanale', 'Ruin', 'Head', 'Medusa', 'Landscape', 'Head', "Michelangelo's", 'Giuliano', 'Medici', 'Gary', 'Moore', 'Got', 'Blues', 'Live', 'Madly', 'New', 'Stereo', 'Mix', 'Microservices', 'Architecture', 'X', 'Service', 'New', 'Haskell', 'Phrasebook', 'page', 'mapping', 'enum', 'type', 'string', '…', 'Set', 'params', 'IPv', 'packet', 'anyone', 'port', 'memset', 'server_addr', ');', 'server_addrsin_addrs_addr', 'htonl', 'INADDR_ANY', ');', 'Michael', 'Jackson', 'Billie', 'Jean', 'Official', 'Video', '✓', 'Data', 'Structures', 'Algorithms', 'repository', '✓', '', 'FAANG', 'Questions', '✓', '', 'Easy', '✓', '', 'Clean', 'Code', 'Star', 'repositor', '…', '✓', 'Data', 'Structures', 'Algorithms', 'repository', '✓', '', 'FAANG', 'Questions', '✓', '', 'Easy', '✓', '', 'Clean', 'Code', 'Star', 'repositor', '…', '✓', 'Data', 'Structures', 'Algorithms', 'repository', '✓', '', 'FAANG', 'Questions', '✓', '', 'Easy', '✓', '', 'Clean', 'Code', 'Star', 'repositor', '…', 'improvement', 'Linux', 'Kernel', 'Defence', 'Map', 'showing', 'Vulnerability', 'class', 'Exploitation', 'technique', 'Bug', 'detection', 'mech', '…', 'Linear', 'Algebra', 'Deep', 'Learning', 'Machine', 'Learning', 'Python', 'Tutorial', 'Linear', 'Algebra', 'Data', 'Science', 'Machine', 'Learning', 'TFW', 'Hello', 'World', 'app', 'Windows', 'Learn', 'C', 'Universal', 'Windows', 'UWP', 'app', 'ht', '…', 'Ruth', 'Bader', 'Ginsburg', 'woman', 'Associate', 'Justice', 'US', 'Supreme', '…', 'Ruth', 'Bader', 'Ginsburg', 'woman', 'Associate', 'Justice', 'US', 'Supreme', '…', 'Nobel', 'Laureate', 'Malala', 'Yousafzai', 'speech', 'access', 'education', 'Unit', '…', '“', 'book', '—', 'Ursula', 'K', 'Le', 'Guin', 'author', 'Berkeley', '…', 'August', 'A', 'Jewish', 'girl', 'Marie', 'Krajzelman', 'Paris', 'July', 'Day', 'Today', 'question', 'graph', 'code', 'year', 'program', 'Afghan', 'woman', 'journalist', '…', 'Cool', 'Future', 'Codex', 'English', 'Into', 'Code', 'Mind', 'Material', 'Robot', 'Strong', 'Artificial', 'Intelligence', 'luck', 'Web', 'App', 'Hacking', 'Burp', 'Suite', 'Part', 'Testing', 'Persistent', 'XSS', 'Web', 'Apps', '…', 'I', 'Create', 'A', 'Pong', 'Game', 'MUSIC', 'Video', 'Artist', 'Name', '…', 'morning', '❤', '', 'Christians', 'Iran', 'year', 'faith', 'Iran', '’', 'system', '…', 'idea', 'OK', 'idea', 'wjat', 'donation', 'troll', '…', 'source', 'project', 'woman', 'Please', 'replie', '…', 'Stand', 'Up', 'Victor', 'Yeimo', 'Stand', 'Up', 'Victor', 'Yeimo', "Let's", 'Support', 'Special', 'Autonomy', 'Welfare', 'Papua', 'Python', 'Basics', 'Hackers', 'Part', 'Exact', 'Location', 'Any', 'IP', 'Address', '…', 'Linux', 'server', 'look', 'Best', 'Tools', 'Your', 'Linux', 'Server', 'Malware', 'Security', 'Flaws', 'http', '…', 'story', '…', 'SpiderFoot', 'Open', 'Source', 'Intelligence', 'OSINT', 'Automation', 'Utilizes',

'Various', 'Data', 'source', 'Range', 'Methods', 'Data', 'A', '…', 'Birthday', 'Erwin', 'Schrödinger', 'founder', 'quantum', 'theory', 'father', 'Schrödinger', 'equa', '…', 'Python', 'DevOps', 'Learn', 'Ruthlessly', 'Effective', 'Automation', 'Link', '…', 'Quote', 'Day', 'Free', 'speech', 'freedom', 'Brutus', 'Exploitation', 'Framework', 'Shipped', 'A', 'Modular', 'Highly', 'Extensible', 'Multi-Tasking', 'Multi-Processing', 'Ar', '…', 'Brutus', 'Exploitation', 'Framework', 'Shipped', 'A', 'Modular', 'Highly', 'Extensible', 'Multi-Tasking', 'Multi-Processing', 'Ar', '…', 'Hacking', 'Backdoor', 'Legitimate', 'Android', 'App', 'http', ':/', '…', 'Japan', 'Exploding', 'Rocket', 'Engine', 'Time', 'Space', 'al-Asaad', 'heroic', 'archaeologist', 'ISIS', 'August', '…', 'power', 'Just', 'meth', '…', 'suggestion', 'resource', 'CommanderGirl', ':)', 'Thank', 'RT', 'RT', 'How', 'Full', 'Stack', 'Developer', 'Retweet', 'share', 'information', '…', 'New', 'detail', 'photo', 'Telegram', 'identity', 'GhostEzra', 'operator', '‘', 'larg', '…', 'Dear', 'developer', 'developer', 'engineer', 'Vintage', 'footage', 'New', 'York', 'K', 'Upgrade', 'city', 'Soloi', 'CyprusVienna', 'Museum', 'Greek', 'Late', 'Classical', 'BC', 'Marble', 'Sarcophagus', '…', 'morning', 'friend', 'week', 'Kate', 'Moss', 'Polaroids', 'Michel', 'Haddi', 'Africans', 'Atlantic', 'Ocean', 'th', 'th', 'century', 'Today', 'International', '…', 'taste', 'Me', '', 'Morning', 'God', 'Bless', 'Greatest', 'Love', 'Story', 'Whitney', 'Houston', 'Dance', 'Somebody', 'Official', 'Video', 'Open', 'Source', 'Intelligence', 'OSINT', '):', 'Finding', 'Breached', 'Email', 'Addresses', 'Passwords', 'Credentials', 'Eurythmics', 'Annie', 'Lennox', 'Dave', 'Stewart', 'Dreams', 'Are', 'Made', 'Eurythmics', 'Annie', 'Lennox', 'Dave', 'Stewart', 'Dreams', 'Are', 'Made', 'Keimpx', 'Check', 'Valid', 'Credentials', 'A', 'Network', 'Over', 'SMB', 'World', 'Thodoris', 'Triantafillou', 'Cj', 'Jeff', 'Remix', 'Amy', 'Winehouse', "I'm", 'No', 'Good', 'Live', 'Acoustic', 'Amy', 'Winehouse', "I'm", 'No', 'Good', 'Live', 'Acoustic', 'Hackers', 'Exotic', 'Programming', 'Languages', 'Malware', 'Development', '…', 'Amy', 'Winehouse', "I'm", 'No', 'Good', 'Live', 'Acoustic', 'Amy', 'Winehouse', "I'm", 'No', 'Good', 'Live', 'Acoustic', 'Open', 'Source', 'Intelligence', 'OSINT', '):', 'Shodan', "World's", 'Most', 'Dangerous', 'Search', 'Engine', '…', 'Open', 'Source', 'Intelligence', 'OSINT', '):', 'Shodan', "World's", 'Most', 'Dangerous', 'Search', 'Engine', '…', 'Earthquake', 'Help', '…', 'Rammstein', 'Du', 'Hast', 'Official', 'Video', 'Sacred', 'Heart', 'Ronnie', 'James', 'Dio', 'A', 'shout', 'wizard', 'sky', '…', 'Scorpions', 'Rock', 'A', 'Hurricane', 'Official', 'Video', 'Great', 'history', 'country', 'Egypt', 'program', 'data', 'sale', '’', 'New', 'document', 'Illinois', 'DOT', 'year', 'location', 'trace', '…', 'NEW', 'speaks', 'Progressive', 'International', 'delegation', 'Brazil', '–', 'threat', '…', 'tornado', 'footage', 'Huludao', 'Province', 'afternoon', 'NetRipper', 'Smart', 'traffic', 'penetration', 'tester', 'Short', 'Guide', '…', 'Monitor', 'Live', 'X', 'Discussions', 'Maltego', 'Disinformation', 'Attacks', 'Tutorial', 'Great', 'OSINT', 'tool', 'X', 'Twint', 'Open', 'Source', 'Intelligence', 'OSINT', 'Part', 'Intelligence', 'X', '…', 'arm', 'Lyrics', 'Crew', "I've", 'mistake', '…', 'Tweep', 'Advanced', 'X', 'Tool', 'Tool', 'Social', 'Media', 'Instagram', 'Facebook', 'Snapchat', 'Github', 'X', 'Geotweet', 'Social', 'engineering', 'tool', 'hacking', 'way', 'X', 'Geotw', 'female', 'name', 'boy', 'yah', 'tweet', 'yah', 'oda', 'acc', 'Sparta', 'da', 'Best', 'OSINT', 'Tracking', 'Flight', 'Aircraft', 'Data', 'Open', 'Source', 'Tools', 'ht', '…', 'morning', 'welcome', 'tea', 'Friday', 'A', 'hug', 'penny', 'ground', 'Haddaway', 'Love', 'Special', 'Extended', 'Mix', 'problem', 'code', 'goddess', 'victory', 'name', 'Winged', 'Goddess', 'daughter', 'Ti', '…', 'Nice', 'catch', 'Master', 'Programmer', 'trainer', 'programmer', 'league', '…', 'Secure', 'Coding', 'Handbook', 'Pitto', 'Shadows', 'Original', 'Mix', 'rush', 'emotion', 'moment', 'word', '…', 'astsu', 'Simple', 'Network', 'Scanner', 'ht', '…', 'technology', 'solution', 'Design', 'Driven', 'Technology', 'Innovations', 'Terralogic', 'world-class', '…', 'Open', 'Source', 'Intelligence', 'OSINT', 'Tracking', 'Marine', 'Traffic', 'World', '…', 'Hurricane', 'IDA', 'IDA', 'Pro', 'Wow', 'project', 'import', 'ex', '…', 'Open', 'Source', 'Intelligence', 'OSINT', '):', 'Shodan', "World's", 'Most', 'Dangerous', 'Search', 'Engine', '…', 'Open', 'Source', 'Intelligence', 'OSINT', '):', 'Shodan', "World's", 'Most', 'Dangerous', 'Search', 'Engine', '…', 'tip', 'IDA', 'Freeware', 'binary', 'Chris', "Eagle's", 'blc', "Ghidra's", 'decompiler', '…', 'Reverse', 'Engineering', 'Malware', 'Part', 'Getting', 'Ida', 'Pro', 'Tina', 'Turner', 'Steamy', 'Windows', 'Live', 'Barcelona',

'Tina', 'Turner', 'Steamy', 'Windows', 'Live', 'Barcelona', 'Tina', 'Turner', 'Steamy', 'Windows', 'Great', 'article', 'history', 'Ada', 'Lovelace', 'Grace', 'Hopper', 'Hedy', 'Lamarr', 'Annie', 'E', '…', 'Cute', 'Fluffy', 'Cat', '', 'Environmant', 'Borland', 'C', 'Good', 'Old', ':-)', 'Port', 'Viewer', 'tool', 'port', 'GUI', 'support', '…', 'Empusa', 'pennata', 'AKA', 'conehead', 'mantis', 'specie', 'Photo', 'Mehmet', 'Karaca', 'Burp', 'Suite', 'Pentester', 'Scan', '…', 'ReconSpider', 'Advanced', 'Open', 'Source', 'Intelligence', 'OSINT', 'Framework', 'IP', 'Addresses', 'Emails', 'Websites', 'Organizati', '…', 'ReconSpider', 'Advanced', 'Open', 'Source', 'Intelligence', 'OSINT', 'Framework', 'IP', 'Addresses', 'Emails', 'Websites', 'Organizati', '…', 'Vuejs', 'framework', 'popularity', 'Than', 'Or', 'photon', 'factory', 'paper', 'resource-efficient', 'protocol', '…', 'Cryptography', 'CICADA', 'Periodic', 'Table', 'Elements', 'Luck', 'algorithm', 'C', '…', 'Cryptography', 'CICADA', 'Periodic', 'Table', 'Elements', 'Luck', 'algorithm', 'C', '…', 'ORNL', 'Develops', 'Advanced', 'Software', 'Framework', 'Programming', 'half', 'world', '’', 'specie', 'risk', 'extinction', 'Most', 'Beautiful', 'Most', 'Beautiful', "Society's", 'emergency', 'relief', 'family', 'Haiti', 'need', 'food', 'shelter', 'assista', '…', 'Open', 'Source', 'Intelligence', 'Part', 'Mining', 'Intelligence', 'Twitter', 'Chill', 'Hip', 'Hop', 'demo', 'M', 'M', 'beat', '✨', 'Open', 'source', 'investment', 'portfolio', 'analysis', 'tool', 'Python', '…', 'Cool', 'clojure-lsp', 'amount', 'reference', 'amount', 'test', 'var', '…', 'Markov', 'Model', 'Pussycat', 'Dolls', 'Hush', 'Hush', 'Hush', 'Hush', 'Official', 'Music', 'Video', 'Dire', 'Straits',
'Sultans', 'AMAZING', 'SOLO', 'Mark', 'Knopfler', 'B', 'Day', 'Today', 'Git', 'Github', 'colla', '…', 'thanks', 'Peter', 'share', 'inclusion', 'board', 'vehicle', '…', 'Modern', 'Talking', "You're", 'My', 'Heart', "You're", 'My', 'Soul', 'Official', 'Music', 'Video', 'Modern', 'Talking', "You're", 'My', 'Heart', "You're", 'My', 'Soul', 'Official', 'Music', 'Video', 'world', 'Handmaid', '’', 'Tale', 'TV', 'Texas', 'woman', 'friend', 'family', 'doctor', 'Uber', 'dri', '…', 'Today', 'moving', 'car', 'h', '…', 'Ask', 'Associated', 'Performer', 'Sertab', 'Duet', 'Mando', 'Composer', 'Lyricist', 'Aysel', 'Gürel', 'Lyricist', 'Demir', 'Demirkaan', 'Producer', 'Yiannis', 'Doulamis', 'Lyricist', 'Eleni', 'Gianatsoulia', 'Honeymoon', 'Song', 'Live', 'BBC', 'Pop', 'Go', 'Beatles', 'th', 'Augus', 'Sam', 'Cooke', 'World', 'Official', 'Lyric', 'Video', 'Sam', 'Cooke', 'World', 'Official', 'Lyric', 'Video', 'Weekend', 'beautiful', 'People', 'Recurrence', 'Elements', 'Helen', 'Hardin', 'artist', 'heritage', 'Recurrence', 'Elements', 'Helen', 'Hardin', 'artist', 'heritage', 'article', 'docker', 'Sir', 'Thomas', 'More', 'Utopia', 'quote', 'quotation', 'Chills', 'Tears', 'Holy', 'shit', 'High-Quality', 'Free', 'Resources', 'Web', 'Development', 'A', 'Master', 'Thread', 'High-Quality', 'Free', 'Resources', 'Web', 'Development', 'A', 'Master', 'Thread', 'people', 'tape', 'cam', 'sense', 'http', '…', 'home', 'Hazara', 'people', 'Badahshani', 'province', 'Afghanistan', 'Many', 'Hazara', 'people', '…', 'Second-longest', 'wall', 'Planet', 'temple', '“', 'Great', 'Wall', 'India', '”', 'fort', '…', 'May', 'memory', 'Please', 'effort', 'Powerful', 'GR', 'Army', 'Web', 'IDE', 'file', 'editor', 'Markdown', 'file', 'preview', 'option', 'yo', '…', 'dot', 'galaxy', 'Sweet', 'Love', 'Club', 'Version', 'Pet', 'Shop', 'Boys', 'Domino', 'Dancing', 'Official', 'Video', 'HD', 'REMASTERED', 'Pet', 'Shop', 'Boys', 'Domino', 'Dancing', 'Official', 'Video', 'HD', 'REMASTERED', 'Frida-Python-Binding', 'Easy', 'Frida', 'python', 'script', 'Android', 'automation', 'Modern', 'Talking', 'Brother', 'Louie', 'Official', 'Music', 'Video', 'Modern', 'Talking', 'Brother', 'Louie', 'Official', 'Music', 'Video', 'Inline', 'bunch', 'compiler', 'feature', 'code', 'New', 'book', 'Creation', 'Me', 'Them', 'Us', 'Spinning', 'Donut', 'Math', 'Maarten', 'De', 'Baecke', 'energy', 'United', 'States', 'week', 'energy', 'United', 'States', 'week', '“', 'Freedom', '”', 'country', 'woman', 'era', 'language', '”', 'Brave', 'woman', '…', 'Peisu', 'known', 'research', 'computer', 'science', 'technology', 'Leading', 'developer', '…', 'Sertab', 'Erener', 'Mando', 'Aşk', 'Fos', 'Video', 'lady', 'Turkey', 'lady', 'Greece', 'voice', 'art', 'politician', 'Sertab', 'Erener', 'Mando', 'Aşk', 'Fos', 'Video', 'lady', 'Turkey', 'lady', '…', 'Always', 'retweet', 'owl', 'REM', 'Shiny', 'People', 'REM', 'Shiny', 'People', 'RT', 'RT', 'Open', 'Port', 'A', 'Service', 'OPaaS', 'option', 'port', '…', 'RT', 'RT', 'Open', 'Port', 'A', 'Service', 'OPaaS', 'option', 'port', '…', 'week', 'Mexico', '’', 'Supreme', 'Court', 'decision', 'right', 'abortion', 'ProtonMail', 'Keeps', 'IP', 'Logs', 'August', 'Jew', 'Avraam',

'Tevet', 'Thessaloniki', '…', 'app', 'APIs', 'image', 'Rover', 'Mars', 'description', 'Th', '…', 'Nobel', 'Prize', 'laureate', 'Malala', 'Yousafzai', 'year', 'peace', 'prize', 'Aft', '…', 'Make', 'Microsoft', 'Edge', 'DevTools', '"', 'article', 'th', '…', 'people', 'Cool', 'documentary', 'Black', 'Holes', 'Side', 'Inifity', 'Watch', 'day', 'http', '…', 'Machine', 'Learning', 'Algorithms', 'Predictions', '…', 'Day', 'time', 'series', 'method', 'model', 'Holt', '…', 'Advanced', 'Financial', 'Analysis', 'Part', '—', 'v', '…', 'link', 'search', 'Social', 'Media', 'Communties', 'Maps', 'Documents', 'Search', 'Engi', '…', 'SubDomainizer', 'Find', 'Hidden', 'Subdomains', 'Secrets', 'Present', 'Websites', 'New', 'Release', 'Alert', 'Kali', 'Linux', 'Updates', '', 'OpenSSL', '⇢', 'New', 'Kali-Tools', 'site', '⇢', 'Better', 'VM', 'support', '…', 'New', 'Release', 'Alert', 'Kali', 'Linux', 'Updates', '', 'OpenSSL', '⇢', 'New', 'Kali-Tools', 'site', '⇢', 'Better', 'VM', 'support', '…', 'New', 'Release', 'Alert', 'Kali', 'Linux', 'Updates', '', 'OpenSSL', '⇢', 'New', 'Kali-Tools', 'site', '⇢', 'Better', 'VM', 'support', '…', 'New', 'Release', 'Alert', 'Kali', 'Linux', 'Updates', '', 'OpenSSL', '⇢', 'New', 'Kali-Tools', 'site', '⇢', 'Better', 'VM', 'support', '…', 'Security', '➞', '', 'Issues', 'Connected', '…', 'performance', 'interface', 'Tests', 'performance', 'interface', 'Tests', 'Best', 'tweet', 'Day', 'Today', 'list', 'account', 'company', 'tool', 'Open', 'Source', 'Software', 'revolution', 'sc', '…', 'New', 'Artificial', 'Intelligence', 'Can', 'Help', 'Diagnose', 'Lung', 'Cancer', 'Year', 'Early', 'effec', '…', 'New', 'Artificial', 'Intelligence', 'Can', 'Help', 'Diagnose', 'Lung', 'Cancer', 'Year', 'Early', 'effec', '…', '✓', 'Top', 'Scala', 'R', 'Libraries', 'language', 'processing', 'CIO', "I'm", 'Sorry', 'Just', 'Coding', 'New', 'Delphi', 'default', 'font', 'Consolas', 'addition', 'installer', 'New', 'malware', 'Windows', 'Subsystem', 'Linux', 'Open', 'Source', 'Intelligence', 'OSINT', '):', 'OSINT', 'Tools', 'Bitcoin', 'Scam', 'Theft', 'Investigations', '…']

{'arts': 8, 'astronomy': 4, 'beach': 3, 'castles': 8, 'dance': 1, 'music': 30, 'minerals': 1, 'sports': 3, 'summer': 4}

User's top 5 preferences are :['music', 'arts', 'castles', 'astronomy', 'summer']

Το πρόγραμμα πραγματοποιεί ένα πέρασμα όλων των λέξεων που προκύπτουν μετά την επεξεργασία των αναρτήσεων του χρήστη και υπολογίζει τη συχνότητα εμφάνισης ουσιαστικών στο κείμενο τις οποίες με τη σειρά τους τις συγκρίνει με τα λεξικά θεματικών όρων τα οποία έχουν δημιουργηθεί και αποτελούν τα λεξιλόγια αναφοράς της κάθε κατηγορίας. Η αναζήτηση με χρήση hashtags είναι πιο στοχευμένη από την αναζήτηση μέσω ροής αναρτήσεων καθώς είναι δυνατή η χρήση συγκεκριμένων λέξεων κλειδιών που κάνουν πιο συγκεκριμένη το εύρος των χρηστών του Χ που .

Ο κώδικας είναι σχεδιασμένος να αγνοεί σύμβολα, τελείες, αποσιωπητικά, κενά και άλλους μη σημαντικούς χαρακτήρες. Παρόλα αυτά, κάποια διάσπαρτα τέτοια στοιχεία που μπορεί να εμφανιστούν στην τελική αλληλουχία λέξεων, δεν επηρεάζουν το τελικό αποτέλεσμα καθώς γίνεται απόλυτη σύγκριση των όρων και προβλέπεται η απόρριψη όλων των ανόμοιων στοιχείων.

Στο τέλος της εκτέλεσης, ένας μετρητής παρέχει μια εικόνα για τις δημοφιλέστερες κατηγορίες που προέκυψαν για το χρήστη και επιλέγει να χρησιμοποιήσει της 5 πρώτες. Με αυτό τον τρόπο, δημιουργείται ένα προφίλ χρήστη που περιγράφει τα ενδιαφέροντα του χρήστη στον τομέα των κατηγοριών που επεξεργάστηκε ο κώδικας.

Είναι σημαντικό να σημειωθεί ότι η ακρίβεια και η απόδοση του κώδικα εξαρτώνται από τη σαφήνεια των αναρτήσεων που καθορίζει και τις επιτυχείς συγκρίσεις με τα λεξιλόγια. Για αυτό το λόγο χρησιμοποιήθηκαν θετικού νοήματος επιρρήματα και επίθετα που συνοδεύουν τα ουσιαστικά σε κάθε πρόταση, χωρίς αυτό να σημαίνει ότι δεν θα μπορούσε να αξιολογηθεί και μια ειρωνική έκφραση με το ακριβώς αντίθετο νόημα. Η ποσοτική και κατ' επέκταση στατιστική αξιολόγηση του αποτελέσματος αποτελεί μια δικλείδα ασφαλείας για να αποσοβούνται τέτοια σφάλματα.

Προκύπτει λοιπόν ένα προφίλ προτιμήσεων του κάθε χρήστη το οποίο δίνει μια εικόνα για τις προτιμήσεις του και μπορεί να αποτελέσει οδηγό για μια στοχοποιημένη ενημέρωση με εκδηλώσεις, τοποθεσίες και δραστηριότητες του ενδιαφέροντος του. Σε περίπτωση που το επιθυμητό αποτέλεσμα είναι διαφορετικός, αν π.χ. αντί επιρροή του κοινού για τουριστικούς σκοπούς είναι η συμμετοχή στην προώθηση τεχνολογικών επιτευγμάτων, τότε τα θεματικά λεξικά θα πρέπει να διαμορφωθούν ανάλογα.

Βιβλιογραφία

1. Μοσκόβης, 1939, *Αριστοτέλους Πολιτικά*
2. Kaplan, A. M., & Haenlein, M. (2009). *Users of the world, unite! The challenges and opportunities of social media.* Business horizons, 53(1), 59-68.
   Ανακτήθηκε από:
   https://www.researchgate.net/publication/222403703_Users_of_the_World_Unite_The_Challenges_and_Opportunities_of_Social_Media
3. Hauben, M., & Hauben, R. (1998). *Netizens: On the history and impact of Usenet and the Internet*. IEEE Computer Society Press (Ch.2)
   Ανακτήθηκε από: http://www.columbia.edu/~rh120/ch106.x10
4. Annamari Martinviita,(2015) *Online community and the personal diary: Writing to connect at Open Diary*, University of Oulu, 4-6
5. Dellarocas, C. (2003). The digitization of word-of-mouth: Promise and challenges of online feedback mechanisms. Management science, 49(10), 1407-1424
   Ανακτήθηκε από: https://dspace.mit.edu/bitstream/handle/1721.1/7346/4296-03.pdf
6. Akehurst G. (2009) User generated content: the use of blogs for tourism organizations and tourism consumers.
   Ανακτήθηκε από: https://link.springer.com/content/pdf/10.1007/s11628-008-0054-2.pdf
7. Hirschberg J., Manning C., (2015) *Advances in natural language processing, Sciencemag*, 349(6245), 264
8. Κατερίνα Γεωργούλη, Τεχνητή Νοημοσύνη (2015), Κεφ. 7.6,
   Ανακτήθηκε από: http://repfiles.kallipos.gr/html_books/93/07a-main.html#
9. Herbrich R., Graepel Thor, 2010, Handbook of Natural Language Processing. Ch 3-5
10. Bird S., Klein E., Loper E. (2009), Natural Language Processing with Python
11. Shreyas P. (2019), Tweet analytics using NLP (online), Ανακτήθηκε από:
    https://medium.com/analytics-vidhya/tweet-analytics-using-nlp-f83b9f7f7349
12. Saunders K., Metadata: A friendly introduction., Toronto Metropolitan University Pressbooks, Ανακτήθηκε από:
    https://pressbooks.library.torontomu.ca/metadataafriendlyintroduction
13. X API Documentation, Standard V1.1, Ανακτήθηκε από:
    https://developer.twitter.com/en/docs/twitter-api/v1/data-dictionary/object-model/entities

14. Tweepy API V1.1 Reference, Ανακτήθηκε από: https://docs.tweepy.org/en/stable/api.html
15. Jablonsky J., Natural Language Processing With Python's NLTK Package. Ανακτήθηκε από: https://realpython.com/nltk-nlp-python
16. Miller G., Beckwith R., Fellbaum C., Gross D., Miller K., 1993 rev., Introduction to WordNet: An On-line Lexical Database. Ανακτήθηκε από: https://wordnetcode.princeton.edu/5papers.pdf

ΠΑΡΑΡΤΗΜΑ

Κώδικας:

Η μέθοδος διασύνδεσης με τη βάση δεδομένων

Αρχείο: mysql_conn.py

```
import configparser
import MySQLdb.cursors

config = configparser.ConfigParser()
config.read('config.ini')

def db_connect():
   return MySQLdb.connect(host=config['MysqlDB_user']['host'],
                          user=config['MysqlDB_user']['user'],
                          passwd=config['MysqlDB_user']['pass'],
                          db=config['MysqlDB_user']['db'],
                          charset=config['MysqlDB_user']['charset'],

                          )

def db_connect_user():
   return MySQLdb.connect()
```

Παρακολούθηση ροής αναρτήσεων σε πραγματικό χρόνο με λέξεις-κλειδιά. Αποθηκεύονται σε βάση τα ονόματα χρηστών

name_stre aming.py

```
import tweepy
from tweepy import Stream
from tweepy.streaming import StreamListener
from authenticator import authentication
from mysql_conn import db_connect
import MySQLdb
import json
from urllib3.exceptions import ProtocolError
from dateutil import parser

stream_loc = [-11.074218749999998, 49.781264058178344, 2.373046875,
59.62332522313024]


def store_db(created_at, user_id, screen_name, tweet, place, location):
   """
```

```
    connect to MySQL database and insert twitter data
    """

    try:
        con = db_connect()
        con.set_character_set('utf8')

        cursor = con.cursor()

        cursor.execute('SET NAMES utf8;')
        cursor.execute('SET CHARACTER SET utf8;')
        cursor.execute('SET character_set_connection=utf8;')

        if con:
            print("Connection successful")

            # Insert twitter data

            query_users = "INSERT INTO USERS (UserId, ScreenName) Values
(%s, %s)"

            cursor.execute(query_users, (user_id, screen_name))

        else:
            print('connection unsuccessful')

        con.commit()
        cursor.close()
        con.close()
        print("MySQL connection is closed")
        return

    except MySQLdb.Error as e:
        print(e)


class OutListener(StreamListener):

    def on_status(self, status):
        print(status.text)
        name = status.user.screen_name
        print(name)

    def on_data(self, data):

        try:
            rawdata = json.loads(data)

            # grab the wanted data from the Tweet

            user_id = rawdata['user']['id']
            screen_name = rawdata['user']['screen_name']
            text = rawdata['text']
```

```
            created_at = parser.parse(rawdata['created_at'])
            location = rawdata['user']['location']

            if rawdata['place'] is not None:

                place = rawdata['place']['country']

            else:
                place = None

            # print out a message to the screen that we have collected a
tweet
            print("Tweet collected at " + str(created_at))
            print(created_at, user_id, screen_name, text, place, location)

            # insert the data into the MySQL database
            store_db(created_at, user_id, screen_name, text, place,
location)

        except MySQLdb.Error as e:
            print(e)

    def on_exception(self, exception):
        print(exception)
        return


if __name__ == "__main__":
    # Get access and key from another class

    # Authentication
    cred = authentication()
    consumer_key = cred.getconsumer_key()
    consumer_secret = cred.getconsumer_secret()
    access_token = cred.getaccess_token()
    access_token_secret = cred.getaccess_token_secret()

    # Authentication
    auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
    auth.secure = True
    auth.set_access_token(access_token, access_token_secret)
    api = tweepy.API(auth)

    # Start streaming

    myListener = OutListener()
    stream = Stream(auth=api.auth, listener=myListener)

    while True:
        try:
            stream.filter(locations=stream_loc, is_async=True)

        except:
```

```
            # Bypass some errors related to urllib3 requests
            (ProtocolError)
            continue
```

Αναζήτηση αναρτήσεων στο twitter με χρήση hashtags και λέξεων κλειδιών. Αποθηκεύονται σε βάση τα ονόματα χρηστών

hash_key_search.py

```
"""Part of the code was based on example #1 from
https://www.programcreek.com/python/example/76301/tweepy.Cursor"""


import MySQLdb
import tweepy
from tweepy import *
from authenticator import authentication
from mysql_conn import db_connect
import time



def search_hashtag(listoftweets, keyword, numoftweets):

   for keyword in tweepy.Cursor(api.search, q=keyword).items(numoftweets):
       diction = {'UserId': keyword.user.id,
           'UserName': keyword.user.name,
           'ScreenName': keyword.user.screen_name,
           'TweetText': keyword.text,
           'UserLocation': keyword.user.location}

       try:
           con = db_connect()
           con.set_character_set('utf8')
           cursr = con.cursor()

           if con:

               userid = diction['UserId']
               username = diction['UserName']
               screen_name = diction['ScreenName']
               tweet_text = diction['TweetText']
               location = diction['UserLocation']


               query_users = "INSERT INTO USERS (UserId, ScreenName) Values
(%s, %s)"


               cursr.execute(query_users, (userid, screen_name))

               listoftweets.append(diction)
```

```
                print(listoftweets)
                print(len(listoftweets))
                con.commit()
                cursr.close()
                con.close()
                print("MySQL connection is closed")

            else:
                print('connection unsaccesful')


        except MySQLdb.Error as e:
            print(e)


if __name__ == "__main__":
    # Authentication

    cred = authentication()
    consumer_key = cred.getconsumer_key()
    consumer_secret = cred.getconsumer_secret()
    access_token = cred.getaccess_token()
    access_token_secret = cred.getaccess_token_secret()

    auth = tweepy.AppAuthHandler(consumer_key, consumer_secret)
    auth.secure = True

    # For user API: auth.set_access_token(access_token, access_token_secret)

    api = tweepy.API(auth, wait_on_rate_limit=True,
                     wait_on_rate_limit_notify=True)

    preferred_hashtags = ["#Chania", "Χανιά", "#Crete"]
    while 1 == 1:
        for hash in preferred_hashtags:
            search_hashtag(listoftweets=[], keyword=hash, numoftweets=10000)
        time.sleep(60*60*24)  # repeat every 24 hours
```

Αναζήτηση των αναρτήσεων των αποθηκευμένων χρηστών και αποθήκευση σε βάσει δεδομένων.
Ισχύει περιορισμός 3400 tweets ανά περίοδο αναζήτησης από το API, το δείγμα ωστόσο είναι επαρκές.

user_tweets_ to_mysql.py

```
"""Ο κώδικας βασίστηκε σε κώδικα που δημιουργήθηκε από τον χρήστη του
Github Yanofsky https://gist.github.com/yanofsky/5436496 ο οποίος έκανε
αποθήκευση σε αρχείο csv και τροποποιήθηκε ώστε να αποθηκεύει σε βάση
δεδομένων"""

import datetime
```

```
import tweepy
from authenticator import authentication
from mysql_conn import db_connect
import MySQLdb

con = db_connect()
curs = con.cursor()


def user_select():

    try:

        curs.execute("SELECT UserId FROM USERS")
        row = curs.fetchone()

        while row is not None:
            print(row)
            row = curs.fetchone()
            return row
        curs.close()
        con.close()

    except MySQLdb.Error as e:
        print(e)


def get_all_tweets(screen_name):
    #Με αυτή τη μέθοδο υπάρχει περιορισμός 3240 tweets από το API

    cred = authentication()
    consumer_key = cred.getconsumer_key()
    consumer_secret = cred.getconsumer_secret()

# Authentication
    auth = tweepy.AppAuthHandler(consumer_key, consumer_secret)
    auth.secure = True


    api = tweepy.API(auth, wait_on_rate_limit=True,
                     wait_on_rate_limit_notify=True)


# initialize a list to hold all the tweepy Tweets
    alltweets = []

    # make initial request for most recent tweets (200 is the maximum
allowed count)
    new_tweets = api.user_timeline(screen_name=screen_name,
tweet_mode="extended", count=200)

# save most recent tweets
    alltweets.extend(new_tweets)
```

```
# save the id of the oldest tweet less one
   oldest = alltweets[-1].id - 1

# keep grabbing tweets until there are no tweets left to grab
   while len(new_tweets) > 0:
       print("getting tweets before %s" % (oldest))

# all subsiquent requests use the max_id param to prevent duplicates
       new_tweets = api.user_timeline(screen_name=screen_name, count=200,
tweet_mode="extended", max_id=oldest)

# save most recent tweets
       alltweets.extend(new_tweets)

# update the id of the oldest tweet less one
       oldest = alltweets[-1].id - 1

       print("Timestamp:%s...  %s tweets downloaded so far" %
(datetime.datetime.now(), len(alltweets)))

       try:

           for tweet in alltweets:
               curs.execute('SET NAMES utf8mb4;')
               curs.execute('SET CHARACTER SET utf8mb4;')
               curs.execute('SET character_set_connection=utf8mb4')
               query = 'INSERT IGNORE INTO TWEETS (ScreenName, tweet_id,
Created_at, TweetText) Values (%s, %s, %s, %s)'
               curs.execute(query, (screen_name, tweet.id_str,
tweet.created_at, tweet.full_text))
               con.commit()

       except MySQLdb.Error as e:
           print(e)
           pass


if __name__ == '__main__':
   count = 0
   con = db_connect()
   curs = con.cursor()
   curs.execute('SET NAMES utf8;')
   curs.execute('SET CHARACTER SET utf8;')
   curs.execute('SET character_set_connection=utf8;')
   user_row = "SELECT ScreenName, UserId FROM USERS"
   curs.execute(user_row)
   screen_names = curs.fetchall()
   print(screen_names)
   screen_name_list = [i[0] for i in screen_names]
   user_id = [j[1] for j in screen_names]
   print(screen_names, user_id, len(screen_names))
```

```
    try:
        for screen_name,user_id in screen_names:
            get_all_tweets(screen_name)
            count += 1
            print(count, " user's tweets processed")
            print(screen_name, user_id)
    except tweepy.TweepError as er:
        if er:
            print(er)
            pass

curs.close()
con.close()
```

Επεξεργασία των αποθηκευμένων στη βάση δεδομένων αναρτήσεων, διαχωρισμός προτάσεων σε μέρη του λόγου (POS), φιλτράρισμα ώστε να μείνουν μόνο τα ουσιαστικά, σύγκριση με θεματικό ευρετήριο και εξαγωγή αποτελεσμάτων.

tweet_process.py

```
"""
Program that cleans a tweet from spare elements and leaves it with
meaningful words that helps us classify each
tweet and make a user profile according to preferences
    """

import nltk
from nltk.stem import WordNetLemmatizer
import json
from nltk.corpus import stopwords
from nltk import pos_tag
from nltk.tokenize.casual import TweetTokenizer
from mysql_conn import db_connect
import langdetect
import string
import re
from nltk.stem import SnowballStemmer
from nltk.stem import WordNetLemmatizer
from dict_data import category_dict
from os import path
#category_dict = {}
key = 0
# Code for tweets in engish
#t_clean_en = []

pos_lem = WordNetLemmatizer()
tTok = TweetTokenizer()

# Make a list of positive adjectives and adverbs to compare with the text
```

```
with open('positive_adj.py') as pos_adj_file:
    pos_adj = pos_adj_file.read().split(',')
    pos_adj_file.close()
    #print(pos_adj)

with open('positive_adv.py') as pos_adv_file:
    pos_adv = pos_adv_file.read().splitlines()
    pos_adv_file.close()
    # print(pos_adv)

positives = sorted(pos_adj + pos_adv)

# This is the most important function for tweet processing
def tweet_process(tweet, t_clean_en):

    # clean text with regular expressions
    tweet = re.sub(r'https?:\/\/.*[\r\n]*', '', tweet)
    # remove retweets
    tweet = re.sub(r'^RT[\s]+', '', tweet)
    # remove mentions
    tweet = re.sub('@[^\s]+', '', tweet)
    # remove punctuation
    tweet = re.sub('@[^\w\s]', '', tweet)
    # remove # from hashtags and $
    tweet = re.sub(r'\#\w*', '', tweet)
    # remove numbers
    tweet = re.sub(r'\d', '', tweet)
    # remove dots
    tweet = re.sub(r'\.+', "", tweet)

    word_tok = tTok.tokenize(tweet)

    tags = pos_tag(word_tok)

    pos_all = [word for word, pos in tags if (pos.startswith('NN') or
pos.startswith('RB') or pos.startswith('JJ'))]

    stop_words_en = set(stopwords.words("english"))

# We use language detection module to determine the language and store
the tweets accordingly
    try:
        if langdetect.detect(tweet) == "en":
            # All selected pos tags (Nouns, adjectives and adverbs)
            for pos_w in pos_all:
                # stemmer = SnowballStemmer("english")
                wnl = WordNetLemmatizer()
                # Select all the tweets where there are adj, adv with
```

```
positive meaning
                if pos_w in positives:
                    # Select only nouns form the positive text
                    pos_noun = [pos_w for pos_w,pos in tags if
(pos.startswith('NN'))]
                    for w in pos_noun:
                        if w not in stop_words_en and w not in
string.punctuation:
                            # t_clean_en.append(stemmer.stem(w))
                            t_clean_en.append(wnl.lemmatize(w))

            return(t_clean_en)
        else:
            return None
            pass
    except:
        return None
        pass

def tweet_selector():
    con = db_connect()
    cursor = con.cursor()
# Select all users' screen names from db as tuple
    user_select_query = "SELECT ScreenName from USERS"
    cursor.execute(user_select_query)
    user_select = cursor.fetchall()
    # print(user_select)
    tw_list = []

# make a list and remove some special characters
    user_select = list(sum(user_select, ()))

    for user in user_select:
        t_clean_en = []
        print("Processing user: " + user)
        cursor.execute("""SELECT TweetText FROM `TWEETS` where ScreenName
= %s""", (user,))

        # fetching user tweets and saving it to a list
        fetchall_usertweets = cursor.fetchall()
        fetchall_usertweets = list(sum(fetchall_usertweets, ()))

        # setting up a counter which counts the tweets of each user
        counter = 0
        # examine each user's tweet separately
        for tweet in fetchall_usertweets:
            tweet_process(tweet, t_clean_en)
            counter += 1
```

```
        usercats = {}
        for k in category_dict.keys():
            usercats[k] = 0
        for word in t_clean_en:
            for k in category_dict.keys():
                for v in category_dict[k]:
                    if v == word:
                        usercats[k] = 1 + usercats[k]

# Sort out the 5 most popular user's preferences

        #print(str(usercats))
        my_keys = sorted(usercats, key=usercats.get, reverse=True)[:5]
        print(t_clean_en)
        print(usercats)
        print("User's top 5 preferences are :" + str(my_keys))

        if counter > 0:
            query_pref = "INSERT IGNORE INTO PREF (`ScreenName`,
`Preference1`, `Preference2`, `Preference3`,`Preference4`,`Preference5`)
Values (%s, %s, %s, %s, %s, %s)"
            cursor1 = con.cursor()
            cursor1.execute(query_pref, (user, my_keys[0], my_keys[1],
my_keys[2], my_keys[3], my_keys[4]))
            cursor1.close()
            con.commit()

if __name__ == '__main__':

    tweet_selector()
```

Ανάγνωση του αρχείου λεξιλογίου με επανάληψη γραμμή προς γραμμή. Σε κάθε επανάληψη η πρώτη γραμμή αφορά την κατηγορία και η δεύτερη τους όρους του λεξιλογίου. Η επανάληψη διακόπτεται όταν ο κώδικας διαπιστώσει κενό.

```
path = "/home/user/PycharmProjects/EventNotifier/main_data.txt"

# Using dictionary to collect data from file
category_dict = {}


# Reading 1st line key, 2nd line value until the end of the text file
with open(path) as cat_file:
```

```
      while True:
      category = cat_file.readline()
      elements = cat_file.readline()
      if category == '':
            break

      category_dict[category.strip()] = [str(classification.strip()) for
classification in elements.strip().split(',')]
      print(category_dict)
```

Θεματικά λεξικά

arts

abstract, acrylic paint, airbrush, animation, architecture, art, art gallery, artist, artist's bridge, artistic, artwork, assemblage, bas-relief, batik, blending, bridge, brightness, brush=wls-id-2>C, calligraphy, canvas, cartoon, carve, casting, cave drawing, ceramics, chalk, charcoal, chisel, clay, cloisonné, collage, color, color wheel, colored pencils, compass, composition, contrast, craft, crayon, create, creativity, critique, crosshatching, decorate, decorative, decoupage, depict, design, draw, drawing, easel, egg tempera, enamel, encaustic, engraving, erase, eraser, etching, exhibit, exhibition, film, form, frame, fresco, gallery, gesso, gilding, glass, glassblowing, glaze, gold leaf, gouache, graffiti, graphic design, graphite, hammer, hatching, high-relief, hue, illustrate, illustration, image, ink, installation, kiln, landscape, latex paint, line, liner brush, linseed oil, lithograph, lost wax method, low-relief, mahl stick, maquette, marble, marbling, marker, masterpiece, mechanical pencil, media, medium, mixed media, mobile, model, mosaic, mural, museum=wls-id-12>O, objets d'art, oil paint, old master, op art, paint, paint roller, paintbrush, painter, painting, palette, palette knife, paper, pastel, pen, pencil, perspective, photo, photograph, pigment, porcelain, portfolio, portrait, portray, poster, pottery, pounce, primary color, print, printing, proportion, quill, quilt, quilting=wls-id-15>R, realism, ruler, scale, screen printing, sculpt, sculptor, sculpture, seascape, secondary color, shade, silk screen, sketch, sketchbook, solvent, stained glass, stencil, still life, stippling, stone, stonecutting, style, t-square, tempera, template, textile arts, tole painting, tone, tools, trompe l'oeil, tube, turpentine, undertone, varnish, vellum, video, visual, watercolor, waterscape, wax, wheel, wood, wood carving, wood engraving, woodcut, woodcut print

astronomy

albedo, Alpha Centauri, apastron, aperture, aphelion, apogee, asterism, asteroid, astronaut, astronomer, astronomical unit, astronomy, axial tilt, azimuth, background radiation, Bailey's beads, big bang theory, binary star, black body, black hole, bolometer , celestial, celestial equator, cislunar, cluster, comet, conjunction, constellation, Coriolis force, corona, cosmic rays, cosmology, cosmonaut, cosmos, crater, crescent moon, dark matter, day, declination, deep space, Deneb, density, docking, Doppler shift, double star,

Drake equation, dust, dwarf planet, dwarf star, Earth, earthbound, eccentricity, eclipse, ecliptic, elliptical orbit, equinox, escape velocity, event horizon, exoplanet, extragalactic, falling star, flare, flyby, free fall, full moon, galaxy, gamma ray, gas giant, gegenschein, geostationary, geosynchronous, gibbous moon, globular cluster, gravitation, gravitational constant, gravitational lens, gravity, H-R diagram, half moon, heliocentric, helium, Hubble telescope, Hubble's law, hydrogen, hyperbolic orbit, hypernova, ice giant, inclination, inertia, inferior planets, inner planets, interstellar, interstellar dust, ionosphere, Jupiter, Kepler's laws, kiloparsec, Kirkwood gaps, Kuiper belt, Lagrange points, lens, light-year, local arm, local group, lunar, magnitude, mare, Mars, mass, Mercury, meteor, meteor shower, meteorite, meteoroid, Milky Way, minor planet, Mir, moon, muttnik, nadir, NASA, nebula, Neptune, neutron star, new moon, north star, nova, observatory, occultation, Oort cloud, opposition, orbit, orbital eccentricity, orbital inclination, outer planets, parallax, parsec, partial eclipse, penumbra, perigee, perihelion, perturbation, phase, plane of the ecliptic, planet, planetary nebula, planetoid, Pluto, pole star, precession, probe, pulsar, quarter moon, quasar, radiant, radiation, red dwarf, red giant star, red shift, revolve, right ascension, rings, Roche limit, rocket, satellite, Saturn, scientific notation, scintillation, shooting star, sidereal, singularity, sky, solar, solar system, solar wind, solstice, space, space exploration, space station, spectroscope, spectrum, Sputnik, star, starlight, sun, sunspot, superior planets, supernova, synodic, syzygy, telemetry, telescope, terminator, terrestrial, total eclipse, totality, transit, translunar, transneptunian, twinkling, umbra, universe, Uranus, vacuum, Van Allen belt, variable star, Venus, vernal equinox, visual magnitude, waning, wavelength, waxing, weightlessness, white dwarf, white giant, wormhole, x-rays, yellow dwarf, zodiac

beach

barnacl, bathing, suit, bay, beach, beachball, bikini, boardwalk, boat, boogie, board, cape, catamaran, clam, clam, bake, coast, conch, cooler, coral, cove, crab, currents, dive, dock, dune, dune, buggy, ebb, tide, family, fins, fish, fishing, Frisbee, gull, hang, five, hat, hermit, crab, high, tide, ice, cream, intertidal, zone, island, jellyfish, kayak, kelp, lagoon, lake, lakeshore, life, jacket, life, preserver, lifeguard, limpet, longboard, low, tide, mangrove, mussels, neap, tide, ocean, paddleboat, palm, tree, pelican, pier, popsicle, reef, relax, rest, rip, current, sail, sailboat, salt, water, salt, water, taffy, sand, sand, dollar, sandals, sandbar, sandcastle, scuba, sea, sea, star, seagull, seashell, seashore, shark, shell, ship, shore, shorebirds, snacks, snorkel, soft, serve, ice, cream, spray, starfish, sun, sun, hat, sunbathe, sunburn, sunglasses, sunscreen, suntan, surf, surfboard, swim, swim, fins, swimming, cap, taffy, tan, tide, tide, pool, towel, trip, trunks, tsunami, umbrella, undertow, underwater, vacation, volleyball, water, water, bottle, waves, weekend, wet, wharf, whitecaps, yacht, zoris

castles

archer, aristocracy, armor, ballista, baron, baroness, baronet, bastion, battering, ram, battle, axe, birthright, castle, catapult, chain, mail, chateau, chieftain, chivalry, count, countess, court, jester, crenelations, crest, crossbow, crown, crown, jewels, curtain, wall, donjon, dragon, drawbridge, duchess, duke, dungeon, dynasty, earl, empire, feudal, fort, fortress, gatehouse, grand, duke, great, hall, heraldry, hereditary, highness, honor, imperial, jester, jousting, keep, king, kingdom, knight, lady, lady-in-waiting,

lance, lord, maid, maiden, mangonel, manor, manor, house, marquis, Middle, Ages, moat, monarch, nobility, noble, onager, palace, peel, tower, peer, prince, princess, queen, realm, reign, Renaissance, royal, royalty, scepter, serf, shield, siege, sovereign, sword, titles, tower, trebuchet, turret, unicorn, viscount, wall, watchtower

dance

abstract, dance, acro, allemande, ballet, ballroom, dance, barn, dance, beguine, belly, dance, bolero, bop, bossa, nova, bourrée, breakdance, bunny, hop, Cajun, dance, cakewalk, can-can, Carioca, ceilidh, cha-cha, Charleston, classical, dance, clogging, conga, contra, dance, country-western, courante, dance, disco, dragon, dance, fandango, flamenco, folk, dance, foxtrot, frug5>G, gavotte, gigue, habanera, Highland, fling, hip, hop, hoedown, hokey-pokey, hop, hula, hustle, ice, dancing, interpretive, dance, Irish, dance, jazz, dance, jig, jitterbug, jive, koftos, krump, lambada, Latin, dance, limbo, Lindy, line, dance, Macarena, mambo, maypole, dance, mazurka, merengue, Mexican, hat, dance, minuet, modern, dance, one-step, pas, de, deux, Pasodoble, pavane, polka, pop, quadrille, quickstep, reel, reggae, rumba, salsa, Scottish, dance, slow, dance, soft, shoe, sousta, square, dance, step, dance, stomp, swing, syrtaki, tango, tap, dance, tarantella, time, warp, toe, dance, troika, twist, two, step, Ukrainian, dance, Viennese, waltz, waltz, Watusi, zeimpekiko, zydeco

music

Acappella, Abrasive, Accidental, Accompaniment, Accordion, Acoustics, Adaptation, Alto, Amateur, Amplification, Applause, Appreciation, Approval, Aria, Array, Artist, Artistic, Assemble, Assurance, Atmosphere, Audience, Audition, Auditory, Awe, Babygrand, Ballet, Band, Barbershopquartet, Baroque, Bar, Bass, Baton, Beat, Bells, Bewitched, Bluegrass, Boisterous, Brass, Bravado, Broadcast, Brushwork, Brusque, Cadence, Calypso, Can-can, Canon, Cantata, Canvas, Chambermusic, Chanting, Charleston, Chime, Choir, Choral, Chord, Chord, Chorus, Christmascarols, Clapping, Classic, Classical, Clef, Collaboration, Collection, Commanding, Commentary, Communication, Complement, Compose, Composer, Composition, Concert, Concertina, Concertina, Concertmaster, Concerto, Conservatory, Console, Continuous, Contribution, Conviction, Cor, Counterpoint, Creative, Crescendo, Critic, Crowd-pleasing, Culture, Damper, Dance, Debut, Dedication, Delivery, Depth, Development, Diffuse, Digital, Dignity, Digression, Display, Dissonant, Diverse, Diversification, Diversity, Do-re-mi-fa-sol-la-ti-do, Dominant, Download, Dramatic, Duet, Duo, Duplication, Duration, Dynamic, Ear, Education, Elaboration, Electronic, Element, Eloquence, Eloquent, Emission, Encore, Enjoyment, Enlightened, Ensemble, Entertainment, Etude, Execution, Experience, Expression, Exquisitely, Extraordinary, Facade, Facets, Falsetto, Fandango, Fanfare, Fantasia, Fantasy, Finale, Flair, Flat, Flourish, Focus, Folkmusic, Foxtrot, Fragment, Freeform, Frenzied, Fret, Fugue, Full blown, Gauge, Generate, Genius, Glare, Gospel, Guitar, Guitarist, Halfstep, Harmony, Hearing, Heritage, Hiatus, Highlight, Humanities, Hymn, Illumination, Imagery, Imitation, Impromptu, Infusion, Innovate, Inspiration, Instrumental, Intensity, Interpretation, Ipod, Jazz, Jingle, Judge, Juxtaposition, Key, Keyboard, Landscape, Libretto, Lightopera, Listening, Literary, Loudness, Lullaby, Luminous, Lyrical, Macarena, Madrigal, Magnify, Major, March, Masterpiece, Masterwork, Matinee, Measure, Medium, Melody, Memorable, Mezzo-soprano, Minor, Minstrel, Modernmedium, Modulate, Momentum, Motive, Movement, Music, Musician, Natural, Notation, Note, Nuance, Numbers, Octave, Ode, Official, On-stage, Opera, Oratorio, Orchestra, Orchestral, Organ, Organist, Outburst, Ovation,

Overtones, Overture, Passage, Patterns, Pedal, Perception, Percussion, Performance, Persistence, Phrase, Piano, Pinnacle, Pitch, Playback, Playing, Pluck, Poetry, Polyphonic, Popular, Portrayal, Practice, Preference, Prelude, Premiere, Presence, Prodigy, Professional, Prolific, Promote, Promotion, Pub, Publish, Quality, Quantity, Quartet, Quaver, Quintessential, Quintet, Range, Rap, Recitation, Recognition, Records, Refrain, Rehearsal, Religious, Represent, Resonance, Rest, Restraint, Rhythm, Rock, Rousing, Scale, Scherzo, Score, Sculpture, Secular, Serenade, Sextet, Sharp, Sheetmusic, Signature, Singing, Smooth, Softness, Solemn, Solo, Soloist, Sonata, Song, Sonnet, Soprano, Sounds, Spiritual, Spontaneity, Spotlight, Staccato, Stage-craft, Stem, Stereophonic, Stop, String, Strings, Strobe, Studio, Style, Stylized, Talent, Technique, Tempo, Tenor, Texture, Theme, Thunderous, Timbre, Timevalue, Tone, Tour, Traditional, Training, Transformation, Treatment, Treble, Trill, Trio, Troubadour, Troupe, Tunes, Undemonstrative, Understanding, Uneventful, Unique, Upright

minerals

agate, alexandrite, amber, amethyst, aquamarine, aventurine, azurite, basalt, beryl, calcite, carat, carnelian, chalcedony, citrine, coal, coral, crystal, density, diamond, dolomite, druze, emerald, facet, feldspar, fire opal, fluorite, fool's gold, fossil, garnet, gem, gemstone, geode, geology, gneiss, gold, granite, gravel, gypsum, hardness, hematite, howlite, igneous rock, inclusion, jade, jadeite, jasper, jet, karat, kunzite, labradorite, lapidary, lapis lazuli, lava, limestone, lithified, loess, luster, magma, magnetite, malachite, metamorphic rock, meteoroid, mica, mineral, moonstone, moss agate, nephrite, obsidian, opal, orthoclase, pebble, peridot, petrified wood, pumice, pyrite, pyroclastic flow, quartz, rock, rock cycle, rose quartz, ruby, salt, sand, sandstone, sapphire, sard, sardonyx, schist, sedimentary rock, seismograph, semi-precious, serpentine, silica, silt, soapstone, sodalite, spinel, stalactite, stalagmite, stone, talc, talus, tiger's eye, topaz, tourmaline, tsavorite, turquoise, unakite, vitreous, volcanic bomb, volcanic glass, volcano, vulcanite, wairakite, xalostocite, xenolith, xyloid jasper, yeatmanite, zircon, zoisite

sports

aerobics, archer, archery, arena, arrow, athlete, athletics, axel, badminton, ball, base, baseball, basketball, bat, baton, batter, batting, biathlon, bicycle, bicycling, bike, biking, billiards, bobsleigh, bocce, boomerang, boules, bow, bowler, bowling, boxer, boxing, bronze medal, bunt, canoe, canoeing, catch, catcher, champion, championship, cleats, club, coach, compete, competing, competition, competitor, crew, cricket, croquet, cross country, cue, curling, cycle, cycling, cyclist, dart, dartboard, deadlifting, decathlon, defense, diamond, discus, dive, diver, diving, dodgeball, doubleheader, dugout, equestrian, equipment, exercise, fencing, field, field hockey, fielder, fielding, figure skating, fitness, football, forward, free throw, Frisbee, game, gear, geocaching, go, goal, goalie, gold medal, golf, golfer, golfing, guard, gym, gymnasium, gymnast, gymnastics, halftime, hammer throw, handball, hang gliding, hardball, helmet, heptathlon, high jump, hitter, hockey, hole-in-one, home, home plate, home run, home team, hoop, horseshoes, huddle, hurdle, ice hockey, ice rink, ice skates, ice skating, infield, infielder, inline skates, inning, jai-alai, javelin, jog, jogger, judo, jump, jump rope, jumper, jumping, karate, kayak, kayaker, kayaking, kickball, kite, kneepads, kung fu, lacrosse, lawn bowling, league, long jump, lose, loser, luge, lutz, major league, mallet,

martial arts, mat, medal, minor league, mitt, mouthguard, move, movement, MVP, net, no-hitter, Nordic skiing, offense, ollie, Olympics, orienteering, out, outfield, outfielder, paddle, paddleball, paddling, paintball, parasailing, parkour, pentathlon, pickleball, ping pong, pitch, pitcher, play, player, playing, playoffs, pogo stick, pole, pole vault, polo, pool, puck, quarter, quarterback, quiver, 17>R, race, racer, racewalking, racing, racket, racquetball, rafting, referee, relay, ride, riding, rink, rock climbing, roller skates, roller skating, row, rower, rowing, rugby, run, runner, running, sailing, score, scoreboard, scuba, scull, sculling, shortstop, shot put, silver medal, skate, skating rink, skeleton, ski, skier, skiing, slalom, sled, sledder, sledding, snorkeling, snowboard, snowboarder, snowboarding, snowshoeing, soccer, softball, somersault, speed skating, sport, sportsmanship, squash, stadium, stick, strike, stroke, Sumo wrestling, surfer, surfing, swim, swimmer, swimming, synchronized swimming, table tennis, taekwondo, tag, target, team, teammate, tee, telemark skiing, tennis, tetherball, throw, throwing, tie, toboggan, track and field, trampoline, triathlete, triathlon, triple jump, triple play, tug of war, ultramarathon, ultramarathoner, umpire, unicycle, unicyclist, uniform, vault, vaulter, vaulting, visiting team, volley, volley ball, wakeboarding, walk, walker, walking, water polo, water ski, water skier, water skiing, weightlifter, weightlifting, weights, wetsuit, white water rafting, wicket, win, windsurfer, windsurfing, winner, winning, World Cup, World Series, wrestler

summer

air conditioner, August, backpacking, baseball, bathing suit, beach, berries, bikini, blistering heat, boating, camp, camping, camper, canoeing, caravan, daisy, diving, ease, fan, flowers, fresh fruit, Frisbee, gardening, grass, heat, hiking, holiday, hot, humidity, ice cream, Independence Day, journey, July, June, lightning, muggy12>O, ocean, outdoors, outings, outside, park, picnic, play, popsicle, recreation, relax, rest, road trip, rose, sailing, sandals, sandcastle, sea, searing heat, seashore, season, shorts, showers, sightseeing, stifling, summer, summer solstice, sun, sundress, sunflower, sunhat, sunny, sunscreen, sweltering, swim, swim fins, swimming cap, swimsuit, tan, thunder, thunderstorm, travel, trip, trunks, vacation, visit, voyage, warm weather, watermelon, waterpark, waterski, zoris