



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
Σχολή Τεχνολογίας
Τμήμα Ψηφιακών Συστημάτων

Μελέτη στην Ανάλυση
συναισθήματος με Python

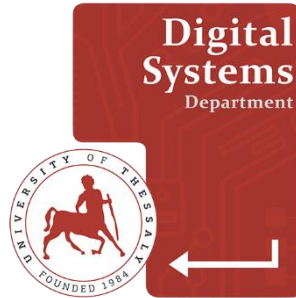
ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Παπαστεργίου Χαρίλαος

ΑΜ: 3120203

Επιβλέπων: ονομ/μο, βαθμίδα

ΛΑΡΙΣΑ, Μήνας Έτος



UNIVERSITY OF THESSALY

School of Technology

Digital Systems Department

Study in Sentiment Analysis with Python

BACHELOR THESIS

Charilaos Papastergiou

(RN: 3120203)

Supervisor: *full name, position*

LARISSA, month year

Υπεύθυνη Δήλωση περί Ακαδημαϊκής Δεοντολογίας και Πνευματικών Δικαιωμάτων

Με πλήρη επίγνωση των συνεπειών του νόμου περί πνευματικών δικαιωμάτων, δηλώνω ρητά ότι η παρούσα πτυχιακή εργασία, καθώς και τα ηλεκτρονικά αρχεία και πηγαίοι κώδικες που αναπτύχθηκαν ή τροποποιήθηκαν στα πλαίσια αυτής της εργασίας, αποτελεί αποκλειστικά προϊόν προσωπικής μου εργασίας, δεν προσβάλλει κάθε μορφής δικαιώματα διανοητικής ιδιοκτησίας, προσωπικότητας και προσωπικών δεδομένων τρίτων, δεν περιέχει έργα/εισφορές τρίτων για τα οποία απαιτείται άδεια των δημιουργών/δικαιούχων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον, και πληρούν τους κανόνες της επιστημονικής παράθεσης. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο, αρχεία ή/και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή.

Αναλαμβάνω πλήρως, ατομικά και προσωπικά, όλες τις νομικές και διοικητικές συνέπειες που δύναται να προκύψουν στην περίπτωση κατά την οποία αποδειχθεί, διαχρονικά, ότι η εργασία αυτή ή τμήμα της δεν μου ανήκει διότι είναι προϊόν λογοκλοπής.

Ο/Η Δηλών/ούσα

(Υπογραφή)

Ονοματεπώνυμο Φοιτητή/-ήτριας
Παπαστεργίου Χαρίλαος

Ημερομηνία
29-11-2024

Περίληψη

Η παρούσα πτυχιακή εργασία δίνει βάση στην τεχνολογία της **ανάλυσης συναισθήματος** και τα επιμέρους στοιχεία που διαθέτει. Συγκεκριμένα μελετήθηκε σε βάθος η ανάλυση συναισθήματος ως έννοια, το πως λειτουργεί, εφαρμογές της στην καθημερινότητα καθώς και τα προβλήματα που αντιμετωπίζει ως τεχνολογία. Περαιτέρω αναλύονται λεπτομερώς οι βασικότερες προσεγγίσεις που ακολουθούν οι χρήστες για την λειτουργία της ανάλυσης συναισθήματος (**λεξιλογική, μηχανικής μάθησης, υβριδική, Aspect based sentiment analysis**) περιλαμβάνοντας στοιχεία όπως το πως ορίζονται, πως λειτουργούν, τι επιπλέον λειτουργίες διαθέτουν, πλεονεκτήματα και μειονεκτήματα αυτών καθώς και που και πως εφαρμόζονται. Επιπρόσθετα επικεντρώνεται στη λειτουργία την ανάλυσης συναισθήματος με την γλώσσα python, πως το sentiment analysis λειτουργεί στο περιβάλλον της καθώς και αναφορά σε **εργαλεία-αλγορίθμους** ανάλυσης συναισθήματος (ασχέτως προσέγγισης) που μπορεί κανείς να χρησιμοποιήσει σε περιβάλλον της python. Τέλος στο πρακτικό μέρος της πτυχιακής εργασίας προβάλλεται η χρήση του sentiment analysis με γλώσσα python χρησιμοποιώντας συγκεκριμένες βιβλιοθήκες όπως **nltk**, **textblob** και **vader** με σκοπό την ανάλυση συναισθήματος σε απλές προτάσεις αλλά και αρχεία text, καθώς και μέτρηση απόδοσης του κώδικα python (accuracy) και μετατροπές που οφείλουν στην αύξηση-μείωση αυτού.

Λέξεις κλειδιά

Ανάλυση συναισθήματος, Αλγόριθμος, Μηχανική μάθηση, Λεξικό συναισθημάτων, Python, NLTK, ABSA,

Abstract

This thesis focuses on the technology of sentiment analysis and its individual components. Specifically, sentiment analysis is examined in depth as a concept, including how it works, its applications in daily life, and the challenges it faces as a technology. Furthermore, the primary approaches used by users to perform sentiment analysis (lexicon-based, machine learning, hybrid, and aspect-based sentiment analysis) are analyzed in detail, covering aspects such as their definitions, mechanisms, additional features, advantages, and disadvantages, as well as where and how they are applied.

Additionally, the thesis emphasizes the operation of sentiment analysis using the Python programming language, exploring how it functions in its environment and referencing sentiment analysis tools and algorithms (regardless of the approach) that can be used for sentiment analysis in Python.

Finally, in the practical part of the thesis, the use of sentiment analysis in Python is demonstrated through specific libraries such as NLTK, TextBlob, and VADER, aiming to analyze sentiment in simple sentences as well as text files. The performance of the Python code (accuracy) is also measured, along with modifications aimed at improving or reducing its performance.

Keywords

Sentiment Analysis, Algorithm, Machine learning, Sentiment Dictionary, Python, NLTK, ABSA,

Ευχαριστίες

Σε αυτό το σημείο θα ήθελα να ευχαριστήσω τους γονείς μου και τους φίλους μου που με ήταν δίπλα μου σε αυτή μου την προσπάθεια και ιδιαίτερα τον επιβλέποντα καθηγητή μου κύριο Όμηρο Ιατρέλλη καθώς το θέμα που επέλεξε να αναλύσω ήταν για εμένα μια θετική πρόκληση αλλά και ευκαιρία να μελετήσω και να λάβω γνώσεις για ένα τόσο ενδιαφέρον αλλά και σημαντικό θέμα-πτυχή στον τομέα της πληροφορικής.

Περιεχόμενα

ΠΕΡΙΛΗΨΗ	4
ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ.....	5
ABSTRACT.....	6
KEYWORDS.....	7
ΕΥΧΑΡΙΣΤΙΕΣ	8
ΠΕΡΙΕΧΟΜΕΝΑ	9
1)ΕΙΣΑΓΩΓΗ	11
ΣΤΟΧΟΙ ΤΗΣ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ	11
2) ΑΝΑΛΥΣΗ ΣΥΝΑΙΣΘΗΜΑΤΟΣ.....	12
2.1 ΟΡΙΣΜΟΣ ΑΝΑΛΥΣΗΣ ΣΥΝΑΙΣΘΗΜΑΤΩΝ.....	12
2.2 ΕΦΑΡΜΟΓΕΣ ΤΗΣ ΑΝΑΛΥΣΗΣ ΣΥΝΑΙΣΘΗΜΑΤΟΣ.....	14
2.3 ΠΡΟΒΛΗΜΑΤΑ ΤΗΣ ΑΝΑΛΥΣΗΣ ΣΥΝΑΙΣΘΗΜΑΤΟΣ	15
2.3.1 Προβλήματα της ανάλυσης συναισθήματος ανάλογα με την προσέγγιση που επιλέγουμε.....	16
3) ΛΕΞΙΛΟΓΙΚΗ ΠΡΟΣΕΓΓΙΣΗ.....	17
3.1 ΟΡΙΣΜΟΣ ΛΕΞΙΛΟΓΙΚΗΣ ΠΡΟΣΕΓΓΙΣΗΣ-ΤΙ ΕΙΝΑΙ ΑΚΡΙΒΩΣ?.....	17
3.1.1 Πρακτικά πως λειτουργεί?	17
3.1.2 Λεξικά συναισθημάτων.....	18
3.2 ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΚΑΙ ΜΕΙΟΝΕΚΤΗΜΑΤΑ ΤΗΣ ΛΕΞΙΛΟΓΙΚΗΣ ΠΡΟΣΕΓΓΙΣΗΣ	26
4) ΠΡΟΣΕΓΓΙΣΗ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ.....	28
4.1 ΟΡΙΣΜΟΣ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ	28
4.2 MACHINE LEARNING VS DEEP LEARNING, ΕΙΝΑΙ ΑΡΑΓΕ ΤΟ ΙΔΙΟ?	29
4.2.1 Παραδείγματα εφαρμογής των Machine και Deep Learning	30
4.3 ΚΑΤΗΓΟΡΙΕΣ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ	31
4.4 ΕΞΟΥΡΥΞΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ	32

4.5 ΜΟΝΤΕΛΑ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ.....	33
4.6 ΕΦΑΡΜΟΓΕΣ ΤΗΣ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ ΣΤΗΝ ΕΠΟΧΗ ΜΑΣ.....	35
5) ΠΡΟΣΕΓΓΙΣΗ ABSA (ASPECT-BASED SENTIMENT ANALYSIS).....	38
5.1 ΟΡΙΣΜΟΣ ABSA.....	38
5.2 ΠΩΣ ΛΕΙΤΟΥΡΓΕΙ?	38
5.3 ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΚΑΙ ΜΕΙΟΝΕΚΤΗΜΑΤΑ	40
5.4 ΧΡΗΣΕΙΣ ΤΟΥΣ ΣΤΗΝ ΚΑΘΗΜΕΡΙΝΟΤΗΤΑ	41
6) PYTHON ΚΑΙ ΑΝΑΛΥΣΗ ΣΥΝΑΙΣΘΗΜΑΤΟΣ	42
6.1 ΟΡΙΣΜΟΣ-ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΤΙ ΕΙΝΑΙ Η PYTHON, ΓΙΑ ΠΟΙΟ ΛΟΓΟ ΧΡΗΣΙΜΟΠΟΙΕΙΤΑΙ ΚΑΙ ΤΙ ΠΕΡΙΛΑΜΒΑΝΕΙ ?.....	42
6.2 PYTHON ΚΑΙ SENTIMENT ANALYSIS, ΜΕ ΠΟΙΟ ΤΡΟΠΟ ΛΕΙΤΟΥΡΓΟΥΝ ΜΑΖΙ? .	43
6.3 ΑΛΓΟΡΙΘΜΟΙ ΤΗΣ PYTHON ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΟΥΝΤΑΙ ΓΙΑ SENTIMENT ANALYSIS	44
7) ΠΡΑΚΤΙΚΟ ΚΟΜΜΑΤΙ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ.....	54
7.1 ΑΠΛΗ ΧΡΗΣΗ ΤΩΝ TEXTBLOB ΚΑΙ VADER ΓΙΑ SENTIMENT ANALYSIS ΜΙΑΣ ΤΥΧΑΙΑΣ ΠΡΟΤΑΣΗΣ.....	54
7.2 ΧΡΗΣΗ ΤΩΝ TEXTBLOB ΚΑΙ VADER ΓΙΑ SENTIMENT ANALYSIS ΣΕ ΕΓΓΡΑΦΑ TEXT.	58
7.3 ΧΡΗΣΗ ΤΩΝ TEXTBLOB ΚΑΙ VADER ΓΙΑ SENTIMENT ANALYSIS ΣΤΑ ΕΓΓΡΑΦΑ POSITIVE-NEGATIVE.TXT ΜΕ ΣΚΟΠΟ ΤΗΝ ΕΜΦΑΝΙΣΗ ΤΟΥ ACCURACY ΠΟΥ ΕΧΟΥΝ ΚΑΙ ΠΩΣ ΑΥΤΗ ΕΠΗΡΕΑΖΕΤΑΙ ΕΦΟΣΟΝ ΑΝΑΛΥΟΥΜΕ ΑΝΑ ΓΡΑΜΜΗ ΤΑ ΕΓΓΡΑΦΑ ΚΑΙ ΔΙΑΦΟΡΟΠΟΙΩΝΤΑΣ ΤΟ THRESHOLD	61
8) ΣΥΜΠΕΡΑΣΜΑ.....	70
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	72

1)Εισαγωγή

Στόχοι της πτυχιακής εργασίας

Βασικός στόχος της πτυχιακής εργασίας είναι να γίνει μια διερεύνηση του πεδίου της **ανάλυσης συναισθήματος (sentiment analysis)**, δίνοντας έμφαση στην κατανόηση της λειτουργίας του, παρουσιάζοντας τις διάφορες εφαρμογές του στην καθημερινότητα και αναλύοντας τα θετικά και αρνητικά στοιχεία των δημοφιλέστερων προσεγγίσεων που ακολουθούνται από τους χρήστες.

Μέσω της ανάγνωσης του θεωρητικού πλαισίου της πτυχιακής εργασίας ο αναγνώστης αποκτάει γνώση για τις μεθόδους και τα εργαλεία που χρησιμοποιούνται στην ανάλυση συναισθήματος, ανεξάρτητα της προσέγγισης που ακολουθείται. Γίνεται ενημέρωση σε βάθος για όλες τις προσεγγίσεις και αξιολογούνται τα πλεονεκτήματα και μειονεκτήματα αυτών.

Στο πρακτικό κομμάτι της πτυχιακής εργασίας στόχος αποτελεί η χρήση της python για ανάπτυξη κώδικα που θα εκτελέσει την λειτουργία της ανάλυσης συναισθήματος σε απλό αλλά και πιο σύνθετο επίπεδο. Συγκεκριμένα ο αναγνώστης σε αυτό το σημείο επιμορφώνεται πάνω στα εξής:

- Στο εντοπισμό συναισθήματος με χρήση python σε μια απλή πρόταση χρησιμοποιώντας τους αλγορίθμους TextBlob και Vader.
- Στην ανάλυση συναισθήματος σε αρχεία κειμένου-dataset.
- Στον υπολογισμό συναισθηματικού σκορ και την αξιολόγηση της ακρίβειας του κώδικα python που αναπτύχθηκε και πώς επηρεάζεται με τροποποιήσεις και βελτιστοποιήσεις του κώδικα.

Συνολικά η πτυχιακή με αυτό τον συνδυασμό του θεωρητικού και πρακτικού πλαισίου στοχεύει να αποτελεί ένα ολοκληρωμένο εργαλείο κατανόησης και εφαρμογής της ανάλυσής συναισθήματος.

2) Ανάλυση Συναισθήματος

2.1 Ορισμός Ανάλυσης Συναισθημάτων

Για να γίνει κατανοητή η χρήση ενός εργαλείου ανάλυσης συναισθημάτων αρχικά κρίνεται απαραίτητο κάποιος να γνωρίζει σε ικανοποιητικό βαθμό το τι είναι η ανάλυση συναισθήματος καθώς και το πως λειτουργεί. Με τον όρο **ανάλυση συναισθήματος**^[1] εννοούμε μια τεχνική επεξεργασίας φυσικής γλώσσας (NLP) που μας βοηθάει να κατανοήσουμε τις απόψεις και τις πεποιθήσεις των ανθρώπων και πιο συγκεκριμένα το αν ένα «περιεχόμενο» (πχ, μια κριτική σε ένα προϊόν, μια δημοσίευση στα social media, ένα άρθρο σε ένα site) γραμμένο στην ανθρώπινη γλώσσα διαθέτει συγκεκριμένα συναισθήματα. Πριν την εκτέλεση της ανάλυσης συναισθήματος είναι σημαντικό να αναφερθεί πως πολλές φορές εκτελούνται και άλλες διαδικασίες όπως πχ τεχνικές NER(τεχνικές αναγνώρισης ονοματικών οντοτήτων). Οι τεχνικές NER χρησιμοποιούνται πριν την εφαρμογή τεχνικών ανάλυσης συναισθήματος με σκοπό την προεπεξεργασία δεδομένων για την κατηγοριοποίηση ονοματικών οντοτήτων σε κείμενα. Αυτό έχει ως αποτέλεσμα την ενίσχυση της ικανότητας ενός συστήματος να αντλεί συμπεράσματα για τα συναισθήματα εντός ενός περιεχομένου που κατηγοριοποιήθηκαν και σχετίζονται με συγκεκριμένες οντότητες ή γεγονότα. Με λίγα λόγια οι NER τεχνικές κάνουν πιο εύκολο το έργο της ανάλυσης συναισθήματος. Ανάλογα με τις προσεγγίσεις και τεχνικές που χρησιμοποιούμε στην ανάλυση συναισθήματος μπορούμε να ανιχνεύσουμε και να εξορύξουμε τα ευρήματά μας, να τα κατανοήσουμε και να τα κατηγοριοποιήσουμε ανάλογα με το συναίσθημα που περιέχουν. Η κατηγοριοποίηση αυτή διαφέρει ανάλογα με τις τεχνικές και τις προσεγγίσεις που υιοθετούμε. Ένα περιεχόμενο (που μπορεί να είναι ένα κείμενο, μια πρόταση, μια λέξη) μπορεί να χωριστεί με βάση του αν είναι μια **θετική, αρνητική ή ουδέτερη έννοια** ή ως ένα συγκεκριμένο **συναίσθημα** (χαρά, λύπη, αγάπη, εκνευρισμός, φόβος).. Υπάρχουν δύο βασικές προσεγγίσεις στην Ανάλυση Συναισθήματος: Η **Λεξιλογική Προσέγγιση** και η **Προσέγγιση Μηχανικής Μάθησης**. Ανάλογα με την χρήση της καθεμίας αλλάζει μέχρι ενός βαθμού η λειτουργία της Ανάλυσης Συναισθήματος. Καθεμία από αυτές διαθέτει αρνητικά και θετικά στοιχεία και επιλέγεται ανάλογα με το τι θέλουμε να μελετήσουμε αν και μπορεί να γίνει και συνδυασμός αυτών (**Υβριδική**

Προσέγγιση) των δύο για να βελτιωθεί η ακρίβεια της Ανάλυσης Συναισθήματος. Συχνά ο όρος **ανάλυση συναισθήματος** ταυτίζεται με την έννοια της **εξόρυξης γνώμης** καθώς τυχαίνει πολλές φορές να χρησιμοποιούνται για τον ίδιο σκοπό. Από την άλλη αν κάποιος αναρωτηθεί αν είναι το ίδιο η απάντηση θα είναι αρνητική καθώς η εξόρυξη γνώμης σε αντίθεση με την ανάλυση συναισθήματος χρησιμοποιείται κυρίως για να χαρακτηριστεί μια έννοια σε ένα κείμενο ως θετική ή αρνητική(πχ το αγαπάω, το μισώ). Η Γενική διαδικασία που πραγματοποιείται κατά την εκτέλεση της ανάλυσης συναισθήματος χωρίς τον ουσιαστικό ρόλο της προσέγγισης που χρησιμοποιήθηκε χωρίζεται σε **3 βήματα**:

- 1) **Την συλλογή δεδομένων** (συλλέγουμε τα δεδομένα στα οποία επιθυμούμε να γίνει η ανάλυση συναισθήματος, οι πηγές των δεδομένων μπορούν να είναι μέσα κοινωνικής δικτύωσης, κριτικές από προϊόντα σε ηλεκτρονικά καταστήματα, άρθρα στο διαδίκτυο), στο συγκεκριμένο βήμα είναι απαραίτητο να ελεγχθεί η ποιότητα των δεδομένων, η αντιπροσωπευτικότητα αυτών καθώς και ο όγκος των δεδομένων. Για την περισυλλογή των δεδομένων αυτών χρησιμοποιείται πληθώρα εργαλείων που υπάρχουν στο διαδίκτυο και η επιλογή ενός συγκεκριμένου γίνεται βάση των δεδομένων που απαιτούνται καθώς και του budget που διατίθεται.
- 2) **Τον Καθορισμό των δεδομένων** που αποτελεί το σημείο στο οποίο γίνεται ενός είδους έλεγχος, εντοπίζονται σφάλματα και αφαιρούνται τυχόν άσχετες πληροφορίες που αντλήθηκαν κατά το πρώτο βήμα. Περιλαμβάνει και ένα είδος ορθογραφικού ελέγχου στον οποίο αφαιρούνται τυχόν άρθρα, σημεία στίξης και ασήμαντα στοιχεία.
- 3) **Την Εξαγωγή χαρακτηριστικών** όπου εξάγονται τα δεδομένα που θα χρησιμοποιηθούν τελικά στην ανάλυση συναισθήματος. Τα δεδομένα αυτά περιλαμβάνουν φράσεις, λέξεις η και κομμάτια κειμένου που συνδέονται με συγκεκριμένα αισθήματα. Στην συνέχεια τα βήματα διαφέρουν ανάλογα με την προσέγγιση που θα πραγματώσει την ανάλυση συναισθήματος. Οι διάφορες προσεγγίσεις και τεχνικές ανάλυσης δεδομένων θα αναφερθούν λεπτομερώς στα επόμενα κεφάλαια.

2.2 Εφαρμογές της Ανάλυσης Συναισθήματος

Στην σημερινή εποχή η Ανάλυση Συναισθήματος εφαρμόζεται σε διάφορες κλάδους της καθημερινότητας μας και παρέχει πολλά πλεονεκτήματα και διευκολύνσεις. Μερικά από τα παραδείγματα εφαρμογής τη είναι τα εξής:

Εξυπηρέτηση Πελατών: Γίνεται χρήση της για να αξιολογηθούν τα συναισθήματα των πελατών μιας εταιρίας σε πραγματικό χρόνο με απώτερο σκοπό την δημιουργία εξατομικευμένων απαντήσεων με βάση την συνομιλία που έχει γίνει μεταξύ των πελατών και ενός αυτόματου βοηθού(bot) σε μια σελίδα ή ένα τηλεφωνικό κέντρο. Με αυτόν τον τρόπο τα θέματα των πελατών αντιμετωπίζονται αποτελεσματικότερα και γρηγορότερα, δίνοντας βάση σε αυτά που διαθέτουν σε μεγαλύτερο βαθμό αρνητικό συναίσθημα(για παράδειγμα θα μπορούσαμε να μιλάμε για μία κλίμα της τάξεως 1-5 που το 5 είναι το πολύ θετικό και το 1 είναι το πολύ αρνητικό) . Επιπλέον δίνεται η δυνατότητα στις εταιρίες μέσω των δεδομένων που συλλέγουν από την ανάλυση συναισθήματος να έχουν πληροφόρηση σχετικά με το αν μένουν οι πελάτες τους ικανοποιημένοι και από ποιες υπηρεσίες-προϊόντα, ποιες χρειάζονται βελτίωση καθώς και το πως θα μπορούσαν να επιλύσουν τα προβλήματα των πελατών τους αποτελεσματικότερα ή ακόμα και πρόληψη μελλοντικών προβλημάτων.

Πρόληψη Κρίσεων: Είναι γνωστό πως τα μέσα κοινωνικής δικτύωσης χρησιμοποιούν πληθώρα αλγορίθμων και έχουν συνεχή προσθήκη καινούργιων τεχνολογιών (πειραματικών ή και μη). Μέσα σε αυτό το πλήθος χρησιμοποιούν και αλγορίθμους Ανάλυσης Συναισθήματος οι οποίοι παρακολουθούν και ελέγχουν το περιεχόμενο που «ανεβάζουν» οι χρήστες στα social media. Οι αλγόριθμοι αυτοί συλλέγουν διάφορες αναφορές προκαθορισμένων λέξεων κλειδιών. Μέσω της μελέτης και χρήσης αυτών μια επιχείρηση που πραγματοποιεί μια διαφημιστική καμπάνια στο διαδίκτυο η και ένας χρήστης μπορεί να συντομεύσει η να αποτρέψει μια κρίση-αρνητική στάση σχετικά με τις δημοσιεύσεις τους στα social media. Επιπλέον πρέπει να σημειωθεί πως η ανάλυση συναισθημάτων παίζει σημαντικό ρόλο και στην ανίχνευση κακόβουλου, ενοχλητικού, ακατάλληλου, και προσβλητικού περιεχομένου στα social media μέσω διάφορων τεχνικών όπως το φιλτράρισμα συναισθήματος, ανίχνευση σεξουαλικού περιεχομένου, αντιμετώπιση καταχρηστικής συμπεριφοράς και πολλά άλλα. Οι διάφορες τεχνικές που προσφέρονται στην

ανάλυση συναισθήματος και ο συνδυασμός αυτών συμβάλουν στην δημιουργία μέτρων αντιμετώπισης τέτοιων κρίσεων στα μέσα κοινωνικής δικτύωσης και οδηγεί στη χρήση αυτοματοποιημένων μηχανισμών ban και φιλτραρίσματος δημοσιεύσεων-περιεχομένου που κρατάει τον χώρο των social media φιλικό και ασφαλές για τον χρήστη.

Έρευνα Αγοράς: Η ανάλυση συναισθήματος πέρα από τη χρήση της από επιχειρήσεις για βελτίωση των εκάστοτε προϊόντων και των υπηρεσιών της καθώς και την παρακολούθηση της πορείας-αποτελεσματικότητας της επωνυμίας τους είναι πολύ χρήσιμο εργαλείο και στην πραγμάτωση της έρευνας αγοράς. Μέσω της ανάλυσης των δεδομένων που γίνεται σε διάφορα social media, forums, κριτικές σε ηλεκτρονικά καταστήματα οι εταιρίες μπορούν να κατανοήσουν την τάση της αγοράς και να δημιουργήσουν αντιπροσωπευτικές στρατηγικές με σκοπό την κάλυψη των αναγκών του αγοραστικού κοινού καθώς και να συμβαδίζουν με τις εξελίξεις και μελλοντικές απαιτήσεις της αγοράς ή ακόμα και να τις προλαμβάνουν.

Ποιότητα Υγείας: Η χρήση της ανάλυσης συναισθήματος στον τομέα της υγείας και ειδικότερα για την ανάλυση της ποιότητας της περίθαλψης των ασθενών μπορεί να κριθεί ως ένα πολύ θετικό πρόσθετο. Με την χρήση της συλλέγονται δεδομένα από τους ασθενείς τα οποία αναλύονται και βοηθούν τους ειδικούς στην υγεία να κατανοήσουν την ποιότητα της περίθαλψης που προσφέρουν στους ασθενείς, αν αυτή είναι ικανοποιητική ή όχι και ανάλογα με αυτό να προβεί σε μέτρα που θα την βελτιώσουν. Έτσι οι ασθενείς θα λαμβάνουν μια ποιοτικότερη περίθαλψη και νιώθουν πως η άποψη τους αποτελεί σημαντικό στοιχείο για την βελτίωση των υπηρεσιών του τομέα υγείας. Επιπλέον μιας και είναι γνωστό το ότι ο τομέας της ανάλυσης συναισθήματος στην πληροφορική έχει σχέση και με τον τομέα της ψυχολογίας είναι βάσιμο να πούμε πως μέσω των εργαλείων που παρέχει η ανάλυση συναισθήματος μπορεί αυτή να βοηθήσει στην πρόωρη και γρηγορότερη διάγνωση της ψυχικής κατάστασης ενός ασθενή. Με την χρήση της από έναν ειδικό στον τομέα της ψυχολογίας μπορεί να αποτελέσει χρήσιμο εργαλείο και να κάνει το έργο του ακόμα ευκολότερο και καλύτερο.

2.3 Προβλήματα της Ανάλυσης Συναισθήματος

Η ανάλυση συναισθήματος παρότι αποτελεί σημαντική λειτουργία και εξέλιξη στον τομέα της τεχνολογίας δεν αναιρεί το γεγονός του ότι παρουσιάζει προβλήματα όπως κάθε τεχνολογία.

Υπάρχουν λοιπόν προβλήματα τα οποία δυσκολεύουν το να εκτελεστεί σωστά. Αυτά μπορούν να έχουν σχέση με το μήκος του κειμένου ή την ύπαρξη πολλαπλών γλωσσών σε ένα κείμενο καθώς και το ότι το συναίσθημα πολλές φορές δεν γίνεται εύκολα κατανοητό-διακριτό, μπορεί να εκφράζεται σε ένα κείμενο με πιο έμμεσο τρόπο. Επιπλέον σε λάθος συμπεράσματα μπορεί να οδηγηθεί η ανάλυση συναισθήματος και λόγω της άποψης του εκφραστή που διατυπώνεται σε ένα κείμενο ή ακόμα και από την διαφορετική σειρά λέξεων που συχνά μπορεί να οδηγήσει σε διαφορετικό συναίσθημα.

2.3.1 Προβλήματα της ανάλυσης συναισθήματος ανάλογα με την προσέγγιση που επιλέγουμε

Ανάλογα με την προσέγγιση που θα επιλέξουμε για την εκτέλεση της ανάλυσης συναισθήματος προκύπτουν και τα αντίστοιχα προβλήματα που παρουσιάζονται παρακάτω :

Λεξιλογική προσέγγιση	Μηχανική Μάθηση
Το πλήθος των λέξεων είναι συγκεκριμένο-πεπερασμένο, σε αντίθεση με την ανάλυση σε ένα δυναμικό μέσο.	Το κόστος είναι μεγαλύτερο σε σχέση με την λεξιλογική προσέγγιση.
Ίδιες λέξεις μπορεί να εκφράζουν διαφορετικό συναίσθημα με βάση και τα διαφορετικά περιεχόμενα στα οποία περιέχονται.	Πολλές φορές υπάρχει δυσκολία στην εύρεση ικανοποιητικού-αντιπροσωπευτικού δείγματος για να εκπαιδευτεί ο ταξινομητής.
Μειωμένη προσαρμοστικότητα καθώς τα λεξικά είναι στατικά και δεν προσαρμόζονται εύκολα σε νέα δεδομένα ή αλλαγές στη γλώσσα.	Απαιτείται παραπάνω χρόνος για την εκπαίδευση του ταξινομητή.
Δυσκολία στην μέτρηση της έντασης του συναισθήματος καθώς τα περισσότερα λεξικά κατατάσσουν τις λέξεις ως θετικές ή αρνητικές.	Οι ταξινομητές δεν γίνεται να εκπαιδευτούν σε παραπάνω από ένα πεδίο.

3) Λεξιλογική προσέγγιση

3.1 Ορισμός λεξιλογικής προσέγγισης-τι είναι ακριβώς?

Ως Λεξιλογική προσέγγιση στο πεδίο της ανάλυσης συναισθήματος ορίζεται η μέθοδος που βασίζεται στην ιδέα του ότι οι λέξεις φέρουν κάποιο συναισθηματικό νόημα το οποίο εξάγεται και κατηγοριοποιείται με διαφόρους τρόπους. Για τον εντοπισμό και την κατηγοριοποίηση μεμονωμένων λέξεων, φράσεων, προτάσεων ή κειμένων που περιέχουν κάποιο συναισθηματικό νόημα χρησιμοποιείται πληθώρα τεχνικών που κάποιες από αυτές έχουν ήδη αναφερθεί παραπάνω όπως είναι τα **λεξικά συναισθημάτων (που εντοπίζει το συναισθηματικό σκορ μιας λέξης-φράσης ενός κειμένου), *keyword spotting, lexical affinity κ.α.** στα οποία θα αναφερθούμε με μεγαλύτερη λεπτομέρεια παρακάτω.

3.1.1 Πρακτικά πως λειτουργεί?

Ας δούμε ένα απλό παράδειγμα λειτουργίας της λεξιλογικής προσέγγισης, ας πούμε πως έχουμε μια κριτική για μια ταινία από έναν τυχαίο χρήστη στο διαδίκτυο, χρησιμοποιώντας για παράδειγμα κάποια έτοιμα λεξικά συναισθημάτων όπως το **LIWC** ή το **SentiWordNet** μπορούμε να αναλύσουμε την συγκεκριμένη κριτική ανάλογα με τις λέξεις που περιέχει. Θα μπορούσε ας πούμε να περιέχει την λέξη «απίστευτη» που θα είχε θετικό συναισθηματικό σκορ 1 και την λέξη «χάλια» που θα είχε αρνητικό συναισθηματικό σκορ -1, κάθε λέξη θα περιέχει και ένα συγκεκριμένο βαθμό πολικότητας που τέθηκε είτε από αυτόν που εκτελεί την ανάλυση συναισθήματος, είτε από ένα λεξικό συναισθημάτων που χρησιμοποιεί, είτε από την μέθοδο που χρησιμοποιείται κατά την έρευνα. Γίνεται λοιπόν ανάλυση όλων των λέξεων που περιέχονται στην κριτική και στο τέλος ερμηνεύονται τα αποτελέσματα. Για παράδειγμα αν έχουμε 10 λέξεις, 5 θετικές και 5 αρνητικές, το αποτέλεσμα του συναισθηματικού σκορ θα ήταν 0 υποδηλώνοντας μια ουδέτερη κριτική, ή ομοίως να έχουμε 10 θετικές λέξεις κατηγοριοποιώντας την παραπάνω κριτική ως θετική ή 10 αρνητικές κατηγοριοποιώντας την παραπάνω κριτική ως αρνητική. Τα

πλαίσια κατηγοριοποίησης μια κριτικής ή ενός κειμένου η λέξεων στην ανάλυση συναισθήματος ασχέτου προσέγγισης καθορίζονται από τον χρήστη που την πραγματώνει. Το παραπάνω παράδειγμα αποτελεί μια από τις πιο απλές περιπτώσεις λειτουργίας της ανάλυσης συναισθήματος καθώς υπάρχουν περισσότερες τεχνικές και μέθοδοι πέρα από τα λεξικά συναισθημάτων όπως είπαμε ήδη οι οποίες λαμβάνουν άλλα περισσότερα, λιγότερα ή άλλα στοιχεία υπόψιν κατά την λειτουργία της ανάλυσης συναισθήματος.

3.1.2 Λεξικά συναισθημάτων

Τα **λεξικά συναισθήματος** (Τα **Sentiment Lexicons**) αποτελούν ένα από τα πιο σημαντικά εργαλεία για την λεξιλογική προσέγγιση στην ανάλυση συναισθήματος. Στις μεθόδους που κάνουν χρήση λεξικών συναισθημάτων περιλαμβάνεται ο υπολογισμός του συναισθηματικού προσανατολισμού ενός κειμένου από τις λέξεις, φράσεις, προτάσεις που το αποτελούν. Τα λεξικά αυτά μπορούν να κατασκευαστούν χειροκίνητα ή και αυτόματα και να περιέχουν κάποιες αρχικές λέξεις (seed words) ώστε να επεκταθεί-ενημερωθεί η λίστα με τις λέξεις. Οι λέξεις που βρίσκονται μέσα στα λεξικά συναισθημάτων διαθέτουν ένα συναισθηματικό σκορ σε μία κλίμακα του τύπου (θετικό-αρνητικό) ή για τον εντοπισμό κάποιου βασικού συναισθήματος. Μέσα στα Sentiment Lexicons μπορεί να περιέχονται λέξεις που σχετίζονται με πληθώρα συναισθημάτων (χαρά, λύπη, θυμός κλπ), ταξινομήσεις λέξεων και το συναίσθημα που φέρουν ή ακόμα και βαθμολογίες που δείχνουν την ένταση του συναισθήματος για την εκάστοτε λέξη που περιέχεται σε αυτό. Στην ουσία λοιπόν τα λεξικά αυτά είναι λίστες από λέξεις και εκφράσεις συναισθημάτων και διάφορα από αυτά βρίσκονται στο διαδίκτυο και είναι διαθέσιμα για χρήση, συνήθως περιέχουν έναν μεγάλο αριθμό από χιλιάδες όρους και είναι αρκετά χρήσιμα ως συναισθηματικές λέξεις και φράσεις (sentiment words, polar words κλπ). Για παράδειγμα μπορεί να περιέχονται οι λέξεις καλός, όμορφος, ευγενικός που κατηγοριοποιούνται ως θετικές λέξεις έχοντας θετικό συναισθηματικό σκορ για ένα κείμενο και λέξεις όπως κακός, απαίσιος, ασεβής που κατηγοριοποιούνται ως αρνητικές λέξεις. Τα λεξικά συναισθημάτων όμως, όπως και τα μοντέλα μηχανικής μάθησης δεν δίνετε να είναι 100% ακριβής και αρκετές φορές μπορεί να περιλαμβάνουν λάθη. Μερικές από τις περιπτώσεις που μπορούμε να συναντήσουμε κατά τη χρήση ενός λεξικού συναισθημάτων ή ενός μοντέλου μηχανικής μάθησης φαίνονται στην παρακάτω εικόνα.

True Positive	False Positive
Reality: I have a valid ticket. Ticket Validator: Yes, it's valid	Reality: I have an invalid ticket. Ticket Validator: Yes, it's valid
False Negative	True Negative
Reality: I have a valid ticket. Ticket Validator: No, it's invalid.	Reality: I have an invalid ticket. Ticket Validator: No, it's invalid

(Στην εικόνα αυτή φαίνονται οι πιο συχνές περιπτώσεις αποτελεσμάτων που μπορούν να φέρουν τα lexicons ή κάποιο μοντέλο μηχανικής μάθησης κατά την εκτέλεση ανάλυσης συναισθήματος σε ένα κείμενο.)

Παραπάνω μας δίνεται ένα απλό, τυχαίο παράδειγμα μιας περίπτωσης που μπορεί να φέρει ως αποτέλεσμα το μοντέλο που χρησιμοποιούμε στην ανάλυση συναισθήματος (**1 περίπτωση από τις 4 αυτές**). Συγκεκριμένα:

1^η περίπτωση: True positive, Αντιπροσωπεύει την περίπτωση που το κείμενο μας έχει θετικό συναίσθημα και το λεξικό-μοντέλο μας το αναγνωρίζει και αυτό ως θετικό.

2^η περίπτωση: False positive, Αντιπροσωπεύει την περίπτωση που το κείμενο μας δεν έχει θετικό συναίσθημα αλλά το λεξικό-μοντέλο μας το ταξινομεί ως θετικό.

3^η περίπτωση: False Negative, Αντιπροσωπεύει την περίπτωση που το κείμενο έχει θετικό συναίσθημα αλλά το λεξικό-μοντέλο μας το ταξινομεί ως αρνητικό .

4^η περίπτωση: True Negative, Αντιπροσωπεύει την περίπτωση που το κείμενο μας δεν έχει θετικό συναίσθημα και το λεξικό-μοντέλο μας το ταξινομεί ως αρνητικό.

Πρέπει να σημειωθεί πως το συγκεκριμένο παράδειγμα **εστιάζει μόνο στην ανάλυση θετικού συναισθήματος**, η λογική που ακολουθείται εδώ βέβαια είναι παρόμοια με αυτή της ανάλυσης κάποιου αρνητικού συναισθήματος με τις αντίστροφες ερμηνείες.

3.1.2.1 Τρόποι Δημιουργίας Λεξικών Συναισθημάτων

Υπάρχουν τρεις τρόποι δημιουργίας Λεξικών Συναισθημάτων:

Χειροκίνητη (μη αυτόματη) προσέγγιση^[5]:

Στην συγκεκριμένη προσέγγιση κατηγοριοποιούμε λέξεις και φράσεις σε κάποια συναισθηματική κατηγορία του τύπου «θετική-αρνητική». Αυτή η προσέγγιση είναι καλή για χρήση όταν η διαδικασία πρόκειται να γίνει μόνο μια φορά.

Corpus-based προσέγγιση^[5]:

Εδώ χρησιμοποιείται συνήθως το **double-propagation** για να αποδοθεί συναισθηματικό σκορ σε λέξεις που δεν υπάρχουν στο ήδη λεξικό. Βασίζεται στην ιδέα του ότι οι λέξεις που εμφανίζονται σε παρόμοια συμφραζόμενα με λέξεις που διαθέτουν γνωστό συναίσθημα πιθανόν θα έχουν παρόμοιο συναισθηματικό σκορ με αυτές τις λέξεις. Αυτή η μέθοδος επιτρέπει την αυτοματοποιημένη δημιουργία λεξικών συναισθημάτων σε μεγάλη κλίμακα αλλά μπορεί να αποτελέσει χρονοβόρα διαδικασία που απαιτεί και μεγάλο όγκο δεδομένων παρότι λαμβάνει υπόψιν το συναίσθημα του «περιβάλλοντος» στο οποίο εμφανίζεται μια λέξη.

Dictionary-based προσέγγιση^[5]:

Εδώ συνήθως γίνεται χρήση ομάδων συνωνύμων και ιεραρχίες ώστε να γίνει εξαγωγή των λέξεων που εκφράζουν κάποιο συναισθηματικό περιεχόμενο. Η μέθοδος αυτή βασίζεται σε λεξικά και γλωσσικούς πόρους για να δώσει συναισθηματικό σκορ σε λέξεις. Το τελικό αποτέλεσμα είναι λέξεις των οποίων η πολικότητα τους δεν εξαρτάται από το γενικό πλαίσιο μέσα στο οποίο βρίσκονται., βασίζονται σε συντακτικά μοτίβα και χρειάζονται μεγάλο όγκο δεδομένων.

3.1.2.2 Παραδείγματα Λεξικών Συναισθημάτων

Μερικά Παραδείγματα Λεξικών αποτελούν τα παρακάτω:

LIWC

Χρησιμοποιείται για την ανάλυση κειμένων για τον εντοπισμό και πιστοποίηση διάφορων ψυχολογικών χαρακτηριστικών της γραφής. Βασίζεται σε μια λίστα λέξεων που είναι ταξινομημένες σε 94 κατηγορίες, οι οποίες αντιπροσωπεύουν διάφορες ψυχολογικές έννοιες. Υπάρχουν κατηγορίες για συναισθήματα(χαρά, θλίψη, θυμός κλπ) , γνωστικές διεργασίες (σκέψη, δράση, αντίληψη κλπ), κοινωνικές αλληλεπιδράσεις(οικογένεια, θρησκεία, εργασία κλπ)και πολλά άλλα που εκφράζονται σε ένα κείμενο, το LIWC μετράει πόσες φορές εμφανίζονται η λέξεις της εκάστοτε κατηγορίας και από τις συχνότητες αυτές υπολογίζει τα ποσοστά που αντιπροσωπεύουν την έκφραση των αντιστοίχων ψυχολογικών χαρακτηριστικών στο κείμενο. Το λεξικό αυτό περιλαμβάνει πάνω από 4000 λέξεις και φράσεις ταξινομημένες στις κατηγορίες τις οποίες έχουμε αναφέρει ήδη και εφαρμόζεται με τα ανάλογα λογισμικά διάφορων platforms (Windows, Linux, Mac) σε διάφορες περιοχές και ερευνητικά πεδία όπως, **ψυχολογία, γλωσσολογία, εκπαίδευση, μάρκετινγκ, ιατρική** και πολλά άλλα. Αποτελεί ένα πολύ χρήσιμο εργαλείο για την ποσοστοποίηση ψυχολογικών χαρακτηριστικών της γραφής που εφαρμόζεται και στην λεξιλογική προσέγγιση της ανάλυσης συναισθήματος

SentiWordNet

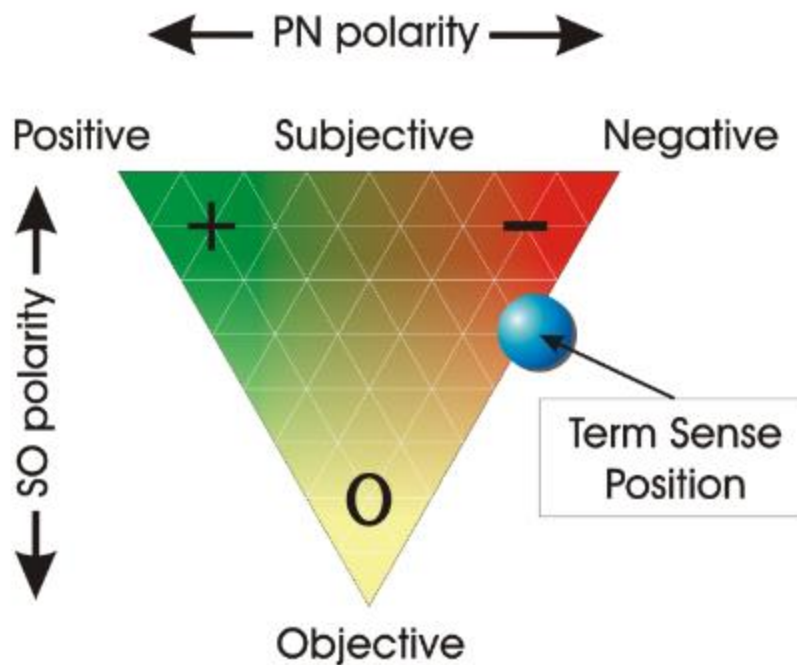
Το SentiWordNet^[3] αποτελεί και αυτό ένα λεξικό συναισθημάτων που βασίζεται όμως στο **WordNet**, ένα ευρέως διαδεδομένο λεξικό αγγλικών λέξεων. Περιλαμβάνει περισσότερες από 11.000 λέξεις και φράσεις από το WordNet έχοντας η κάθε μια ένα ή περισσότερα συναισθηματικά σκορ. Στην ουσία το SentiWordNet αναθέτει συναισθηματικά σκορ σε **σύνολα (synets)** λέξεων και φράσεων από το WordNet με τη διαφορά όμως του ότι σε σχέση με το LIWC τα συναισθηματικά αυτά σκορ εκφράζουν κάποιου βαθμού θετικότητα, αρνητικότητα ή ουδετερότητα μια λέξης-πρότασης σε ένα κείμενο που μελετάμε. Εφόσον το λεξικό μας λειτουργεί με σύνολα λέξεων-φράσεων του WordNet θα αποδώσει διαφορετικό συναισθηματικό σκορ σε μια λέξη ανάλογα με το νόημα αυτής. Για παράδειγμα αν έχουμε την λέξη «**attractive**» μπορεί να έχει τα δύο παρακάτω νοήματα.

1) «pleasing to the eye or mind especially through beauty and charm»

2) «having the properties of a magnet; the ability to draw or pull»

Είναι προφανές ότι το πρώτο νόημα της λέξης attractive θα κατέχει κάποιο θετικό συναισθηματικό σκορ ενώ το δεύτερο ένα ουδέτερο συναισθηματικό σκορ.

Όπως και το LIWC το SentiWord έχει και αυτό εφαρμογή σε διάφορες ερευνητικές και εφαρμοσμένες περιοχές πέρα της ανάλυσης συναισθημάτων όπως: **Εξόρυξη γνώμης**(Εξαγωγή απόψεων και αξιολογήσεων από κείμενα), **Συστήματα σύστασης** (για προτάσεις προϊόντων ή υπηρεσιών που ταιριάζουν στις προτιμήσεις του εκάστοτε χρήστη) , **Μηχανική μετάφραση** (λαμβάνοντας υπόψιν και τα συναισθήματα κατά τη μετάφραση ενός κειμένου) και πολλά ακόμα.



(Αυτή η εικόνα αποτελεί την γραφική αναπαράσταση που υιοθετήθηκε από το Senti-WordNet για την αναπαράσταση των ιδιοτήτων που σχετίζονται με τις ιδιότητες της έννοιας μιας λέξης.)

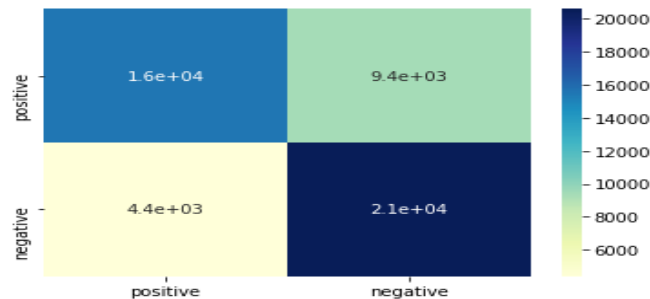
AFINN LEXICON

Θεωρείται ένα από τα πιο απλά και διάσημα λεξικά για την ανάλυση συναισθήματος. Η πιο πρόσφατη έκδοση του AFINN^[3] είναι η AFINN-en-165.txt που περιέχει 3382 λέξεις μαζί με τα σκορ πολικότητας τους. Το λεξικό αυτό βασίζεται σε λίστες λέξεων που έχουν συλλεχθεί από πληθώρα πηγών στο διαδίκτυο όπως blogs, tweets, και άλλα λεξικά συναισθημάτων και έχει ως σκοπό να αναθέτει συναισθηματικά σκορ σε λέξεις της αγγλικής γλώσσας. Αντίθετα με τα παραπάνω λεξικά στο λεξικό AFINN κάθε λέξη-φράση έχει ένα σκορ από -5 έως 5 (όπου το -5 αντιπροσωπεύει το πολύ αρνητικό συναίσθημα, το 0 αντιπροσωπεύει ουδέτερο συναίσθημα, και το 5 αντιπροσωπεύει το πολύ θετικό συναίσθημα). Μερικές από τις εφαρμογές του λεξικού αυτού πέραν της ανάλυσης συναισθήματος είναι η εξόρυξη γνώμης, συστήματα σύστασης, μηχανική μετάφραση και πολλά ακόμα. Τα θετικά του λεξικού αυτού είναι το ότι είναι απλό στην χρήση, διατίθεται υπό ελεύθερη άδεια δηλαδή, μπορεί να χρησιμοποιηθεί χωρίς περιορισμούς και έχει μεγάλη αποτελεσματικότητα κατά την εφαρμογή του στην ανάλυση συναισθήματος έχοντας **72%** περισσότερη ακρίβεια από άλλα λεξικά μοντέλα. Από την άλλη μεριά όμως ένα αρνητικό του είναι πως περιέχει ένα μικρότερο εύρος λέξεων σε σχέση με άλλα λεξικά.

Accuracy: 0.72308
Precision: 0.7325462842911575
Recall: 0.72308
F1 Score: 0.7202328730646428
Model Report:

	precision	recall	f1-score	support
negative	0.78	0.62	0.69	25000
positive	0.69	0.82	0.75	25000
accuracy			0.72	50000
macro avg	0.73	0.72	0.72	50000
weighted avg	0.73	0.72	0.72	50000

CONFUSION MATRIX:

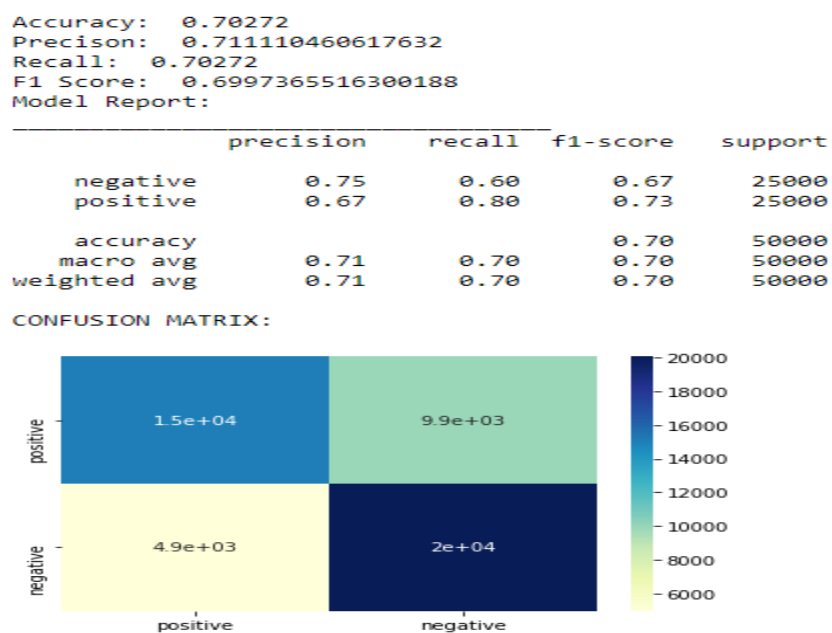


Εικόνα χαρακτηριστικών απόδοσης του AFINN Lexicon.

VADER (Valence Aware Dictionary and Sentiment Reasoner)

Το Vader ^[3] αποτελεί ένα λεξικό και εργαλείο στην ανάλυση συναισθήματος που λειτουργεί βάση κανόνων στην ανάλυση συναισθήματος και ειδικεύεται στην μελέτη συναισθημάτων που εκφράζονται μέσα από κείμενα στα social media (μέσα κοινωνικής δικτύωσης). Το Vader lexicon είναι ικανό να εντοπίσει την **πόλωση** και την **ένταση** ενός συναισθήματος σε ένα κείμενο στο οποίο εφαρμόζεται ανάλυση συναισθήματος με χρήση αυτού. Περιλαμβάνει περισσότερες από 8000 λέξεις και φράσεις της αγγλικής γλώσσας, έχοντας ένα η περισσότερα συναισθηματικά σκορ για αυτές. Όσον αφορά τον τρόπο λειτουργίας του το Vader lexicon χρησιμοποιεί **κανόνες** και **στατικά μοντέλα** για να αναθέσει σκορ σε λέξεις, φράσεις και κείμενα λαμβάνοντας υπόψιν και το λεξιλόγιο (εμφάνιση συγκεκριμένων λέξεων), τη γραμματική (σημεία στίξης, κεφαλαία), μορφολογία (emojis, σύμβολα) και την σύνταξη (δομή προτάσεων). Διατίθεται λογισμικό για την ενσωμάτωση του Vader σε διάφορες πλατφόρμες (Python, Java, R) και έχει εφαρμογές παρόμοιες με τα προηγούμενα μοντέλα που παρουσιάστηκαν όπως την Εξόρυξη γνώμης, συστήματα σύστασης, ανάλυση μέσων κοινωνικής

δικτύωσης, ανάλυση μάρκετινγκ και πολλά ακόμα. Κάποια από τα πλεονεκτήματα του Vader είναι το ότι είναι εύκολο στη χρήση, ακριβές (έχει αποδειχθεί αποτελεσματικό σε διάφορες εφαρμογές ανάλυσης συναισθήματος), και αποτελεί ένα λεξικό **πλήρους ανοιχτού κώδικα** που μπορεί να βρεθεί στο πακέτο **NLTK** και μπορεί να εφαρμοστεί κατευθείαν σε **μη ταξινομημένα δεδομένα**. Αντίθετα τώρα, κάποια από τα αρνητικά του λεξικού αυτού αποτελεί το περιορισμένο λεξιλόγιο του (περιλαμβάνει μικρότερο αριθμό λέξεων σε σύγκριση με άλλα λεξικά που υπάρχουν) καθώς και το ότι το λεξικό αυτό είναι χρήσιμο μόνο αν η εφαρμογή του γίνεται σε κάποιο κείμενο στην αγγλική γλώσσα (κρίνεται απαραίτητο για την λειτουργία του το αντικείμενο που μελετάμε να είναι στην αγγλική γλώσσα).



Εικόνα χαρακτηριστικών απόδοσης του εργαλείου VADER.

3.2 Πλεονεκτήματα και μειονεκτήματα της Λεξιλογικής Προσέγγισης

Όπως είδαμε ήδη παραπάνω μπορούμε να έχουμε διαφορετικές προσεγγίσεις κατά τη χρήση της ανάλυσης συναισθήματος, (λεξιλογική προσέγγιση, προσέγγιση μηχανικής μάθησης ή κάποιου είδους υβριδική προσέγγιση). Κάθε προσέγγιση χρησιμοποιείται από τους χρήστες για διάφορων ειδών μελέτες-λειτουργίες έχοντας πάντα ως στόχο τα καλύτερα δυνατά αποτελέσματα στην ανάλυση συναισθήματος είτε αφορά την ανάλυση κάποιου κείμενου, είτε για ποιο πολύπλοκες λειτουργίες στα πεδία της υγείας, της τεχνολογίας, του μάρκετινγκ και πολλά άλλα. Όλες οι προσεγγίσεις έχουν τα θετικά και τα αρνητικά σημεία τους όσον αφορά την ανάλυση συναισθήματος και παρότι καμιά από αυτές δεν φέρνει 100% εγκυρότητα όσον αφορά τα αποτελέσματα όλες χρησιμοποιούνται σε ικανοποιητικό βαθμό από τους χρήστες σήμερα. Η λεξιλογική προσέγγιση παρότι δεν είναι όσο ακριβής είναι η προσέγγιση μηχανικής μάθησης συνεχίζει να χρησιμοποιείται από τους χρήστες για διάφορους λόγους. Παρακάτω παρουσιάζονται τα πλεονεκτήματα και μειονεκτήματα από τη χρήση της λεξιλογικής προσέγγισης στην ανάλυση συναισθήματος.

Πλεονεκτήματα της Λεξιλογικής προσέγγισης:

- 1) Οι μέθοδοι λεξιλογικής προσέγγισης και τα lexicons γενικότερα είναι πιο εύκολα προσβάσιμα και διαθέσιμα στο ευρύ κοινό (ένα παράδειγμα αυτού αποτελεί το SentiWordNet).
- 2) Η προσέγγιση αυτή είναι οικονομικότερη καθώς δεν περιλαμβάνει την χρήση εξελιγμένων αλγορίθμων ανάλυσης συναισθήματος που χρησιμοποιούνται στα μοντέλα μηχανικής μάθησης.
- 3) Δεν απαιτείται η εκπαίδευση ενός μοντέλου όπως στην μηχανική μάθηση, τα λεξικά πέρα από τι λέξεις που περιέχουν πολλές φορές είναι ταξινομημένα, διαθέτουν συναισθηματικό σκορ, κάποια ετικέτα καθώς και κάποιον αριθμό που δείχνει την ένταση ενός συναισθήματος είτε έχουν κατηγοριοποιηθεί ως (θετικά, αρνητικά, ουδέτερα) ή με βάση συγκεκριμένων αισθημάτων (χαρά, λύπη, οργή).
- 4) Είναι ευκολότερη η χρήση της σε σχέση με την προσέγγιση μηχανικής μάθησης (λόγο της χρήσης λεξικών που δεν περιλαμβάνουν κάποια δραστηριότητα επιμόρφωσης ενός μοντέλου για την ανάλυση συναισθήματος).

Μειονεκτήματα της Λεξιλογικής προσέγγισης:

- 1) Οι μέθοδοι ανάλυσης συναισθήματος στην λεξιλογική προσέγγιση αντιμετωπίζουν δυσκολία στον εντοπισμό σαρκασμού, ορθογραφικών λαθών, γραμματικά λάθη, η ειρωνεία, Λόγο αυτού δεν είναι αρκετά έμπιστα για την ανάλυση δεδομένων που έχουν ως πηγή το διαδίκτυο και κυρίως τα μέσα κοινωνικής δικτύωσης.
- 2) Λόγο του ότι η κατηγοριοποίηση δεδομένων εδώ βασίζεται σε ετικέτες και κανόνες κάποια εταιρία που θα την χρησιμοποιήσει θα πρέπει να έχει στην διάθεση της ένα ικανοποιητικό μέγεθος δεδομένων για την δημιουργία ενός αξιόπιστου lexicon (σε περίπτωση που δεν θέλει η δεν μπορεί να χρησιμοποιήσει κάποιο που υπάρχει ήδη).
- 3) Λόγου του ότι τα δεδομένα κατηγοριοποιούνται και παίρνουν κάποια ετικέτα μηχανικά (δηλαδή από κάποιον άνθρωπο και όχι αυτόματα όπως στα μοντέλα μηχανικής μάθησης) η προετοιμασία των δεδομένων στα lexicons μπορεί να είναι αρκετά χρονοβόρα.
- 4) Είναι αρκετά επιρρεπής στις προκαταλήψεις και τις γνώσεις του ανθρώπου που δημιουργεί ένα λεξικό συναισθημάτων που χρησιμοποιείται στην λεξιλογική προσέγγιση. Εάν τα άτομα που ετοιμάζουν το λεξικό δεν διαθέτουν επαρκές γνώσεις σε κάποιο τομέα που αναλύονται τα αποτελέσματα της ανάλυσης συναισθήματος δεν μπορούν να θεωρηθούν ακριβή.

4) Προσέγγιση Μηχανικής Μάθησης

4.1 Ορισμός Μηχανικής Μάθησης

Για να γίνει κατανοητή η προσέγγιση μηχανικής μάθησης στο πεδίο της ανάλυσης συναισθήματος πρέπει να κατανοήσει κανείς την έννοια της μηχανικής μάθησης^[11]. Ως **μηχανική μάθηση** αποκαλείται το πεδίο της επιστήμης των υπολογιστών που αναπτύχθηκε από τη μελέτη της **αναγνώρισης προτύπων** και της **υπολογιστικής θεωρίας μάθησης** στην **τεχνητή νοημοσύνη**. Συγκεκριμένα το 1959 ο **Άθρουρ Σάμουελ** (Αμερικανός πρωτοπόρος στην τεχνητή νοημοσύνη) ήταν ο άνθρωπος που επινόησε πρώτος τον όρο τεχνητή νοημοσύνη δίνοντας στον όρο αυτό το εξής νόημα «πεδίο μελέτης που δίνει την δυνατότητα στους ηλεκτρονικούς υπολογιστές να μαθαίνουν **χωρίς να έχουν προγραμματιστεί ρητά** από κάποιο άνθρωπο». Συγκεκριμένα η μηχανική μάθηση αποτελεί υποκομμάτι της **τεχνητής νοημοσύνης** που εκείνη αναφέρεται γενικότερα στην ικανότητα μιας μηχανής να μπορεί να αναπαράγει κάποιες γνωστικές λειτουργίες που διαθέτει ο άνθρωπος, όπως (η μάθηση, η σχεδίαση και η δημιουργικότητα). Η τεχνητή νοημοσύνη δίνει τη δυνατότητα στους υπολογιστές να μπορούν να κατανοήσουν το περιβάλλον τους, να λύνουν προβλήματα και να μπορούν να ολοκληρώνουν τους στόχους που έχουν θέσει. Με απλά λόγια λοιπόν ένας υπολογιστής δέχεται δεδομένα, τα επεξεργάζεται με διάφορους τρόπους (ειδικούς αλγορίθμους, χρησιμοποιώντας κάποιο συγκεκριμένο εκπαιδευμένο μοντέλο μηχανικής μάθησης) ή λαμβάνονται ήδη επεξεργαζόμενα και στη συνέχεια αναλύονται από τον υπολογιστή ώστε να λάβει αποφάσεις σχετικά με της επόμενες ενέργειες του καθώς και για παρόμοιες μελλοντικές αποφάσεις. Η μηχανική μάθηση (Machine Learning) βοηθάει σε μεγάλο βαθμό στην διαχείριση και σωστή χρήση ενός μεγάλου όγκου δεδομένων τα οποία ονομάζουμε σήμερα ως **Big Data** με την χρήση των ανάλογων μοντέλων. Αυτά τα μοντέλα μηχανικής μάθησης τροφοδοτούνται με διάφορα δεδομένα τα οποία αναλύουν, επεξεργάζονται και κατηγοριοποιούν. Στη συνέχεια τα μοντέλα αυτά εξάγουν τα συμπεράσματα τους και κρατάνε τα δεδομένα αυτά που τα βοηθάνε να βελτιώνονται (μαθαίνουν από αυτά ώστε να φέρουν πιο ακριβή αποτελέσματα σε μικρότερο χρονικό διάστημα στο μέλλον κατά τη χρήση τους). Οι υπολογιστές μέσα από την διαδικασία αυτή μαθαίνουν να αναγνωρίζουν συγκεκριμένα μοτίβα και να ανεξαρτητοποιούνται (δηλαδή λειτουργούν με βάση το μοντέλο μηχανικής μάθησης που εφαρμόζουν αντί κάποιου συγκεκριμένου κώδικα ορισμένο από κάποιον προγραμματιστή) εκτελώντας έτσι λειτουργίες με

δική τους πρωτοβουλία, σύμφωνα με τα δεδομένα που μελετήθηκαν και την ενέργεια που απαιτείται να εκτελέσουν. Υπάρχουν διάφορα μοντέλα μηχανικής μάθησης καθώς και κατηγορίες αυτής που αναφέρονται σε βάθος παρακάτω.

4.2 Machine Learning vs Deep Learning, είναι άραγε το ίδιο?

Στην επιστήμη της πληροφορικής η **μηχανική μάθηση** (Machine Learning) όπως και η **βαθιά μάθηση**^[12] (Deep Learning) αποτελούν κομμάτια της τεχνητής νοημοσύνης. Πολλές φορές μπορεί κάποιος να αναφερθεί στην μια χρησιμοποιώντας την άλλη αλλά στην πραγματικότητα η βαθιά μάθηση αποτελεί μετεξέλιξη (μια αναβάθμιση) της μηχανικής μάθησης. Η μηχανική μάθηση αποτελεί μια τεχνολογία που έχει ως στόχο την ανάπτυξη αλγορίθμων και τεχνικών που επιτρέπουν στους υπολογιστές να «μαθαίνουν» από τα δεδομένα, από την άλλη πλευρά στην βαθιά μάθηση χρησιμοποιούνται αρκετά πιο πολύπλοκοι αλγόριθμοι, γίνεται χρήση **νευρωνικών δικτύων** και η λειτουργία τους μοιάζει πολύ με το πως θα λειτουργούσε ένας ανθρώπινος εγκέφαλος, αυτό γίνεται με τη χρήση ενός αλγορίθμου που έχει αναπτυχθεί με βάση τον ανθρώπινο εγκέφαλο και ονομάζεται **ANN** (Artificial Neural Network). Η χρήση των νευρωνικών δικτύων στην βαθιά μάθηση την καθιστά εξαιρετικά αποτελεσματική για εργασίες όπως αναγνώριση εικόνας, επεξεργασία φυσικής γλώσσας(NLP), ανάλυση φωνής κ.α.

Σύγκριση Μηχανικής Μάθησης και Βαθιάς Μάθησης:

Μηχανική Μάθηση	Βαθιά Μάθηση
Η μηχανική μάθηση χρησιμοποιεί (κατηγοριοποιημένα-δομημένα) δεδομένα.	Η βαθιά μάθηση χρησιμοποιεί το δίκτυο ANN για να εξάγει αποτελέσματα.
Στην μηχανική μάθηση από την στιγμή που χρησιμοποιείται ένας αλγόριθμος πολλές φορές υπάρχει επίβλεψη από κάποιον data analyst που παρατηρεί τις διάφορες μεταβλητές.	Οι αλγόριθμοι στο deep learning έχουν μεγαλύτερη ανεξαρτησία κατά την ανάλυση δεδομένων.

Οι αλγόριθμοι στο machine learning δημιουργούνται αρκετά γρήγορα έχοντας ως αποτέλεσμα η αποτελεσματικότητά τους να μην είναι και πολύ υψηλή.	Οι αλγόριθμοι στο deep learning χρειάζονται παραπάνω χρόνο για να δημιουργηθούν αλλά είναι αποτελεσματικότεροι σε σχέση με τους αλγορίθμους machine learning και εξελίσσονται συνεχώς με τα δεδομένα που λαμβάνουν.
Οι αλγόριθμοι εδώ χρειάζονται λιγότερο χρόνο εκτέλεσης ενός μοντέλου αλλά αρκετό χρόνο για να δοκιμαστεί.	Οι αλγόριθμοι του deep learning χρειάζονται περισσότερο χρόνο εκτέλεσης του μοντέλου αλλά η δοκιμή του μοντέλου γίνεται αρκετά σύντομα.
Στο machine learning η λύση ενός προβλήματος γίνεται ως εξής. Ο αλγόριθμος σπάει το πρόβλημα σε μικρότερα κομμάτια, λύνει το καθένα ξεχωριστά με αποτέλεσμα στο τέλος να έχει λυθεί όλο το πρόβλημα.	Στο deep learning ο αλγόριθμος ενημερώνεται σχετικά με το πρόβλημα και δίνει κατευθείαν την τελική λύση (εφόσον την διαθέτει).

4.2.1 Παραδείγματα εφαρμογής των Machine και Deep Learning

Ας δούμε κάποια παραδείγματα εφαρμογής του machine και deep learning στην καθημερινότητα μας. Ένα παράδειγμα χρήσης του machine learning αποτελούν οι ηλεκτρονικοί μας υπολογιστές που λειτουργούν κάνοντας χρήση των δεδομένων που εμείς κατηγοριοποιούμε και είναι σε θέση να τα αναγνωρίζουν. Μέσω της εξέλιξης των υπολογιστών και τη συνεχή βελτίωση του machine learning κρίνεται στο άμεσο μέλλον πως αυτή η ανάγκη για κατηγοριοποίηση των δεδομένων για την αναγνώριση τους να μην είναι απαραίτητη.

Ένα παράδειγμα εφαρμογής του deep learning στην καθημερινότητα αποτελούν τα αυτοκίνητα που χρησιμοποιούμε και διαθέτουν αισθητήρες που μας προειδοποιούν σε περίπτωση κάποιου σφάλματος ή εμποδίου. Αυτή η διαδικασία πραγματοποιείται με τη χρήση του deep learning. Η

διαδικασία αυτή δεν γίνεται βέβαια απευθείας. Ο αλγόριθμος αρχικά μέσω της αναγνώρισης αρκετών εμποδίων «εκπαιδεύεται» και μπορεί να αναγνωρίζει μέχρι ένα βαθμό κάποιο εμπόδιο. Μέσω του deep learning και τη συνεχή ανάγνωση δεδομένων σταδιακά μαθαίνει να αναγνωρίζει και άλλα εμπόδια και με την πάροδο του χρόνου βελτιώνεται ακόμα περισσότερο.

4.3 Κατηγορίες Μηχανικής Μάθησης

Στον τομέα αυτό έχουμε διάκριση των μεθόδων της μηχανικής μάθησης σύμφωνα με τον τρόπο τον οποίο δίνονται οι πληροφορίες σε ένα σύστημα. Η μηχανική μάθηση λοιπόν χωρίζεται στις εξής κατηγορίες:

1)**Επιβλεπόμενη Μάθηση^[15] (Supervised learning)** : Ο υπολογιστής εδώ δέχεται κάποια δεδομένα εισόδου (από κάποιον που επιτηρεί αυτή τη διαδικασία) μαζί με τα αντίστοιχα επιθυμητά δεδομένα εξόδου. Στόχος του υπολογιστή είναι να μάθει ένα γενικό κανόνα προκειμένου να μπορέσει να αντιστοιχήσει της εισόδους με τα αποτελέσματα. Ο στόχος είναι λοιπόν ο αλγόριθμος να μπορεί να μαθαίνει συγκρίνοντας τις πραγματικές εξόδους που έχει με αυτές που του δόθηκαν για τον εντοπισμό σφαλμάτων και την πραγματοποίηση μετατροπών που απαιτούνται σύμφωνα με το μοντέλο που έχει δημιουργηθεί.

Χρησιμοποιείται συχνά σε προβλήματα ταξινόμησης(classification), πρόγνωσης(prediction), διερμηνείας(Interpretation) κ.α.

2)**Μη Επιβλεπόμενη Μάθηση^[15](Unsupervised learning)**: Εδώ ο αλγόριθμος που χρησιμοποιείται δεν λαμβάνει κάποιες επιθυμητές εξόδους, δημιουργεί ένα μοντέλο που δέχεται ένα σύνολο δεδομένων με την μορφή παρατηρήσεων. Σκοπός του λοιπόν είναι να βρει τη δομή των δεδομένων εισόδου χωρίς να διαθέτει κάποια εμπειρία ο αλγόριθμος μάθησης. Η μη επιβλεπόμενη μάθηση πολλές φορές αποτελεί αυτοσκοπός (ανακαλύπτει κρυμμένα μοτίβα σε δεδομένα) η μέσο για ένα τέλος (γεγονός που αποτελεί χαρακτηριστικό της μάθησης).

Χρησιμοποιείται συχνά σε προβλήματα ανάλυσης συσχετισμών(Association Analysis) και ομαδοποίησης (Clustering).

3)Ενισχυτική Μάθηση^[15] (Reinforcement Learning): Εδώ ένα πρόγραμμα υπολογιστή μαθαίνει μια στρατηγική ενεργειών μέσα από την αλληλεπίδραση του με το περιβάλλον γύρω του για την επίτευξη ενός στόχου χωρίς την ενημέρωση του από κάποιον «δάσκαλο»-επιτηρητή για το πόσο κοντά είναι στην επίτευξη αυτού του στόχου. Ένα παράδειγμα χρήσης αυτής θα αποτελούσε η εκμάθηση ενός παιχνιδιού και η επίτευξη της νίκης απέναντι σε κάποιον αντίπαλο.

Χρησιμοποιείται συχνά σε προβλήματα σχεδιασμού (Planning) όπως είναι για παράδειγμα ο έλεγχος κίνησης ενός ρομπότ.

4.4 Εξόρυξη Δεδομένων και Μηχανική Μάθηση

Πολύ συχνά όταν αναφερόμαστε στην μηχανική μάθηση ακούμε την έννοια της **εξόρυξης δεδομένων^[16]** και αυτό δεν συμβαίνει τυχαία. Στην πραγματικότητα η μηχανική μάθηση κατέχει μια σημαντική σχέση με την εξόρυξη δεδομένων. Όπως είδαμε και παραπάνω στην μηχανική μάθηση χρησιμοποιούμε ή δημιουργούμε αλγορίθμους που διαβάζουν δεδομένα για την πρόβλεψη αποτελεσμάτων από τα μοντέλα μηχανικής μάθησης που έχουν εκπαιδευτεί και κάνουν χρήση των αλγορίθμων αυτών. Για να μπορούν να υπάρξουν τα αποτελέσματα αυτά θα πρέπει να υπάρχουν τα ανάλογα δεδομένα που εξάγονται από τις πληροφορίες που χρειαζόμαστε. Εδώ λοιπόν έρχεται η εξόρυξη δεδομένων που αναλαμβάνει αυτή τη διαδικασία και αποτελεί με λίγα λόγια ένα σύνολο χρήσιμων τεχνικών (αλγορίθμων) για την ανάλυση μεγάλων συλλογών δεδομένων και την εξόρυξη των πληροφοριών που χρειαζόμαστε από αυτές.

Κάποια βασικά παραδείγματα αλγορίθμων εξόρυξης δεδομένων αποτελούν οι εξής:

C4.5, k-means ,EM(expectation maximization), PageRank, AdaBoost ,KNN, Naïve Bayes, και **Cart** που καλύπτουν τα σημαντικότερα πεδία της εξόρυξης δεδομένων, δηλαδή τα πεδία της ταξινόμησης, ομαδοποίησης, παλινδρόμησης, στατικής μάθησης, ανάλυσης συνδέσεων και εξόρυξης δεδομένων, (καθένα από αυτά τα πεδία έχει τη δική του χρησιμότητα και εφαρμογή στην διαδικασία της εξόρυξης δεδομένων και βοηθά στην κατανόηση και αξιοποίηση των δεδομένων με διάφορους τρόπους. Αναφορικά μπορούμε να πούμε πως η ταξινόμηση κάνει πρόβλεψη

κατηγοριών, η ομαδοποίηση δημιουργεί ομάδες δεδομένων με βάση τα κοινά τους στοιχεία, η παλινδρόμηση χρησιμοποιείται για την πρόβλεψη αριθμητικών τιμών, η στατική μάθηση κάνει ανάλυση και πρόβλεψη με χρήση στατικών μοντέλων μαζί με μηχανική μάθηση, η ανάλυση συνδέσεων ανακαλύπτει σχέσεις μεταξύ συνόλων δεδομένων, και η εξόρυξη δεδομένων είναι μια γενική διαδικασία που μπορούμε να πούμε πως συνδυάζει όλες τις παραπάνω τεχνικές για εξαγωγή χρήσιμης γνώσης από το σύνολο των δεδομένων που μελετιούνται.

4.5 Μοντέλα Μηχανικής Μάθησης

Η μηχανική μάθηση χρησιμοποιεί διαφόρων ειδών μοντέλα και τεχνικές που κάθε μια από αυτές κατέχει διαφορετική προσέγγιση των δεδομένων που μελετιούνται και διαφορετικό τρόπο χρήσης αυτών. Τα μοντέλα και οι τεχνικές αυτές χρησιμοποιούνται και ποικίλουν με βάση το είδος της μηχανικής μάθησης που χρησιμοποιείται σε ένα μοντέλο (επιβλεπόμενη μάθηση, μη επιβλεπόμενη μάθηση, ενισχυτική μάθηση). Ας δούμε μερικά παραδείγματα μοντέλων μηχανικής μάθησης, πως λειτουργούν καθώς και πιο είδος εκμάθησης μοντέλου υποστηρίζουν από τα παραπάνω:

Εκμάθηση με δέντρο απόφασης (Decision Tree Learning): Το μοντέλο αυτό χρησιμοποιεί ένα δέντρο απόφασης ως προγνωστικό μοντέλο που αντιστοιχίζει παρατηρήσεις σχετικά με ένα στοιχείο σε συμπεράσματα σχετικά με την τιμή στόχο ενός αντικειμένου. Το μοντέλο αυτό χρησιμοποιεί την επιβλεπόμενη μάθηση και χρησιμοποιείται συχνά σε προβλήματα ταξινόμησης και παλινδρόμησης.

Τεχνητά Νευρωνικά Δίκτυα (Artificial Neural Networks): Το μοντέλο αυτό είναι εμπνευσμένο από τη δομή και τη λειτουργία των βιολογικών νευρώνων. Αποτελούνται από διασυνδεδεμένους «νευρώνες» που επεξεργάζονται τις εισόδους και παράγουν εξόδους. Εδώ η δομή των υπολογισμών που γίνονται βασίζεται σε μια ομάδα εσωτερικά διασυνδεδεμένων τεχνητών νευρώνων οι οποίοι επεξεργάζονται μια πληροφορία και εκτελούν υπολογισμούς επικοινωνώντας μεταξύ τους. Αυτό το μοντέλο υποστηρίζει την επιβλεπόμενη μάθηση και την ενισχυτική μάθηση. Χρησιμοποιείται συχνά για την μοντελοποίηση σύνθετων σχέσεων μεταξύ των δεδομένων εισόδου-εξόδου καθώς και για την ανακάλυψη προτύπων στα δεδομένα.

Μηχανές Διανυσμάτων Υποστήριξης(Support Vector Machine-SVM): Το μοντέλο αυτό περιλαμβάνει ένα σύνολο μεθόδων επιτηρούμενης μάθησης που χρησιμοποιούνται για τα στάδια της ταξινόμησης και της παλινδρόμησης. Εδώ δίνεται ένα σύνολο παραδειγμάτων εκπαίδευσης που κάθε φορά δηλώνεται σε ποια από τις δύο κατηγορίες ανήκει το παράδειγμα. Μια μηχανή διανυσμάτων υποστήριξης λοιπόν κατασκευάζει ένα μοντέλο που προβλέπει αν το νέο-επόμενο παράδειγμα ανήκει στην μια κατηγορία ή στην άλλη. Υποστηρίζει την επιβλεπόμενη μάθηση και γίνεται χρήση του σε προβλήματα ταξινόμησης και παλινδρόμησης.

Δίκτυα Bayes (Bayesian Networks): Το μοντέλο αυτό διαθέτει πιθανοθεωρητικά γραφικά μοντέλα ή δίκτυα εμπιστοσύνης όπως αλλιώς λέγονται που απεικονίζουν ένα σύνολο τυχαίων μεταβλητών και τις μεταξύ τους υποθετικές ανεξαρτησίες μέσω ενός **κατευθυνόμενου άκακου γράφου**. Υποστηρίζει την επιβλεπόμενη και μη επιβλεπόμενη μάθηση και ένα παράδειγμα εφαρμογής του θα μπορούσε να αποτελέσει η μοντελοποίηση πιθανολογικών σχέσεων και ο υπολογισμός πιθανοτήτων όπως γίνεται για παράδειγμα σε ιατρικές διαγνώσεις και σε μία ανάλυση κινδύνου.

*Στην θεωρία των γράφων ένας κατευθυνόμενος άκακος γράφος είναι ένας κατευθυνόμενος γράφος που δεν περιέχει κύκλους, δηλαδή απλά μονοπάτια που ξεκινούν και τελειώνουν στον ίδιο κόμβο.

Γενετικοί Αλγόριθμοι (Genetic Algorithms): Στο μοντέλο αυτό πραγματοποιείται μια **ευρετική αναζήτηση** που μιμείται την διαδικασία της φυσικής επιλογής χρησιμοποιώντας τεχνικές όπως είναι η μετάλλαξη, διασταύρωση και επιλογή για την δημιουργία νέων γονοτύπων έχοντας την ελπίδα εύρεσης αποτελεσματικών λύσεων. Υποστηρίζει την επιβλεπόμενη και μη επιβλεπόμενη μάθηση ανάλογα με τον τρόπο εφαρμογής του. Χρησιμοποιούνται κυρίως για την βελτιστοποίηση και την αναζήτηση λύσεων σε διάφορα προβλήματα όπως στον προγραμματισμό και στην σχεδίαση μηχανών.

4.6 Εφαρμογές της Μηχανικής Μάθησης στην εποχή μας

Η μηχανική μάθηση τα τελευταία χρόνια έχει αποκτήσει σημαντική θέση και εφαρμόζεται σε διάφορους τομείς της καθημερινότητας μας λόγω της ικανότητας της να αναλύει ένα τεράστιο όγκο δεδομένων και να λαμβάνει αποφάσεις ή να πραγματοποιεί προβλέψεις χωρίς να έχει προγραμματιστεί ρητά με κάποιο τρόπο. Η τεχνολογία αυτή έχει βρει εφαρμογή σε διάφορους τομείς και βιομηχανίες φέρνοντας επανάσταση στον τρόπο εκτέλεσης των εργασιών και ανοίγοντας νέες δυνατότητες. Ας δούμε λοιπόν κάποια παραδείγματα εφαρμογής της μηχανικής μάθησης:

Υγειονομική Περίθαλψη: Με την χρήση της μηχανικής μάθησης στον τομέα της υγείας αναπτύσσονται προγνωστικά μοντέλα που εκτελούν της λειτουργίες της διάγνωσης, πρόγνωσης και σχεδιασμού θεραπείας διαφόρων ασθενειών. Οι αλγόριθμοι της μηχανικής μάθησης διαβάζοντας διάφορα ιατρικά δεδομένα για την σωστή εκπαίδευση και λειτουργία τους έχουν την δυνατότητα να αναλύσουν ιατρικές εικόνες (ακτινολογικές-μαγνητικές εικόνες) για να ανιχνεύσουν τυχόν ανωμαλίες με στόχο να βοηθήσουν τους ακτινολόγους να κάνουν πιο ακριβείς διαγνώσεις. Επίσης εδώ η μηχανική μάθηση μπορεί να χρησιμοποιηθεί και για την πρόβλεψη των αποτελεσμάτων ενός ασθενή (βάση των εξετάσεων και ιστορικού ενός ασθενή) καθώς και για να εντοπίσει άτομα που έχουν αυξημένο κίνδυνο εμφάνισης μιας ασθένειας, επιτρέποντας έτσι την έγκυρο εντοπισμό αυτής και την δημιουργία εξατομικευμένων θεραπειών για τον εκάστοτε ασθενή.

Λιανικό Εμπόριο: Η μηχανική μάθηση έχει προσφέρει καινούργιες δυνατότητες στον τομέα του εμπορίου λιανικής κάνοντας δυνατή την ύπαρξη του εξατομικευμένου μάρκετινγκ καθώς και παρέχει της δυνατότητες της πρόβλεψη ζήτησης της αγοράς καθώς και της διαχείρισης αποθεμάτων των εταιριών-βιομηχανιών. Οι αλγόριθμοι μηχανικής μάθησης αναλύοντας δεδομένα πελάτων μπορούν να κατανοήσουν τις προτιμήσεις και την συμπεριφορά των πελατών, δίνοντας έτσι τη δυνατότητα στους λιανοπωλητές να δημιουργήσουν εξατομικευμένες διαφημίσεις και προτάσεις για να προσελκύσουν περισσότερο και πιο στοχευμένα το αγοραστικό κοινό τους. Κάποιες ακόμα λειτουργίες που παρέχει η χρήση αυτής της τεχνολογίας στις επιχειρήσεις είναι

πρόβλεψη ζήτησης προϊόντων, βελτίωση των επιπέδων αποθέματος καθώς και μείωση του κόστους που σχετίζεται με την υπερτροφοδότηση ή τις εξαντλήσεις.

Χρηματοοικονομικά: Ο χρηματοοικονομικός κλάδος έχει αφομοιώσει και χρησιμοποιεί την τεχνολογία της μηχανικής μάθησης για διάφορες λειτουργίες όπως αυτή του εντοπισμού απάτης, βαθμολόγηση πιστώσεων, αλγοριθμική διαπραγμάτευση κ.α. Στον τομέα αυτό οι αλγόριθμοι μηχανικής μάθησης μελετούν ένα μεγάλο μέγεθος οικονομικών δεδομένων με σκοπό να μπορούν να εντοπίζουν πρότυπα και ανωμαλίες που σχετίζονται με δόλιες δραστηριότητες βοηθώντας έτσι τα χρηματοοικονομικά ιδρύματα να ελαχιστοποιήσουν-αντιμετωπίσουν-προλαμβάνουν τέτοιους κινδύνους. Επιπλέον κάποια μοντέλα μηχανικής μάθησης χρησιμοποιούνται εδώ και για να αξιοποιήσουν την πιστοληπτική ικανότητα ενός ατόμου αναλύοντας παράγοντες όπως είναι το ιστορικό, εισόδημα, δημογραφικές πληροφορίες οδηγώντας έτσι τα ιδρύματα αυτά να λαμβάνουν πιο ακριβείς πιστωτικές αποφάσεις. Περαιτέρω η μηχανική μάθηση δίνει τη δυνατότητα στα κέντρα αυτά για σχεδιασμό κερδοφόρων στρατηγικών μέσω μελέτης ιστορικών δεδομένων καθώς και για μελλοντικές προβλέψεις σε πραγματικό χρόνο.

Κατασκευή: Η μηχανική μάθηση κατέχει πληθώρα εφαρμογών στον κατασκευαστικό τομέα προσφέροντας βελτίωση της παραγωγής και του ποιοτικού ελέγχου. Εδώ οι αλγόριθμοι μηχανικής μάθησης αναλύοντας δεδομένα αισθητήρων γραμμών παραγωγής μπορούν να εντοπίσουν ανωμαλίες καθώς και να προβλέψουν μελλοντικές βλάβες που πρόκειται να εμφανιστούν στον εξοπλισμό παραγωγής, επιτρέποντας έτσι την προληπτική συντήρηση του καθώς και ελαχιστοποίηση του χρόνου διακοπής λειτουργίας. Επιπλέον παρέχει σημαντικές βελτιώσεις όσον αφορά τα χρονοδιαγράμματα παραγωγής και την κατανομή πόρων, οδηγώντας έτσι στη βελτιστοποίηση της παραγωγικότητας και την μείωση του κόστους παραγωγής.

Μεταφορά: Στον τομέα αυτό θα μπορούσε να πει κάποιος πως η μηχανική μάθηση έχει φέρει την επανάσταση μέσω της δυνατότητας δημιουργίας αυτόνομων οχημάτων (όπως tesla) καθώς και εφαρμογών και συσκευών για την πρόβλεψη της κυκλοφορίας και βελτιστοποίηση διαδρομής. Τα αυτόνομα οχήματα χρησιμοποιούν τους αλγόριθμους μηχανικής μάθησης για να κατανοήσουν και να ερμηνεύσουν σωστά το περιβάλλον γύρω τους κάνοντας έτσι δυνατή τη λήψη αποφάσεων σε

πραγματικό χρόνο για μια ασφαλή πλοήγηση. Επιπλέον, μέσω της μηχανικής εκμάθησης μπορεί να αναλύσει ιστορικά δεδομένα κυκλοφορίας για τον εντοπισμό μοτίβων κυκλοφοριακής συμφοράς βοηθώντας έτσι τους οδηγούς και τις αρχές μεταφορών στην λήψη σωστών αποφάσεων. Στην διαδικασία αυτή της πρόβλεψης της βέλτιστης διαδρομής οι αλγόριθμοι μηχανικής μάθησης μελετούν και παράγοντες όπως είναι καιρικές συνθήκες, προτιμήσεις πελατών και κυκλοφοριακές συνθήκες οδηγώντας έτσι τους ανθρώπους σε πιο αποτελεσματικά δίκτυα μεταφορών.

5) Προσέγγιση ABSA (Aspect-Based Sentiment Analysis)

5.1 Ορισμός ABSA

Πέρα από τις δύο βασικές προσεγγίσεις (και τον συνδυασμό αυτών) που παρουσιάζονται στην παρούσα πτυχιακή εργασία σωστό αποτελεί το να γίνει αναφορά και στο **Aspect Based Sentiment Analysis**^[36] το οποίο αποτελεί μια λειτουργία που ενδιαφέρει σε μεγάλο βαθμό το περιβάλλον και την κοινότητα της τεχνολογίας και διαθέτει αρκετά μεγάλο ερευνητικό ενδιαφέρον. Το ABSA είναι μια εξειδικευμένη μορφή ανάλυσης συναισθήματος η οποία επικεντρώνεται στον εντοπισμό και την κατανόηση συναισθημάτων με βάση συγκεκριμένων **πτυχών (aspects)** σε ένα κείμενο που μπορεί να αφορά ένα προϊόν μια υπηρεσία ή γενικά κάποιο θέμα.

5.2 Πως λειτουργεί?

Η λειτουργία του ABSA περιλαμβάνει τα στάδια που παρατηρούνται και στις άλλες προσεγγίσεις δηλαδή αυτά της συλλογής δεδομένων, επεξεργασίας αυτών πριν την ανάλυση και προσθήκη ετικετών συναισθήματος η συναισθηματικού score, ανάλυση συναισθήματος και αξιολόγηση. Παρόλα αυτά η βασική διαφορά του με αυτές είναι το γεγονός του ότι το ABSA επικεντρώνεται στις πτυχές που βρίσκονται σε ένα κείμενο και όχι στο γενικότερο συναίσθημα αυτού. Συγκεκριμένα περιλαμβάνει δύο βασικά στάδια:

1) Τον εντοπισμό πτυχών (Aspect Extraction):

Μέσω αυτού εντοπίζονται οι πτυχές ή χαρακτηριστικά που αναφέρονται σε ένα κείμενο. Για παράδειγμα αν έχουμε μία κριτική εστιατορίου οι πτυχές μπορεί να είναι η ποιότητα του φαγητού η εξυπηρέτηση και η ατμόσφαιρα.

2)Ανάλυση Συναισθήματος (Sentiment Analysis):

Εδώ προσδιορίζετε το συναίσθημα (θετικό, αρνητικό, ουδέτερο) που εκφράζεται στην κάθε πτυχή που αναλύεται. Για παράδειγμα αν πρέπει να αναλυθεί η πρόταση **«Το φαγητό ήταν εξαιρετικό , αλλά η εξυπηρέτηση αργή»**, το συναίσθημα για το φαγητό είναι θετικό ενώ για την εξυπηρέτηση είναι αρνητικό. Με αυτό τον τρόπο το ABSA δεν μας δίνει απλά μια πιο λεπτομερή ανάλυση συναισθήματος πάνω στο κείμενο που αναλύεται αλλά και πιο στοχευμένες και χρήσιμες πληροφορίες και αποτελέσματα.

Πιο συγκεκριμένα το ABSA λειτουργεί μέσω **συνδυασμού τεχνικών επεξεργασίας φυσικής γλώσσας και αλγορίθμων μηχανικής μάθησης**. Μερικά από τα κύρια βήματα περιλαμβάνουν:

1. **Προεπεξεργασία Δεδομένων:** εδώ γίνεται ο καθαρισμός των δεδομένων που θα μελετηθούν. Περιλαμβάνει αφαίρεσή stop words, τυποποίηση λέξεων και άλλα). Επιπλέον γίνεται και εντοπισμός keyword(λέξεων κλειδιών) και συντακτικών σχέσεων.
2. **Aspect Detection:** γίνεται χρήση μοντέλων βασισμένων σε κανόνες (όπως για παράδειγμα εξαγωγή ουσιαστικών) ή άλλων αλγορίθμων με σκοπό τον εντοπισμό απλών και περίπλοκων πτυχών.
3. **Sentiment Classification:** με αυτή τη λειτουργία καθορίζεται το συναίσθημα για κάθε πτυχή μέσω μεθόδων μηχανικής μάθησης όπως (SVMs, Random Forests etc..) ή ακόμα και με τη χρήση νευρωνικών δικτύων όπως SNN, RNN ή και πιο πρόσφατα όπως είναι τα BERT η GPT για **ταξινόμηση συναισθήματος με βάση το συμφραζόμενο**. Η ταξινόμηση συναισθήματος με βάση το συμφραζόμενο είναι μια μέθοδος ανάλυσης συναισθήματος που λαμβάνει υπόψη τα συμφραζόμενα (**context**) για να κατανοήσει το ακριβές συναίσθημα που εκφράζεται σε ένα κείμενο και αποτελεί χρήσιμο εργαλείο γιατί πολλές λέξεις ή φράσεις αποκτούν **διαφορετική συναισθηματική σημασία** ανάλογα με το περιβάλλον στο οποίο βρίσκονται.
4. **Εξαγωγή Αποτελεσμάτων:** Αυτό το βήμα παρουσιάζει το τελικό αποτέλεσμα που περιλαμβάνει των συνδυασμό πτυχών του κειμένου και των αντίστοιχων συναισθημάτων αυτών.

5.3 Πλεονεκτήματα και Μειονεκτήματα

Ας δούμε μερικά από τα πλεονεκτήματα και μειονεκτήματα του ABSA

Πλεονεκτήματα:

- Λεπτομερής εστίαση σε πτυχές του κειμένου: επιτρέπει την κατανόηση συγκεκριμένων θεμάτων η χαρακτηριστικών του κειμένου αντί μιας γενικής ανάλυσης συναισθήματος του κειμένου.
- Αυτοματοποιημένη επεξεργασία μεγάλου όγκου δεδομένων: το ABSA μπορεί να επεξεργαστεί μεγάλους όγκους δεδομένων με γρήγορο ρυθμό προσφέροντας έτσι στο χρήστη μια πιο αποδοτική λύση σε σχέση με την χειροκίνητη ανάλυση.
- Διαχείριση πολυπλοκότητας κειμένου: μπορεί να διαχειριστεί περιπτώσεις όπου διαφορετικές πτυχές ενός κειμένου διαθέτουν διαφορετικό συναισθηματικό φορτίο (για παράδειγμα η πρόταση καλό φαγητό, αργή εξυπηρέτηση δίνοντας στην πτυχή φαγητό θετικό συναισθηματικό φορτίο και στην εξυπηρέτηση αρνητικό συναισθηματικό φορτίο).
- Ευελιξία σε εφαρμογές: το ABSA μπορεί να εφαρμοστεί σε διάφορους τομείς όπως αξιολογήσεις, κοινωνικά δίκτυα, έρευνες πελατών και επιχειρηματική στρατηγική.

Μειονεκτήματα:

- Απαίτηση μεγάλου όγκου δεδομένων: η εκπαίδευση ενός μοντέλου ABSA συχνά απαιτεί σχολιασμένα δεδομένα με πτυχές και τα αντίστοιχα συναισθήματα που είναι κάτι ακριβό και χρονοβόρο.
- Εξάρτηση από την ποιότητα των δεδομένων: Οι λανθασμένες ετικέτες ή έλλειψη ποικιλίας δεδομένων μπορεί να επηρεάσει αρνητικά την απόδοση του μοντέλου.
- Περιορισμοί σε contextual understanding: παρά τις βελτιώσεις στα σύγχρονα μοντέλα ABSA μπορεί να υπάρξει δυσκολία όσον αφορά περίπλοκα συμπεραζόμενα όπως ειρωνεία, σαρκασμός ή πολιτιστικές ιδιαιτερότητες.
- Προσαρμογή σε νέα θέματα ή γλώσσες: Η χρήση του ABSA σε διαφορετικά domain ή γλώσσες απαιτεί επιπλέον εκπαίδευση και βελτιστοποίηση, κάτι που συχνά αυξάνει σημαντικά το κόστος και την πολυπλοκότητα στην εφαρμογή και λειτουργία του.

5.4 Χρήσεις τους στην καθημερινότητα

Κάποιες από τις εφαρμογές του ABSA στην καθημερινότητα και σε διάφορους τομείς είναι οι εξής:

1. **Επιχειρηματική ανάλυση:** χρησιμοποιείται για την καλύτερη κατανόηση της αντίληψης των πελατών για συγκεκριμένα προϊόντα-υπηρεσίες που οδηγεί στη βελτίωση αυτών.
2. **Εξυπηρέτηση Πελατών:** το ABSA εδώ χρησιμεύει στην αυτόματη κατηγοριοποίηση παραπόνων σε κριτικές και την δυνατότητα παραχώρησης προτεραιότητας σε πτυχές που χρειάζονται βελτίωση.
3. **Εμπόριο και Μάρκετινγκ:** δίνει την δυνατότητα της προσαρμογής των στρατηγικών μάρκετινγκ με βάση τις πτυχές που επαινούνται ή επικρίνονται περισσότερο.
4. **Διαχείριση φήμης:** παρέχει σε έναν χρήστη την δυνατότητα παρακολούθησης της δημόσιας εικόνας μέσω ανάλυσης των σχολίων που δέχεται στα μέσα κοινωνικής δικτύωσης.

6) Python και ανάλυση συναισθήματος

6.1 Ορισμός-περιγραφή του τι είναι η python, για ποιο λόγο χρησιμοποιείται και τι περιλαμβάνει ?

Η python γενικά αποτελεί μια διερμηνευόμενη γενικού σκοπού και υψηλού επιπέδου γλώσσα προγραμματισμού. Υποστηρίζει **δομημένο προγραμματισμό** (procedural programming) αλλά και **αντικειμενοστραφή προγραμματισμό** (Object Oriented) και αποτελεί μια δυναμική γλώσσα προγραμματισμού. Η python δημιουργήθηκε από τον Ολλανδό **Guido Van Rosum** και κυκλοφόρησε για το κοινό για πρώτη φορά το 1991. Κύριος στόχος της python αποτελεί η ευκολία στην χρήση της και η αναγνωσιμότητα που έχει ο κώδικας της. Το συντακτικό της γλώσσας αυτής δίνει τη δυνατότητα στους προγραμματιστές να μπορούν να εκφράσουν έννοιες σε λιγότερες γραμμές κώδικα σε σχέση με άλλες γλώσσες προγραμματισμού όπως (c,c++,java). Επιπλέον ξεχωρίζει ως γλώσσα προγραμματισμού καθώς διαθέτει πληθώρα από βιβλιοθήκες που διευκολύνουν σε μεγάλο βαθμό συγκεκριμένες εργασίες για τους προγραμματιστές καθώς και για την ευκολία με την οποία μπορεί να γίνει η εκμάθηση αυτής. Παρόλαυτα ένα μειονέκτημα αυτής αποτελεί το ότι είναι πιο αργή από άλλες μεταγλωττιζόμενες γλώσσες όπως (c,c++) λόγω του ότι είναι διερμηνευόμενη και για το λόγο αυτό δεν θεωρείται κατάλληλη για γραφή-δημιουργία λειτουργικών συστημάτων. Ο διερμηνευτής της python είναι διαθέσιμος για χρήση σε διάφορα λειτουργικά συστήματα επιτρέποντας έτσι την εκτέλεση κώδικα python σε πληθώρα διαφορετικών συστημάτων. Η python έχει αναπτυχθεί ως λογισμικό **ανοιχτού κώδικα** (open source) και η διαχείριση αυτής γίνεται από ένα μη κερδοσκοπικό οργανισμό, το Python Software Foundation. Κατά την δημιουργία της η python αποτελούσε μια γλώσσα σεναρίων που χρησιμοποιήθηκε στο κατανεμημένο λειτουργικό σύστημα **Amoeba**, ικανή και για κλήσεις συστήματος. Θεωρήθηκε ως ο διάδοχος της γλώσσας προγραμματισμού **ABC** η οποία υπήρξε βασική πηγή έμπνευσης όπως αναφέρθηκε από τον δημιουργό της. Στο πέραςμα του χρόνου και από τη δημιουργία αυτής το 1989 στο ερευνητικό κέντρο Centrum Wiskunde & Informatica(CWI) υπήρξαν διάφορες εκδόσεις αυτής που έγιναν διαθέσιμες στο ευρύ κοινό, η κάθε μια παρέχοντας αλλαγές, μετατροπές και διορθώσεις σε λάθη που έχουν βρεθεί με την τελευταία έκδοση python να είναι η 3.13 που δημοσιεύτηκε στις 8 Μάιου 2024.

6.2 Python και Sentiment Analysis, με ποιο τρόπο λειτουργούν μαζί?

Αρχικά πρέπει να αναφερθούμε στο ότι η ανάλυση συναισθήματος με απλά λόγια αποτελεί μια τεχνική που στην ουσία εντοπίζει το συναίσθημα πίσω από μια πρόταση ή ένα κείμενο και μπορούμε να πούμε πως χρησιμοποιείται στα πλαίσια της γλώσσας python με τη χρήση διάφορων αλγορίθμων ή βιβλιοθηκών που υπάρχουν ή μπορεί να εντάξει ο χρήσης στο περιβάλλον αυτής ανάλογα με το σκοπό που θέλει να πετύχει. Για να γίνει λοιπόν η χρήση του sentiment analysis εύστοχα πάνω σε κάποια δεδομένα που έχουμε συλλέξει (από διάφορες πηγές που μπορεί να είναι κριτικές προϊόντων, σχόλια στα social media ή γενικά text από κάποιο αρχείο ή άρθρο) απαιτούνται κάποια συγκεκριμένα βήματα όπως αυτό της **προεπεξεργασίας των δεδομένων**, **εξαγωγής χαρακτηριστικών** κ.α. ασχέτως της προσέγγισης που ακολουθείται (λεξιλογική ή μηχανικής μάθησης) και των μοντέλων αυτών. Ας δούμε αναλυτικότερα για την python:

Κάποια από τα βασικά βήματα κατά τη χρήση του Sentiment Analysis στην Python είναι τα παρακάτω:

1ο) Συλλογή δεδομένων: Αρχικά γίνεται η συλλογή των δεδομένων που θα χρησιμοποιηθούν για να εκπαιδεύσουν κάποιο μοντέλο ή θα μελετηθούν με τη χρήση κάποιων τεχνικών ή αλγορίθμων NLP(Natural Language Processing).

2ο) Προεπεξεργασία Κειμένου: Εδώ εκτελείται μια διαδικασία καθαρισμού του κειμένου που περιλαμβάνει την αφαίρεση μη σχετικών πληροφοριών όπως ειδικοί χαρακτήρες, διάφορα σημεία στίξης και τα λεγόμενα stop words.

3ο)Tokenization(κατακερματισμός): Το κείμενο που μελετιέται εδώ διαιρείται σε λέξεις ή μονάδες(tokens) για να διευκολυνθεί περαιτέρω η διαδικασία της ανάλυσης.

4ο)Εξαγωγή χαρακτηριστικών: Η διαδικασία της εξαγωγής περιλαμβάνει την ανάκτηση σημαντικών χαρακτηριστικών από το κείμενο όπως λέξεις n-grams ή ακόμα και μέρη του λόγου.

5ο)Ταξινόμηση συναισθήματος: Στο σημείο αυτό χρησιμοποιούνται αλγόριθμοι μηχανικής μάθησης ή προκαθορισμένα μοντέλα που κατηγοριοποιούν το συναίσθημα κάθε δείγματος του κειμένου. Αυτό επιτυγχάνεται μέσω της επιβλεπόμενης μάθησης (όπου τα μοντέλα εκπαιδεύονται με δεδομένα που διαθέτουν ετικέτες) ή με προεκπαιδευμένα μοντέλα που κατανοούν μοτίβα συναισθημάτων μέσω της μελέτης μεγάλων συνόλων δεδομένων που τους παρέχονται προς μελέτη.

6ο)Μετα-Επεξεργασία: Εδώ τα αποτελέσματα που λαμβάνονται από την εκτέλεση της ανάλυσης συναισθήματος ενδέχεται να υποβληθούν σε περαιτέρω επεξεργασία, όπως είναι η συγκέντρωση των βαθμολογιών συναισθήματος ή η εφαρμογή κανόνων για την κατάταξη των συναισθημάτων ως θετικά, αρνητικά ή ουδέτερα.

7ο)Αξιολόγηση: Ως τελευταίο βήμα οι ερευνητές αξιολογούν την απόδοση του μοντέλου ανάλυσης συναισθήματος που χρησιμοποιήθηκε χρησιμοποιώντας μετρικές αξιολόγησης όπως είναι η ακρίβεια, η ακρίβεια προβλέψεων (precision), η ανάκληση (recall) και το **F1-score**.

6.3 Αλγόριθμοι της Python που χρησιμοποιούνται για Sentiment Analysis

Η python αποτελεί ένα σημαντικό εργαλείο στην ανάλυση συναισθήματος και μια γλώσσα προγραμματισμού η οποία διαθέτει πληθώρα αλγορίθμων και βιβλιοθηκών που κάνουν δυνατή την ανάλυση συναισθήματος. Παρακάτω δίνονται μερικές από τις σημαντικότερες βιβλιοθήκες που διαθέτει η python για την ανάλυση συναισθήματος:

- **Textblob**

Το textblob αποτελεί μια εξαιρετική επιλογή για την ανάλυση συναισθήματος. Είναι μια απλή βιβλιοθήκη της γλώσσας python που υποστηρίζει σύνθετες αναλύσεις και λειτουργίες σε δεδομένα κειμένου. Ακολουθώντας τη λεξιλογική προσέγγιση η βιβλιοθήκη αυτή ορίζει ένα συναίσθημα με βάση το σημασιολογικό του προσανατολισμό

και την ένταση κάθε λέξης που βρίσκεται σε μια πρόταση, η οποία απαιτεί ένα προκαθορισμένο λεξικό που ταξινομεί αρνητικά και θετικά τις λέξεις. Το `textblob` χρησιμοποιεί μια προσεγγιστική μέθοδο για την ανάλυση συναισθήματος η οποία βασίζεται στην πολικότητα και την υποκειμενικότητα του κειμένου. Η βιβλιοθήκη αυτή προσδίδει μεμονωμένες βαθμολογίες σε όλες τις λέξεις μελετώντας λεξικά συναισθημάτων τα οποία διαθέτουν λέξεις με συναισθηματικό σκορ και αφού γίνει μια σύγκριση με αυτά υπολογίζεται ένα τελικό συναισθηματικό σκορ. Στη συνέχεια επιστρέφει την πολικότητα και την υποκειμενικότητα μιας πρότασης με ένα εύρος πολικότητας που ορίζεται από αρνητικό έως θετικό. Η βιβλιοθήκη αυτή διαθέτει σημασιολογικές ετικέτες που βοηθούν στην ανάλυση συμπεριλαμβανομένων των emotions, των θαυμαστικών των emojis και άλλων σημείο στίξης.

Μερικά από τα κύρια χαρακτηριστικά του `TextBlob` είναι τα εξής:

1. Αποτελεί μια απλή βιβλιοθήκη της Python.
2. Υποστηρίζει διάφορες σύνθετες αναλύσεις και λειτουργίες σε δεδομένα κειμένου.
3. Εκχωρεί ατομικές βαθμολογίες συναισθήματος.
4. Μπορεί και επιστρέφει την πολικότητα και την υποκειμενικότητα μιας πρότασης που αναλύθηκε.

- **Vader**

Μια ακόμα κορυφαία επιλογή για ανάλυση συναισθήματος αποτελεί το `Vader` το οποίο είναι μια προκατασκευασμένη βιβλιοθήκη ανάλυσης συναισθήματος ανοιχτού κώδικα βασισμένη σε κανόνες και βρίσκεται μέσα στο `NLTK` (Natural Language Toolkit). Το εργαλείο αυτό έχει σχεδιαστεί ειδικά για την ανάλυση συναισθημάτων που εκφράζονται στα μέσα κοινωνικής δικτύωσης και χρησιμοποιεί ένα συνδυασμό του λεξικού και μιας λίστας λεξιλογικών χαρακτηριστικών που γενικά επισημαίνονται σύμφωνα με το σημασιολογικό τους προσανατολισμό είτε ως θετικά είτε αρνητικά. Το `Vader` αρχικά υπολογίζει το συναίσθημα του κειμένου και επιστρέφει την πιθανότητα μια δεδομένη πρόταση εισαγωγής να είναι θετική αρνητική ή ουδέτερη. Βασίζεται σε ένα προκαθορισμένο λεξικό που περιέχει λέξεις και φράσεις με συσχετισμένες βαθμολογίες συναισθήματος. Κάθε λέξη ή φράση έχει μια βαθμολογία που εκφράζει ένα συναίσθημα

θετικό αρνητικό ή ουδέτερο. Παρ' όλα αυτά η διαφορά του με το προηγούμενο μοντέλο(TextBlob) είναι ότι αυτό δεν περιορίζεται μόνο στην απλή αναγνώριση λέξεων με θετικό ή αρνητικό συναίσθημα, χρησιμοποιεί επιπλέον κανόνες ενίσχυσης ή μείωσης της συναισθηματικής έντασης με βάση το πλαίσιο που αναλύεται. Κάποιοι από αυτούς τους κανόνες μπορεί να είναι οι εξής(αρνητικές λέξεις, η παρουσία σχετικών λέξεων όπως είναι για παράδειγμα πολύ-πάρα πολύ, κεφάλαια γράμματα, σημεία στίξης και σύνδεση εννοιών δηλαδή κατανόηση λέξεων που εμφανίζονται συχνά για τον υπολογισμό μιας συναισθηματικής τιμής για αυτές). Το τελικό σκορ υπολογίζεται στο μοντέλο αυτό ως εξής, έχουμε το ποσοστό θετικού συναισθήματος (pos) το ποσοστό ουδέτερο (neu) συναισθήματος και το ποσοστό αρνητικού συναισθήματος (neg). Πέρα από αυτά τα 3 στο τέλος έχουμε και μια συνολική βαθμολογία (compound) που κυμαίνεται -1 (έντονα αρνητικό) έως 1 (έντονα θετικό) και αποτελεί την κύρια μετρική που χρησιμοποιείται για να καθορίσει το γενικό συναίσθημα του κειμένου. Το εργαλείο μπορεί να αναλύσει δεδομένα από κάθε είδους πλατφόρμας κοινωνικών μέσων όπως Twitter και Facebook και αποτελεί μια εξαιρετική επιλογή για γρήγορη και απλή ανάλυση συναισθήματος για ανάλυση κοινωνικών δεδομένων ή μικρών κειμένων.

Μερικά από τα κύρια χαρακτηριστικά του Vader είναι τα εξής:

1. Δεν απαιτεί δεδομένα εκπαίδευσης καθώς λειτουργεί με κάποιο προκαθορισμένο λεξικό συναισθήματος.
2. Κατανοεί το συναίσθημα του κειμένου που περιέχει emoticon,αργκό, συνδέσμους, σημεία στίξης.
3. Αποτελεί εξαιρετική επιλογή για κείμενο στα μέσα κοινωνικής δικτύωσης.
4. Αποτελεί μια βιβλιοθήκη ανοιχτού κώδικα.

- **Flair**

Το Flair αποτελεί μια βιβλιοθήκη της Python που χρησιμοποιείται για επεξεργασία φυσικής γλώσσας(NLP) και είναι ιδιαίτερα γνωστή για την ικανότητά της στην ανάλυση συναισθήματος, αναγνώριση οντοτήτων και άλλες εργασίες με χρήση τεχνητής νοημοσύνης. Το μοντέλο αυτό προσφέρει μια απλή διεπαφή για την αξιοποίηση ισχυρών προ εκπαιδευμένων μοντέλων γλώσσας που βασίζονται σε βαθιές νευρωνικές δικτύωσης.

Η βιβλιοθήκη Flair βασίζεται σε προ εκπαιδευμένα νευρωνικά δίκτυα και ενσωμάτωση λέξεων τα οποία βοηθούν στην κατανόηση του περιεχομένου του κειμένου σε μεγαλύτερο βάθος σε σχέση με άλλα μοντέλα. Οι λέξεις κωδικοποιούνται σε διανύσματα που αντιπροσωπεύουν τη σημασία τους. Η βιβλιοθήκη αυτή χρησιμοποιεί κυρίως contextual embeddings (ενσωματώσεις περιεχομένου) γεγονός που σημαίνει ότι η σημασία μιας λέξης μπορεί να αλλάξει ανάλογα με το περιβάλλον στο οποίο βρίσκεται. Για παράδειγμα η αγγλική λέξη “bank” μπορεί να αναφέρεται είτε σε τράπεζα είτε σε όχθη ποταμού και το Flair έχει τη δυνατότητα να διακρίνει τη σωστή σημασία από το context του κειμένου που αναλύεται. Επιπλέον το μοντέλο αυτό προσφέρει έτοιμα μοντέλα που έχουν ήδη εκπαιδευτεί σε μεγάλα σύνολα δεδομένων γεγονός που σημαίνει ότι μπορεί να τα χρησιμοποιήσει κανείς άμεσα χωρίς να χρειάζονται περαιτέρω εκπαίδευση με το επιπλέον πλεονέκτημα ότι κανείς μπορεί να κάνει προσαρμογή των μοντέλων που παρέχει αυτή η βιβλιοθήκη στα δικά του δεδομένα, αν έχει συγκεκριμένες απαιτήσεις.

Μερικά από τα κύρια χαρακτηριστικά του Flair είναι τα εξής:

1. Το Flair είναι ανοιχτού κώδικα και παρέχει εργαλεία που επιτρέπουν σε ένα χρήστη να προσαρμόζει ή να δημιουργεί τα δικά του μοντέλα με βάση τα υπάρχοντα.
2. Υποστηρίζει πληθώρα γλωσσών.
3. Είναι απλό στη χρήση.
4. Διαθέτει μια μεγάλη σειρά προεκπαιδευμένων μοντέλων πάνω στην ανάλυση συναισθήματος.

• Pattern

Μια από τις καλύτερες βιβλιοθήκες Python για ανάλυση συναισθήματος αποτελεί το Pattern το οποίο είναι μια βιβλιοθήκη πολλαπλών χρήσεων που μπορεί να χειριστεί και NLP. Η βιβλιοθήκη αυτή διαθέτει ένα ευρύ φάσμα χαρακτηριστικών που περιλαμβάνει την εύρεση υπερθετικών και συγκριτικών. Επιπλέον μπορεί να πραγματοποιήσει ανίχνευση γεγονότων και απόψεων, γεγονός που την κάνει κορυφαία επιλογή ανάμεσα στις βιβλιοθήκες για ανάλυση συναισθήματος. Το Pattern διαθέτει έναν αρκετά εύκολο και πολύ αποτελεσματικό τρόπο για την ανάλυση συναισθήματος κειμένου επιτρέποντας την κατηγοριοποίηση της συναισθηματικής πολικότητας ενός κειμένου ως θετικό ή αρνητικό.

Η βιβλιοθήκη αυτή χρησιμοποιεί ένα **λεξικό συναισθημάτων** και **μεθόδους κανόνα** για την εκτίμηση της πολικότητας χωρίς να βασίζεται σε εξελιγμένα νευρωνικά δίκτυα ή προ εκπαιδευμένα μοντέλα που χρησιμοποιούν άλλες βιβλιοθήκες. Ο τρόπος με τον οποίο λειτουργεί εδώ η ανάλυση συναισθήματος είναι εξής. Η βιβλιοθήκη αυτή αξιολογεί την πολικότητα η οποία θα είναι ένας αριθμός που κυμαίνεται από το -1 έως 1 (-1 αρνητικό συναίσθημα, 0 ουδέτερο συναίσθημα, 1 θετικό συναίσθημα) και την υποκειμενικότητα από 0 έως 1 (0 υποδηλώνει ότι το κείμενο είναι αντικειμενικό και βασίζεται σε γεγονότα, 1 υποδηλώνει ότι το κείμενο είναι πιο υποκειμενικό και βασίζεται σε προσωπικές απόψεις και συναισθήματα). Η πολικότητα υπολογίζεται με βάση μια λίστα θετικών και αρνητικών λέξεων στο κείμενο (με χρήση λεξικού λέξεων που διαθέτουν συναισθηματικά σκορ) και την βαρύτητα αυτών ενώ η υποκειμενικότητα με βάση αν το κείμενο περιέχει όρους που εκφράζουν γνώμη η συναίσθημα.

Μερικά από τα κύρια χαρακτηριστικά του Pattern είναι τα εξής:

1. Αποτελεί μια βιβλιοθήκη πολλαπλών χρήσεων.
2. Μέσω αυτής γίνεται δυνατή η εύρεση υπερθετικών και συγκριτικών.
3. Διαθέτει τη δυνατότητα επιστροφής της πολικότητας και της υποκειμενικότητας ενός κειμένου.
4. Η πολικότητα κυμαίνεται από πολύ θετική έως εξαιρετικά αρνητική.

- **spaCy**

Το spaCy αποτελεί μια από τις πιο δημοφιλείς βιβλιοθήκες για επεξεργασία φυσικής γλώσσας (NLP) ανοιχτού κώδικα στην Python. Η βιβλιοθήκη αυτή επιτρέπει στους προγραμματιστές να αναλύουν και να κατανοούν μεγάλους όγκους κειμένου με διάφορους τρόπους και χρησιμεύει στην κατασκευή συστημάτων κατανόησης φυσικής γλώσσας και συστημάτων εξαγωγής πληροφοριών. Είναι κατασκευασμένο για ταχύτητα και προσφέρει έτοιμα εκπαιδευμένα μοντέλα για πολλαπλές γλώσσες όπως αγγλικά, γαλλικά, ισπανικά, ελληνικά και πολλές ακόμα. Η συγκεκριμένη βιβλιοθήκη χρησιμοποιεί έτοιμα εκπαιδευμένα μοντέλα, ένα μοντέλο περιέχει πληροφορίες που επιτρέπει στο spaCy να αναγνωρίζει λεξιλόγιο, POS tags, οντότητες και πολλά ακόμα. Το spaCy από μόνο του δεν παρέχει έτοιμο μοντέλο για ανάλυση συναισθήματος όπως άλλες βιβλιοθήκες ωστόσο

μπορεί να χρησιμοποιηθεί για ανάλυση συναισθήματος σε συνδυασμό με άλλα εργαλεία ή εκπαιδευμένα μοντέλα καθώς και να προσαρμοστεί στο μοντέλο ενός συγκεκριμένου χρήστη για εκπαίδευση.

Μερικά από τα κύρια χαρακτηριστικά του spaCy είναι τα εξής:

1. Είναι γρήγορο και εύκολο στη χρήση.
2. Αποτελεί την ιδανική λύση για αρχάριους προγραμματιστές.
3. Έχει τη δυνατότητα να επεξεργαστεί τεράστιους όγκους δεδομένων.
4. Μπορεί να εκτελέσει τη λειτουργία της ανάλυσης συναισθήματος έχοντας ένα ευρύ φάσμα πηγών.

- **Scikit-learn**

Το Scikit-learn αποτελεί μια αυτόνομη βιβλιοθήκη Python που ήταν αρχικά μια επέκταση τρίτου μέρους στη βιβλιοθήκη SciPy. Είναι ιδιαίτερα χρήσιμη για κλασικούς αλγορίθμους μηχανικής μάθησης που χρησιμοποιούνται για την ανίχνευση ανεπιθύμητων μηνυμάτων και την αναγνώριση εικόνων αλλά μπορεί να χρησιμοποιηθεί και για εργασίες NLP συμπεριλαμβανομένης της ανάλυσης συναισθήματος. Το Scikit-learn δεν διαθέτει ειδικά προ εκπαιδευμένα μοντέλα για ανάλυση συναισθήματος αλλά διαθέτει όλα τα απαραίτητα εργαλεία για να δημιουργήσει κανείς το δικό του μοντέλο ανάλυσης συναισθήματος χρησιμοποιώντας αλγορίθμους μηχανικής μάθησης όπως είναι **Logistic Regression**, **Naïve Bayes**, **Support Vector Machines** και διάφορους άλλους για να εκπαιδεύσει ένα μοντέλο στην πρόβλεψη συναισθήματος ενός κειμένου. Μια συνοπτική επισκόπηση του πώς λειτουργεί χρησιμοποιώντας ένα μη εκπαιδευμένο μοντέλο θα ήταν η εξής: Αρχικά μετατρέπει το κείμενο σε χαρακτηριστικά που μπορούν να χρησιμοποιηθούν από αλγόριθμους μηχανικής μάθησης (TF-IDF Vectorization) στη συνέχεια διαχωρίζει τα δεδομένα σε 2 μέρη για εκπαίδευση αξιολόγηση(Training and Test Split) περαιτέρω χρησιμοποιεί αλγορίθμους όπως **Logistic Regression** ή **Support Vector Machines** για να εκπαιδευτεί το μοντέλο και στο τέλος της διαδικασίας αυτής γίνεται η αξιολόγηση ακρίβειας του μοντέλου χρησιμοποιώντας μετρικές όπως το accuracy για να αξιολογηθεί η απόδοσή του. Η βιβλιοθήκη αυτή μας βοηθάει να πραγματοποιήσουμε ανάλυση συναισθήματος για να αναλύουμε απόψεις ή συναισθήματα μέσω δεδομένων

εκπαιδεύοντας ένα μοντέλο που μπορεί να κατανοήσει αν το κείμενο είναι θετικό ή αρνητικό.

Μερικά από τα κύρια χαρακτηριστικά του Scikit-learn είναι τα εξής:

1. Είναι χτισμένο πάνω σε SciPy και NumPy.
2. Είναι αποδεδειγμένο με πρακτικές εφαρμογές.
3. Διαθέτει μια ποικιλία μοντέλων και αλγορίθμων που μπορεί να χρησιμοποιηθούν με αυτό.
4. Έχει χρησιμοποιηθεί από μεγάλες εταιρείες όπως το Spotify.

- **Polyglot**

Μια ακόμα εξαιρετική βιβλιοθήκη και ανάλυση συναισθήματος αποτελεί το Polyglot το οποίο είναι μια βιβλιοθήκη της Python ανοιχτού κώδικα, χτισμένη πάνω σε NumPy, που χρησιμοποιείται για πληθώρα λειτουργιών NLP (Natural Language Processing) όπως εντοπισμός γλώσσας (Language Detection), Τμηματοποίηση προτάσεων (Sentence Segmentation), Διάσπαση σε λέξεις (Tokenization), ανάλυση συναισθήματος (Sentiment Analysis), Αναγνώριση ονομάτων οντοτήτων (Named Entity Recognition-NER), Μετάφραση και εύρεση λέξεων κλειδιών (Translation and keyword extraction) και πολλές ακόμα. Το βασικό πλεονέκτημα αυτής της βιβλιοθήκης είναι ότι υποστηρίζει πάνω από **130 γλώσσες** καθιστώντας την ιδανική για πολυγλωσσικές εφαρμογές και επιλέγεται συχνά για έργα που περιλαμβάνουν γλώσσες που δεν υποστηρίζονται από άλλες βιβλιοθήκες όπως το spaCy. Η ανάλυση συναισθήματος εδώ βασίζεται σε λεξικά που περιέχουν λέξεις με θετική ή αρνητική πολικότητα για πληθώρα γλωσσών. Τα λεξικά αυτά χρησιμοποιούνται για να υπολογίσουν τη συνολική πολικότητα του κειμένου Μέσω ανάλυσης λέξεων (κάθε λέξη του κειμένου συγκρίνεται με το λεξικό συναισθήματος για να δούμε αν έχει θετική αρνητική ή ουδέτερη πολικότητα) και μέσω αυτού γίνεται ο υπολογισμός της συνολικής πολικότητας του κειμένου(η πολικότητα όλων των λέξεων στο κείμενο συνδυάζεται για να προκύψει ένα συνολικό σκορ που στην ουσία αντιπροσωπεύει το γενικό συναίσθημα του κειμένου).

Μερικά από τα κύρια χαρακτηριστικά του Polyglot είναι τα εξής:

1. Είναι πολύγλωσσο με 136 γλώσσες που υποστηρίζονται για ανάλυση συναισθήματος.
2. Είναι χτισμένο πάνω στο NumPy.
3. Αποτελεί βιβλιοθήκη ανοιχτού κώδικα.
4. Είναι αποτελεσματικό, γρήγορο και απλό στη χρήση.

- **PyTorch**

Το PyTorch αποτελεί μια βιβλιοθήκη της Python ανοιχτού κώδικα. Είναι μια βιβλιοθήκη που χρησιμεύει για λειτουργίες μηχανικής μάθησης και deep learning και βασίζεται στη γλώσσα προγραμματισμού Python. Δημιουργήθηκε από την ερευνητική ομάδα τεχνητής νοημοσύνης (Facebook AI Research) και δίνει τη δυνατότητα σε κάποιον να πραγματοποιήσει πολλές διαφορετικές εφαρμογές που περιλαμβάνουν την ανάλυση συναισθήματος για εντοπισμό σε ένα κείμενο ή μια πρόταση και κατηγοριοποίηση του συναισθήματος ως θετικό ή αρνητικό. Η βασική λειτουργία του PyTorch είναι η διαχείριση τενσοροειδών δεδομένων-tensors(τα οποία είναι πολυδιάστατοι πίνακες που μοιάζουν με τα arrays του NumPy αλλά είναι προσαρμοσμένα για γρήγορους υπολογισμούς σε GPU) και η εκπαίδευση μοντέλων μέσω του αυτόματου υπολογισμού παραγώγων-**autograd** (που αποτελεί η λειτουργία του αυτόματου υπολογισμού παραγώγων) και είναι κρίσιμο για την εκπαίδευση νευρωνικών δικτύων καθώς επίσης και την αυτόματη εκτέλεση της μεθόδου της οπισθόδρομης διάδοσης(back propagation). Η διαδικασία ανάλυσης συναισθήματος με χρήση αυτής της βιβλιοθήκης γίνεται αξιοποιώντας τη χρήση **νευρωνικών δικτύων** όπως (**LSTMs** ή **BERT**) για την εκπαίδευση μοντέλων ανάλυσης συναισθήματος που μπορούν να αναγνωρίσουν αν ένα κείμενο εκφράζει θετικά αρνητικά ή ουδέτερα συναισθήματα. Αρχικά έχουμε τη διαδικασία της προετοιμασίας των δεδομένων όπου και έχουμε τη συλλογή των δεδομένων τα οποία αποτελούνται από κείμενα με ετικέτες συναισθήματος και στη συνέχεια η προεπεξεργασία των δεδομένων δηλαδή μετατροπή των δεδομένων σε μια μορφή που μπορεί να επεξεργαστεί η βιβλιοθήκη μας χρησιμοποιώντας τεχνικές όπως(tokenization, μετατροπή σε ακολουθίες αριθμών,padding). Στη συνέχεια γίνεται μετατροπή των δεδομένων σε ενσωματώσεις-**Embeddings**(Τα Embeddings μετατρέπουν τις λέξεις σε διανυσματικούς χώρους όπου παρόμοιες λέξεις έχουν κοντινές τιμές). Το επόμενο βήμα αποτελεί η δημιουργία ενός

ηλεκτρονικού δικτύου ή η χρήση ενός όπως για παράδειγμα **LSTM** ή **BERT** για καλύτερη κατανόηση των αλληλουχιών μεταξύ των λέξεων. Περαιτέρω έχουμε επιλογή συνάρτησης απώλειας και βελτιστοποίησή τα οποία καθορίζουν τον τρόπο εκπαίδευσης του μοντέλου, τέλος ακολουθεί η διαδικασία εκπαίδευσης των μοντέλων και η τελική αξιολόγηση της αποδοτικότητας του.

Μερικά από τα κύρια χαρακτηριστικά του PyTorch είναι τα εξής:

1. Είναι εξαιρετικά γρήγορο.
2. Διαθέτει ένα στιβαρό πλαίσιο.
3. Μπορεί να λειτουργήσει από επιμέρους επεξεργαστές, CPU ή GPU.
4. Δημιουργεί δυναμικά γραφήματα υπολογισμών των νευρωνικών δικτύων γεγονός που το κάνει πιο ευέλικτο για πειραματισμό και debugging.

- **CoreNLP**

Το CoreNLP αποτελεί μια ακόμα βιβλιοθήκη Python ανοιχτού κώδικα που έχει αναπτυχθεί από το Stanford NLP Group. Η βιβλιοθήκη αυτή διαθέτει πληθώρα εργαλείων και αλγορίθμων για βασικές αλλά και προχωρημένες εργασίες επεξεργασίας φυσικής γλώσσας όπως αναγνώρισης συναισθήματος, ανάλυση συντακτικού-οντοτήτων και υποστηρίζει πληθώρα γλωσσών ανάμεσα στις οποίες είναι: Αγγλικά, Αραβικά, Γερμανικά, Κινέζικα, Γαλλικά και Ισπανικά. Το CoreNLP σε αντίθεση με άλλες βιβλιοθήκες ανάλυσης συναισθήματος υποστηρίζει και της δύο προσεγγίσεις (λεξιλογική-μηχανικής μάθησης) χρησιμοποιώντας είτε κάποιο προ εκπαιδευμένο μοντέλο χωρίς να παρέχεται στο χρήστη η δυνατότητα εκπαίδευσης του μοντέλου (μέσα στο περιβάλλον CoreNLP) αυτού είτε ενσωματώνοντας κάποιο εξωτερικό λεξικό συναισθημάτων για την διεργασία της ανάλυσης συναισθήματος. Η διαδικασία του sentiment analysis εδώ ακολουθεί την ανάλογη λειτουργία που ακολουθεί η κάθε προσέγγιση ανάλογα με το ποια έχει επιλέξει κάποιος χρήστης έχοντας κάποιο συγκεκριμένο στόχο.

Μερικά από τα κύρια χαρακτηριστικά του CoreNLP είναι τα εξής:

1. Ενσωματώνει εργαλεία StanfordNLP.
2. Υποστηρίζει πληθώρα γλωσσών.

3. Έχει τη δυνατότητα να αναλύει κείμενο προσθέτοντας «συναίσθημα».
4. Γραμμή εντολών υποστήριξης και μοντέλο υποστήριξης εκπαίδευσης.

- **NLTK**

Το NLTK (Natural Language Toolkit) είναι μια βιβλιοθήκη της γλώσσας προγραμματισμού python για επεξεργασία φυσικής γλώσσας η οποία προσφέρει εργαλεία και αλγόριθμους για ανάλυση κειμένου. Συγκεκριμένα όσον αφορά την ανάλυση συναισθήματος το NLTK ακολουθεί κυρίως στη **λεξιλογική προσέγγιση**(δηλαδή την χρήση κάποιου λεξικού συναισθήματος για τον προσδιορισμό συναισθήματος σε κείμενο) όπως είναι το **VADER** το οποίο περιλαμβάνεται στη βιβλιοθήκη ως επέκταση αυτής αλλά μπορεί να υποστηρίξει και μοντέλα μηχανικής μάθησης για ανάλυση συναισθήματος. Παρόλα αυτά δεν περιλαμβάνει κάποια προ εκπαιδευμένα μοντέλα ή εξειδικευμένα εργαλεία για εκπαίδευση τέτοιων μοντέλων εκ των προτέρων. Χρησιμοποιώντας ενσωματωμένο εργαλεία όπως **Naïve Bayes** και **SVM Classifiers** μπορεί κάποιος χρήστης να δημιουργήσει και να εκπαιδεύσει ένα μοντέλο ανάλυσης συναισθήματος σε σύνολα δεδομένων, όπως για παράδειγμα είναι το **Movie Reviews Dataset** που περιλαμβάνεται στο NLTK.

Μερικά από τα κύρια χαρακτηριστικά του NLTK είναι τα εξής:

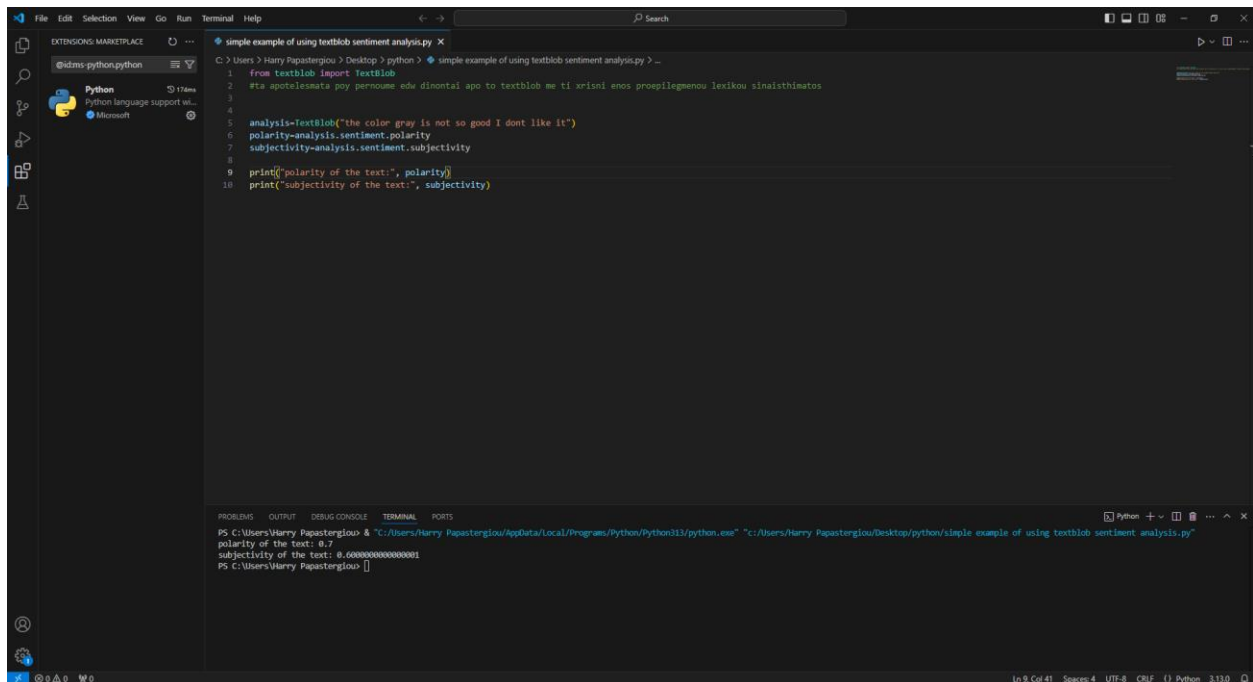
1. Χρησιμοποιεί για ανάλυση συναισθήματος σε σύντομα κείμενα και text σε κοινωνικά δίκτυα εργαλεία όπως το **VADER**.
2. Διαθέτει τη δυνατότητα της δημιουργίας μοντέλου ανάλυσης συναισθήματος μηχανικής μάθησης (**Custom Machine Learning**) χρησιμοποιώντας τους αλγορίθμους του NLTK (όπως **Naïve Bayes**).
3. Παρέχει εργαλεία για δημιουργία λεξικών από τους χρήστες και τη δυνατότητα χρήσης λεξικών συναισθήματος για την ανάλυση συναισθήματος.
4. Διαθέτει μια μεγάλη συλλογή από βιβλία οδηγούς και παραδείγματα διευκολύνοντας τους χρήστες στην χρήση του εργαλείου αυτού για επεξεργασία φυσικής γλώσσας.

7) Πρακτικό κομμάτι πτυχιακής εργασίας

7.1 Απλή χρήση των TextBlob και Vader για Sentiment Analysis μιας τυχαίας πρότασης.

Αρχικά στο πρακτικό μέρος της πτυχιακής εργασίας δίνονται παραδείγματα χρήσης κάποιων αλγορίθμων της python που χρησιμοποιούνται από τους χρήστες για την διεργασία της ανάλυσης συναισθήματος. Συγκεκριμένα παρουσιάζεται πρακτικά με χρήση κώδικα python σε ανάλογο IDE (Visual Studio Code) το **Textblob**^[49] και **VaderSentimentAnalyzer**^[48] τα οποία μπορούν να χρησιμοποιηθούν για ανάλυση συναισθήματος έχοντας το καθένα από αυτά τα δικά του πλεονεκτήματα και μειονεκτήματα σε σύγκριση μεταξύ τους. Σημαντικό είναι επίσης να αναφερθεί το γεγονός πως και τα δύο αυτά ακολουθούν **την λεξιλογική προσέγγιση ανάλυσης συναισθήματος** έχοντας την απλούστερη μορφή χρήσης τους (δηλαδή χωρίς περαιτέρω συγκεκριμένες οδηγίες από έναν χρήστη) , λειτουργώντας και παράγοντας αποτελέσματα με τη χρήση κάποιο προεπιλεγμένου **λεξικού συναισθήματος**. Προϋπόθεση χρήσης και των δυο είναι η εγκατάσταση της βιβλιοθήκης **NLTK** και περαιτέρω του **Textblob** και **Vader Sentiment** μέσω του **cmd** η ενός installer από τα ανάλογα official site.

Απλό παράδειγμα χρήσης του **TextBlob**:



Στο συγκεκριμένο παράδειγμα βλέπουμε την πιο απλή χρήση του **TextBlob**. Δημιουργήθηκε ένα αντικείμενο textblob που περιλαμβάνει την πρόταση (**The color gray is not so good I don't like it**) που αποθηκεύεται στην μεταβλητή analysis και η οποία πρόταση θα αναλυθεί με χρήση των ιδιοτήτων του textblob(**polarity: range from -1.0 to 1.0 with 0.0 neutral , subjectivity: range from 0.0 to 1.0**) . Στο τέλος του προγράμματος εμφανίζονται τα σκορ αυτά.

```
from textblob import TextBlob

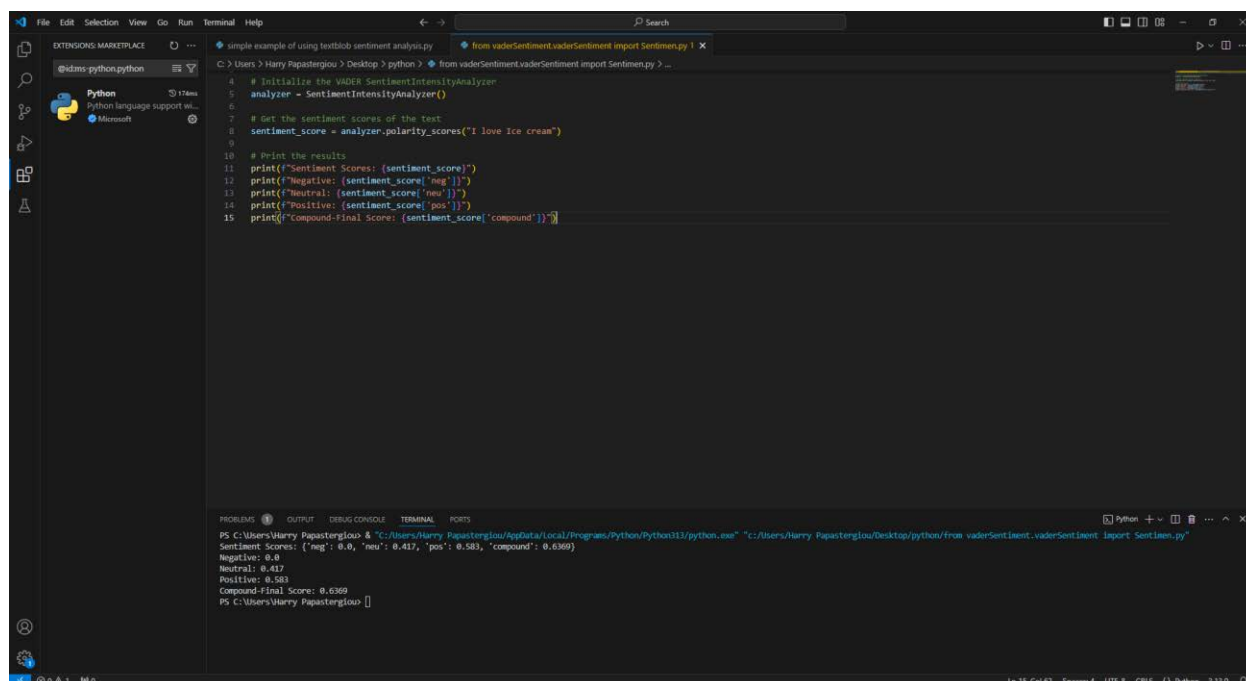
#ta apotelesmata poy pernoume edw dinontai apo to textblob me
ti xrisni enos proepilegmenou lexikou sinaisthimatos

analysis=TextBlob("the color gray is not so good I dont like it")
polarity=analysis.sentiment.polarity
subjectivity=analysis.sentiment.subjectivity

print("polarity of the text:", polarity)

print("subjectivity of the text:", subjectivity)
```

Απλό παράδειγμα χρήση του VaderSentimentAnalyzer:



```
4 # Initialize the VADER SentimentIntensityAnalyzer
5 analyzer = SentimentIntensityAnalyzer()
6
7 # Get the sentiment scores of the text
8 sentiment_score = analyzer.polarity_scores("I love Ice cream")
9
10 # Print the results
11 print("Sentiment Scores: (sentiment_score)")
12 print("Negative: (sentiment_score['neg'])")
13 print("Neutral: (sentiment_score['neu'])")
14 print("Positive: (sentiment_score['pos'])")
15 print("Compound-Final Score: (sentiment_score['compound'])")
```

```
PS C:\Users\Harry Papastergiou & "C:\Users\Harry Papastergiou\AppData\Local\Programs\Python\Python113\python.exe" "C:\Users\Harry Papastergiou\Desktop\python\from vaderSentiment-vaderSentiment import Sentimen.py"
Sentiment Scores: {'neg': 0.0, 'neu': 0.417, 'pos': 0.583, 'compound': 0.6369}
Negative: 0.0
Neutral: 0.417
Positive: 0.583
Compound-Final Score: 0.6369
PS C:\Users\Harry Papastergiou >
```

Σε αυτό το παράδειγμα βλέπουμε την πιο απλή χρήση του **VaderSentimentAnalyzer**. Αρχικά γίνεται η εκκίνηση του **VaderSentimentIntensityAnalyzer** στην μεταβλητή **Analyzer**. Στη συνέχεια δημιουργείται μια μεταβλητή με όνομα **sentiment_score** που περιλαμβάνει την εντολή **analyzer.polarity_scores()** που επιστρέφει το θετικό, αρνητικό, ουδέτερο και **compound** σκορ της πρότασης που βρίσκεται μέσα στην παρένθεση που στο συγκεκριμένο παράδειγμα είναι η **(I love Ice cream)**. Στην συνέχεια με εντολή **print** και τις ανάλογες **sentiment** εντολές εμφανίζονται τα παραπάνω σκορ. Τα όρια των **score** αυτών είναι τα εξής (**Positive**: range from 0.0 to 1.0, **Negative** range from 0.0 to 1.0, **Neutral**: range from 0.0 to 1.0, **Compound**: range ≥ 0.05 is positive, range ≤ -0.05 is negative, **Neutral** when compound score is between -0.5 and 0.5).

Σημαντικό είναι να αναφερθεί πως το **compound score** δεν είναι ένα τελικό συναισθηματικό σκορ του κειμένου που προκύπτει από τα θετικά, αρνητικά και ουδέτερα στοιχεία του. Το **compound score** λαμβάνει μια τιμή μετά από αντιστοίχιση όλων των λέξεων του κειμένου με συναισθηματικό σκορ (κλίμακας -4 έως 4) του προεπιλεγμένου λεξικού που χρησιμοποιεί το **Vader** λαμβάνοντας υπόψιν και παράγοντες όπως αρνητικές λέξεις (πχ **not**), ενισχυτικές λέξεις (πχ **very**), σημεία στίξης, θαυμαστικά και γενικά δομές που αλλάζουν την βαρύτητα μιας λέξης. Αθροίζει τις βαθμολογίες όλων των λέξεων στην συνέχεια και πριν δώσει το αποτέλεσμα εκτελεί μια

μαθηματική κανονικοποίηση για την μετατροπή του τελικού αποτελέσματος στην κλίμακα 1.0(πλήρως αρνητικό) έως 1.0(πλήρως θετικό) . **To compound score** τελικά αποτελεί μια μέτρηση του γενικού συναισθήματος του κειμένου.

```
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer

#τα αποτελεσματα που περνουμε edw dinontai apo to vader me ti xrisni enos proepilegmenou
lexikou sinaisthimatos

# Initialize the VADER SentimentIntensityAnalyzer
analyzer = SentimentIntensityAnalyzer()

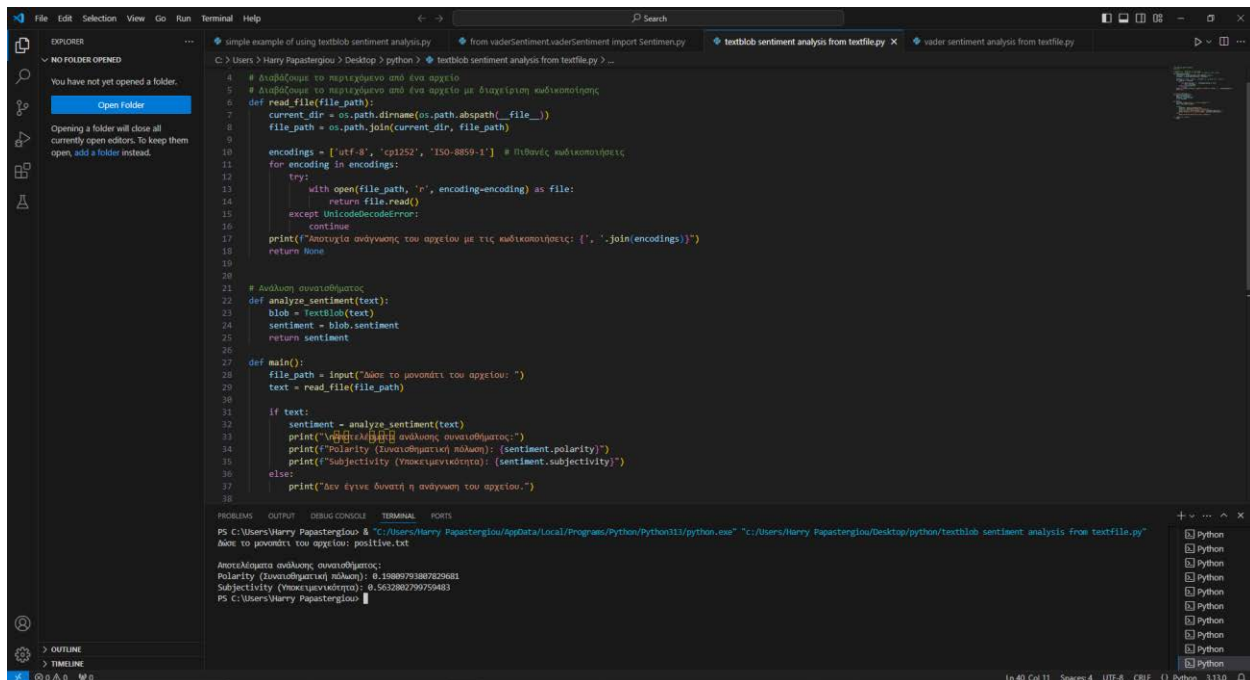
# Get the sentiment scores of the text
sentiment_score = analyzer.polarity_scores("I love Ice cream")

# Print the results
print(f'Sentiment Scores: {sentiment_score}')
print(f'Negative: {sentiment_score['neg']}')
print(f'Neutral: {sentiment_score['neu']}')
print(f'Positive: {sentiment_score['pos']}')
print(f'Compound Score: {sentiment_score['compound']}')
```

7.2 Χρήση των TextBlob και Vader για Sentiment Analysis σε έγγραφα text.

Στα παραδείγματα παρακάτω πραγματοποιείται ανάλυση συναισθήματος με τη χρήση των **TextBlob** και **VaderSentimentAnalyzer** ανιχνεύοντας το συναίσθημα σε δύο αρχεία που διαθέτουν περίπου **5300 γραμμές text** με θετικές και αρνητικές κριτικές ταινιών στα ανάλογα αρχεία (**positive.txt** και **negative.txt**).

Παράδειγμα **TextBlob** με το αρχείο **positive.txt** :



```
4 # Διαβάζουμε το περιεχόμενο από ένα αρχείο
5 # Διαβάζουμε το περιεχόμενο από ένα αρχείο με διαχείριση κωδικοποίησης
6 def read_file(file_path):
7     current_dir = os.path.dirname(os.path.abspath(__file__))
8     file_path = os.path.join(current_dir, file_path)
9
10    encodings = ['utf-8', 'cp1252', 'ISO-8859-1'] # Πιθανές κωδικοποιήσεις
11    for encoding in encodings:
12        try:
13            with open(file_path, 'r', encoding=encoding) as file:
14                return file.read()
15        except UnicodeDecodeError:
16            continue
17    print(f"Ασυνεχής ανάγνωση του αρχείου με τις κωδικοποιήσεις: {', '.join(encodings)}")
18    return None
19
20
21 # Ανάλυση συναισθήματος
22 def analyze_sentiment(text):
23     blob = TextBlob(text)
24     sentiment = blob.sentiment
25     return sentiment
26
27 def main():
28     file_path = input("Δώσε το μονοδρόμιο του αρχείου: ")
29     text = read_file(file_path)
30
31     if text:
32         sentiment = analyze_sentiment(text)
33         print("\nΑνάλυση συναισθήματος:")
34         print(f"Πολarity (Συναισθηματική πόλωση): {sentiment.polarity}")
35         print(f"Subjectivity (Υποκειμενικότητα): {sentiment.subjectivity}")
36     else:
37         print("Δεν έγινε δυνατή η ανάγνωση του αρχείου.")
38
39 if __name__ == "__main__":
40     main()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\Harry Papastergiou\AppData\Local\Programs\Python\Python311\python.exe "C:\Users\Harry Papastergiou\Desktop\python\textblob sentiment analysis from textfile.py"

Δώσε το μονοδρόμιο του αρχείου: positive.txt

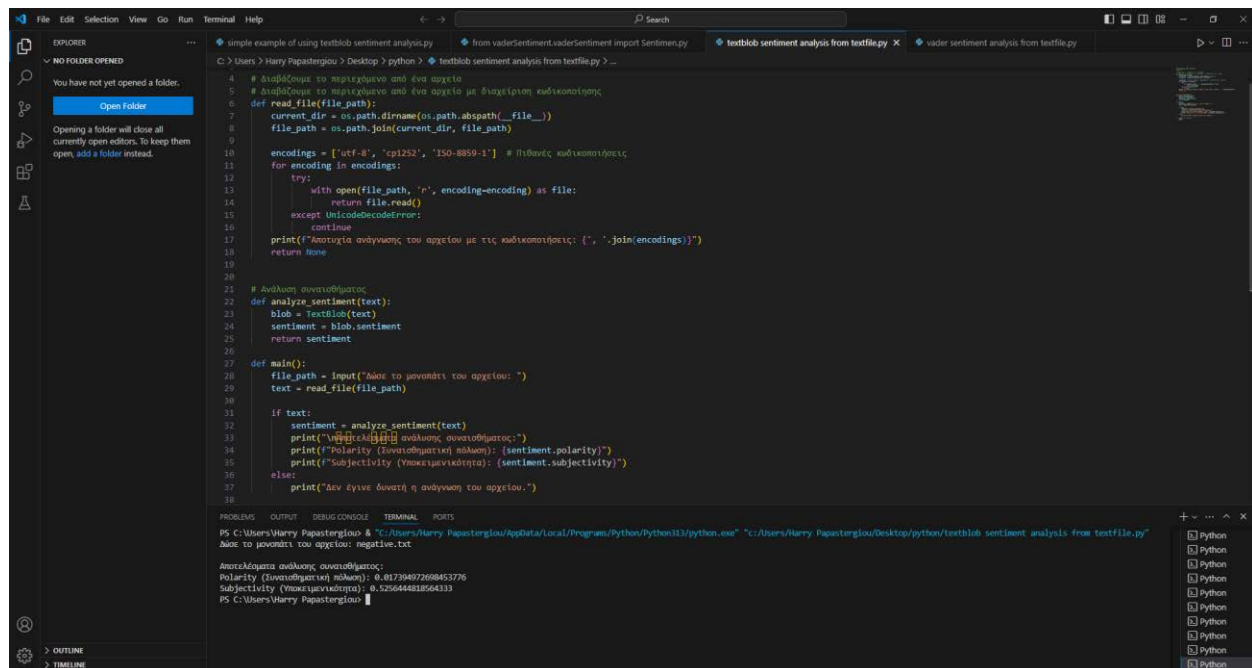
Αποτελέσματα ανάλυσης συναισθήματος:

Polarity (Συναισθηματική πόλωση): 0.19809793807629681

Subjectivity (Υποκειμενικότητα): 0.5632802799759483

PS C:\Users\Harry Papastergiou

Παράδειγμα TextBlob με το αρχείο negative.txt:



```
4 # διευκρινίζω το περιεχόμενο του αρχείου
5 # διευκρινίζω το περιεχόμενο του αρχείου με διαχωρισμό κωδικοποίησης
6 def read_file(file_path):
7     current_dir = os.path.dirname(os.path.abspath(__file__))
8     file_path = os.path.join(current_dir, file_path)
9
10    encodings = ['utf-8', 'cp1252', 'ISO-8859-1'] # πιθανές κωδικοποιήσεις
11    for encoding in encodings:
12        try:
13            with open(file_path, 'r', encoding=encoding) as file:
14                return file.read()
15        except UnicodeDecodeError:
16            continue
17    print(f"Ασυναγνώστη ανάγνωση του αρχείου με τις κωδικοποιήσεις: {', '.join(encodings)}")
18    return None
19
20
21 # ανάλυση συναισθήματος
22 def analyze_sentiment(text):
23     blob = TextBlob(text)
24     sentiment = blob.sentiment
25     return sentiment
26
27 def main():
28     file_path = input("Δώσε το μοναδικό του αρχείου: ")
29     text = read_file(file_path)
30
31     if text:
32         sentiment = analyze_sentiment(text)
33         print(f"Ανάλυση συναισθήματος:")
34         print(f"Πολικότητα (Συναίσθημα): {sentiment.polarity}")
35         print(f"Υποκειμενικότητα (Υποκειμενικότητα): {sentiment.subjectivity}")
36     else:
37         print("Δεν έγινε δυνατή η ανάγνωση του αρχείου.")
38
```

```
PS C:\Users\Harry Papastergiou & "C:\Users\Harry Papastergiou\AppData\Local\Programs\Python\Python311\python.exe" "C:\Users\Harry Papastergiou\Desktop\python\textblob sentiment analysis from textfile.py"
Δώσε το μοναδικό του αρχείου: negative.txt

Αποτελέσματα ανάλυσης συναισθήματος:
Polarity (Συναίσθημα): 0.017394072098453776
Subjectivity (Υποκειμενικότητα): 0.526444818564333
PS C:\Users\Harry Papastergiou >
```

Από τα αποτελέσματα και των δύο παραδειγμάτων παρατηρείται πως το TextBlob παράγει κάποια βάσιμα σκορ με το **positive.txt** να είναι **0.2 περίπου μονάδες** πάνω από το **negative.txt** γεγονός που δείχνει πως κατανοεί την διαφορά του συναισθήματος που υπάρχει στα δύο αρχεία και την **αντικειμενικότητα** με όριο από 0.0 έως 1.0 αν είναι κάπου στη μέση στο **0.50** και για τις δύο περιπτώσεις. Ωστόσο τα αποτελέσματα αυτά δίνονται με βάση τις προκαθορισμένες οδηγίες λειτουργίας του **TextBlob** και χωρίς κάποιο **threshold** που θα έκανε μια πιο λεπτομερή ανάλυση στο text του αρχείου.

*Οι παρακάτω δύο εικόνες αποτελούν το ίδιο κομμάτι κώδικα χωρισμένο σε (1),(2)

Παράδειγμα **VaderSentiment Analyzer** με κείμενο txt (1):

```
C:\Users\Harry Papastergiou\Desktop> python > vader sentiment analysis from textfile.py > main
1 from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
2 import os
3
4 # Συνάρτηση για να διαβάσει το περιεχόμενο του αρχείου με διαφορετικές κωδικοποιήσεις και να αγνοεί τα σφάλματα
5 def read_file(file_name):
6     current_dir = os.path.dirname(os.path.abspath(__file__))
7     file_path = os.path.join(current_dir, file_name)
8
9     # Δοκιμή με διάφορες κωδικοποιήσεις
10    encodings = ['utf-8', 'cp1252', 'ISO-8859-1', 'utf-16', 'latin1']
11
12    for encoding in encodings:
13        try:
14            # Διαβάζουμε το αρχείο αγνοώντας τα σφάλματα με την παράμετρο 'ignore'
15            with open(file_path, 'r', encoding=encoding, errors='ignore') as file:
16                print(f"Το αρχείο '{file_name}' διαβάστηκε επιτυχώς με την κωδικοποίηση: {encoding}")
17                return file.read()
18        except Exception as e:
19            print(f"Σφάλμα κατά την ανάγνωση του αρχείου με κωδικοποίηση {encoding}: {e}")
20
21    print("Αποτυχία ανάγνωσης του αρχείου με τις δοκιμασμένες κωδικοποιήσεις.")
22    return None
23
24 # Συνάρτηση για την ανάλυση συναισθήματος με VADER
25 def analyze_sentiment_vader(text):
26     analyzer = SentimentIntensityAnalyzer()
27     return analyzer.polarity_scores(text)
28
29 # Κύρια συνάρτηση για την εκτέλεση του προγράμματος
30 def main():
31     file_name = input("Δώσε το όνομα του αρχείου (π.χ. positive.txt): ")
32     text = read_file(file_name)
33
34     if text:
35         print("Ανάλυση συναισθήματος σε εξέλιξη...")
```

Παράδειγμα **VaderSentiment Analyzer** με κείμενο txt (2):

```
simple example of using textblob sentiment analysis.py  from vaderSentiment.vaderSentiment import Sentimen.py  textblob sentiment analysis from textfile.py  vader sentiment analysis from textfile.py X
C:\Users\Harry Papastergiou\Desktop> python > vader sentiment analysis from textfile.py > main
25 def analyze_sentiment_vader(text):
26     analyzer = SentimentIntensityAnalyzer()
27     return analyzer.polarity_scores(text)
28
29 # Κύρια συνάρτηση για την εκτέλεση του προγράμματος
30 def main():
31     file_name = input("Δώσε το όνομα του αρχείου (π.χ. positive.txt): ")
32     text = read_file(file_name)
33
34     if text:
35         print("Ανάλυση συναισθήματος σε εξέλιξη...")
36         sentiment_scores = analyze_sentiment_vader(text.strip()) # Ανάλυση ολόκληρου του κειμένου
37
38         # Εκτύπωση των αποτελεσμάτων
39         print(f"Τελικά αποτελέσματα συναισθήματος:")
40         print(f"Αρνητικό: {sentiment_scores['neg']}")
41         print(f"Ουδέτερο: {sentiment_scores['neu']}")
42         print(f"Θετικό: {sentiment_scores['pos']}")
43         print(f"Σκορ Συναισθήματος (Τύνθικο): {sentiment_scores['compound']}")
44     else:
45         print(f"Το πρόγραμμα τερματίζει λόγω σφάλματος.")
46
47 # Εκτέλεση του προγράμματος
48 if __name__ == "__main__":
49     main()
```

Αποτελέσματα της ανάλυσης στο αρχείο **hopecore.txt**

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\Harry Papastergiou\AppData\Local\Programs\Python\Python313\python.exe "C:\Users\Harry Papastergiou\Desktop\python\vader sentiment analysis from textfile.py"
Δώσε το όνομα του αρχείου (π.χ. positive.txt): hopecore.txt
Το αρχείο 'hopecore.txt' διαβάστηκε επιτυχώς με την κωδικοποίηση: utf-8
Ανάλυση συναισθήματος σε εξέλιξη...

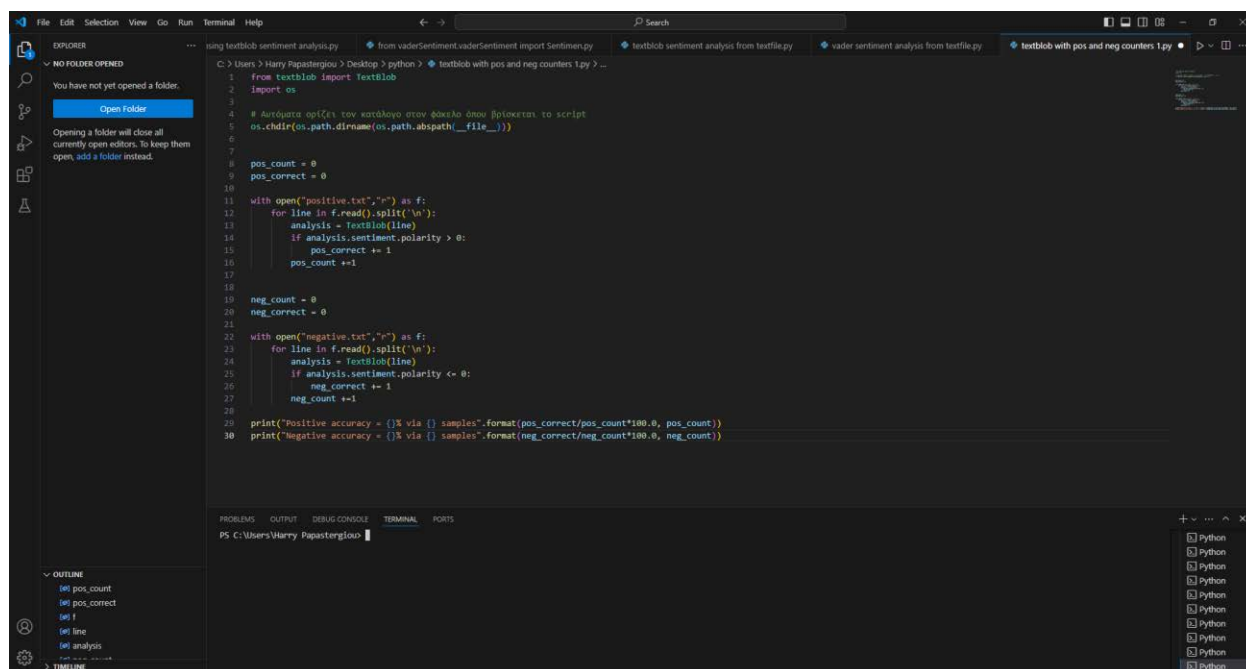
Τελικά αποτελέσματα συναισθήματος:
Αρνητικό: 0.00
Ουδέτερο: 0.686
Θετικό: 0.254
Σκορ Συναισθήματος (Τύνθικο): 0.0478
PS C:\Users\Harry Papastergiou>
```

Εδώ σε αντίθεση με το **textblob** παρατηρείται πως έχουμε σκορ που περιλαμβάνει το θετικό, αρνητικό και ουδέτερο συναίσθημα καθώς και το compound που δείχνει το **γενικότερο συναίσθημα** του κειμένου (όπου το βασικό όριο του είναι από -1 έως 1) που εδώ φαίνεται να είναι θετικό.

7.3 Χρήση των TextBlob και Vader για Sentiment Analysis στα έγγραφα positive-negative.txt με σκοπό την εμφάνιση του accuracy που έχουν και πως αυτή επηρεάζεται εφόσον αναλύουμε ανά γραμμή τα έγγραφα και διαφοροποιώντας το threshold .

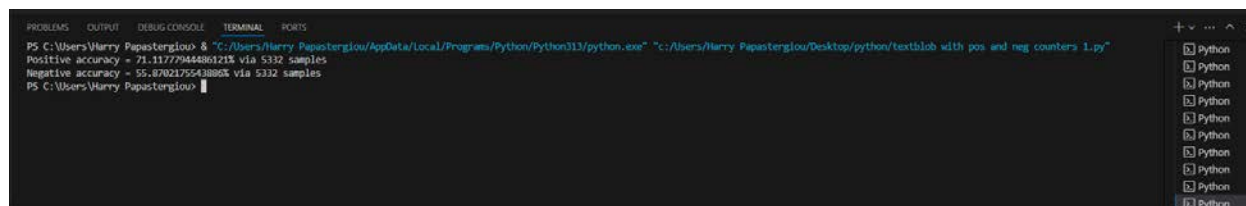
Στο συγκεκριμένο μέρος της πτυχιακής εργασίας γίνεται μια προσπάθεια υπολογισμού του **accuracy** των δύο μοντέλων στα δύο **dataset**, **negative** και **positive** (που παράγεται από την πράξη μέτρησης θετικών-αρνητικών στοιχείων/πλήθος επιβεβαιωμένων θετικών-αρνητικών στοιχείων *100) καθώς και τροποποίηση των **threshold** στο εκάστοτε **polarity-subjectivity-neg-cos** και **neu score**.

Αρχικά έχουμε τον παρακάτω κώδικα που κάνει χρήση του **Textblob**:



```
1 from textblob import TextBlob
2 import os
3
4 # Αυτόματα επιλέγει τον κατάλογο στον φάκελο όπου βρίσκεται το script
5 os.chdir(os.path.dirname(os.path.abspath(__file__)))
6
7
8 pos_count = 0
9 neg_count = 0
10
11 with open("positive.txt", "r") as f:
12     for line in f.read().split('\n'):
13         analysis = TextBlob(line)
14         if analysis.sentiment.polarity > 0:
15             pos_count += 1
16
17
18 neg_count = 0
19 neg_count = 0
20
21
22 with open("negative.txt", "r") as f:
23     for line in f.read().split('\n'):
24         analysis = TextBlob(line)
25         if analysis.sentiment.polarity <= 0:
26             neg_count += 1
27
28
29 print("Positive accuracy = {}% via {} samples".format(pos_count/pos_count*100, pos_count))
30 print("Negative accuracy = {}% via {} samples".format(neg_count/neg_count*100, neg_count))
```

Διαθέτει τα εξής αποτελέσματα:



```
PS C:\Users\Harry Papastergiou> python "c:/Users/Harry Papastergiou/Desktop/python/textblob with pos and neg counters 1.py"
Positive accuracy = 71.1177944486121% via 5332 samples
Negative accuracy = 55.8702175543886% via 5332 samples
PS C:\Users\Harry Papastergiou>
```

Παρατηρούμε πως έχουμε

Positive Accuracy=71.1177944486121% via 5332 samples

Negative accuracy = 55.8702175543886% via 5332 samples

Έχοντας ως μοναδικό μας threshold το polarity που εντοπίζεται στα δύο dataset, αν είναι μεγαλύτερο του 0 αυξάνονται οι 2 μετρητές του positive ενώ αν είναι και μικρότερο του 0 αυξάνονται οι μετρητές του negative. Τα ποσοστά του accuracy και στα δύο βέβαια δεν βρίσκονται σε αρκετά ικανοποιητικό επίπεδο ακόμα.

Τι θα συμβεί άραγε αν μεγαλώσουμε το threshold του polarity από 0 σε 0.2?

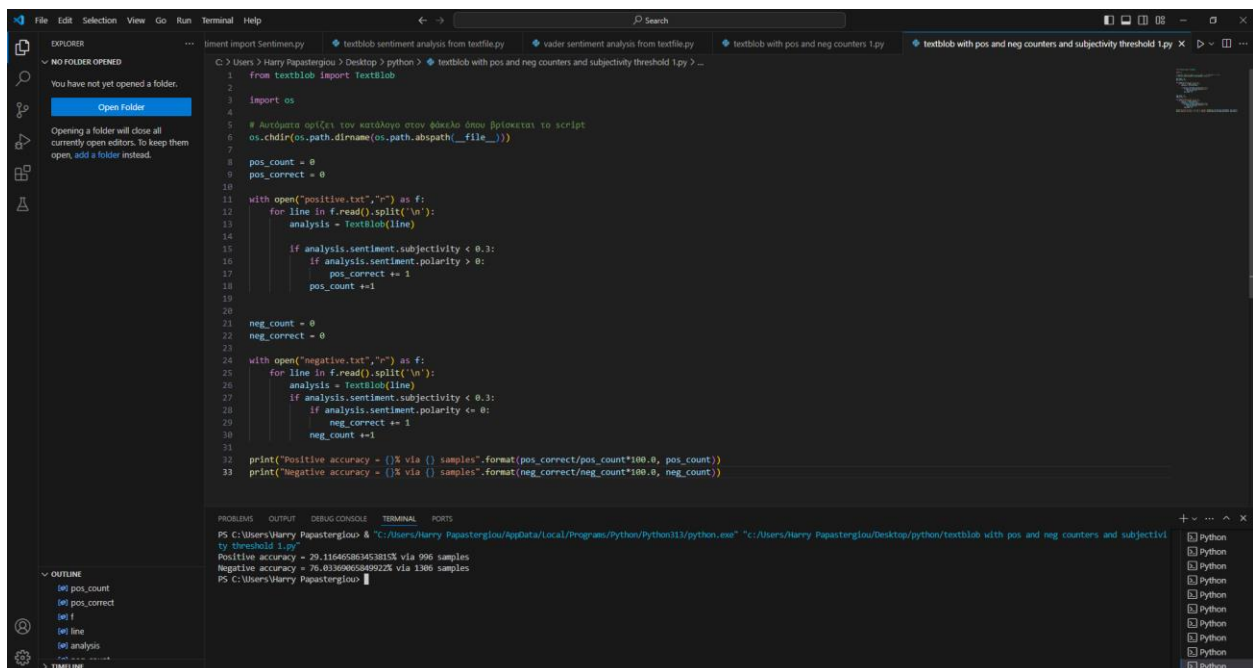
```
Positive accuracy = 46.19279819954989% via 5332 samples
Negative accuracy = 80.1012753188297% via 5332 samples
PS C:\Users\Harry Papastergiou>
```

Παρατηρούμε πως το ποσοστό του negative accuracy ανέβηκε υπερβολικά αλλά χάθηκε αρκετό από το positive accuracy οπότε και αυτή η περίπτωση δεν είναι καλή για χρήση από έναν χρήστη. Τι θα συμβεί αν μεγαλώσουμε το threshold του polarity από 0 σε 0.1?

```
Positive accuracy = 60.483870967741936% via 5332 samples
Negative accuracy = 68.39834958739685% via 5332 samples
PS C:\Users\Harry Papastergiou>
```

Παρατηρούμε ξανά πως το ποσοστό και στα δύο accuracy παραμένει χαμηλό.

Τι θα συμβεί αν βάλουμε και το **subjectivity** ως threshold στο παραπάνω κώδικα?



```
1 from textblob import TextBlob
2
3 import os
4
5 # Αρμόδια ορίζει τον κατάλογο στον οποίο είναι αποθηκευμένα τα αρχεία
6 os.chdir(os.path.dirname(os.path.abspath(__file__)))
7
8 pos_count = 0
9 pos_correct = 0
10
11 with open("positive.txt", "r") as f:
12     for line in f.read().split('\n'):
13         analysis = TextBlob(line)
14
15         if analysis.sentiment.subjectivity < 0.1:
16             if analysis.sentiment.polarity > 0:
17                 pos_correct += 1
18             pos_count += 1
19
20 neg_count = 0
21 neg_correct = 0
22
23 with open("negative.txt", "r") as f:
24     for line in f.read().split('\n'):
25         analysis = TextBlob(line)
26
27         if analysis.sentiment.subjectivity < 0.1:
28             if analysis.sentiment.polarity <= 0:
29                 neg_correct += 1
30             neg_count += 1
31
32 print("Positive accuracy = {}% via {} samples".format(pos_correct/pos_count*100.0, pos_count))
33 print("Negative accuracy = {}% via {} samples".format(neg_correct/neg_count*100.0, neg_count))
```

```
PS C:\Users\Harry Papastergiou> python .\textblob_with_pos_and_neg_counters_and_subjectivity_threshold_1.py
Positive accuracy = 29.116465863453815% via 996 samples
Negative accuracy = 76.83369065849225% via 1306 samples
PS C:\Users\Harry Papastergiou>
```

Παρατηρούμε πως έχουμε:

Positive Accuracy= 29.116465863453815% via 996 samples

Negative accuracy = 76.03369065849922% via 1306 samples

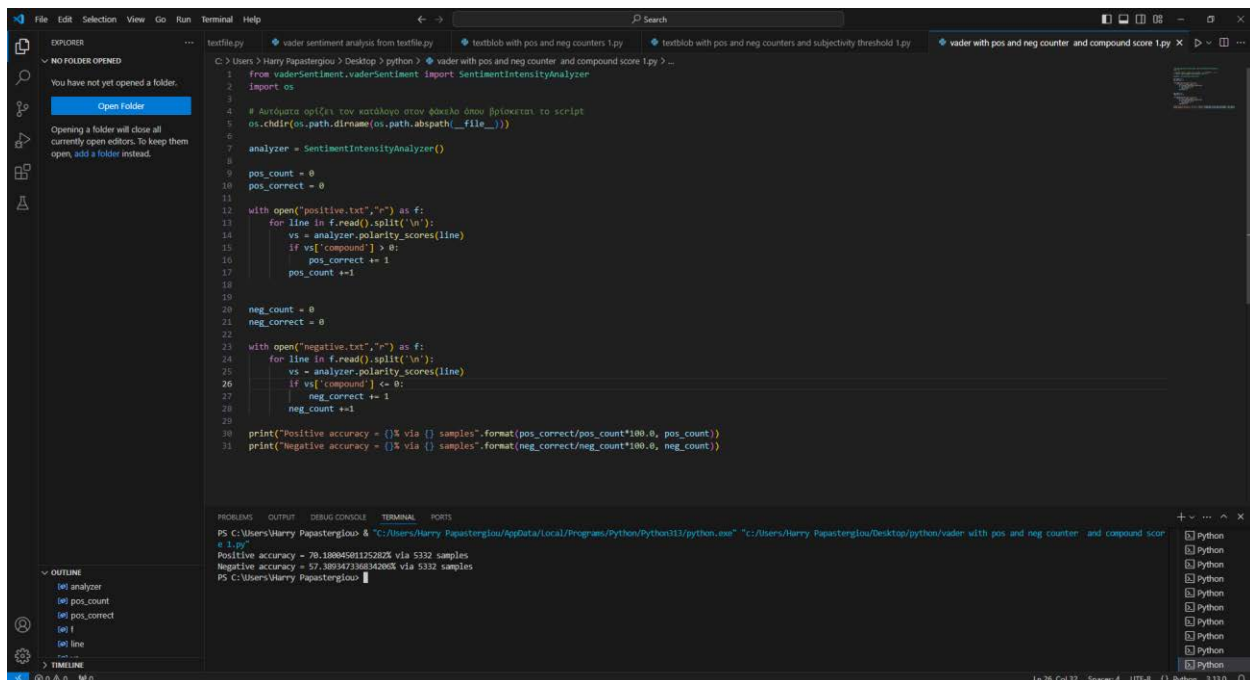
Το ποσοστό του Positive accuracy εδώ είναι πολύ χαμηλό σε σχέση με το negative και έχουμε πολύ μικρότερο δείγμα αποδεκτών sample από το αρχικό που είναι περίπου 5300 και στα δύο datasets. Τα αποτελέσματα παραμένουν μη ικανοποιητικά.

Τι συμβαίνει αν αυξήσουμε τον βαθμό αντικειμενικότητας-subjectivity από 0.3 σε 0.9?

```
Positive accuracy = 70.46771523178808% via 4832 samples
Negative accuracy = 55.10121457489878% via 4940 samples
PS C:\Users\Harry Papastergiou>
```

Εδώ παρατηρείται πάλι μια σημαντική αύξηση του θετικού accuracy αλλά και μια σημαντική μείωση του negative accuracy αλλά και τα δύο δεν πλησιάζουν σε ένα ποσοστό έστω πάνω από 80-90% που θα μπορούσε να θεωρηθεί αρκετά ικανοποιητικό.

Ίσως η ίδια διαδικασία με το VaderSentimentAnalyzer να φέρει καλύτερα αποτελέσματα.



```
1 from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
2 import os
3
4 # Αυτόματα ορίζει τον κατάλογο στον φάκελο όπου βρίσκεται το script
5 os.chdir(os.path.dirname(os.path.abspath(__file__)))
6
7 analyzer = SentimentIntensityAnalyzer()
8
9 pos_count = 0
10 pos_correct = 0
11
12 with open("positive.txt", "r") as f:
13     for line in f.read().split('\n'):
14         vs = analyzer.polarity_scores(line)
15         if vs['compound'] > 0:
16             pos_correct += 1
17             pos_count += 1
18
19 neg_count = 0
20 neg_correct = 0
21
22 with open("negative.txt", "r") as f:
23     for line in f.read().split('\n'):
24         vs = analyzer.polarity_scores(line)
25         if vs['compound'] <= 0:
26             neg_correct += 1
27             neg_count += 1
28
29 print("Positive accuracy = {}% via {} samples".format(pos_correct/pos_count*100.0, pos_count))
30 print("Negative accuracy = {}% via {} samples".format(neg_correct/neg_count*100.0, neg_count))
```

```
PS C:\Users\Harry Papastergiou\ & "C:\Users\Harry Papastergiou\AppData\Local\Programs\Python\Python311\python.exe" "C:\Users\Harry Papastergiou\Desktop\python\vader with pos and neg counter and compound scor
e 1.py"
Positive accuracy = 70.18804561125282% via 5332 samples
Negative accuracy = 57.38947733613065% via 5332 samples
PS C:\Users\Harry Papastergiou>
```

Παρατηρούμε πως εδώ έχουμε:

Positive Accuracy= 70.18004501125282% via 5332 samples

Negative accuracy = 57.389347336834206% via 5332 samples

Στο συγκεκριμένο παράδειγμα έχουμε ως threshold το compound score να είναι μεγαλύτερο ή μικρότερο του 0.

Τι γίνεται όταν ταξινομούμε το θετικά αρνητικό η ουδέτερο συναίσθημα σύμφωνα με τα όρια του compound score που δίνει το **documentation** του Vader?

Συγκεκριμένα:

- Positive sentiment: compound score ≥ 0.5
- Neutral sentiment: (compound score > -0.5) and (compound score < 0.5)
- Negative sentiment: compound score ≤ -0.5

```
and neg counters 1.py  textblob with pos and neg counters and subjectivity threshold 1.py  vader with pos and neg counter and compound score 1.py  vader with pos and neg counter and compound score 2 with a 0.5 threshold.py  ...
C:\Users\Harry Papastergiou> python 2 vader with pos and neg counter and compound score 2 with a 0.5 threshold.py ...
1
2 # Αυτόματη οπίζει τον κατάλογο στον φάκελο όπου βρίσκεται το script
3 os.chdir(os.path.dirname(os.path.abspath(__file__)))
4
5 analyzer = SentimentIntensityAnalyzer()
6
7
8 pos_count = 0
9 pos_correct = 0
10
11 threshold = 0.5
12
13
14 with open("positive.txt", "r") as f:
15     for line in f.read().split("\n"):
16         vs = analyzer.polarity_scores(line)
17
18         if vs['compound'] >= threshold or vs['compound'] <= -threshold:
19             if vs['compound'] > 0:
20                 pos_correct += 1
21             pos_count += 1
22
23
24 neg_count = 0
25 neg_correct = 0
26
27
28 with open("negative.txt", "r") as f:
29     for line in f.read().split("\n"):
30         vs = analyzer.polarity_scores(line)
31
32         if vs['compound'] >= threshold or vs['compound'] <= -threshold:
33             if vs['compound'] <= 0:
34                 neg_correct += 1
35             neg_count += 1
36
37
38 print("Positive accuracy = {}% via {} samples".format(pos_correct/pos_count*100.0, pos_count))
39 print("Negative accuracy = {}% via {} samples".format(neg_correct/neg_count*100.0, neg_count))
```

```
Positive accuracy = 87.66037735849056% via 2650 samples
Negative accuracy = 49.56140350877193% via 1824 samples
PS C:\Users\Harry Papastergiou>
```

Η διαφορά με το προηγούμενο κομμάτι κώδικα αποτελεί το γεγονός του ότι σε αυτή την περίπτωση για να θεωρηθεί ένα δείγμα ως θετικό η αρνητικό και να μετρηθεί από τους εκάστοτε counters συγκρίνεται και με την μεταβλητή threshold που έχει τεθεί και είναι **0.5** (αποκλείοντας δείγματα ως ουδέτερα ανάμεσα στις τιμές **-0.5** έως **0.5**) . Παρατηρούμε πως έχει ανέβει το θετικό

accuracy όμως το αρνητικό παραμένει χαμηλό καθώς και ότι έχουμε αποκλείσει ένα μεγάλο μέρος των δειγμάτων των dataset μας.

Τα αποτελέσματα όλων των παραπάνω διεργασιών ανάλυσης συναισθήματος με χρήση του **Vader** είναι αρκετά όμοια με εκείνα του **Textblob** χωρίς κάποια σημαντική βελτίωση, τι θα γινόταν άραγε αν η διαδικασία ορισμού ενός θετικού η αρνητικού συναισθήματος γινόταν πιο περίπλοκη-αυστηρή?

Θα υπήρχε μεγάλη βελτίωση στο accuracy του παραπάνω κώδικα αν απαιτούνταν για παράδειγμα να θεωρηθεί ένα δείγμα ως θετικό μόνο όταν το αντίθετο δηλαδή το neg να είναι μικρότερο από 0.1 ή ανύπαρκτο?

Προσαρμόζοντας των κώδικα για να εκτελεστεί ακριβώς αυτό το σενάριο έχουμε το παρακάτω τμήμα κώδικα:

```
vader with pos and neg counter and compound score 1.py vader with pos and neg counter and compound score 2 with a 0.5 threshold.py vader with pos and neg counter and compound score 3 with thresholds in neg and pos.py
C:\Users\Harry Papastergiou\Desktop>python .\vader with pos and neg counter and compound score 3 with thresholds in neg and pos.py
4
5 # Αυτόματα ορίζει τον κατάλογο στον φάκελο όπου βρίσκεται το script
6 os.chdir(os.path.dirname(os.path.abspath(__file__)))
7
8 import time
9
10 analyzer = SentimentIntensityAnalyzer()
11
12 pos_count = 0
13 pos_correct = 0
14
15 with open("positive.txt", "r") as f:
16     for line in f.read().split('\n'):
17         vs = analyzer.polarity_scores(line)
18         if not vs['neg'] > 0.1:
19             if vs['pos'] - vs['neg'] > 0:
20                 pos_correct += 1
21                 pos_count += 1
22
23
24 neg_count = 0
25 neg_correct = 0
26
27 with open("negative.txt", "r") as f:
28     for line in f.read().split('\n'):
29         vs = analyzer.polarity_scores(line)
30         if not vs['pos'] > 0.1:
31             if vs['pos'] - vs['neg'] <= 0:
32                 neg_correct += 1
33                 neg_count += 1
34
35 print("Positive accuracy = {}% via {} samples".format(pos_correct/pos_count*100.0, pos_count))
36 print("Negative accuracy = {}% via {} samples".format(neg_correct/neg_count*100.0, neg_count))
Positive accuracy = 81.0302734375% via 4096 samples
Negative accuracy = 89.26862611073138% via 2926 samples
PS C:\Users\Harry Papastergiou>
```

Από τα αποτελέσματα του κώδικα παρατηρείται μια υπερβολικά καλή βελτιώση και στα δύο accuracy χωρίς ιδιαίτερα άσχημο ποσοστό sample.

Τι γίνεται στην περίπτωση που χρησιμοποιούμε το TextBlob για ανάλυση συναισθήματος με τιμές από -0.5 έως 0.5 να είναι ουδέτερες όπως ορίζει και το documentation του Vader?

```
bound score 2 with a 0.5 threshold.py  vader with pos and neg counter and compound score 3 with thresholds in neg and pos.py  textblob with pos and neg counters final with a threshold from minus 0.5 till plus 0.5 being neutral samples.py > ...
C:\Users\Harry Papastergiou\Desktop> python > textblob with pos and neg counters final with a threshold from minus 0.5 till plus 0.5 being neutral samples.py > ...
1  from textblob import TextBlob
2
3  import os
4
5  # Αυτόματα ορίζει τον κατάλογο στον φάκελο όπου βρίσκεται το script
6  os.chdir(os.path.dirname(os.path.abspath(__file__)))
7
8  pos_count = 0
9  pos_correct = 0
10
11  with open("positive.txt", "r") as f:
12      for line in f.read().split('\n'):
13          analysis = TextBlob(line)
14
15          if analysis.sentiment.polarity >= 0.5:
16              if analysis.sentiment.polarity > 0:
17                  pos_correct += 1
18                  pos_count += 1
19
20  neg_count = 0
21  neg_correct = 0
22
23  with open("negative.txt", "r") as f:
24      for line in f.read().split('\n'):
25          analysis = TextBlob(line)
26
27          if analysis.sentiment.polarity <= -0.5:
28              if analysis.sentiment.polarity < -0.5:
29                  neg_correct += 1
30                  neg_count += 1
31
32  print("Positive accuracy = {}% via {} samples".format(pos_correct/pos_count*100.0, pos_count))
33  print("Negative accuracy = {}% via {} samples".format(neg_correct/neg_count*100.0, neg_count))
```

```
Positive accuracy = 100.0% via 766 samples
Negative accuracy = 100.0% via 282 samples
PS C:\Users\Harry Papastergiou>
```

Με αυτές τις αλλαγές λοιπόν του polarity να είναι ≥ 0.5 για αύξηση των pos μετρητών και ≤ -0.5 για αύξηση των neg μετρητών πετυχαίνουμε 100% accuracy έχοντας βέβαια το ελάτωμα του να χάνουμε αρκετά sample.

Αν μειώσουμε το polarity στις πρώτες εντολές if από **0.5** και **-0.5** σε **0.1** και **-0.1** τι αποτελέσματα θα έχουμε?

```
Positive accuracy = 100.0% via 3309 samples
Negative accuracy = 100.0% via 1498 samples
```

Αν το **0.1** και **-0.1** γίνει ακόμα μικρότερο δλδ **0.01** και **-0.01**

```
Positive accuracy = 100.0% via 3770 samples
Negative accuracy = 100.0% via 2047 samples
```

Από τις παραπάνω αλλαγές γίνεται κατανοητό πως **όσο μικραίνει το threshold μας** που είναι οι δυο αρχικές εντολές if για την κατηγοριοποίηση sample ως pos και neg **τόσο μεγαλώνει το ποσοστό των samples χωρίς να επηρεάζεται το accuracy που έχουμε.**

Τελικά αποτελέσματα.

```
C:\Users\Harry Papastergiou\Desktop> python > textblob with pos and neg counters final with a threshold from minus 0.5 till plus 0.5 being neutral samples.py > ...
1 from textblob import TextBlob
2
3 import os
4
5 # Αυτόματα ορίζει τον κατάλογο στον φάκελο όπου βρίσκεται το script
6 os.chdir(os.path.dirname(os.path.abspath(__file__)))
7
8 pos_count = 0
9 pos_correct = 0
10
11 with open("positive.txt", "r") as f:
12     for line in f.read().split('\n'):
13         analysis = TextBlob(line)
14
15         if analysis.sentiment.polarity >= 0.01:
16             if analysis.sentiment.polarity > 0:
17                 pos_correct += 1
18                 pos_count += 1
19
20
21 neg_count = 0
22 neg_correct = 0
23
24 with open("negative.txt", "r") as f:
25     for line in f.read().split('\n'):
26         analysis = TextBlob(line)
27
28         if analysis.sentiment.polarity <= -0.01:
29             if analysis.sentiment.polarity < 0:
30                 neg_correct += 1
31                 neg_count += 1
32
33 print("Positive accuracy = {}% via {} samples".format(pos_correct/pos_count*100.0, pos_count))
34 print("Negative accuracy = {}% via {} samples".format(neg_correct/neg_count*100.0, neg_count))
```

```
Positive accuracy = 100.0% via 3770 samples
Negative accuracy = 100.0% via 2047 samples
```

Ο συγκεκριμένος κώδικας textblob, με τα παραπάνω output και τις ανάλογες τιμές στις εντολές που αφορούν το polarity για κατηγοριοποίηση ως **neg** η **pos** θεωρείται αξιόπιστος και ικανοποιητικός για έναν χρήστη.

Όσον αφορά το **Vader** ο παρακάτω κώδικας είναι εκείνος που έφερε το καλύτερο accuracy χωρίς να χάσει υπερβολικά μεγάλο αριθμό sample:

```

C:\Users\Harry Papastergiou > Desktop > python > vader with pos and neg counter and compound score 1 with thresholds in neg and pos.py > ...
4
5 # Αυτήματα επιλέγει τον κατάλογο στον φάκελο όπου βρίσκεται το script
6 os.chdir(os.path.dirname(os.path.abspath(__file__)))
7
8 import time
9
10 analyzer = SentimentIntensityAnalyzer()
11
12 pos_count = 0
13 pos_correct = 0
14
15 with open("positive.txt","r") as f:
16     for line in f.read().split('\n'):
17         vs = analyzer.polarity_scores(line)
18         if not vs['neg'] > 0.05:
19             if vs['pos']-vs['neg'] > 0:
20                 pos_correct += 1
21                 pos_count += 1
22
23
24 neg_count = 0
25 neg_correct = 0
26
27 with open("negative.txt","r") as f:
28     for line in f.read().split('\n'):
29         vs = analyzer.polarity_scores(line)
30         if not vs['pos'] > 0.05:
31             if vs['pos']-vs['neg'] <= 0:
32                 neg_correct += 1
33                 neg_count += 1
34
35 print("Positive accuracy = {}% via {} samples".format(pos_correct/pos_count*100.0, pos_count))
36 print("Negative accuracy = {}% via {} samples".format(neg_correct/neg_count*100.0, neg_count))

```

```

Positive accuracy = 80.57996485061511% via 3414 samples
Negative accuracy = 98.87031179394488% via 2213 samples

```

Προκαλώντας τροποήσεις στα επιτρεπόμενα neg pos και στις ανάλογες εντολές if αλλάζει το τελικό accuracy και ο αριθμός των sample που έχουμε στο output.

Συμπέρασμα

Από όλα τα παραπάνω γίνεται αντιληπτό το γεγονός του ότι και τα δύο μοντέλα (**Vader-TextBlob**) ικανοποιούν σε καλό βαθμό την διεργασία της ανάλυσης συναισθήματος έχοντας υψηλό accuracy και ικανοποιητικό αριθμό δεικτών samples από τα δύο dataset (positive-negative.txt με 5300+ γραμμές text). Και τα δύο έχουν τα αρνητικά και θετικά τους σημεία και συγκριτικά κάποιος χρήστης οφείλει να επιλέξει τον αλγόριθμο ανάλυσης συναισθήματος που τον συμφέρει σύμφωνα με το project που θέλει να υλοποιήσει.

8) Συμπέρασμα

Ως μέρος της διατριβής μου, πραγματοποίησα ανάλυση συναισθήματος χρησιμοποιώντας τη γλώσσα προγραμματισμού Python. Αρχικά, προβλήθηκαν σε θεωρητικό επίπεδο οι έννοιες, τα χαρακτηριστικά, οι εφαρμογές της ανάλυσης συναισθημάτων και τα προβλήματα που αντιμετωπίζει ως τεχνολογία. Στη συνέχεια, μελετώνται τέσσερις βασικές προσεγγίσεις: λεξιλογική, μηχανικής μάθησης, υβριδική και ανάλυση συναισθήματος βάσει πτυχών.

Περαιτέρω ένα συγκεκριμένο μέρος αυτής της εργασίας επικεντρώνεται στη χρήση της Python ως εργαλείο για την υλοποίηση της ανάλυσης συναισθημάτων. Αναλύονται οι δυνατότητες της γλώσσας και εισάγονται δημοφιλείς βιβλιοθήκες και αλγόριθμοι που υποστηρίζουν διάφορες προσεγγίσεις, όπως NLTK, VADER και TextBlob.

Στο πρακτικό μέρος, δημιουργήθηκε και προβλήθηκε κώδικας Python για την εκτέλεση ανάλυσης συναισθημάτων. Ξεκίνησε με απλές προτάσεις για να γίνει έλεγχος της απόδοσης του εργαλείου και, στη συνέχεια, επεκτείνεται η ανάλυση σε δύο σύνολα δεδομένων κειμένου-dataset. Κατά τη διάρκεια της υλοποίησης της πρακτικής διαδικασίας πραγματοποιήθηκαν «πειράματα-μετατροπές» για τη βελτίωση της ακρίβειας του κώδικα και προσαρμόζοντας διαφορετικά σημεία του παρατηρήθηκαν-καταγράφηκαν οι αλλαγές στα αποτελέσματα.

Προκλήσεις και Διδάγματα

Κατά τη διάρκεια της εργασίας προέκυψαν αρκετές προκλήσεις. Η μεγαλύτερη δυσκολία σίγουρα αποτέλεσε ο καθορισμός των ορίων για την επίτευξη ισορροπημένων αποτελεσμάτων σε όλο το σύνολο δεδομένων, καθώς και οι διαφορετικές ακρίβειες που απαιτούν δοκιμές και ανάλυση. Επιπλέον μια ακόμη πρόκληση για εμένα αποτέλεσε και η κατανόηση και εφαρμογή των εργαλείων VADER και TextBlob, που προσφέρουν διαφορετικές μεθόδους και λειτουργίες και απαιτεί χρόνο, εξοικείωση και εξάσκηση.

Αν ήθελα να συνεχίσω ξανά αυτή την πτυχιακή εργασία , πιθανότατα θα αφιέρωνα περισσότερο χρόνο μελετώντας τη **μηχανική μάθηση** και θα χρησιμοποιούσα πιο **σύνθετους αλγόριθμους**, όπως **νευρωνικούς ταξινομητές**, που θα μπορούσαν να βελτιώσουν την ακρίβεια του κώδικα στα αποτελέσματα που φέρει. Ωστόσο, η εργασία με βοήθησε στην κατανόηση του θέματος και στο να αποκτήσω πρακτική εμπειρία, η οποία ενίσχυσε το ενδιαφέρον μου για το θέμα και την έρευνα πάνω στο συγκεκριμένο πεδίο της πληροφορικής.

Μελλοντική Εξέλιξη

Αυτή η εργασία μπορεί να χρησιμεύσει ως βάση για περαιτέρω έρευνα. Οι μελλοντικοί μαθητές ή εγώ ο ίδιος θα μπορούσαμε να επεκτείνουμε την ανάλυση συναισθήματος χρησιμοποιώντας **νευρωνικά δίκτυα** και **μοντέλα βαθιάς μάθησης** όπως το BERT ή το GPT. Επιπλέον, η **πολύγλωσση ανάλυση συναισθημάτων** μπορεί να συνδυαστεί για να ελέγξει την ακρίβεια και την αποτελεσματικότητα διαφορετικών γλωσσικών εργαλείων.

Συνολικά, αυτή η πτυχιακή εργασία αποτέλεσε ένα πολύτιμο ταξίδι γνώσης και πρακτικής εμπειρίας καθώς και μια ενδιαφέρον πρόκληση που θεωρώ πως έφερα εις πέρας και που ίσως μελλοντικά να μου επιτρέψει να συνεχίσω να εμβαθύνω στο πεδίο της ανάλυσης συναισθήματος.

Βιβλιογραφία

- [1] Bible DataAcademy, 25 Οκτωβρίου 2023. <https://bigblue.academy/gr/analusi-sunaisthimatos>
- [2] Ανάλυση των συναισθημάτων της κοινής γνώμης στα μέσα κοινωνικής δικτύωσης, Κωνσταντίνου, Μαρία, 2021-05. <https://kypseli.ouc.ac.cy/handle/11128/4992>
- [3] Sentiment Analysis-Lexicon Models vs Machine Learning, Abhinandan Roul, Nerd for Tech, published in January 4, 2024. <https://medium.com/nerd-for-tech/sentiment-analysis-lexicon-models-vs-machine-learning-b6e3af8fe746>
- [4] Text Mining with R: A Tidy Approach, written by Julia Silge and David Robinson, built on 2024-02-02. <https://www.tidytextmining.com/sentiment>
- [5] Sentiment Analysis Methods in 2024: Overview, Pros & Cons, April 4 2024, written by Cem Dilmegani. <https://research.aimultiple.com/sentiment-analysis-methods/>
- [6] Lexicon-based vs Machine Learning-based Sentiment Analysis For Contact Centers, by GennadiyBezko, 16-7-2023. <https://blog.miarec.com/lexicon-based-vs-ml-based-sentiment-analysis>
- [7] SentiWordNet Papers, by Andrea Esuli, 2019. <https://github.com/aesuli/SentiWordNet>
- [8] Sentiment Analysis with AFINN Lexicon, by Himansu Loiya, 30-6-2018. <https://himanshulohiya.medium.com/sentiment-analysis-with-afinn-lexicon-930533dfe75b>
- [9] Μηχανική Μάθηση, τί είναι και γιατί έχει σημασία?, by SAS analytics. https://www.sas.com/el_gr/insights/analytics/machine-learning.html
- [10] Τεχνητή Νοημοσύνη(TN)-Artificial Intelligence(AI), by Ελληνικό κέντρο ασφαλούς διαδικτύου ITE. https://saferinternet4kids.gr/hot-topics/ai_tn/
- [11] Τι είναι το machine learning, οδηγός για αρχάριους 2023, by Big Blue data academy, 7-5-2023. <https://bigblue.academy/gr/ti-einai-to-machine-learning>

- [12] Deep Learning vs Machine Learning: Οι Διαφορές, by Big Blue data academy, 14-8-2022.
<https://bigblue.academy/gr/deep-learning-vs-machine-learning>
- [13] Μηχανική Μάθηση, Wikipedia.
https://el.wikipedia.org/wiki/%CE%9C%CE%B7%CF%87%CE%B1%CE%BD%CE%B9%CE%BA%CE%AE_%CE%BC%CE%AC%CE%B8%CE%B7%CF%83%CE%B7
- [14] Επιβλεπόμενη Μάθηση: Ορισμός και Παραδείγματα(2023), by Big Blue data academy, 15-8-2023.
<https://bigblue.academy/gr/epivlepomeni-mathisi>
- [15] Τα είδη της Μηχανικής Μάθησης, by elementsofai.com.
<https://course.elementsofai.com/el/4/1>
- [16] Data Mining: Έννοια, Οφέλη και Τεχνικές (2023), by Big Blue data academy, 7-2-2023.
<https://bigblue.academy/gr/data-mining>
- [17] Τι είναι η μηχανική μάθηση, by sap.com.
<https://www.sap.com/greece/products/artificial-intelligence/what-is-machine-learning.html>
- [18] Ποιες είναι μερικές από τις εφαρμογές της μηχανικής μάθησης σε διαφορετικούς τομείς και κλάδους, by EITCA (Ευρωπαϊκή Ακαδημία Πιστοποίησης Τεχνολογιών Πληροφοριών), 2-9-2023.
<https://el.eitca.org/artificial-intelligence/eitc-ai-gcml-google-cloud-machine-learning/introduction/what-is-machine-learning/examination-review/what-are-some-of-the-applications-of-machine-learning-in-different-fields-and-industries/>
- [19] Sentiment Analysis with python by pythongeeksteam, by PythonGeeks.
<https://pythongeeks.org/sentiment-analysis-with-python/>
- [20] Sentiment Analysis Using Python, by Survat, 07-10-20 2024, by Analytics Vidhya.
<https://www.analyticsvidhya.com/blog/2022/07/sentiment-analysis-using-python/>
- [21] 10 καλύτερες βιβλιοθήκες Python για ανάλυση συναισθήματος, by Άλεξ ΜακΦάρλαντ, 16-01-2024, by Unite.ai.
[Link](#)
- [22] NLTK Tutorial: Σε τι βρίσκεται η NLTK Library Python?, by Έβελιν Κλαρκ, 12-6-2024.
<https://www.guru99.com/el/nltk-tutorial.html>
- [23] Vader Sentiment Analysis documentation in Github. <https://github.com/cjhutto/vaderSentiment#about-the-scoring>
- [24] Pattern GitHub documents. <https://github.com/clips/pattern>
- [25] Nltk GitHub wiki. <https://github.com/nltk/nltk/wiki>
- [26] Quick intro to Aspect-Based Sentiment Analysis, by Fabio Chiusano, 19-08-2022.
<https://medium.com/nlplanet/quick-intro-to-aspect-based-sentiment-analysis-c8888a09eda7>

- [27] A systematic review of aspect-based sentiment analysis: domains, methods, and trends, by Yan Cathy Hua Paul Denny, Jörg Wicker & Katerina Taskova, 17-9-2024. <https://link.springer.com/article/10.1007/s10462-024-10906-z>
- [28] A survey on sentiment analysis methods, applications, and challenges, Published: 07 February 2022. <https://link.springer.com/article/10.1007/S10462-022-10144-1#Sec15>
- [29] Tools and Techniques for Lexicon Driven Sentiment Analysis: A Review , January 2017, Munir Ahmad¹, Shabib Aftab², Syed Shah Muhammad³ and Usman Waheed. <http://www.ijmse.org/Volume8/Issue1/paper4.pdf>
- [30] An Overview of Lexicon-Based Approach For Sentiment Analysis, February 2018, Azeema Sadia¹, Fariha Khan² and Fatima Bashir³. http://iecc.neduet.edu.pk/2018/Papers_2018/15.pdf
- [31] Twitter Sentiment Analysis: Lexicon Method, Machine Learning Method and Their Combination, September 18 2015, Olga Kolchyna¹, Tharsis T. P. Souza¹, Philip C. Treleaven. <https://arxiv.org/abs/1507.00955>
- [32] An Approach To Sentiment Analysis Using Lexicons With Comparative Analysis of Different Techniques, May-June-2016, Tanvi Hardeniya¹, D. A. Borikar². <https://www.iosrjournals.org/iosr-jce/papers/Vol18-issue3/Version-1/J1803015357.pdf>
- [33] Climate Change Sentiment Analysis Using Lexicon, Machine Learning and Hybrid Approaches, by Nabila Mohamad Sham, Azlinah Mohamed, 14 April 2022. <https://www.mdpi.com/2071-1050/14/8/4723>
- [34] Lexicon and Deep Learning-Based Approaches in Sentiment Analysis on Short Texts, January 2024, by Taminul Islam, Md. Alif Sheakh, Md. Rezwane Sadik, Mst. Sazia Tahosin, Md. Musfiqur Rahman Foysal, Jannatul Ferdush, Mahbuba Begum. <https://www.scirp.org/journal/paperinformation?paperid=130385>
- [35] A Comparative study of Lexicon based and Machine learning based classifications in Sentiment Analysis, by T. Nikil Prakash, A. Aloysius, 01-june-2020. https://www.researchgate.net/publication/337830303_A_Comparative_study_of_Lexicon_based_and_Machine_learning_based_classifications_in_Sentiment_analysis
- [36] ASPECT-BASED SENTIMENT ANALYSIS METHODS IN RECENT YEARS, by ZOHREH MADHOUSHI, ABDUL RAZAK HAMDAN, SUHAILA ZAINUDIN, published June 2019. <http://journalarticle.ukm.my/14153/1/28204-107928-1-PB.pdf>
- [37] Bigdata & Sentiment Analysis Using Python, by Akshansh Malik, Aman Mahal, Aman Kamboj, Abhishek Sharma, published 26 June 2020.

https://www.academia.edu/43451699/Bigdata_and_Sentiment_Analysis_Using_Python

[38] Study of Twitter Sentiment Analysis using Machine Learning Algorithms on Python, by Bhumika Gupta, PhD Assistant Professor, C.S.E.DG.B.P.E.C, Pauri, Uttarakhand, India Monika Negi, Kanika Vishwakarma, Gold Rawat, Priyanka Badhani B.Tech, C.S.E.D G.B.P.E.C Uttarakhand, India

published may 2017. [link](#)

[39] Decoding Emotions: Harnessing the Power of Python for Sentiment Analysis in Social Media, by Mohammed RIZVI, published 2023-10-30.

<https://epess.net/index.php/epess/article/view/742>

[40] Sentiment analysis algorithms and applications: A survey, by Walaa Medhat, Ahmed Hassan and Hoda Korashy, published 27-5-2014.

<https://www.sciencedirect.com/science/article/pii/S2090447914000550>

[41] Sentiment Analysis: An Overview Comprehensive Exam Paper, by Yelena Mejova , published November 16 2009.

https://www.academia.edu/291678/Sentiment_Analysis_An_Overview

[42] A comprehensive survey on sentiment analysis: Approaches, challenges and trends, by Marouane Birjali, Mohammed Kasri, Abderrahim Beni-Hssane, published 17-08-2021.

<https://www.sciencedirect.com/science/article/abs/pii/S095070512100397X#coi1>

[43] Sentiment analysis using product review data, by Xing Fang & Justin Zhan , published 16-06-2015.

<https://link.springer.com/article/10.1186/s40537-015-0015-2>

[44] Hutto, C., & Gilbert, E. (2014). VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. Proceedings of the International AAAI Conference on Web and Social Media, 8(1), 216-225.

<https://ojs.aaai.org/index.php/icwsm/article/view/14550>

[45] Sentiment analysis on Twitter data integrating TextBlob and deep learning models: The case of US airline industry, Wajdi Aljedaani a, Furqan Rustam b, Mohamed Wiem Mkaouer c, Abdullatif Ghallab d, Vaibhav Rupapara e, Patrick Bernard Washington f, Ernesto Lee g, Imran Ashraf h, published 14 November 2022.

<https://www.sciencedirect.com/science/article/abs/pii/S0950705122009017#sec3>

[46] Study of Python libraries for NLP, by Anand Khandare, Nipun Agarwal, Amruta Bodhankar, Ankur Kulkarni and Ishaan Mane, published 25-07-2023.

<https://www.inderscienceonline.com/doi/abs/10.1504/IJDATS.2023.132564>

[47] Sentiment analysis using Twitter data: a comparative application of lexicon- and machine-learning-based approach, by Yuxing Qi & Zahratu Shabrina, published 09-02-2023.

<https://link.springer.com/article/10.1007/s13278-023-01030-x>

[48] Vader Official Documentation Site, <https://vadersentiment.readthedocs.io/en/latest/>

[49] Text Blob Official Documentation Site, <https://textblob.readthedocs.io/en/dev/>

[50] What is named entity recognition , <https://www.ibm.com/think/topics/named-entity-recognition>