ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

“ΤΕΧΝΟΛΟΓΙΕΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ”

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Ανίχνευση Σαρκασμού στο Twitter με την Αξιοποίηση Τεχνικών Βαθιάς Μάθησης

Δήμητρα Δ. Γεωργίου
Μαρίνα Κ. Καρανίκα

Επιβλέπουσα: **Χριστίνα Αλεξανδρή,** Καθηγήτρια

ΑΘΗΝΑ

ΟΚΤΩΒΡΙΟΣ 2022

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Ανίχνευση Σαρκασμού στο Twitter με την Αξιοποίηση Τεχνικών Βαθιάς Μάθησης

Δήμητρα Δ. Γεωργίου
Α.Μ.: IC1180001

Μαρίνα Κ. Καρανίκα
Α.Μ.: IC1180004

ΕΠΙΒΛΕΠΟΥΣΑ: **Χριστίνα Αλεξανδρή,** Καθηγήτρια

ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ: **Χριστίνα Αλεξανδρή,** Καθηγήτρια
Μανόλης Κουμπαράκης, Καθηγητής
Γιάννης Παναγάκης, Αναπληρωτής Καθηγητής

Οκτώβριος 2022

ΠΕΡΙΛΗΨΗ

Η παρούσα έρευνα ασχολείται με το πρόβλημα της ανίχνευσης του σαρκασμού, σε τίτλους ειδήσεων του κοινωνικού δικτύου *Twitter*.

Η μεταφορική γλώσσα, η οποία αποτελείται από στοιχεία όπως η ειρωνεία, ο σαρκασμός και η σάτιρα, έχει θεωρηθεί διάχυτο φαινόμενο σε πλατφόρμες κοινωνικής δικτύωσης όπως το *Twitter* [1]. Συγκεκριμένα, η σατιρική / σαρκαστική ειδησεογραφία είναι ιδιαίτερα δημοφιλής στα κοινωνικά δίκτυα, στα οποία είναι σχετικά εύκολο να μιμηθεί κανείς μια αξιόπιστη πηγή ειδήσεων, και συχνά μπορεί να συγχέεται με αληθινές ειδήσεις. Η ανίχνευση του σαρκασμού είναι, επομένως, θεμελιώδης για ορισμένες προσεγγίσεις σε πεδία όπως η Ανάλυση Συναισθημάτων, καθώς οι εκφράσεις με ειρωνεία / σαρκασμό μπορούν να παίξουν το ρόλο των αντιστροφέων πολικότητας.

Αρκετές υπολογιστικές προσεγγίσεις των οποίων ο στόχος είναι να εντοπιστεί η ειρωνεία, ο σαρκασμός και η σάτιρα έχουν εμφανιστεί τα τελευταία χρόνια και σε πολλές από αυτές οι όροι θεωρούνται συνώνυμοι. Η παρούσα έρευνα ακολουθεί αυτήν την προσέγγιση. Η σάτιρα είναι ένα σημαντικό γλωσσικό φαινόμενο που αποτελείται από τη χρήση του χιούμορ και της ειρωνείας για την κριτική και τη γελοιοποίηση κάποιου προσώπου ή μιας κατάστασης.

Η σάτιρα ειδήσεων είναι ιδιαίτερα δημοφιλής στα κοινωνικά δίκτυα στα οποία είναι σχετικά εύκολο να μιμηθεί κανείς μια αξιόπιστη πηγή ειδήσεων. Ωστόσο, η σάτιρα ειδήσεων συχνά συγχέεται με αληθινές ειδήσεις, ειδικά όταν διαχωρίζεται από την αρχική της πηγή. Μερικές ελληνικές σατιρικές πηγές ειδήσεων είναι «*Το Βατράχι*» και «*Το Κουλούρι*».

Για την ανίχνευση του σαρκασμού στα δεδομένα κειμένου στα ελληνικά, συλλέχθηκαν με το χέρι, μέσω του Twitter API, τίτλοι ειδήσεων από τις προαναφερθείσες πηγές σατιρικών ειδήσεων καθώς και από τους ειδησεογραφικούς ιστότοπους «*CNN Greece*» και «*HuffPost Greece*». Στη συνέχεια, δημιουργήθηκαν 7 μοντέλα νευρωνικών δικτύων τα οποία ήταν ένας συνδυασμός από Νευρωνικά Δίκτυα Μακράς Βραχυπρόθεσμης Μνήμης (LSTM), Αμφίδρομα Δίκτυα Μακράς Βραχύχρονης Μνήμης (BiLSTM), τα οποία τροφοδοτήθηκαν με προ-εκπαιδευμένες αναπαραστάσεις λέξεων (word embeddings). Αυτές οι αναπαραστάσεις έχουν δημιουργηθεί από τα γλωσσικά μοντέλα Word2Vec, FastText και Greek-BERT. Εφαρμόζεται και μία σειρά μεθόδων που χρησιμοποιούνται ως συγκριτικό σημείο αναφοράς (benchmark), όπως οι Bag-of-Words και οι TF-IDF σε συνδυασμό με παραδοσιακούς ταξινομητές μηχανικής μάθησης, όπως η Λογιστική Παλινδρόμηση, ο πολυωνυμικός Naïve Bayes ταξινομητής και Μηχανές Διανυσμάτων Υποστήριξης. Τα μοντέλα αυτά αξιολογήθηκαν με διάφορες μετρικές όπως η ακρίβεια (accuracy) και το F1 Score.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Ανίχνευση Σαρκασμού

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: ανάλυση συναισθήματος, επεξεργασία φυσικής γλώσσας, μηχανική μάθηση, βαθιά νευρωνικά δίκτυα, προ-εκπαιδευμένες αναπαραστάσεις λέξεων

ABSTRACT

The present research concerns the detection of sarcasm in media headlines and in social media and the issues involving its processing. In the present Thesis, the detection of sarcasm as a Natural Language Processing (NLP) application is approached with the use of Artificial Neural Networks (NNNs) utilizing distributed representations of words (word embeddings).

Figurative language, which consists of elements such as irony, sarcasm, and satire, has been considered a pervasive phenomenon in social networking platforms such as *Twitter* [1]. Figurative language detection is, therefore, fundamental in some approaches in fields such as Sentiment Analysis, as ironic expressions can play the role of polarity reversers.

Several computational approaches whose goal is to detect sarcasm and irony have emerged in recent years. However, sarcasm in combination with satire are issues that are seldom addressed in current research. Satire is an important linguistic phenomenon consisting of the use of humor and irony to criticize and ridicule a person or situation. Parody, burlesque, exaggeration, juxtaposition, comparison, analogy, and double entendre are often used in satirical speech and writing. Satirical journalism is a genre of parody that is presented as a typical form of mainstream journalism/news and is called satire because of its content.

News satire is particularly popular on the web, and in particular on social networks where it is relatively easy to impersonate a credible news source and stories can achieve wide distribution from almost any site. However, news satire is often confused with real news, especially when separated from its original source, such as when a tweet from a satirical *Twitter* account is recommended via Facebook. The present research includes the processing of data from Modern Greek. The present research includes the processing of data from Modern Greek. Some Greek satirical news sources are "*To Koulouri*" and "*To vatraxi*".

To detect sarcasm in Greek text data, 7 neural network models were created which were a combination of Long Short-Term Memory (LSTM) Neural Networks, Bi-directional Long Short-Term Memory (BiLSTM) Networks, which were fed with pre-trained word representations (word embeddings). These representations are generated by the Word2Vec, FastText and Greek-BERT language models. A number of benchmark methods were also implemented, such as Bag-of-Words and TF-IDF in combination with traditional machine learning classifiers, such as Logistic Regression, polynomial Naïve Bayes classifier and Support Vector Machines. These models were evaluated with various metrics such as accuracy and F1 Score.

SUBJECT AREA: Sarcasm Detection

KEYWORDS: sentiment analysis, natural language processing, machine learning, deep neural networks, pretrained word embeddings

Στις οικογένειές μας.

ΕΥΧΑΡΙΣΤΙΕΣ

Θα θέλαμε να ευχαριστήσουμε από καρδιάς την επιβλέπουσα καθηγήτριά μας, Χριστίνα Αλεξανδρή, για την ευκαιρία που μας έδωσε να ασχοληθούμε με αυτό το αντικείμενο και την καθοδήγηση που μας παρείχε καθ΄ όλη τη διάρκεια της έρευνάς μας. Μας στήριξε σε όλη την προσπάθειά μας και μας ενθάρρυνε από την πρώτη στιγμή. Ήταν πάντα διαθέσιμη, πρόθυμη να συζητήσει μαζί μας και να μας κατευθύνει. Είμαστε ευγνώμονες που είχαμε την ευκαιρία να συνεργαστούμε μαζί της.

ΠΕΡΙΕΧΟΜΕΝΑ

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

ΠΡΟΛΟΓΟΣ

Η εργασία αυτή διεξήχθη στο πλαίσιο του Προγράμματος Μεταπτυχιακών Σπουδών «Τεχνολογίες Πληροφορικής και Επικοινωνιών» του τμήματος Τεχνολογίες Πληροφορικής και Επικοινωνιών.

Επιλέχθηκε ένα θέμα με μεγάλο ερευνητικό ενδιαφέρον, η Ανάλυση Συναισθήματος, και πιο συγκεκριμάνα η ανίχνευση σαρκασμου στα κοινωνικά δίκτυα. Η παρούσα μελέτη υλοποιήθηκε σε προγραμματιστικό περιβάλλον Python με χρήση βοηθητικών βιβλιοθηκών.

1. ΕΙΣΑΓΩΓΗ

1.1 Ανάλυση συναισθήματος και κοινωνικά δίκτυα

Με την ραγδαία ανάπτυξη της χρήσης μέσων κοινωνικής δικτύωσης όπως το *Twitter*, το *Reddit*, το *Facebook* καθώς και των εταιρειών ηλεκτρονικού εμπορίου όπως η *Amazon*, έχει δημιουργηθεί ένας τεράστιος όγκος δεδομένων - μεγάλα δεδομένα. Ένα σημαντικό ποσό αυτών των δεδομένων που δημιουργούνται από τον χρήστη είναι σε μορφή κειμένου όπως κριτικές, tweets και blogs και παρέχουν πολλές προκλήσεις καθώς και ευκαιρίες σε ερευνητές του τομέα Επεξεργασίας Φυσικής Γλώσσας - ΕΦΓ (Natural Language Processing - NLP) για την ανακάλυψη σημαντικών πληροφοριών που προέρχονται από διάφορες εφαρμογές.

Οι διαθέσιμες πληροφορίες κειμένου είναι δύο τύπων: γεγονότα και δηλώσεις γνώμης. Τα γεγονότα είναι αντικειμενικές προτάσεις για τις οντότητες. Από την άλλη πλευρά, οι απόψεις είναι υποκειμενικής φύσης και γενικά περιγράφουν τα συναισθήματα των ανθρώπων απέναντι σε οντότητες και γεγονότα.

Η έρευνα σχετικά με την επεξεργασία των υποκειμενικών προτάσεων είναι ένας από τους ενεργούς και δημοφιλείς ερευνητικούς τομείς λόγω του μεγάλου αριθμού των προκλήσεων που εμπλέκονται καθώς και του μεγάλου φάσματος επιχειρηματικών και κοινωνικών εφαρμογών. Η Ανάλυση Συναισθήματος - ΑΣ, ως ερευνητικός κλάδος παρέχει μια αυτοματοποιημένη υπολογιστική προσέγγιση για την επεξεργασία και την ανακάλυψη αυτών των στοιχείων και απόψεων από το κείμενο. Είναι η μελέτη που αναλύει τη γνώμη και το συναίσθημα των ανθρώπων απέναντι σε οντότητες όπως προϊόντα, υπηρεσίες, άτομα και οργανισμούς που υπάρχουν στο κείμενο. Η αυτοματοποιημένη ανάλυση του διαδικτυακού περιεχομένου για την εξαγωγή της γνώμης απαιτεί από τις μηχανές να δημιουργήσουν μια βαθιά κατανόηση του φυσικού κειμένου.

Η ανάλυση συναισθημάτων αποδεικνύεται εξαιρετικά χρήσιμη στην παρακολούθηση των κοινωνικών μέσων, καθώς δίνει τη δυνατότητα επισκόπησης της ευρύτερης κοινής γνώμης πίσω από ορισμένα θέματα. Ακολουθούν κάποιες πολύ δημοφιλείς εφαρμογές χρήσης ανάλυσης συναισθημάτων στα κοινωνικά δίκτυα:

- Κυβερνήσεις, καταναλωτές και επωνυμίες εκμεταλλεύονται αυτές τις πλατφόρμες για να μοιράζονται προωθητικές προσφορές, να ανταλλάσσουν ιδέες, να αυξάνουν την ευαισθητοποίηση σε κοινωνικά θέματα και να προωθούν προϊόντα και υπηρεσίες.
- Δίνεται η δυνατότητα σύγκρισης της απόδοσης της εκάστοτε επωνυμίας και της απόδοσης των καμπανιών αυτής στα κοινωνικά κανάλια με εκείνες των ανταγωνιστών και εντοπίζονται οι τομείς που χρειάζονται βελτίωση χρησιμοποιώντας συγκριτική αξιολόγηση ανταγωνιστών.
- Μέσω των εφαρμογών ανάλυσης συναισθήματος μπορεί να δοθεί προτεραιότητα σε συγκεκριμένους χρήστες στα μέσα κοινωνικής δικτύωσης. Συγκεκριμένα, γίνεται φιλτράρισμα των μη αναγνωσμένων ή των νέων αναφορών βάσει του συναισθήματος. Με αυτόν τον τρόπο δεν δίνεται σημασία σε χρήστες χαμηλότερης προτεραιότητας, ενώ γίνεται εστίαση σε πράγματα που πρέπει πρώτα να αντιμετωπιστούν. Με αυτό τον τρόπο είναι εφικτό να απαντηθούν μηνύματα δυσαρεστημένων πελατών που χρειάζονται βοήθεια ή απάντηση σε αρνητικά σχόλια τρίτων.
- Ο υπολογισμός τη φήμης μίας εταιρίας γίνεται με χρήση εφαρμογών ανάλυσης συναισθήματος στα κοινωνικά δίκτυα. Μπορεί να ανιχνευθεί το συναίσθημα που

προκύπτει από κάθε μεμονωμένη αναφορά κάποιου χρήστη, αλλά και να δημιουργηθεί μία γενική εικόνα της φήμης της επωνυμίας. Οι αναφορές κατηγοριοποιούνται βάσει του συναισθήματος και μπορούν να βοηθήσουν στην ανίχνευση κατά πόσο θετικά ή αρνητικά συναισθήματα δημιουργεί το brand μίας εταιρίας βάσει του τόνου των αναφορών στα μέσα κοινωνικής δικτύωσης.

1.2 Ορίζοντας τον σαρκασμό

Ο σαρκασμός είναι μια μορφή επικοινωνίας όπου το άτομο δηλώνει το αντίθετο από αυτό που υπονοείται. Είναι μια έκθεση παιχνιδιάρικης στάσης στην ανάλυση της συμπεριφοράς ή της προσέγγισης κάποιου στη ζωή. Χρησιμοποιείται για να υποτιμήσει, να γελοιοποιήσει, να εκφράσει δυσαρέσκεια και να περιφρονήσει ένα συγκεκριμένο άτομο ή πράγμα, άμεσα ή έμμεσα. Συχνά ένα σαρκαστικό σχόλιο μπορεί να είναι ιδιαίτερα σκληρό, καυστικό και χαιρέκακο σε σημείο προσβολής και να επηρεάσει τον παραλήπτη.

Το ρήμα σαρκάζω χρησιμοποιήθηκε αρχικά με την σημασία «σχίζω την σάρκα» και «κόβω χόρτα, αφού τα αποσπάσω με σφιγμένα χείλη», από όπου προήλθε η σημασία «δαγκώνω τα χείλη μου από θυμό» και, κατ' επέκταση, «μιλώ ειρωνικά, χλευάζω».

Ο σαρκασμός καταγράφηκε για πρώτη φορά σε ένα αγγλικό βιβλίο που ονομάζεται "The Shepheardes Calender" από τον Edmund Spenser το 1579 (Oxford English Dictionary).

- *Tom Piper: an ironicall Sarcasmus, spoken in derision of these rude wits, which make more account of a ryming rybaud, then of skill grounded vpon learning and iudgment.*

Ωστόσο, η λέξη sarcastic, που σημαίνει «Χαρακτηρίζεται από ή περιλαμβάνει σαρκασμό· δίνεται στη χρήση του σαρκασμού· πικρόχολη ή καυστική», εμφανίζεται μόλις το 1695 (Oxford English Dictionary).

Η ειρωνεία και ο σαρκασμός είναι δύο όροι οι οποίοι συγχέονται και τα όρια που τους διαχωρίζουν είναι δυσδιάκριτα. Έχει διατυπωθεί ότι συνιστούν διαφορετικές έννοιες ([2], [3]), ότι είναι συνώνυμοι όροι που περιγράφουν το ίδιο φαινόμενο [4] και ότι πρόκειται για αλληλεπικαλυπτόμενες έννοιες ([5], [6]). Σε κάθε περίπτωση, παρά την απουσία μιας ευρέως αποδεκτής άποψης, όλοι οι ερευνητές συγκλίνουν στην παραδοχή της γειτνίασης των δύο φαινομένων [7].

Σύμφωνα με τον Colston, ο στόχος του σαρκασμού είναι η άσκηση δριμείας κριτικής. Συνιστά μια αρνητική μορφή ειρωνείας και μάλιστα η αρνητικότητα που εκφράζει είναι εντονότερη από την άμεση κριτική που εκφράζουμε κυριολεκτικά, χωρίς την χρήση σχηματικού λόγου. Επιπλέον ο Colston τονίζει, πως η έκφραση μιας αρνητικής διάθεσης είναι μια από της πρωταρχικές λειτουργίες της λεκτικής ειρωνείας, εντάσσοντας έτσι τον σαρκασμό στις προτυπικές μορφές ειρωνείας [8].

Η Φαρακλού [9] προσθέτει ότι ο σαρκασμός συνδυάζει την πρόθεση κριτικής και στηλίτευσης με το κυριολεκτικό σχόλιο και δεν αφήνει περιθώρια αμφισημίας. Υπογραμμίζει ότι ο αποδέκτης - στόχος είναι ένα πραγματικό πρόσωπο, με φυσική παρουσία στην συνομιλία, σε αντίθεση με την ειρωνεία, η οποία μπορεί να αναφέρεται σε ιδέες, απόψεις και άτομα παρόντα ή απόντα από την συνομιλία. Αντίστοιχα, και ο Attardo [6] υποστηρίζει ότι ο σαρκασμός είναι μια επιθετική μορφή ειρωνείας, με ισχυρούς δείκτες που τον σηματοδοτούν και έναν φανερό στόχο στον οποίο απευθύνεται.

Οι Lee & Katz [10] δίνουν έμφαση στην διάθεση γελοιοποίησης που είναι φανερή στον σαρκασμό, ενώ δεν εμφανίζεται έντονη στην ειρωνεία. Ο Haiman [11] μιλά για την

σκοπιμότητα, την συνειδητή πρόθεση (κριτικής) που διακρίνει τον σαρκασμό και προσθέτει ότι εμφανίζεται μόνο με γλωσσική μορφή, σε αντίθεση με την ειρωνεία που μπορεί να είναι και καταστασιακή, τραγική κ.λπ.

Στις αναπτυξιακές έρευνες, ο σαρκασμός χρησιμοποιείται ως συνώνυμος όρος με την ειρωνεία και προτιμάται λόγω της εξοικείωσης των συνομιλητών με αυτόν [12]. Οι ψυχογλωσσολογικές προσεγγίσεις έχουν φέρει στο προσκήνιο τον ρόλο της οπτικής στην ανάλυση του σαρκασμού. Δηλαδή, για να αναλύσουμε τα χαρακτηριστικά του σαρκασμού πρέπει να λάβουμε υπόψη μας τις τρεις οπτικές μέσα από τις οποίες μπορεί να ιδωθεί. Την οπτική του ομιλητή, του ακροατή και του κοινού. Πειραματικές μελέτες έχουν δείξει, ότι ο ομιλητής που παράγει ένα σαρκαστικό εκφώνημα θεωρεί τον εαυτό του ικανό και επιδέξιο. Ο ακροατής- θύμα αξιολογεί τον σαρκασμό ως μια δραστική μορφή κριτικής απέναντί του και τέλος, η ύπαρξη ακροατηρίου εντείνει την οξύτητα του σαρκασμού ([13], [14]).

Από την άλλη πλευρά, οι Littman & Mey [2], βλέπουν τον σαρκασμό ως μια γλωσσική πράξη με στόχο την προσβολή του προσώπου στο οποίο απευθύνεται. Σημειώνουν την εγγύτητά του με την ειρωνεία και το χιούμορ, αλλά τον κατατάσσουν στις γλωσσικές πράξεις που εκφράζουν επιθετικότητα, όπως την επίπληξη, τις ύβρεις και την προσβολή των πεποιθήσεων των άλλων. Τονίζουν, ότι ένα σαρκαστικό εκφώνημα μπορεί να περιέχει ειρωνεία, αλλά αυτό δεν είναι απαραίτητο. Για να αποσαφηνίσουμε αυτή την διατύπωση, θα λέγαμε ότι ένα σαρκαστικό εκφώνημα μπορεί να επιτυγχάνει την άσκηση δριμείας κριτικής μέσω της ειρωνείας, μέσω μιας μορφής αντιστροφής. Για παράδειγμα, το εκφώνημα «Δύο ειλικρινείς ανθρώπους ξέρω, εσένα και τον Πινόκιο, είναι σαρκαστικό γιατί στοχεύει στην τρώση του θετικού προσώπου του ακροατή, εφόσον αμφισβητεί την ειλικρίνειά του. Ωστόσο, ένα σαρκαστικό εκφώνημα μπορεί να μην περιέχει ειρωνεία, αλλά να συνιστά ένα κυριολεκτικό σχόλιο. Σε ένα περικείμενο όπου λόγου χάρη δύο αγόρια καυγαδίζουν, και ειπωθεί το εκφώνημα: «Νομίζεις ότι είσαι μεγάλος άντρας; Μάθε λοιπόν ότι οι μεγάλοι άντρες, τρώνε μεγάλες γροθιές.» Το αγόρι που παράγει ένα σαρκαστικό εκφώνημα έχει στόχο να προσβάλει τον συνομιλητή του και να το παρουσιάσει αδύναμο, με έλλειψη θάρρους, αλλά καμία από τις λέξεις που χρησιμοποιεί δεν αντιστρέφεται σημασιολογικά. Όλες χρησιμοποιούνται με τη σημασία με την οποία εκφράστηκαν. Την ίδια άποψη εκφράζει και ο Brown (2006), υποστηρίζοντας ότι είναι διακριτά φαινόμενα. Άρα, υπάρχει και η άποψη ότι ο σαρκασμός δεν ταυτίζεται με την ειρωνεία.

Ο σαρκασμός συχνά χρησιμοποιείται και ως συνώνυμο της σάτιρας. Στη λογοτεχνία, οι συγγραφείς χρησιμοποιούν το σαρκασμό και τη σάτιρα για να διασκεδάσουν τον αναγνώστη καθώς και για να μεταφέρουν διαφορετικά χαρακτηριστικά αυτών στα έργα τέχνης τους. Ο σαρκασμός αναφέρεται στη χρήση ειρωνείας για να χλευάσουμε ή να μεταδώσουμε περιφρόνηση. Η σάτιρα, από την άλλη πλευρά, μπορεί να θεωρηθεί ως η χρήση χιούμορ και ειρωνείας για κριτική και γελοιοποίηση άλλων.

Η σάτιρα κάνει συχνά μια προσπάθεια να δημιουργήσει ένα εποικοδομητικό σημείο ή δύο. Στόχος του είναι να αποδείξει τον παραλογισμό μιας δεδομένης κατάστασης, πολιτικής ή κοινωνικής. Αυτός είναι πρωτίστως ο λόγος που η σάτιρα θεωρήθηκε πρωταρχική μέθοδος κριτικής των πολιτικών και κοινωνικών καταστάσεων από τους ποιητές του προηγούμενου έτους. Ο Αλέξανδρος Πάπας ήταν ένας μεγάλος σατιριστής. Είναι ενδιαφέρον να σημειωθεί ότι ένα άτομο που εμπλέκεται στη σάτιρα είτε γραπτά είτε μιλώντας ονομάζεται σατιριστής. Ένας σατιριστής ασχολείται κυρίως με την περιγραφή των αυξανόμενων αλλαγών στον τρόπο ζωής και την κοινωνική συμπεριφορά.

Στα πλαίσια της παρούσας έρευνας ο σαρκασμός, η ειρωνεία και η σάτιρα θεωρούνται συνώνυμοι όροι, όπως και σε μεγάλο μέρος της βιβλιογραφίας που αφορά στην ανίχνευσή τους σε κείμενα με χρήση αυτόματων μεθόδων.

Μερικά παραδείγματα σαρκασμού

- «*Γιατί βρίζεις, νεαρέ; Αυτά σου μαθαίνουν στο σχολείο; Συγχαρητήρια! Πολύ εγκόσμια η γλώσσα σου!*» Το κωμικό στοιχείο το αντιλαμβάνεται είτε ο ίδιος ο νεαρός, εφόσον το γλωσσικό του επίπεδο το επιτρέπει, είτε κάποιος τρίτος που ακούει τον επίδοξο γλωσσαμύντορα να πέφτει στο γλωσσικό ολίσθημα.
- Βαθιά θρησκευόμενος διηγείται πως είδε το εικόνισμα της Παναγίας να δακρύζει. Και ο είρωνας, προσποιούμενος έκπληξη και θαυμασμό: «*Τι μου λες; Τελικά, αργά ή γρήγορα, η πραγματική πίστη πάντα ανταμείβεται...*» Για τον είρωνα, η πραγματική πίστη δεν είναι άλλο από την τυφλή/δογματική πίστη και η ανταμοιβή, απλώς μια απάτη.
- «*Ωραίο άρωμα! Πόσο καιρό μαρινάρεσαι σε αυτό;*» Όταν κάποιος βάζει πολύ άρωμα.
- Όταν κάποιος λέει κάτι που είναι πολύ προφανές: «*Αλήθεια, Σέρλοκ; Όχι! Είσαι πάρα πολύ έξυπνος*.»
- «*Είναι ώρα για τα φάρμακά σου ή τα δικά μου;*» Όταν ο συγκάτοικός σου φέρεται παράξενα.

Μερικά διάσημα σαρκαστικά αποσπάσματα

- «*Κάθε φορά που σε κοιτάζω με πιάνει μια έντονη επιθυμία να είμαι μοναχικός*.» - Oscar Levant
- «*Αναγνώστη, ας υποθέσουμε ότι ήσασταν ηλίθιος. Τώρα υποθέστε ότι ήσαστε μέλος του Κογκρέσου. Αλλά επαναλαμβάνω τον εαυτό μου*.» - Mark Twain
- «*Δεν ξεχνώ ποτέ ένα πρόσωπο, αλλά στην περίπτωσή σας θα χαρώ να κάνω μια εξαίρεση*.» - Groucho Marx
- «*Νιώθω τόσο άθλια χωρίς εσένα, είναι σχεδόν σαν να σε έχω εδώ*.» - Stephen Bishop

Σαρκαστικά αποσπάσματα από βιβλία, ταινίες και σειρές

- Ο Χάρι Πότερ και το Τάγμα του Φοίνικα της J.K. Rowling

  - *Τι κάνεις κάτω από το παράθυρό μας, νεαρέ;*

  - *Ακούω τις ειδήσεις*, είπε ο Χάρι με παραιτημένη φωνή.

  Η θεία του και ο θείος του αντάλλαξαν βλέμματα αγανάκτησης.

  - *Ακούς τα νέα; Πάλι;*

  - *Να, βλέπετε, αλλάζουν κάθε μέρα!*, είπε ο Χάρι.

  Ο Χάρι Πότερ κάνει ξεκάθαρο πόσο χαζό θεωρεί τον θείο του Βέρνον.

- Η Διάσωση του Andy Weir

- Αστείο, είπε ο Βένκατ. *Να το παίζεις έξυπνος σε έναν άντρα επτά επίπεδα πάνω από σένα στην εταιρεία σου. Δες πώς γίνεται αυτό*.

- Ω, όχι, είπε η Μίντι, υπάρχει περίπτωση να χάσω τη δουλειά μου ως διαπλανητικός παρατηρητής; Μάλλον τότε θα πρέπει να χρησιμοποιήσω το μεταπτυχιακό μου για κάτι άλλο.

- Το λάθος αστέρι του John Greene

  - Σας ευχαριστώ που μου εξηγήσατε ότι ο καρκίνος των ματιών μου δεν πρόκειται να με κάνει κωφό. Αισθάνομαι τόσο τυχερός που ένας πνευματικός γίγαντας σαν εσάς θα ήθελε να με χειρουργήσει.

  Το βιβλίο εμβαθύνει σε αρκετά βαριά θέματα όπως ο θάνατος και ο παιδικός καρκίνος. Αλλά αυτό δεν σημαίνει ότι απαγορεύεται να προστεθεί λίγος σαρκασμός, όπως φαίνεται και από το απόσπασμα του Ισαάκ.

- Αγώνες Πείνας της Suzanne Collins

  - Συνοικία 12. Εκεί που μπορείς να πεθάνεις από την πείνα με ασφάλεια., μουρμουρίζω. Μετά ρίχνω μια ματιά γρήγορα πίσω μου. Ακόμα κι εδώ, στη μέση του πουθενά, ανησυχεί κανείς ότι κάποιος μπορεί να τον ακούσει.

  Η Κάτνις Έβερντιν πέρασε μια σκληρή ζωή στην Συνοικία 12 αλλά είναι φανερός ο σαρκασμός και η ειρωνεία στα λόγια της.

- The Big Bang Theory Αμερικανική κωμική σειρά των Chuck Lorre και Bill Prady (2007 - 2019)

  Λέοναρντ: «*Γεια, Πένυ. Πώς είναι η δουλειά;*»

  Πένυ: «*Τέλεια! Ελπίζω να είμαι σερβιτόρα στο Cheesecake Factory για όλη μου τη ζωή!*»

  Σέλντον: «*Ήταν η απάντησή σου σαρκαστική;*»

  Πένυ: «*Όχι.*»

  Σέλντον: «*Ήταν αυτή η απάντησή σου σαρκαστική;*»

  Πένυ: «*Ναι.*»

  Η Πένυ είναι εξαιρετική στο είναι σαρκαστική με τον εαυτό της. Αυτό είναι ιδιαίτερα αστείο αφού οι περισσότεροι από τους έξυπνους φίλους της δεν την αντιλαμβάνονται.

Ο σαρκασμός βρήκε και τον δρόμο του στο Διαδίκτυο, όπου κέρδισε δημοτικότητα. Απαντάται συχνά σε κοινωνικά δίκτυα, φόρουμ συζήτησης και ιστολόγια. Εκεί, ο σαρκασμός σταδιακά παραμορφώθηκε και μετατράπηκε σε “τρολάρισμα”. Τα άτομα που ασχολούνται με το τρολ λέγονται προφανώς “trols”. Τα τρολ γράφουν προκλητικά μηνύματα στο Διαδίκτυο, εξοργίζοντας τους άλλους χρήστες.

- «*Οι επιστήμονες έκαναν ένα πείραμα. Πήραν δύο λιοντάρια: το ένα το τάιζαν πολύ κρέας, και το άλλο πολλά λαχανικά. Μια εβδομάδα αργότερα, το χορτοφαγικό λιοντάρι πέθανε*».

Το σύνολο δεδομένων που δημιουργήθηκε είναι συλλογή τίτλων ειδήσεων από σατιρικές δημοσιογραφικές πηγές που χρησιμοποιούν σαρκαστικό και ειρωνικό ύφος.

1.3 Ανίχνευση του σαρκασμού στα κοινωνικά δίκτυα

Σε εργασίες Επεξεργασίας Φυσικής Γλώσσας όπως η ανάλυση συναισθημάτων στα κοινωνικά δίκτυα, η παρουσία του σαρκασμού στα κείμενα θα πρέπει να γίνεται αντιληπτή από τις αυτοματοποιημένες υπολογιστικές προσεγγίσεις.

Μία από τις μεγαλύτερες προκλήσεις της ανίχνευσης του σαρκασμού στα κοινωνικά δίκτυα είναι η διφορούμενη φύση του αφού δεν υπάρχει προδιαγεγραμμένος ορισμός του σαρκασμού. Μια άλλη σημαντική πρόκληση είναι το αυξανόμενο μέγεθος των γλωσσών. Κάθε μέρα εκατοντάδες νέες λέξεις αργκό δημιουργούνται και χρησιμοποιούνται σε αυτούς τους ιστότοπους. Ως εκ τούτου, το υπάρχον σώμα θετικών και αρνητικών συναισθημάτων μπορεί να μην είναι ακριβές στην ανίχνευση του σαρκασμού.

Επίσης, οι πρόσφατες εξελίξεις στα κοινωνικά δίκτυα επιτρέπουν στους χρήστες του να χρησιμοποιούν ποικίλα είδη emoticon με το κείμενο. Αυτά τα emoticon μπορεί να αλλάξουν την πολικότητα του κειμένου και να το κάνουν σαρκαστικό. Λόγω αυτών των δυσκολιών και της εγγενώς δύσκολης φύσης του σαρκασμού, συνήθως αγνοείται κατά την ανάλυση των κοινωνικών δικτύων. Έτσι, η ανίχνευση σαρκασμού αποτελεί ένα από τα πιο κρίσιμα προβλήματα που πρέπει να ξεπεράσουμε. Η ανίχνευση σαρκαστικού περιεχομένου είναι ζωτικής σημασίας για διάφορα συστήματα που βασίζονται σε εργασίες Επεξεργασίας Φυσικής Γλώσσας.

1.4 Αντικείμενο της παρούσας έρευνας

Ο στόχος εδώ είναι η έρευνα στο πεδίο της ανίχνευσης σαρκασμού και η ανάπτυξη μοντέλων, για τον εντοπισμό του σε τίτλους ειδήσεων από το *Twitter*. Χρησιμοποιήθηκαν τίτλοι ειδήσεων / tweets από 4 διαφορετικούς ελληνικούς ειδησεογραφικούς ιστότοπους: *HuffPost Greece, CNN Greece, Το Κουλούρι, Το Βατράχι*, από τους οποίους οι δύο τελευταίοι επικεντρώνονται σε σατιρικές / σαρκαστικές ειδήσεις.

Η ΑΣ και συγκεκριμένα η ανίχνευση σαρκασμού είναι ένα ερευνητικό πεδίο με μεγάλο ενδιαφέρον, τόσο από την επιστημονική κοινότητα, όσο και από την βιομηχανία, λόγω της πληθώρας των εφαρμογών και προκλήσεων. Η ΑΣ στο *Twitter* είναι ακόμη δυσκολότερο πρόβλημα, εξαιτίας της ιδιαιτερότητας του μέσου. Για την ανάπτυξη των αντίστοιχων μοντέλων, αρχικά απαιτείται μία κατανόηση της ΑΣ σε θεωρητικό επίπεδο και στη συνέχεια των πιο σημαντικών προσεγγίσεων, οι οποίες έχουν δοκιμαστεί από την επιστημονική κοινότητα.

Πιο αναλυτικά, οι βασικοί στόχοι της παρούσας προσέγγισης είναι:

- Σύνοψη της ΑΣ και της ανίχνευσης σαρκασμού. Αυτό αφορά μια μικρή παρουσίαση του θεωρητικού πλαισίου σχετικά με την ΑΣ και την ανίχνευση σαρκασμού και μία εκτεταμένη ιστορική αναδρομή αναφορικά με τις τεχνικές που έχουν χρησιμοποιηθεί για την υλοποίηση μοντέλων ανίχνευσης σαρκασμού.
- Έρευνα στις τεχνικές μηχανικής μάθησης για την ανάπτυξη μοντέλων ΕΦΓ. Αυτό αφορά την έρευνα στους αλγορίθμους μηχανικής μάθησης για την δημιουργία μοντέλων. Η μεγαλύτερη βαρύτητα θα δοθεί στην έρευνα σε τεχνικές βαθιών τεχνητών νευρωνικών δικτύων σε συνδυασμό με προ-εκπαιδευμένες αναπαραστάσεις λέξεων με τεχνικές μηχανικής μάθησης, καθώς τα τελευταία χρόνια έχουν κυριαρχήσει στο πεδίο της ΕΦΓ.
- Ανάπτυξη μοντέλων ανίχνευσης σαρκασμού για μηνύματα στο *Twitter*. Αυτός είναι ο απώτερος στόχος της παρούσας έρευνας. Γίνεται σύγκριση των μοντέλων

βαθιάς μάθησης με τις παραδοσιακές προσεγγίσεις για να διαπιστωθεί η αποτελεσματικότητά τους.

1.5 Δομή της εργασίας

Το πρώτο τμήμα της παρούσας εργασίας ασχολείται με την σύνοψη του πεδίου της ΑΣ καθώς και πιο συγκεκριμένα με την ανίχνευση του σαρκασμού (Κεφ. 1) όπου δίνονται οι ορισμοί των σχετικών εννοιών. Στην συνέχεια αναλύονται οι στόχοι και η δομή της εργασίας.

Στο δεύτερο κεφάλαιο (Κεφ. 2), γίνεται μια συστηματική βιβλιογραφική ανασκόπηση όλων των εργασιών που καταφέραμε να συγκεντρώσουμε σχετικά με τον ορισμό και εντοπισμό το σαρκασμού, στα πλαίσια κυρίως εργασιών της ΕΦΓ. Συγκεντρώθηκαν οι γλώσσες για τις οποίες έχουν γίνει έρευνες, τα σύνολα δεδομένων που χρησιμοποιήθηκαν, οι μέθοδοι και τα μοντέλα καθώς και οι τεχνικές που δημιουργήθηκαν για την εργασία της ανίχνευσης του σαρκασμού.

Στο επόμενο κεφάλαιο (Κεφ. 3), βρίσκεται το θεωρητικό υπόβαθρο όλων των τεχνικών που χρησιμοποιούνται στο κεφάλαιο 4. Αρχικά παρουσιάζονται τα βασικά χαρακτηριστικά της Ανάλυσης Συναισθήματος. Στη συνέχεια, αναλύονται κάποιοι παραδοσιακοί αλγόριθμοι μηχανικής μάθησης και βασικές έννοιες για τα τεχνητά νευρωνικά δίκτυα. Τέλος, γίνεται παρουσίαση των βασικών εννοιών που αφορούν στη διανυσματική αναπαράσταση λέξεων και μιας ιστορικής αναδρομής της εξέλιξης των μοντέλων που χρησιμοποιούνται για την δημιουργία τέτοιων αναπαραστάσεων.

Συνεχίζοντας στο κεφάλαιο 4, παρουσιάζονται ορισμένα μοντέλα, τα οποία υλοποιήθηκαν για την ανίχνευση του σαρκασμού σε μηνύματα από το Twitter και αναλύεται ο τρόπος που εκπαιδεύτηκαν. Γίνεται σύγκριση των κατηγοριοποιητών που χρησιμοποιούν νέες αρχιτεκτονικές νευρωνικών δικτύων συνδυαστικά με τη χρήση της διανυσματικής αναπαράστασης λέξεων, σε σχέση με τις πιο παραδοσιακές προσεγγίσεις.

Στο κεφάλαιο 5 παρουσιάζονται τα αποτελέσματα της απόδοσης των κατηγοριοποιητών με χρήση γραφημάτων και πινάκων.

Τέλος, στα κεφάλαια 6 και 7 παρουσιάζονται συνοπτικά τα συμπεράσματα και με βάση αυτά κάποιες προτάσεις για μελλοντική έρευνα.

2. ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΑΝΑΣΚΟΠΗΣΗ

Σε αυτό το κεφάλαιο παρουσιάζεται αναλυτικά η υπάρχουσα βιβλιογραφία γύρω από το θέμα της ανίχνευσης του σαρκασμού.

2.1 Εισαγωγή

Ο σαρκασμός είναι μια μεγάλη πρόκληση για τα μοντέλα ανάλυσης συναισθήματος [15], κυρίως επειδή ο σαρκασμός επιτρέπει σε έναν ομιλητή ή συγγραφέα να κρύψει την πραγματική του πρόθεση περιφρόνησης και αρνητικότητας υπό το πρόσχημα της έκδηλης θετικής αναπαράστασης. Έτσι, η αναγνώριση του σαρκασμού και της λεκτικής ειρωνείας είναι κρίσιμη για την κατανόηση των πραγματικών συναισθημάτων και πεποιθήσεων των ανθρώπων [16].

Η έρευνα πάνω στην ανίχνευση του σαρκασμού είναι ένας δημοφιλής τομέας έρευνας τα τελευταία χρόνια λόγω της δημοτικότητας των πλατφορμών κοινωνικής δικτύωσης (π.χ. *Amazon*) και των ιστότοπων micro-blogging (π.χ. *Twitter*) και κυρίως λόγω των εγγενών δυσκολιών διάκρισής του μέσα στο κείμενο. Κατά συνέπεια, ένα ευρύ φάσμα ερευνητικών εργασιών που επικεντρώνεται στην ανίχνευση σαρκασμού και ειρωνείας σε δεδομένα κειμένου μπορεί να βρεθεί στη βιβλιογραφία. Οι έρευνες ποικίλλουν από τη συλλογή συνόλων δεδομένων έως τη δημιουργία συστημάτων ανίχνευσης. Ωστόσο, οι ερευνητές και οι γλωσσολόγοι δεν μπορούν ακόμη να συμφωνήσουν σε έναν συγκεκριμένο ορισμό του τι θεωρείται σαρκασμός.

2.2 Γλώσσες

Οι περισσότερες έρευνες για την ανίχνευση σαρκασμού υπάρχουν για τα Αγγλικά. Ωστόσο, έχουν γίνει έρευνες και για τις παρακάτω γλώσσες: Ιταλικά ([17], [18], [19]), Ολλανδικά [20], Ελληνικά ([21], [22]), Ινδονησιακά ([23], [24]), Αραβικά ([25], [26], [27], [28], [29]), Γαλλικά ([26], [30]), Ισπανικά ([31], [32], [33]), Πορτογαλικά [34], Γερμανικά [35], Βιετναμέζικα [36] και Χίντι [37]. Η ανίχνευση δίγλωσσου σαρκασμού με ένα μοντέλο ανά γλώσσα έχει επίσης διερευνηθεί, όπως για τα Αγγλικά-Τσεχικά [38] και τα Αγγλικά-Κινεζικά [39] αλλά όχι υπό μια διαγλωσσική οπτική.

Το 2020 προτάθηκε από τους [27] το πρώτο πολύγλωσσο (Γαλλικά, Αγγλικά και Αραβικά) και πολυπολιτισμικό (ινδοευρωπαϊκές γλώσσες έναντι λιγότερο πολιτιστικά κοντινών γλωσσών) σύστημα ανίχνευσης σαρκασμού / ειρωνείας.

2.3 Σύνολα Δεδομένων για την ανίχνευση σαρκασμού

Στην ενότητα αυτή παρουσιάζονται συνοπτικά τα σύνολα δεδομένων που έχουν χρησιμοποιηθεί από τους ερευνητές για το θέμα της ανίχνευσης του σαρκασμού σε κείμενα.

Τα κοινωνικά δίκτυα είναι πολύ δημοφιλή μέσα για τη συλλογή δεδομένων στις έρευνες για την ανίχνευση του σαρκασμού / ειρωνείας. Την τελευταία δεκαετία, η έκφραση ειρωνείας ευδοκιμεί στα κοινωνικά δίκτυα και ιδιαίτερα στο *Twitter*, όπου λόγω του περιορισμού των 140 χαρακτήρων στις ενημερώσεις κατάστασης είναι ιδανικό για την “συμπυκνωμένη” έκφραση των ιδεών των χρηστών. Ως δημόσιο μέσο κοινωνικής δικτύωσης, οι χρήστες συνειδητοποιούν ότι τα γραπτά τους μπορεί να διαβαστούν και να αναπαραχθούν από δυνητικά όλους, κερδίζοντας δημοτικότητα και οπαδούς. Όμως

αυτή η δημοσιότητα, σε αντίθεση με την πολιτική του *Facebook* για το πραγματικό όνομα, συχνά δεν έχει άμεσες συνέπειες στην καθημερινότητά τους, αφού η πλειοψηφία συμμετέχει ανώνυμα, χρησιμοποιώντας ένα avatar και ένα ψευδώνυμο. Αυτή η κατάσταση χωρίς λογοκρισία συμβάλλει στην ελευθερία έκφρασης προσωπικών σκέψεων για πολυδιάστατα, περίπλοκα, ταμπού, μη δημοφιλή ή αμφιλεγόμενα ζητήματα, μέρος των οποίων περιέχει σάτιρα. Το *Reddit* και το *Amazon* χρησιμοποιούνται επίσης σε πολλές έρευνες.

Δύο μέθοδοι χρησιμοποιούνται κυρίως για την επισήμανση κειμένων ως σαρκαστικά ή όχι στα παραπάνω μέσα κοινωνικής δικτύωσης: η εξ αποστάσεως επίβλεψη (distant supervision) και η επισήμανση από ειδικούς (manual labeling). Η εξ αποστάσεως επίβλεψη είναι μακράν η πιο κοινή μέθοδος. Τα κείμενα θεωρούνται θετικά παραδείγματα (σαρκαστικά) εάν πληρούν προκαθορισμένα κριτήρια, όπως να περιέχουν συγκεκριμένες ετικέτες (hashtags), όπως #irony, #sarcasm, #not, #humor, #satire για δεδομένα *Twitter* ([40], [41], [38], [42], [20], [43], [44]) και /s για δεδομένα *Reddit* ([45], [23]) ή όταν αναρτώνται από συγκεκριμένους λογαριασμούς μέσων κοινωνικής δικτύωσης ([46], [35], [31], [30] (FreSaDa dataset)). Τα αρνητικά παραδείγματα είναι συνήθως τυχαίες αναρτήσεις που δεν ταιριάζουν με τα κριτήρια. Το κύριο πλεονέκτημα της εξ αποστάσεως επίβλεψης είναι ότι επιτρέπει τη δημιουργία μεγάλων συνόλων δεδομένων με ετικέτα χωρίς ιδιαίτερη προσπάθεια. Ωστόσο, οι ετικέτες που παράγονται μπορεί να είναι πολύ θορυβώδεις ή να περιέχουν λάθη ([47], [48]).

Μια εναλλακτική λύση στην εξ αποστάσεως επίβλεψη είναι η επισήμανση από σχολιαστές. Η Filatova [49] ζητά από τους σχολιαστές να βρουν ζευγάρια κριτικών του *Amazon* όπου η μία είναι σαρκαστική και η άλλη όχι. Οι Abercrombie και Hovy [25] χρησιμοποιούν ανθρώπους σχολιαστές για να επισημάνουν αναρτήσεις στο *Twitter*, αντίστοιχα. Οι Riloff κ.ά. [50] χρησιμοποιούν μια υβριδική προσέγγιση, όπου συλλέγουν tweets με ή χωρίς ετικέτα σαρκασμού και αφού αυτή αφαιρείται, αυτά παρουσιάζονται σε μια ομάδα σχολιαστών για τελική επισήμανση (Riloff dataset). Μια παρόμοια προσέγγιση χρησιμοποιείται από τους Van Hee κ.ά. [51] που παρουσίασαν το σύνολο δεδομένων τους ως μέρος μιας κοινής εργασίας SemEval για την ανίχνευση του σαρκασμού (SemEval-2018). Οι Oprea και Magdy [48] ζήτησαν από τους χρήστες του *Twitter* να παρέχουν τόσο σαρκαστικά όσο και μη σαρκαστικά tweets που είχαν δημοσιεύσει στο παρελθόν (iSarcasm dataset). Οι ετικέτες προσδιορίστηκαν από τους ίδιους τους συγγραφείς.

Άλλα σύνολα δεδομένων όπου έγινε επισήμανση από σχολιαστές και σε άλλες γλώσσες είναι το GRGE dataset από τους Tsakalidis κ.ά. [22], το ArSarcasm dataset από τους Abu Farha κ.ά. [29] κ.λπ.

2.4 Μέθοδοι ανίχνευσης σαρκασμού

Οι Sarsam κ.ά. [52] και Yaghoobian κ.ά. [53] έκαναν βιβλιογραφικές ανασκοπήσεις για την ανίχνευση του σαρκασμού σε κείμενα. Η πρώτη ομάδα συγκέντρωσε 31 έρευνες με θέμα την ανίχνευση σαρκασμού με χρήση αλγορίθμων μηχανικής μάθησης. Η δεύτερη ομάδα συγκέντρωσε έρευνες και τις χώρισε σε 3 βασικές κατηγορίες προσέγγισης του προβλήματος. 1) ημιεποπτευόμενη εξαγωγή μοτίβων για τον εντοπισμό υπονοούμενων συναισθημάτων, 2) χρήση εποπτείας βασισμένη σε ετικέτες και 3) ενσωμάτωση γενικού πλαισίου (context) πέρα από το κείμενο-στόχο.

Σε αυτήν την ενότητα, με βάση την εργασία των Yaghoobian κ.ά. [53], παρουσιάζονται συνοπτικά οι μέθοδοι ανίχνευσης σαρκασμού που βρέθηκαν στη βιβλιογραφία.

2.4.1 Μέθοδοι βασισμένες στο περιεχόμενο (Content-based methods)

Μέθοδοι βασισμένες σε κανόνες (Rule-Based Methods)

Οι Veale και Hao [54] αναζήτησαν σαρκαστικές παρομοιώσεις οι οποίες ακολουθούν να συγκεκριμένο μοτίβο «*as * as **» (π.χ. «*as private as a park-bench*») σε αναζητήσεις στο Google και χρησιμοποιώντας μια προσέγγιση εννέα βημάτων προκύπτει ότι το 18% των παρομοιώσεων είναι ειρωνικές / σαρκαστικές.

Οι Bharti κ.ά. [55] χρησιμοποίησαν έναν συνδυασμό δύο προσεγγίσεων στη μελέτη τους πάνω στον σαρκασμό. Προτείνουν έναν αλγόριθμο που αναζητά tweets που είναι συναισθηματικά φορτισμένα και προσδιορίζει τον σαρκασμό με τις μορφές μιας αντίφασης αρνητικού (ή θετικού) συναισθήματος και θετικής (ή αρνητικής) κατάστασης. Ψάχνουν επίσης για τη συνύπαρξη υπερβολικών λέξεων όπως «*wow*», «*yay*» κ.λπ. στην αρχή των tweets, και άλλων λέξεων όπως «*absolutely*», «*huge*», π.χ., «*Wow, that's a huge discount, I'm not buying anything!!* #sarcasm.»

Ομοίως, οι Riloff κ.ά. [50] εντόπισαν μια θετική / αρνητική αντίθεση μεταξύ ενός συναισθήματος και μιας κατάστασης, η οποία υποδηλώνει ένδειξη σαρκασμού, π.χ., «*I'm so pleased mom woke me up with vacuuming my room this morning. :)*». Ομοίως, οι Van Hee κ.ά. [56] εικάζουν ότι η ασυμφωνία συναισθημάτων μέσα σε μια δήλωση (utterance) υποδηλώνει σαρκασμό. Για το σκοπό αυτό, συγκεντρώνουν όλες τις έννοιες (concepts) του πραγματικού κόσμου που συνοδεύονται από ένα συναίσθημα και τις χαρακτηρίζουν είτε με "θετικό" ή "αρνητικό" συναίσθημα. Για παράδειγμα, το «*going to the dentist*» συνδέεται συχνά με αρνητικό συναίσθημα. Αν και το μοντέλο τους δεν καταφέρνει μεγάλη ακρίβεια, υπογραμμίζουν τη δυσκολία και τη σημασία της ενσωμάτωσης της ανίχνευσης σαρκασμού σε ταξινομητές συναισθημάτων. Θεωρούν τις προσπάθειές τους ως επέκταση της θεμελιώδους δουλειάς των Greene και Resnik [57] οι οποίοι χρησιμοποιήσαν μια έννοια που ονομάζεται συντακτική συσκευασία (syntactic packaging) για να καταδείξουν την επιρροή των συντακτικών επιλογών στο αντιληπτό υπονοούμενο συναίσθημα των τίτλων ειδήσεων.

Ένα από τα πρώτα έργα είναι των Tepperman κ.ά. [58] που προσδιορίζει τον σαρκασμό στους προφορικούς διαλόγους και βασίζεται σε μεγάλο βαθμό σε ενδείξεις (cues) όπως το γέλιο, οι παύσεις και το φύλο του ομιλητή. Τα δεδομένα τους περιορίζονται σε σαρκαστικές εκφράσεις που περιέχουν την έκφραση «*yeah-right*». Οι Carvalho κ.ά. [59] βελτιώνουν την ακρίβεια του μοντέλου σαρκασμού τους χρησιμοποιώντας προφορικές ή χειρονομιακές ενδείξεις στα σχόλια των χρηστών, όπως emoticons, ονοματοποιητικές εκφράσεις (π.χ. *achoo, haha, grr, ahem*) για γέλιο, βαριά σημεία στίξης, εισαγωγικά και θετικές παρεμβολές. Οι Davidov κ.ά. [40], οι Tsur κ.ά. [60] χρησιμοποιούν συντακτικά και βασισμένα σε πρότυπα γλωσσικά χαρακτηριστικά για να κατασκευάσουν τα διανύσματα χαρακτηριστικών (feature vectors). Οι Barbieri κ.ά. [46] ακολουθούν μια παρόμοια προσέγγιση και επεκτείνουν την προηγούμενη εργασία τους βασιζόμενοι στην εσωτερική δομή των δηλώσεων όπως το απροσδόκητο, η ένταση των όρων ή η ανισορροπία μεταξύ των συναισθημάτων.

Σύνολα Χαρακτηριστικών (Feature sets)

Σε αυτήν την ενότητα, εξετάζουμε τα κύρια χαρακτηριστικά του κειμένου που χρησιμοποιούνται αποτελεσματικά για την ανίχνευση του σαρκασμού. Οι περισσότερες μελέτες χρησιμοποιούν σε κάποιο βαθμό το μοντέλο bag-of-words. Ωστόσο, εκτός από αυτό, έχει αναφερθεί η χρήση πολλών άλλων συνόλων χαρακτηριστικών. Συζητάμε χαρακτηριστικά συμφραζομένων (δηλαδή, χαρακτηριστικά που εξαρτώνται από την

κωδικοποίηση των πληροφοριών που παρουσιάζονται πέρα από το κείμενο) στην Ενότητα 3.

Οι Reyes κ.ά. [61] εισάγουν ένα σύνολο χαρακτηριστικών (feature set) τα οποία εξαρτώνται από το χιούμορ ή την ειρωνεία και σχετίζονται με την ασάφεια, το απροσδόκητο και κάποιο συναισθηματικό σενάριο. Τα χαρακτηριστικά ασάφειας καλύπτουν τη δομική, μορφοσυντακτική, σημασιολογική ασάφεια, ενώ τα χαρακτηριστικά του απροσδόκητου μετρούν τη σημασιολογική συνάφεια. Οι Riloff κ.ά. [50], εκτός από έναν ταξινομητή βασισμένο σε κανόνες, χρησιμοποίησαν και ένα σύνολο μοτίβων, συγκεκριμένα θετικά ρήματα και φράσεις αρνητικής κατάστασης, ως χαρακτηριστικά. Οι Liebrecht κ.ά. [20] χρησιμοποίησαν διγράμματα (bi-grams) και τριγράμματα (tri-grams) και ομοίως, οι Reyes κ.ά. [62] χρησιμοποίησαν skip-grams και χαρακτηριστικά σε επίπεδο χαρακτήρων (character-level features). Οι Barbieri κ.ά. [18] συμπεριέλαβαν επτά σύνολα χαρακτηριστικών όπως παραδείγματος χάριν, τη μέγιστη / ελάχιστη ένταση επιθέτων και επιρρημάτων (maximum / minimum of intensity of adjectives and adverbs), το κενό έντασης επιθέτων και επιρρημάτων (gap of intensity of adjectives and adverbs), τον μέγιστο / ελάχιστο / μέσο αριθμό συνωνύμων για τις λέξεις (max / min / average number of synonyms) ενός κειμένου-στόχου και ούτω καθεξής. Οι Buschmeier κ.ά. [63] ενσωμάτωσαν έλλειψη, υπερβολή και ανισορροπία στο σύνολο των χαρακτηριστικών τους. Οι Joshi κ.ά. [42] χρησιμοποίησαν χαρακτηριστικά που αντιστοιχούν στη γλωσσική θεωρία της δυσαρμονίας (linguistic theory of incongruity). Τα χαρακτηριστικά ταξινομήθηκαν σε δύο σύνολα: χαρακτηριστικά που βασίζονται στην υπονοούμενη και στην ρητή δυσαρμονία (implicit and explicit incongruity-based features).

Οι Mishra κ.ά. [64] πρότειναν μια νέα προσέγγιση για τη διερεύνηση των κύριων χαρακτηριστικών του σαρκασμού στο κείμενο. Σχεδίασαν ένα σύνολο χαρακτηριστικών που βασίζονται στο βλέμμα των αναγνωστών, όπως η μέση διάρκεια σταθεροποίησης του βλέμματος στο κείμενο, ο αριθμός παλινδρόμησης, ο αριθμός παράβλεψης κ.λπ., με βάση σχολιασμούς από τα πειράματά τους όπου παρακολουθούνταν τα μάτια αναγνωστών. Επιπλέον, χρησιμοποίησαν σύνθετα χαρακτηριστικά βλέμματος όπως παραδείγματος χάριν, το γρήγορο άλμα του βλέμματος μεταξύ δύο θέσεων ανάπαυσης.

2.4.2 Μέθοδοι βασισμένες στη μάθηση (Learning-based methods)

Παρακάτω συγκεντρώνονται και περιγράφονται μέθοδοι μηχανικής μάθησης τις οποίες χωρίζουμε στις παρακάτω κατηγορίες: εποπτευόμενη μάθηση (supervised learning), ημι-εποπτευόμενη μάθηση(semi-supervised learning), μη εποπτευόμενη (unsupervised learning).

Εποπτευόμενη μάθηση (Supervised learning)

Στις παραδοσιακές προσεγγίσεις μηχανικής μάθησης, οι περισσότερες εργασίες για τη στατιστική ανίχνευση του σαρκασμού έχουν βασιστεί σε διάφορες συνδυαστικές μορφές τεχνικών όπως το Τυχαίο Δάσος (Random Forest), Μηχανές Διανυσματικής Υποστήριξης (Support Vector Machines), τα Δέντρα Αποφάσεων (Decision Trees), ο απλοϊκός ταξινομητής Bayes (Naive Bayes) και τα Νευρωνικά Δίκτυα (Neural Networks) ([40], [42], [58], [60], [65], [66], [67]). Για παράδειγμα, οι González-Ibáñez κ.ά. [68] χρησιμοποίησαν SVM με Διαδοχική Ελάχιστη Βελτιστοποίηση (Sequential Minimal Optimization) και Λογιστική Παλινδρόμηση (Logistic Regression), τα οποία συνήθως χρησιμοποιούνται για την ανάλυση συναισθήματος, για τον εντοπισμό διακριτικών χαρακτηριστικών. Οι Riloff κ.ά. [50] χρησιμοποίησαν ένα υβριδικό σύστημα SVM που ξεπέρασε τις επιδόσεις του απλού ταξινομητή SVM. Ομοίως, οι Liebrecht κ.ά., [20]

έκαναν χρήση Ισορροπημένων Αλγορίθμων Winnow. Οι Reyes κ.ά., [62] έκαναν χρήση Naive Bayes και Decision Trees για πολλαπλά ζεύγη ετικετών μεταξύ ειρωνείας, χιούμορ, πολιτικής και εκπαίδευσης και για ασαφή ομαδοποίηση ανίχνευσης σαρκασμού (Mukherjee και Bala, [69]). Οι Bamman και Smith [70] παρουσιάζουν τη χρήση της δυαδικής Λογιστικής Παλινδρόμησης για την ταξινόμηση των tweets σε σαρκαστικά ή μη. Ομοίως, οι Joshi κ.ά. [42] αναφέρουν ότι οι αλγόριθμοι επισήμανσης ακολουθιών είναι πιο χρήσιμοι για δεδομένα συνομιλίας σε αντίθεση με τις μεθόδους ταξινόμησης. Χρησιμοποιούν SVM-HMM και SEARN ως αλγόριθμους επισήμανσης ακολουθιών. Οι Liu κ.ά. [71] παρουσιάζουν μια προσέγγιση εκμάθησης συνόλου πολλαπλών στρατηγικών (MSELA) που περιλαμβάνει Bagging, Boosting, κ.λπ., για να χειριστούν την ανισορροπία μεταξύ σαρκαστικών και μη σαρκαστικών δειγμάτων.

Ενώ οι προσεγγίσεις που βασίζονται σε κανόνες στηρίζονται κυρίως σε λεξιλογικές πληροφορίες και δεν απαιτούν εκπαίδευση, η μηχανική μάθηση χρησιμοποιεί πάντα δεδομένα εκπαίδευσης και εκμεταλλεύεται διαφορετικούς τύπους πηγών (ή χαρακτηριστικών) πληροφοριών, όπως bag-of-words αναπαραστάσεις, συντακτικά μοτίβα, πληροφορίες συναισθήματος ή σημασιολογική συνάφεια. Οι πρώτες προσπάθειες προς αυτήν την κατεύθυνση χρησιμοποιούν ομοιότητα μεταξύ διανυσμάτων λέξεων ως χαρακτηριστικό για την ανίχνευση σαρκασμού. Οι Ghosh και Veale [72] χρησιμοποιούν έναν συνδυασμό Συνελικτικών Νευρωνικών δικτύων, LSTM που ακολουθείται από ένα DNN. Οι Van Hee κ.ά. [51] προτείνουν ένα μοντέλο που προσδιορίζει τα σαρκαστικά tweets και στη συνέχεια διαφοροποιεί τον τύπο (από τέσσερις κατηγορίες) του εκφρασμένου σαρκασμού. Τα συστήματα που υποβλήθηκαν και για τις δύο υπο-εργασίες αντιπροσωπεύουν μια ποικιλία προσεγγίσεων που βασίζονται σε νευρωνικά δίκτυα (δηλαδή, CNN, RNN και (bi-)LSTMs) που εκμεταλλεύονται ενσωματώσεις λέξεων και χαρακτήρων καθώς και χαρακτηριστικά που ενσωματώνονται χειροκίνητα.

Ημι-εποπτευόμενη μάθηση (Semi-supervised learning)

Αυτή η μορφή μηχανικής μάθησης, η οποία εμπίπτει μεταξύ της εποπτευόμενης και μη μάθησης, χρησιμοποιεί μια ελάχιστη ποσότητα δεδομένων με ετικέτα και μια μεγάλη ποσότητα δεδομένων χωρίς ετικέτα κατά τη διάρκεια της εκπαίδευσης [60]. Η παρουσία των μη επισημασμένων συνόλων δεδομένων και η ανοιχτή πρόσβαση στα μη επισημασμένα σύνολα δεδομένων είναι το χαρακτηριστικό που διαφοροποιεί την ημι-εποπτευόμενη από την εποπτευόμενη μάθηση. Οι Davidov κ.ά. [40] χρησιμοποιούν μια ημι-εποπτευόμενη προσέγγιση μάθησης για την αυτόματη αναγνώριση σαρκασμού χρησιμοποιώντας δύο διαφορετικές μορφές κειμένου, tweets από το *Twitter* και κριτικές προϊόντων από την *Amazon*. Στη μελέτη τους συλλέγεται ένας συνολικός αριθμός 66.000 προϊόντων και κριτικών βιβλίων και εξάγονται τόσο συντακτικά όσο και βασισμένα σε μοτίβα χαρακτηριστικά. Η πολικότητα συναισθήματος από 1 έως 5 επιλέγεται στη φάση της εκπαίδευσης για όλα τα δεδομένα εκπαίδευσης. Οι συγγραφείς αναφέρουν απόδοση 77% ακρίβειας (precision).

Μη εποπτευόμενη μάθηση (Unsupervised learning)

Η μη εποπτευόμενη μάθηση στον αυτόματο εντοπισμό του σαρκασμού είναι ακόμη σε αρχικό στάδιο και οι περισσότερες προσεγγίσεις, οι οποίες ισχύουν κυρίως για την αναγνώριση προτύπων, βασίζονται στην ομαδοποίηση (clustering). Παρακινούμενοι από τους περιορισμούς και τις δυσκολίες που ενυπάρχουν στην επισήμανση των συνόλων δεδομένων (δηλαδή, το γεγονός ότι αποτελεί χρονοβόρα και απαιτητική διεργασία) στις εποπτευόμενες μεθόδους μάθησης, οι ερευνητές επιδιώκουν να

εξαλείψουν τέτοιες προσπάθειες εστιάζοντας στην ανάπτυξη μοντέλων χωρίς επίβλεψη. Οι Nozza κ.ά. [73] προτείνουν ένα πλαίσιο χωρίς επίβλεψη για ανεξάρτητη από τον τομέα ανίχνευση ειρωνείας. Βασίζονται σε πιθανολογικά θεματικά μοντέλα που αρχικά ορίστηκαν για ανάλυση συναισθήματος. Αυτά τα μοντέλα είναι προεκτάσεις του γνωστού μοντέλου Latent Dirichlet Allocation (LDA) (Blei κ.ά., [74]). Προτείνουν το μοντέλο θέματος-ειρωνείας (TIM), το οποίο είναι σε θέση να μοντελοποιήσει την ειρωνεία προς διαφορετικά θέματα σε ένα πλήρως μη εποπτευόμενο περιβάλλον, επιτρέποντας σε κάθε λέξη σε μια πρόταση να δημιουργηθεί από την ίδια διανομή θέματος ειρωνείας. Εμπλουτίζουν το μοντέλο τους με ένα λεξικό νευρωνικής γλώσσας που προέρχεται από ενσωματώσεις λέξεων. Σε μια παρόμοια προσπάθεια, οι Mukherjee και Bala [69] χρησιμοποιούν τόσο εποπτευόμενες όσο και μη εποπτευόμενες ρυθμίσεις. Χρησιμοποιούν Naive Bayes για ομαδοποίηση εποπτευόμενης και Fuzzy C-means (FCM) για μάθηση χωρίς επίβλεψη. Δικαιολογημένα, η FCM δεν λειτουργεί τόσο αποτελεσματικά όσο η NB.

2.4.3 Μοντέλα βασισμένα στο συγκείμενο πλαίσιο (Context-based models)

Η κατανόηση των σαρκαστικών εκφράσεων εξαρτάται σε μεγάλο βαθμό από το υπόβαθρο γνώσης και τις εξαρτήσεις από τα συμφραζόμενα που είναι τυπικά ποικίλες. Για παράδειγμα, μια σαρκαστική ανάρτηση από το *Reddit*, «*I'm sure Hillary would've done that, lmao.*» απαιτεί προηγούμενη γνώση για το συμβάν, δηλαδή εξοικείωση με τη συνήθη συμπεριφορά της Χίλαρι Κλίντον τη στιγμή που έγινε η ανάρτηση. Ομοίως, σαρκαστικές αναρτήσεις όπως «*But atheism, yeah *that's* a religion*!» απαιτούν βασικές γνώσεις, ακριβώς λόγω της φύσης θεμάτων όπως ο αθεϊσμός που συχνά υπόκειται σε εκτενή επιχειρηματολογία και είναι πιθανό να προκαλέσει σαρκαστική ερμηνεία. Τα προτεινόμενα μοντέλα σε αυτήν την ενότητα χρησιμοποιούν πληροφορίες περιεχομένου και συμφραζομένων (contextual) που απαιτούνται για την ανίχνευση σαρκασμού. Επιπλέον, υπάρχει αυξανόμενο ενδιαφέρον για τη χρήση νευρωνικών μοντέλων γλώσσας για προεκπαίδευση για διάφορες εργασίες στην επεξεργασία φυσικής γλώσσας.

Οι Wallace κ.ά. [75] ισχυρίζονται ότι οι άνθρωποι που βάζουν ετικέτες βασίζονται σταθερά σε πληροφορίες συμφραζομένων για να αποφασίσουν σχετικά με την σαρκαστική πρόθεση ή μη του συγγραφέα. Αντίστοιχα, πρόσφατες μελέτες προσπαθούν να αξιοποιήσουν διάφορες μορφές πληροφοριών με βάση τα συμφραζόμενα ανεξάρτητα από το κείμενο, για τον πιο αποτελεσματικό προσδιορισμό του σαρκασμού. Διαισθητικά, στην περίπτωση των κριτικών προϊόντων της *Amazon*, το να γνωρίζουμε το είδος των βιβλίων που συνήθως αρέσουν σε ένα άτομο μπορεί να μας βοηθήσει να κρίνουμε: κάποιος που διαβάζει και αξιολογεί κυρίως τον Ντοστογιέφσκι είναι στατιστικά ειρωνικός αν γράφει μια εγκωμιαστική κριτική για το *Twilight*. Προφανώς, πολλοί άνθρωποι απολαμβάνουν πραγματικά να διαβάζουν το *Twilight*, και έτσι, αν η κριτική είναι γραμμένη με διακριτικότητα, πιθανότατα θα είναι δύσκολο να διακρίνει κανείς την πρόθεση του συγγραφέα χωρίς αυτό το υπόβαθρο του συγγραφέα. Ως εκ τούτου, οι Mukherjee και Bala [69] αναφέρουν ότι η συμπερίληψη στοιχείων ανεξάρτητα από το κείμενο οδηγεί στη βελτίωση της απόδοσης των μοντέλων σαρκασμού. Για το σκοπό αυτό, οι μελέτες λαμβάνουν τρεις μορφές πλαισίου ως χαρακτηριστικό: 1) πλαίσιο συγγραφέα (Hazarika κ.ά. [76], Bamman και Smith [70]), 2) πλαίσιο συνομιλίας (Wang κ.ά. [77]) και 3) τοπικό πλαίσιο (Ghosh και Veale [43]). Μια άλλη δημοφιλής γραμμή έρευνας χρησιμοποιεί διάφορες τεχνικές ενσωμάτωσης χρηστών που κωδικοποιούν τα χαρακτηριστικά της προσωπικότητας των χρηστών για να βελτιώσουν τα μοντέλα ανίχνευσης σαρκασμού (Hazarika κ.ά. [76]) με το μοντέλο τους, CASCADE. Όταν χρησιμοποιείται μαζί με εξαγωγείς χαρακτηριστικών που

βασίζονται σε περιεχόμενο, όπως τα Συνελικτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks), επιτυγχάνεται μια σημαντική βελτίωση στην απόδοση ταξινόμησης σε ένα σύνολο δεδομένων από το *Reddit*.

Οι Agrawal κ.ά. [78] διατυπώνουν το έργο της ανίχνευσης σαρκασμού ως ένα πρόβλημα ταξινόμησης ακολουθίας αξιοποιώντας τις φυσικές αλλαγές σε διάφορα συναισθήματα κατά τη διάρκεια ενός κειμένου. Οι Li κ.ά. [79] προτείνουν μια ημι-εποπτευόμενη μέθοδο για την ανίχνευση σαρκασμού με βάση τα συμφραζόμενα σε διαδικτυακά φόρουμ συζήτησης. Υιοθετούν την προηγούμενη προτίμηση του συγγραφέα και του θέματος για σαρκασμό ως διανύσματα πλαισίου (conteDevlinxt embedding) που παρέχουν μια απλή αλλά αντιπροσωπευτική γνώση του υποβάθρου. Οι Nimala κ.ά. [80] προτείνουν επίσης ένα μη εποπτευόμενο πιθανοτικό σχεσιακό μοντέλο για τον εντοπισμό κοινών θεμάτων σαρκασμού με βάση την κατανομή συναισθημάτων των λέξεων στα tweets.

Ανίχνευση σαρκασμού με χρήση προ-εκπαιδευμένων γλωσσικών μοντέλων

Δεδομένης της τονισμένης σημασίας του πλαισίου ή των συμφραζομένων για τον εντοπισμό μεταφορικών γλωσσικών φαινομένων και της δυσκολίας για την δημιουργία ετικετών στα δεδομένα, οι προσεγγίσεις μεταφοράς μάθησης (transfer learning) κερδίζουν ολοένα και μεγαλύτερη προσοχή. Συγκεκριμένα, η χρήση προ-εκπαιδευμένων διανυσματικών αναπαραστάσεων λέξεων (pre-trained word embeddings) όπως το Global Vectors (GloVe) (Pennington κ.ά. [81]) και το ELMo (Peters κ.ά. [82]) ή η αξιοποίηση μεθόδων Transformer seq2seq όπως το BERT (Bidirectional Encoder Representations from Transformers) (Devlin κ.ά [83]), RoBERTa (Liu κ.ά. [84]) και XLNet (Yang κ.ά. [85]) κ.λπ. παρουσιάζουν άνοδο. Οι Potamias κ.ά. [86] προτείνουν το μοντέλο Recurrent CNN RoBERTA (RCNN-RoBERTa), μία υβριδική αρχιτεκτονική που συνδυάζει την αρχιτεκτονική RoBERTA, η οποία και ενισχύεται περαιτέρω με τη χρήση και ενός Επαναλαμβανόμενου Συνελικτικού Νευρωνικού Δικτύου (Recurrent Convolutional Neural Network). Αναφέρουν μια απόδοση με ακρίβεια 79% στο σύνολο δεδομένων SARC (Khodak κ.ά. [45]). Ομοίως, οι Dadu και Pant [87] χρησιμοποιούν ένα σύνολο RoBERTa και ALBERT (Lan κ.ά. [88]) στο σύνολο δεδομένων #GetitOffMyChest (Jaidka κ.ά. [89]) επιτυγχάνουν απόδοση με ακρίβεια 85% με F1 Score: 0.55. Οι Javdan κ.ά. [90] χρησιμοποιούν το BERT μαζί με ανάλυση συναισθήματος που βασίζεται σε πτυχές (aspect-based) για να εξάγουν τη σχέση μεταξύ του διαλόγου και απάντησης επί του κειμένου. Λαμβάνουν F1 Score: 0,73 σε σύνολο δεδομένων από *Twitter* και 0,73 σε σχέση με σύνολο δεδομένων από το *Reddit*.

3. ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ

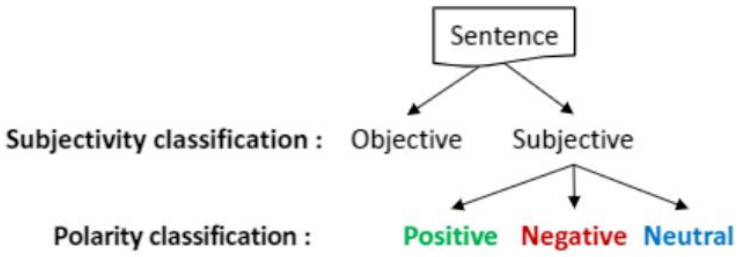
3.1 Ανάλυση συναισθήματος

Η ανάλυση συναισθημάτων είναι ένα ευρύ και πολύπλοκο πεδίο έρευνας. Στη συνέχεια του κεφαλαίου αυτού, αναλύονται τα κύρια χαρακτηριστικά που συνιστούν τον τομέα της ανάλυσης συναισθημάτων.

3.1.1 Χαρακτηριστικά Ανάλυσης Συναισθήματος

Κατηγοριοποίηση Συναισθήματος

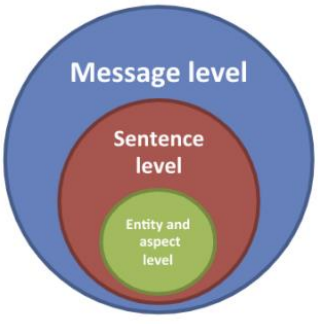
Ο πρώτος στόχος όσον αφορά στην ανάλυση συναισθημάτων συνίσταται συνήθως στη διάκριση μεταξύ υποκειμενικών και αντικειμενικών προτάσεων. Εάν μια δεδομένη πρόταση ταξινομείται ως αντικειμενική, δεν απαιτούνται άλλες θεμελιώδεις διεργασίες, ενώ εάν η πρόταση ταξινομείται ως υποκειμενική, πρέπει να εκτιμηθεί η πολικότητά της, Αυτή μπορεί να είναι θετική, αρνητική ή ουδέτερη (όπως φαίνεται και στην εικόνα 1). Η ταξινόμηση υποκειμενικότητας είναι η εργασία που διακρίνει προτάσεις που εκφράζουν αντικειμενικές (ή πραγματικές) πληροφορίες και ονομάζονται αντικειμενικές προτάσεις από προτάσεις που εκφράζουν υποκειμενικές απόψεις, που αποτελούν τις υποκειμενικές προτάσεις.



Εικόνα 1: Ροή διεργασιών ανάλυσης συναισθημάτων

Επίπεδα Ανάλυσης

Ο στόχος της ανάλυσης συναισθημάτων είναι να «ορίσει αυτόματα εργαλεία ικανά να εξάγουν υποκειμενικές πληροφορίες από κείμενα στη φυσική γλώσσα». Η πρώτη επιλογή όταν κάποιος εφαρμόζει ανάλυση συναισθημάτων είναι να καθορίσει τι σημαίνει το κείμενο (δηλαδή, το αντικείμενο που αναλύθηκε) στην περίπτωση της μελέτης που εξετάζεται. Σε γενικές γραμμές, η ανάλυση συναισθημάτων στα κοινωνικά δίκτυα μπορεί να διερευνηθεί κυρίως σε τρία επίπεδα (παρουσιάζεται γραφικά στο ακόλουθο σχήμα):



Εικόνα 2: Επίπεδα Ανάλυσης

- **Επίπεδο μηνύματος**: Στόχος είναι να ταξινομηθεί η πολικότητα ενός ολόκληρου μηνύματος. Για παράδειγμα, με δεδομένη μια κριτική ενός προϊόντος, το σύστημα καθορίζει εάν το μήνυμα εξ ολοκλήρου εκφράζει μια γενικά θετική, αρνητική ή ουδέτερη γνώμη για το προϊόν. Η υπόθεση είναι ότι ολόκληρο το μήνυμα εκφράζει μόνο μία γνώμη για μία οντότητα (π.χ. ένα μεμονωμένο προϊόν).
- **Επίπεδο προτάσεων**: Στόχος είναι να προσδιοριστεί η πολικότητα κάθε πρότασης που περιέχεται σε ένα μήνυμα κειμένου. Η υπόθεση είναι ότι κάθε πρόταση, σε ένα δεδομένο μήνυμα, υποδηλώνει μια ενιαία γνώμη για μία οντότητα.
- **Επίπεδο οντότητας και άποψης**: Εκτελεί μια πιο λεπτομερή ανάλυση από το επίπεδο μηνυμάτων και προτάσεων. Βασίζεται στην ιδέα ότι μια γνώμη αποτελείται από ένα συναίσθημα και έναν στόχο (της γνώμης). Παράδειγμα μίας τέτοιας περίπτωσης είναι η ακόλουθη κριτική: «*Αυτό το κινητό είναι πολύ καλό, αλλά πρέπει να βελτιωθούν κάποια θέματα όπως η ασφάλεια και η διάρκεια της μπαταρίας*». Στο παράδειγμα αξιολογούνται τρεις πτυχές: το κινητό (θετικό), η διάρκεια ζωής της μπαταρίας (αρνητικό) και η ασφάλεια (αρνητικό).

Ο ρόλος των σαφών και ασαφών απόψεων

Μεταξύ των διαφορετικών αποχρώσεων που μπορεί να υποτεθούν για μια γνώμη, πρέπει να διακρίνουμε τις σαφείς και ασαφείς απόψεις:

- Σαφείς απόψεις: Μια σαφής γνώμη είναι μια υποκειμενική δήλωση που δείχνει απευθείας τη γνώμη (πχ. «*Η φωτεινότητα της οθόνης του κινητού είναι φοβερή*»).
- Ασαφείς απόψεις: Μια ασαφής γνώμη είναι μια αντικειμενική δήλωση που συνήθως εκφράζει ένα επιθυμητό ή ανεπιθύμητο γεγονός. Παράδειγμα: «*Η πρώτη ταινία που είδα ήταν περισσότερο βίαιη από τη δεύτερη*». Με αυτή την έκφραση δεν είναι φανερή η γνώμη του χρήστη απέναντι στην πρώτη ταινία. Για μερικούς ανθρώπους, η βία σε πολεμικές ταινίες θα μπορούσε να είναι ένα καλό χαρακτηριστικό που καθιστά την ταινία πιο ρεαλιστική, ενώ θα μπορούσε να είναι ένα αρνητικό χαρακτηριστικό για άλλους.

Ο ρόλος της σημασιολογίας των λέξεων

Η σημασιολογία της γλώσσας που χρησιμοποιείται στα κοινωνικά δίκτυα είναι θεμελιώδης για την ακριβή ανάλυση των εκφράσεων των χρηστών. Το περιεχόμενο μιας έκφρασης κειμένου είναι ένα κρίσιμο στοιχείο που πρέπει να ληφθεί υπόψη για την κατάλληλη αντιμετώπιση του υποκείμενου συναισθήματος. Μια πρόταση «που λαμβάνεται ως έχει» μπορεί να φαίνεται αρνητική ή θετική, αλλά αν αναλυθεί σωστά από σημασιολογική άποψη μπορεί να είναι εντελώς διαφορετική. Για παράδειγμα, η φράση «*Είδα την πιο καταπληκτική ταινία τρόμου. Ήταν σαν πραγματικός εφιάλτης! ΠΑΝΙΚΌΣ!*» μπορεί αρχικά να ερμηνευτεί ως αρνητική, αλλά λαμβάνοντας υπόψιν που συνήθως εκφράζονται τέτοιου είδους απόψεις, όπως μια κοινότητα χρηστών που προτιμούν ταινίες τρόμου, καθώς επίσης και ορισμένα λεκτικά στοιχεία που είναι χαρακτηριστικά της γλώσσας του κοινωνικού δικτύου, μπορούμε να αντλήσουμε μία (πραγματική) θετική κρίση. Ως εκ τούτου, καθίσταται αναγκαία η βαθιά κατανόηση της σημασιολογίας της φυσικής γλώσσας που χρησιμοποιείται στα κοινωνικά δίκτυα.

Ο ρόλος των σχημάτων του λόγου

Τα σχήματα λόγου θεωρούνται οποιαδήποτε απόκλιση από τον συνηθισμένο τρόπο ομιλίας ή γραφής. Παράδειγμα των πιο “προβληματικών” σχημάτων λόγου είναι η ειρωνεία και ο σαρκασμός. Η ειρωνεία χρησιμοποιείται συχνά για να τονίσει περιστατικά που αποκλίνουν από τα αναμενόμενα, όπως ανατροπές της μοίρας, ενώ ο σαρκασμός χρησιμοποιείται συνήθως για να μεταφέρει έμμεση κριτική με ένα συγκεκριμένο θύμα ως στόχο.

Ένα εγγενές χαρακτηριστικό των σαρκαστικών και ειρωνικών λεκτικών πράξεων είναι ότι μερικές φορές είναι δύσκολο να αναγνωριστούν, αρχικά από τον άνθρωπο και ύστερα από τις μηχανές. Η δυσκολία στην αναγνώριση του σαρκασμού και της ειρωνείας προκαλεί παρεξηγήσεις στην καθημερινή επικοινωνία και δημιουργεί προβλήματα σε πολλά συστήματα επεξεργασίας φυσικής γλώσσας. Στο πλαίσιο της ανάλυσης συναισθημάτων (όπου ο σαρκασμός και η ειρωνεία που θεωρούνται συνήθως συνώνυμα) όταν μια σαρκαστική / ειρωνική πρόταση ανιχνεύεται ως θετική, πιθανότατα είναι αρνητική, καθώς και το αντίστροφο.

3.1.2 Ανάλυση Συναισθήματος με χρήση τεχνικών Μηχανικής Μάθησης

Πριν από την αύξηση του ενδιαφέροντος γύρω από τις τεχνικές βαθιάς μάθησης, οι ερευνητές είχαν εστιάσει την προσοχή τους σε ένα σύνολο αλγορίθμων μηχανικής μάθησης που περιλάμβαναν πιθανοτικούς ταξινομητές όπως οι: Naive Bayes (NB), Bayesian network (BN), maximum entropy (MaxEnt), multinomial Naive Bayes (MNB), conditional random field (CRF), γραμμικούς ταξινομητές όπως είναι το logistic regression (LR), τα Support Vector Machines (SVM) και τα δέντρα αποφάσεων όπως ο Random Forest (RF).

Οι αλγόριθμοι αυτοί εκπαιδεύονται συνήθως αξιοποιώντας ένα σύνολο γνωρισμάτων, με το μεγαλύτερο μέρος της εργασίας να σχετίζεται με την εύρεση των βέλτιστων από αυτά γνωρίσματα. Ως εκ τούτου, η απόδοση αυτών επηρεάζεται έντονα από εκείνα που επιλέχθηκαν. Η εύρεση και κυρίως, η επιλογή των χαρακτηριστικών είναι αυτή που παίζει τον σημαντικότερο ρόλο στη βελτίωση της ακρίβειας της ταξινόμησης.

3.2 Μηχανική Μάθηση

Σε αυτό το σημείο θα παρουσιάσουμε τους αλγορίθμους μηχανικής μάθησης που θα χρησιμοποιηθούν στην παρούσα προσέγγιση για την Ανίχνευση Σαρκασμού.

3.2.1 Naive Bayes

Ο Naive Bayes (“Αφελής” Μπέυζ) είναι ένας πολύ δημοφιλής ταξινομητής σε προβλήματα επεξεργασίας φυσικής γλώσσας. Ένα από τα πρώτα προβλήματα στα οποία εφαρμόστηκε ήταν αυτό της αναγνώρισης μηνυμάτων ανεπιθύμητης ηλεκτρονικής αλληλογραφίας (spam) (Sahami κ.ά. [91], Androutsopoulos κ.ά. [92]). Διακρίνεται για την απλότητα και την αποδοτικότητά του.

Όπως υποδηλώνει και το όνομά της μεθόδου, βασίζεται στο θεώρημα πιθανότητας του Bayes.

$$P(c \mid d) = \frac{P(d \mid c)P(c)}{P(d)}$$

- $P(c)$, είναι η πιθανότητα εμφάνισης της κλάση c. Καλείται εκ των προτέρων πιθανότητα (prior probability) της c. Υπολογίζεται ως εξής: $P(c_i) = \frac{N_{c_i}}{N_c}$, όπου N_{c_i}, το πλήθος των παρατηρήσεων που ανήκουν στην κλάση c_i και N_c το πλήθος των κλάσεων.
- $P(d)$, είναι η πιθανότητα εμφάνισης του εγγράφου d. Καλείται εκ των προτέρων πιθανότητα(prior probability) του d.
- $P(c \mid d)$, είναι η πιθανότητα το έγγραφο d να ανήκει στην κλάση c. Καλείται εκ των υστέρων πιθανότητα (posterior probability) της c . Αυτό είναι το ζητούμενο.
- $P(d \mid c)$, είναι η πιθανότητα να παρατηρηθεί το έγγραφο d, με δεδομένο ότι ανήκει στην κλάση c. Αυτό καλείται πιθανοφάνεια (likelihood). Διαισθητικά, αυτή η τιμή είναι η απάντηση στην ερώτηση: “ποια είναι η πιθανότητα να παρατηρηθεί ένα κείμενο με τα χαρακτηριστικά (λέξεις) του κειμένου d, με δεδομένο ότι ανήκει στην κλάση c”.

Σύμφωνα με τα παραπάνω, για να υπολογίσουμε σε ποια κλάση ανήκει ένα έγγραφο d το οποίο αποτελείται από ένα σύνολο από χαρακτηριστικά $d = x_1, x_2, \dots, x_n$ (δηλαδή λέξεις $\text{word}_1, \text{word}_2, \dots, \text{word}_n$), πρέπει να υπολογίσουμε την κλάση με τη μέγιστη εκ των υστέρων πιθανότητα (Maximum A-posteriori Probability):

$$c_{MAP} = \arg\max_{c \varepsilon C} P(c \mid d)$$

$$= \arg\max_{c \varepsilon C} \frac{P(d \mid c)P(c)}{P(d)}$$

$$= \arg\max_{c \varepsilon C} P(d \mid c)P(c)$$

$$= \arg\max_{c \varepsilon C} P(x_1, x_2, \dots, x_n \mid c)P(c)$$

Το πρόβλημα είναι ο υπολογισμός του $P(x_1, x_2, \dots, x_n \mid c)$. Για να υπολογιστεί, απαιτείται ο υπολογισμός όλων των συνδυασμών των επιμέρους δεσμευμένων πιθανοτήτων, το οποίο (1) είναι πάρα πολύ χρονοβόρο και (2) αυτό σημαίνει ότι χρειάζονται πολύ περισσότερα δεδομένα για την εκτίμηση των πιθανοτήτων. Κάνοντας όμως τις παραδοχές, (1) ότι η σειρά των λέξεων δεν έχει σημασία (bag-of-words) και (2) ότι τα ενδεχόμενα είναι ανεξάρτητα μεταξύ τους, μπορούμε να υπολογίσουμε την παράσταση ως απλά το γινόμενο των επιμέρους δεσμευμένων πιθανοτήτων.

Αυτή η παραδοχή είναι ο λόγος που η τεχνική ονομάζεται “Αφελής” (Naive) Μπέυζ.

$$P(x_1, x_2, \dots, x_n|c) = P(x_1|c) \times P(x_2|c) \times \dots \times P(x_n|c)$$

Έτσι, για τον ταξινόμηση ενός εγγράφου κάνουμε:

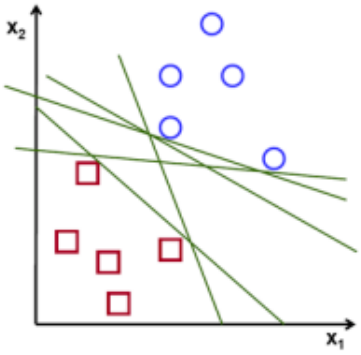
$$c_{MAP} = \arg\underset{c\varepsilon C}{max}\, P(x_1, x_2, \dots, x_n \mid c)P(c)$$

$$c_{MAP} = \arg\underset{c\varepsilon C}{max} \prod_{x\varepsilon X} P(x_i \mid c)P(c)$$

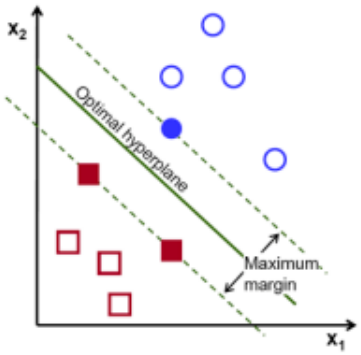
Αυτό πρακτικά σημαίνει ότι η ποσότητα $P(x_i \mid c)$, μετράει πόσο ισχυρή ένδειξη είναι ο όρος t_i στην σωστή ταξινόμηση της κλάσης c.

3.2.2 Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machines)

Οι Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machines - SVM), είναι μία οικογένεια αλγορίθμων επιτηρούμενης μάθησης, για την ταξινόμηση παρατηρήσεων σε δύο ή περισσότερες κλάσεις. Ο τρόπος με τον οποίο κατηγοριοποιούν τις παρατηρήσεις, είναι με την εύρεση ενός (ή και περισσότερων) υπερεπίπεδου (γενίκευση της ευθείας για διαστάσεις) στον χώρο των χαρακτηριστικών, το οποίο να τις διαχωρίζει γραμμικά. Επιπλέον, ένα SVM προσπαθεί να βρει εκείνο το υπερεπίπεδο, το οποίο θα έχει την μέγιστη δυνατή απόσταση από τις παρατηρήσεις κάθε κλάσης.



Εικόνα 3: Πιθανά υπερεπίπεδα διαχωρισμού.



Εικόνα 4: Ένας γραμμικός ταξινομητής μέγιστου

Ας υποθέσουμε ότι έχουμε ένα δυαδικό πρόβλημα ταξινόμησης με γραμμικά μοντέλα της μορφής:

$$f(x) = w^T.\varphi(x) + b$$

όπου $\varphi(x)$ είναι ένα μετασχηματισμός στο χώρο χαρακτηριστικών και η παράμετρος b (bias) έχει οριστεί. Το σύνολο δεδομένων εκπαίδευσης αποτελείται από N διανύσματα εισόδου $x_1, x_2, \dots, x_n$ με αντίστοιχες τιμές εξόδου $y_1, y_2, \dots, y_n$ όπου $y_i \in \{1,1\}$, και νέα δείγματα x ταξινομούνται ανάλογα με το πρόσημο της $f(x)$.

Υποθέτουμε ότι τα δεδομένα εκπαίδευσης είναι γραμμικά διαχωρίσιμα στο χώρο χαρακτηριστικών, ούτως ώστε εξ ορισμού να υπάρχει τουλάχιστον μία επιλογή παραμέτρων w και b τέτοια ώστε μία συνάρτηση της παραπάνω μορφής να ικανοποιεί την ανισότητα $f(x_i) > 0$ για δεδομένα που έχουν $y_i = +1$ και την $f(x_i) < 0$ για δεδομένα που έχουν $y_i = -1$, έτσι ώστε $y_i f(x_i) > 0$ για όλα τα δεδομένα εκπαίδευσης.

Ένα παράδειγμα δεδομένων εκπαίδευσης φαίνεται στην Εικόνα 3, όπου τα δείγματα που ανήκουν στην πρώτη κατηγορία είναι κόκκινα και τετράγωνα, ενώ τα δείγματα που ανήκουν στην δεύτερη κατηγορία είναι μπλε και κυκλικά.

Υπάρχει άπειρο πλήθος πιθανών ευθειών που διαχωρίζουν τις δύο κλάσεις. Ο στόχος του αλγορίθμου SVM είναι να βρει τον πιο γενικό ταξινομητή. Με άλλα λόγια, ο αλγόριθμος SVM προσπαθεί να βρει το υπερεπίπεδο για το οποίο η ελάχιστη απόσταση μεταξύ των δύο κλάσεων (περιθώριο - margin) έχει την μέγιστη δυνατή τιμή. Το υπερεπίπεδο που ικανοποιεί την παραπάνω απαίτηση είναι το βέλτιστο υπερεπίπεδο και μπορεί να παρατηρηθεί στην Εικόνα 4.

Αν η $f(x)$ διαχωρίζει τα δείγματα, η γεωμετρική απόσταση μεταξύ ενός σημείου x_i και του υπερεπιπέδου $f(x) = 0$ είναι ίση με $\frac{|f(x_i)|}{\|w\|}$. Επιπλέον, ενδιαφερόμαστε μόνο για λύσεις για τις οποίες όλα τα δείγματα ταξινομούνται σωστά, ούτως ώστε $y_i f(x_i) > 0$ για κάθε i . Έπειτα, η απόσταση μεταξύ ενός σημείου x_i και του βέλτιστου υπερεπιπέδου δίνεται από:

$$\frac{y_i f(x_i)}{||w||} = \frac{y_i(w^T \phi(x_i) + b)}{||w||}$$

Το περιθώριο δίνεται από την κάθετη απόσταση στο κοντινότερο σημείο x_n από το σύνολο δεδομένων, και επιθυμούμε να βελτιστοποιήσουμε τις παραμέτρους w και b για

να μεγιστοποιήσουμε αυτή την απόσταση. Οπότε, το μέγιστο περιθώριο βρίσκεται λύνοντας την

$$L(w,b) = \arg\max_{w,b}\left\{\frac{1}{||w||}\min_i\{y_i(w^T\varphi(x_i)+b)\}\right\}$$

Η μεγιστοποίηση $\frac{1}{||w||}$ ισοδυναμεί με ελαχιστοποίηση της $\frac{1}{2}\parallel w \parallel^2$. Το πρόβλημα τώρα μετασχηματίζεται ως εξής:

$$L(w) = \arg\min_{w,b}\left\{\frac{1}{2}\parallel w \parallel\right\}$$

$$y_i(w^T\varphi(x_i)+b) \geq 1, i = 1, \dots, N.$$

Η λύση της εξίσωσης αυτής δίνεται από τους πολλαπλασιαστές Lagrange.

3.2.3 Αλγόριθμος Μέγιστης Εντροπίας

Ο αλγόριθμος μέγιστης εντροπίας (Maximum Entropy) που ονομάζεται και αλγόριθμος λογιστικής παλινδρόμησης (Logistic Regression) υλοποιεί παρά την ονομασία του ένα γραμμικό μοντέλο με σκοπό την ταξινόμηση και όχι την παλινδρόμηση. Πρόκειται για έναν πιθανοτικό ταξινομητή του οποίου οι πιθανότητες εξόδου μοντελοποιούνται κάνοντας χρήση μιας λογιστικής συνάρτησης, δηλαδή συνάρτησης της μορφής:

$$f(x) = \frac{L}{1+e^{-k(x-x_0)}}$$

Η γενίκευση της λογιστικής συνάρτησης στην περίπτωση πολλών εισόδων ονομάζεται Softmax Function και ορίζεται παρακάτω.

Όπως υποδηλώνει το όνομά του, ο MaxEntropy βασίζεται στην αρχή της μέγιστης εντροπίας, σύμφωνα με την οποία μεταξύ όλων των μοντέλων που ταιριάζουν με τα δεδομένα επιλέγεται εκείνο που δεν κάνει καμία άλλη υπόθεση πέρα των περιορισμών που επιβάλλονται από το training set και συνεπώς η κατανομή είναι όσο το δυνατόν ομοιόμορφη. Ο ταξινομητής MaxEntropy χρησιμοποιείται σε πολλά προβλήματα ταξινόμησης κειμένου, όπως ανίχνευση γλώσσας, ταξινόμηση με βάση το θέμα του κειμένου, ανάλυση συναισθήματος και άλλα.

Ανόμοια με τον επίσης πιθανοτικό ταξινομητή Naive Bayes, που αναπτύχθηκε στην προηγούμενη ενότητα, ο MaxEntropy δεν υποθέτει ότι τα χαρακτηριστικά είναι υπό συνθήκη ανεξάρτητα μεταξύ τους. Το γεγονός ότι ο αλγόριθμος μέγιστης εντροπίας δεν κάνει κάποια υπόθεση ανεξαρτησίας μεταξύ των χαρακτηριστικών τον καθιστά ιδιαίτερα αποδοτικό σε προβλήματα ταξινόμησης κειμένου, όπου τα χαρακτηριστικά-λέξεις δεν είναι προφανώς ανεξάρτητα μεταξύ τους. Το μειονέκτημά του σε σχέση με τον Naive Bayes είναι ο μεγαλύτερος χρόνος εκπαίδευσης λόγω του προβλήματος βελτιστοποίησης που πρέπει να επιλυθεί προκειμένου να προσδιοριστούν οι παράμετροί του.

Ας θεωρήσουμε την κλασική BoW αναπαράσταση και έστω $\{w_1, w_2, \dots, w_n\}$ οι λέξεις του λεξιλογίου. Ο στόχος είναι να κατασκευάσουμε ένα στοχαστικό μοντέλο που θα δέχεται

στην είσοδο ένα κείμενο x και θα το αντιστοιχεί σε μία κατηγορία c_j (θετική ή αρνητική για το πρόβλημά μας). Αρχικά από το training set που έχουμε στη διάθεσή μας, υπολογίζουμε με MLE την εμπειρική πιθανότητα το τυχαίο κείμενο $\mathbf{x}$ να ανήκει στην κατηγορία c:

$$\tilde{P}(\mathbf{x}, c) = \frac{\text{times that sample } (\mathbf{x}, c) \text{ appears in the training set}}{\text{training set size}}$$

Ορίζουμε στη συνέχεια την παρακάτω Boolean συνάρτηση:

$$f_i(\mathbf{x}, c) = \begin{cases} 1, & \text{if } c = c_j \text{ and word } w_i \text{ in } \mathbf{x} \\ 0, & \text{otherwise} \end{cases}$$

η οποία στην βιβλιογραφία του MaxEntropy ονομάζεται χαρακτηριστικό.
Ορίζουμε και τις ακόλουθες δύο προσδοκίες χαρακτηριστικών (feature expectations):

- Αναμενόμενη τιμή χαρακτηριστικού ως προς την εμπειρική κατανομή $\tilde{P}(\mathbf{x}, c)$:

$$E(f_i) = \sum_{(\mathbf{x},c)} \tilde{P}(\mathbf{x}, c) f_i(\mathbf{x}, c)$$

- Αναμενόμενη τιμή χαρακτηριστικού ως προς το μοντέλο $P(c \mid \mathbf{x})$:

$$E(f_i) = \sum_{\mathbf{x},c} \tilde{P}(\mathbf{x}) P(c \mid \mathbf{x}) f_i(\mathbf{x}) f_i(\mathbf{x}, c)$$

όπου $\tilde{P}(\mathbf{x})$είναι η εμπειρική κατανομή του x στο training set και συνήθως:

$$\tilde{P}(\mathbf{x}) = \frac{1}{\text{training set size}}$$

Επιβάλλοντας τον περιορισμό η αναμενόμενη τιμή να είναι ίση με την εμπειρική τιμή, έχουμε:

$$\sum_{\mathbf{x},c} \tilde{P}(\mathbf{x}) P(c \mid \mathbf{x}) f_i(\mathbf{x}, c) = \sum_{(\mathbf{x},c)} \tilde{P}(\mathbf{x}, c) f_i(\mathbf{x}, c)$$

Η εξίσωση αυτή καλείται περιορισμός και έχουμε έναν περιορισμό για κάθε χαρακτηριστικό f_i.

Οι παραπάνω περιορισμοί μπορούν να ικανοποιηθούν από άπειρα στοχαστικά μοντέλα. Κάνοντας χρήση της αρχής της μέγιστης εντροπίας, ο αλγόριθμος επιλέγει το μοντέλο που είναι κατά το δυνατόν ομοιόμορφο. Δηλαδή επιλέγει το μοντέλο P^*:

$$P^* = \arg_{P \varepsilon C} max \left(- \sum_{\mathbf{x},c} \tilde{P}(\mathbf{x}) P(c \mid \mathbf{x}) \log P(c \mid \mathbf{x}) \right)$$

σύμφωνα με τους περιορισμούς *C*:

$$P(c \mid \mathbf{x}) \geq 0 \forall \mathbf{x}, c$$
$$\sum_{c} P(c \mid \mathbf{x}) = 1 \forall \mathbf{x}$$
$$\sum_{\mathbf{x},c} \tilde{P}(c, \mathbf{x}) f_i(\mathbf{x}, c) = \sum_{\mathbf{x},c} \tilde{P}(\mathbf{x}, c) f_i(\mathbf{x}, c), i = 1, \dots, n$$

Το παραπάνω πρόβλημα μετατρέπεται στο δυικό πρόβλημα χωρίς περιορισμούς κάνοντας χρήση των πολλαπλασιαστών $\lambda_1, \lambda_2, \dots, \lambda_n$. Η εκτίμηση των παραμέτρων λ_i απαιτεί τη χρησιμοποίηση ενός επαναληπτικού αλγορίθμου κλιμάκωσης, όπως ο GIS (Generalized Iterative Scaling) ή ο IIS (Improved Iterative Scaling). Αποδεικνύεται ότι αφού βρούμε τους πολλαπλασιαστές Lagrange, η εκ των υστέρων πιθανότητα το κείμενο x να ανήκει στην κατηγορία c_j γίνεται από την συνάρτηση softmax και είναι:

$$P(c_j \mid x) = \frac{\exp\left(\sum_i \lambda_i f_i(\mathbf{x}, c_j)\right)}{\sum_c \exp\left(\sum_i \lambda_i f_i(\mathbf{x}, c_j)\right)}$$

Η παράμετρος λ_i δηλώνει το βάρος του χαρακτηριστικού i στην επιλογή της κλάσης c_j.

Μεγάλη θετική τιμή δηλώνει ότι η λέξη i πιθανότατα σχετίζεται με την κλάση c_j, ενώ μεγάλη αρνητική τιμή σημαίνει ότι η λέξη i πιθανότατα δεν σχετίζεται με την κλάση c_j.

3.3 Τεχνητά Νευρωνικά Δίκτυα

Τα ΤΝΔ είναι μία οικογένεια αλγορίθμων μηχανικής μάθησης, εμπνευσμένα από τον τρόπο λειτουργίας των νευρώνων του εγκεφάλου. Ο νευρώνας είναι ένα κύτταρο, το οποίο αποτελεί δομικό μέρος του εγκεφάλου και κατά συνέπεια του νευρικού συστήματος. Ένας νευρώνας συνδέεται με άλλους νευρώνες μέσω ειδικών συνδέσεων, τις συνάψεις. Κάθε νευρώνας μπορεί να έχει πολλές εισόδους, δηλαδή να λαμβάνει σήματα από πολλούς άλλους νευρώνες, αλλά έχει μία μόνο έξοδο.

Όταν το άθροισμα των σημάτων ενός νευρώνα, ξεπεράσει ένα συγκεκριμένο κατώφλι, τότε ο νευρώνας ενεργοποιείται, βγάζοντας ένα σήμα εξόδου, το οποίο δίνεται ως είσοδος σε άλλους νευρώνες οι οποίοι είναι συνδεδεμένοι με αυτόν. Αυτό είναι ένα υπεραπλουστευμένο μοντέλο της λειτουργίας του ανθρώπινου νευρωνικού δικτύου. Όμως, αρκετές φορές μοντέλα εμπνευσμένα από τη βιολογία, ακόμα και πολύ απλοϊκά, μπορούν να φανούν χρήσιμα σε πολλά προβλήματα.

Στο πλαίσιο των ΤΝΔ, ο νευρώνας είναι μία υπολογιστική μονάδα. Ο πρώτος τέτοιος αλγόριθμος είναι ο Perceptron (Rosenblatt, [93]), ο οποίος είναι ένας γραμμικός

ταξινομητής. Ένας νευρώνας δέχεται ως είσοδο σήματα από τις συνάψεις (συνδέσεις εισόδου), ορίζοντας κάποια βάρη σε κάθε μία από αυτές. Τα σήματα σταθμισμένα ως προς τα βάρη, δίνονται ως είσοδο σε μία συνάρτηση ενεργοποίησης (activation function). Όταν η τιμή της συνάρτησης ενεργοποίησης πάρει τιμή πάνω από ένα ορισμένο κατώφλι, τότε βγάζει ως έξοδο την τιμή 1 (ο νευρώνας ενεργοποιείται). Ένας νευρώνας μπορεί να έχει διάφορες συναρτήσεις ενεργοποίησης που επηρεάζουν την συμπεριφορά του.

Με στόχο την μάθηση σύνθετων μη-γραμμικών συναρτήσεων, είναι δυνατό να σχεδιαστούν αρχιτεκτονικές που συνδυάζουν πολλούς νευρώνες, στοιχισμένους σε επίπεδα. Οι αρχιτεκτονικές αυτές είναι γνωστές και ως Multi-Layer Perceptron (MLP). Αντί για τον νευρώνα τύπου Perceptron, μπορεί να χρησιμοποιηθεί κάποια άλλη υπολογιστική μονάδα στους κόμβους του δικτύου. Μία συνήθης αρχιτεκτονική είναι αυτή του Feed-Forward Neural Network (FFNN), στο οποίο κάθε νευρώνας συνδέεται με όλους τους νευρώνες του προηγούμενου επιπέδου. Στην αρχιτεκτονική αυτή δεν υπάρχουν συνδέσεις μεταξύ των νευρώνων του ίδιου επιπέδου. Τα επίπεδα τα οποία μεσολαβούν ανάμεσα στα επίπεδα εισόδου και εξόδου, ονομάζονται κρυφά επίπεδα. Αν ένα δίκτυο έχει περισσότερα από ένα κρυφά επίπεδα, τότε το δίκτυο λέγεται ότι είναι βαθύ. Τα δίκτυα αυτά ονομάζονται Βαθιά Νευρωνικά Δίκτυα ή Deep Neural Networks (DNN). Αυξάνοντας το βάθος του δικτύου και με την χρήση μη-γραμμικών συναρτήσεων ενεργοποίησης το δίκτυο μπορεί να μάθει να εκτελεί πιο σύνθετες εργασίες.

Ένα FFNN με τουλάχιστον ένα κρυφό επίπεδο και πεπερασμένο πλήθος νευρώνων, έχει αποδειχθεί ότι είναι ένας καθολικός προσεγγιστής (universal approximator) (Hornik κ.ά. [94]). Αυτό σημαίνει ότι μπορεί να προσεγγίσει οποιαδήποτε συνεχή συνάρτηση. Στην πράξη όμως, για δίκτυα με ένα μόνο κρυφό δίκτυο, αυτό δεν ισχύει για τα προβλήματα που μας ενδιαφέρουν. Για τον λόγο αυτό τα τελευταία χρόνια η επιστημονική κοινότητα έχει στραφεί στη δημιουργία βαθιών νευρωνικών δικτύων.

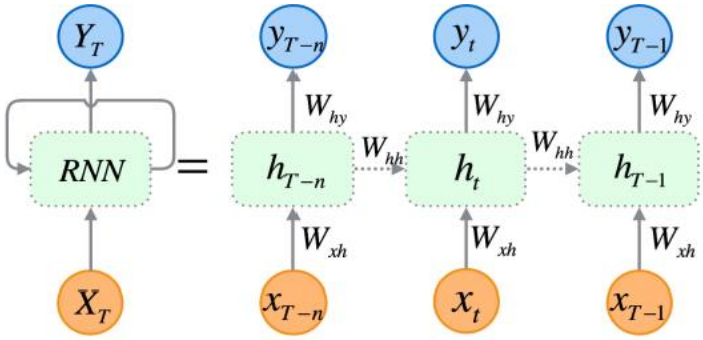
Στο πλαίσιο της έρευνας θα ασχοληθούμε με δύο σύνθετες αρχιτεκτονικές οι οποίες τα τελευταία χρόνια έχουν πετύχει πολύ καλά αποτελέσματα σε προβλήματα ΕΦΓ. Το είδος δικτύου το οποίο θα εξετάσουμε είναι τα Επαναλαμβανόμενα Νευρωνικά Δίκτυα (Recurrent Neural Networks, RNN) και πιο συγκεκριμένα τα Δίκτυα Μακράς Βραχύχρονης Μνήμης (Long short-term memory, LSTM).

3.3.1 Επαναλαμβανόμενα Νευρωνικά Δίκτυα (Recurrent Neural Networks)

Τα επαναλαμβανόμενα νευρωνικά δίκτυα (Lipton κ.ά. [95]) είναι ένα είδος τεχνητού νευρωνικού δικτύου σχεδιασμένο να αναγνωρίζει μοτίβα σε ακολουθίες δεδομένων όπως χρηματιστηριακές τιμές, κείμενα, ηχητικά δεδομένα κ.ά.

Αυτό που διαφοροποιεί τα επαναλαμβανόμενα (RNNs) από τα υπόλοιπα δίκτυα είναι ότι είναι προσαρμοσμένα να λειτουργούν για δεδομένα χρονοσειρών ή δεδομένα που περιλαμβάνουν ακολουθίες.

Η αρχιτεκτονική του RNN επεξεργάζεται την ακολουθία επαναληπτικά για κάθε στοιχείο της και διατηρώντας μια κατάσταση που περιέχει πληροφορίες σχετικά με αυτό που έχει δει, καταγράφει τη συσχέτιση μεταξύ του τρέχοντος χρονικού βήματος και των προηγούμενων. Η τυπική αρχιτεκτονική του RNN, ενώ και το εσωτερικό μίας μονάδας αυτού αναπαριστώνται στα επόμενα γραφήματα:



Εικόνα 5: Αρχιτεκτονική RNN (αριστερά) και εσωτερική δομή μιας μονάδας (cell) του δικτύου

Ενώ στα τυπικά νευρωνικά δίκτυα τροφοδοσίας κάθε στοιχείο διανύσματος εισόδου έχει τα βάρη του, στα επαναλαμβανόμενα δίκτυα, κάθε στοιχείο μοιράζεται τα ίδια βάρη.

3.3.2 Δίκτυα Μακράς Βραχύχρονης Μνήμης (Long short-term memory, LSTM)

Αν μια ακολουθία είναι μεγάλη, η μεταφορά πληροφορίας από προηγούμενα χρονικά βήματα σε μεταγενέστερα αποτελεί απαιτητικό εγχείρημα. Τα δίκτυα μακράς βραχυπρόθεσμης μνήμης δημιουργήθηκαν για να δώσουν λύση σε αυτό ακριβώς το πρόβλημα. Έχουν εσωτερικούς μηχανισμούς που ονομάζονται πύλες (gates) που παρέχουν τη δυνατότητα ρύθμισης της ροής της πληροφορίας. Με αυτό τον τρόπο “μαθαίνουν” ποια δεδομένα είναι χρήσιμα σε μια σειρά και μπορούν να διαβιβάσουν σχετικότερες πληροφορίες με σκοπό να επιτευχθούν ακριβέστερες προβλέψεις (Staudemeyer και Morris [96]). Ως επακόλουθο, τα LSTM χρησιμοποιούνται σε εφαρμογές αναγνώρισης και σύνθεσης ομιλίας και κειμένου.

3.3.3 Αμφίδρομα Δίκτυα Μακράς Βραχύχρονης Μνήμης (Bidirectional Long short-term memory, BiLSTM)

Ένα αμφίδρομο δίκτυο μακράς βραχύχρονης μνήμης, είναι ένα μοντέλο επεξεργασίας ακολουθίας που αποτελείται από δύο δίκτυα μακράς βραχύχρονης μνήμης. Το ένα παίρνει την είσοδο σε μια κατεύθυνση προς τα εμπρός και το άλλο σε μια κατεύθυνση προς τα πίσω. Τα Bi-LSTM αυξάνουν αποτελεσματικά τον όγκο των πληροφοριών που είναι διαθέσιμες στο δίκτυο, βελτιώνοντας το σημασιολογικό και εννοιολογικό πλαίσιο που είναι διαθέσιμο στον αλγόριθμο, γνωρίζοντας ποιες λέξεις ακολουθούν αμέσως και προηγούνται μιας λέξης σε μια πρόταση (Karpathy, Johnson και Fei-Fei Li [97]). Για περαιτέρω κατανόηση χρησιμοποιείται το εξής παράδειγμα: «*Ο Γιώργος είναι ευχάριστο άτομο, τον έχεις γνωρίσει;*», «*Στη χημεία και στη φυσική το μικρότερο σωματίδιο ενός χημικού στοιχείου είναι το άτομο, το οποίο παραμένει αμετάβλητο κατά την εξέλιξη ενός χημικού φαινομένου.*»

Η λέξη άτομο έχει αμφίσημη σημασία στο παράδειγμα. Χωρίζοντας τις προτάσεις στη μέση και παρατηρώντας τι ακολουθεί και τι προηγείται, εκτιμώνται με μεγαλύτερη ακρίβεια τα συμφραζόμενα.

3.4 Διανυσματική Αναπαράσταση Λέξεων (Word Embeddings)

Στην Επεξεργασία Φυσικής Γλώσσας, η Διανυσματική Αναπαράσταση Λέξεων είναι η αναπαράσταση λέξεων ως ένα διάνυσμα πραγματικών αριθμών σε έναν πολυδιάστατο (m-διαστάσεις) διανυσματικό χώρο $\Re^m$.

Από τις πιο απλές μορφές τέτοιων αναπαραστάσεων είναι η “one-hot” ή “1-από-Ν” κωδικοποίηση. Σε μια τέτοια αναπαράσταση, ο διανυσματικός χώρος έχει τον ίδιο αριθμό διαστάσεων με τον αριθμό των λέξεων στο λεξιλόγιο όπου κάθε διανυσματική αναπαράσταση λέξης αποτελείται κυρίως από μηδενικά, με ένα «1» στην αντίστοιχη διάσταση της λέξης.

Μια προσπάθεια επιλογής των διαστάσεων της αναπαράστασης με μη αυτόματο τρόπο, έτσι ώστε να αποτυπώνονται οι σχέσεις μεταξύ των λέξεων, θα “επιβραβευόταν” με μερικά πλεονεκτήματα:

- Το σύνολο των αναπαραστάσεων θα είναι πιο αποτελεσματικό, κάθε λέξη θα αντιπροσωπεύεται από ένα διάνυσμα μικρότερου μήκους.
- Οι αναπαραστάσεις αυτές δηλαδή θα κωδικοποιούν το νόημα των λέξεων, με τρόπο που λέξεις που βρίσκονται πιο κοντά στο διανυσματικό χώρο αναμένεται να έχουν κοινά μορφολογικά, σημασιολογικά ή συντακτικά χαρακτηριστικά.
- Πιθανή αύξηση του λεξιλογίου δεν θα είχε ως συνέπεια την αύξηση και των διαστάσεων.

Αρχικά, οι λέξεις κωδικοποιούνται μέσω στατιστικών μεθόδων και τεχνικών. Κάποιες από αυτές είναι το Bag-Of-Words, TF-IDF, η Διάσπαση Ιδιόμορφων Τιμών (Singular Value Decomposition) η Λανθάνουσα Σημασιολογική Δεικτοδότηση (Latent Semantic Indexing) και η Λανθάνουσα Κατανομή Dirichlet (Latent Dirichlet Allocation).

Εδώ, θα χρησιμοποιήσουμε τις τεχνικές Bag-of-Words και Term Frequency–Inverse Document Frequency.

3.4.1 Bag-of-Words (BoW)

Το BoW είναι ένας ευρέως χρησιμοποιούμενος αλγόριθμος στην Επεξεργασία της Φυσικής Γλώσσας, ο οποίος βασίζεται στη συχνότητα των λέξεων του κειμένου. Χρησιμοποιείται για την αναπαράσταση του κειμένου ως σάκος των λέξεων που το αποτελούν, παραβλέποντας τη σειρά και τη σημασιολογία τους, αλλά αποθηκεύοντας την πολλαπλότητα εμφάνισης της κάθε λέξης. Συγκεκριμένα, σε προβλήματα επεξεργασίας φυσικής γλώσσας με την τεχνική BoW δημιουργείται μια αναπαράσταση που μετατρέπει το αυθαίρετο κείμενο σε διανύσματα σταθερού μήκους μετρώντας πόσες φορές εμφανίζεται κάθε λέξη. Παράδειγμα του συγκεκριμένου αλγορίθμου παρουσιάζεται παρακάτω.

Ακολουθούν δύο απλά έγγραφα κειμένου:

- **Έγγραφο 1ο**: John likes to watch movies. Mary likes movies too.
- **Έγγραφο 2ο**: Mary also likes to watch football games.

Βάσει αυτών των εγγράφων δημιουργείται το ακόλουθο λεξιλόγιο για κάθε ένα από αυτά:

- **Λεξιλόγιο 1ο**: “John”,“likes”,“to”,“watch”,“movies”,“Mary”,“likes”,“movies”,“too”
- **Λεξιλόγιο 2ο**: “Mary”,“also”,“likes”,“to”,“watch”,“football”,“games”

Οπότε για κάθε ένα πό αυτά τα έγγραφα δημιουργείται το αντίστοιχο BoW:

- **BoW 1ο** = {“John”:1,“likes”:2,“to”:1,“watch”:1,“movies”:2,“Mary”:1,“too”:1};
- **BoW 2ο** = {“Mary”:1,“also”:1,“likes”:1,“to”:1,“watch”:1,“football”:1,“games”:1}

Για να μπορεί να χρησιμοποιηθεί η συγκεκριμένη τεχνική από έναν κατηγοριοποιητή πρέπει το έγγραφο να μετατραπεί σε διάνυσμα όπου κάθε τιμή του παίρνει την συχνότητα εμφάνισης της κάθε λέξης.

Στο παράδειγμα προκύπτουν τα εξής διανύσματα:

Πίνακας 1: BoW διανυσματική αναπαράσταση των εγγράφων

	John	likes	to	watch	movies	Mary	too	also	football	games
Έγγραφο 1ο	1	2	1	1	2	1	1	0	0	0
Έγγραφο 2ο	0	1	1	1	0	1	0	1	1	1

3.4.2 TF-IDF (Term Frequency–Inverse Document Frequency)

Σύμφωνα με αυτή την μετρική, αξιολογείται πόσο σχετική είναι μια λέξη σε ένα έγγραφο σε μια συλλογή εγγράφων. Αυτό γίνεται πολλαπλασιάζοντας δύο μετρικές: πόσες φορές μια λέξη εμφανίζεται σε ένα έγγραφο (Term Frequency ή TF) και την αντίστροφη συχνότητα εγγράφου της λέξης σε ένα σύνολο εγγράφων (Inverse Document Frequency ή IDF).

Ο υπολογισμός της μετρικής TF γίνεται βάσει του αριθμού των φορών που εμφανίζεται μια λέξη σε ένα έγγραφο διαιρούμενος με τον συνολικό αριθμό των λέξεων στο έγγραφο. Ο τύπος που υπολογίζει τη σχέση αυτή είναι ο παρακάτω:

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k^{n_{i,j}}}$$

Η μετρική IDF ισούται με τον λογάριθμο του αριθμού των εγγράφων διαιρούμενο με τον αριθμό των εγγράφων που περιέχουν τη λέξη w. Η αντίστροφη συχνότητα δεδομένων καθορίζει το βάρος των σπάνιων λέξεων σε όλα τα έγγραφα του σώματος. Η μαθηματική σχέση που περιγράφηκε δίνεται παρακάτω:

$$idf(W) = \log\left(\frac{N}{df_t}\right)$$

Η τελική σχέση που περιγράφει την τεχνική του TF-IDF είναι ουσιαστικά ο πολλαπλασιασμός της μετρικής TF με την IDF και αναγράφεται στην επόμενη σχέση:

$$w_{i,j} = tf_{i,j} \times \log\left(\frac{N}{df_i}\right)$$

Για τα προηγούμενα δύο έγγραφα τα διανύσματα θα προέκυπταν ως ακολούθως:

Πίνακας 2: TF-IDF διανυσματική αναπαράσταση των εγγράφων

Λέξη	TF		IDF	TF-IDF	
	Έγγραφο 1ο	Έγγραφο 2ο		Έγγραφο 1ο	Έγγραφο 2ο
John	1/9	0/7	log(2/1)=0.3	0.033	0
likes	2/9	1/7	log(2/2)=0	0	0
to	1/9	1/7	log(2/2)=0	0	0
watch	1/9	1/7	log(2/2)=0	0	0
movies	2/9	0/7	log(2/1)=0.3	0.067	0
Mary	1/9	1/7	log(2/2)=0	0	0
too	1/9	0/7	log(2/1)=0.3	0.033	0
also	0/9	1/7	log(2/1)=0.3	0	0.043
football	0/9	1/7	log(2/1)=0.3	0	0.043
games	0/9	1/7	log(2/1)=0.3	0	0.043

Ωστόσο, ένα βασικό μειονέκτημα στη στατιστική μοντελοποίηση γλώσσας είναι ότι δημιουργούνται προβλήματα που σχετίζονται με την “κατάρα της διάστασης” (curse of dimensionality). Ένα τέτοιο πρόβλημα, παραδείγματος χάριν, είναι πως στις περιπτώσεις που αλλάζει το λεξικό οι τιμές των διανυσμάτων πρέπει να υπολογίζονται εκ νέου. Επίσης, οι στατιστικές μέθοδοι είναι πολύ ευαίσθητες στην ανισορροπία της συχνότητας των λέξεων, επομένως συχνά είναι απαραίτητη η προ-επεξεργασία των εγγράφων με ληματοποίηση (lemmatization), αποκατάληξη (stemming) κ.ά.

Στη συνέχεια, ξεκίνησαν οι προσπάθειες για την χρήση μεθόδων μηχανικής μάθησης κυρίως με τη βοήθεια νευρωνικών δικτύων τα οποία εκπαιδεύονται σε μεγάλες συλλογές κειμένων (corpus). Δημιουργήθηκαν έτσι διανυσματικές αναπαραστάσεις λέξεων που έχουν προκύψει από προ-εκπαιδευμένα μοντέλα οι οποίες χρησιμοποιούνται σε διάφορες εργασίες (downstream tasks) που χρησιμοποιούν μικρές ποσότητες δεδομένων με ετικέτα (labeled data).

Οι διανυσματικές αναπαραστάσεις λέξεων με χρήση μεθόδων μηχανικής μάθησης είναι μία από τις λίγες επί του παρόντος επιτυχημένες εφαρμογές στην εκπαίδευση σε εργασία Επεξεργασίας Φυσικής Γλώσσας. Το κύριο πλεονέκτημά τους είναι αναμφισβήτητα ότι δεν απαιτούν ακριβό σχολιασμό - επισήμανση (annotation).

Χωρίζουμε τις προσπάθειες σε δύο κατηγορίες ανάλογα με το αν λαμβάνουν υπόψιν το συγκείμενο πλαίσιο (context) ή όχι.

3.4.3 Παραδοσιακές Τεχνικές Διανυσματικής Αναπαράστασης Λέξεων (Traditional / Static Word Embeddings)

Ο όρος Διανυσματική Αναπαράσταση Λέξεων προτάθηκε από τους Bengio κ.ά. [98] ως “κατανεμημένη αναπαράσταση για λέξεις” (distributed representation for words) για την αντιμετώπιση της οικογένειας προβλημάτων που είναι γνωστή ως “η κατάρα της διάστασης”, όπως περιγράφηκε παραπάνω. Η αρχιτεκτονική τους αποτελεί σε μεγάλο βαθμό το πρωτότυπο πάνω στο οποίο οι τρέχουσες προσεγγίσεις έχουν σταδιακά βελτιωθεί.

Οι Collobert και Weston [99] ήταν αναμφισβήτητα οι πρώτοι που επέδειξαν τη δύναμη των προ-εκπαιδευμένων διανυσματικών αναπαραστάσεων λέξεων, όπου τις καθιερώνουν ως ένα εξαιρετικά αποτελεσματικό εργαλείο όταν χρησιμοποιούνται σε εργασίες (task) μη εποπτευόμενης μάθησης, ενώ επίσης ανακοίνωσαν μια

αρχιτεκτονική νευρωνικών δικτύων πάνω στην οποία βασίστηκαν πολλές από τις σημερινές προσεγγίσεις.

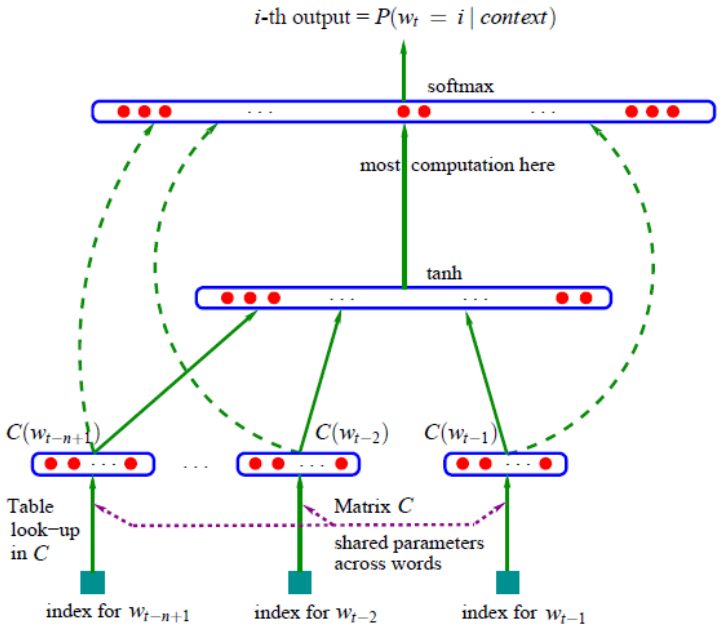
Παρόλα αυτά, ήταν οι Mikolov κ.ά. [100] στη Google που έφεραν πραγματικά την διανυσματική αναπαράσταση στο προσκήνιο, λύνοντας ζητήματα προηγούμενων προσεγγίσεων, μέσω της δημιουργίας του **Word2Vec**, μιας εργαλειοθήκης που επιτρέπει την εκπαίδευση και τη χρήση προ-εκπαιδευμένων αναπαραστάσεων. Ένα χρόνο αργότερα, οι Pennington κ.ά. [81] παρουσίασαν το **GloVe**, ένα ανταγωνιστικό σύνολο προ-εκπαιδευμένων αναπαραστάσεων.

Οι Bojanowski κ.ά. στην Facebook [101] επέκτειναν τα μοντέλα τύπου Word2Vec προσθέτοντας πληροφορίες υπολέξεων (sub-word information) δημιουργώντας το **FastText**.

Στη συνέχεια, θα σημειωθούν τα βασικά χαρακτηριστικά των παραπάνω μοντέλων, σύμφωνα με τον Ruder [102].

Κλασικό νευρωνικό μοντέλο γλώσσας

Το κλασικό νευρωνικό μοντέλο γλώσσας που πρότειναν οι Bengio κ.ά. [98] αποτελείται από ένα Δίκτυο Πρόσθιας Τροφοδότησης (Feed-Forward Neural Network) με ένα κρυφό στρώμα (one hidden layer) που προβλέπει την επόμενη λέξη με μια ακολουθία όπως φαίνεται στην παρακάτω εικόνα.



Εικόνα 6: Classic neural language model (Bengio κ.ά. [98])

Για ένα σώμα εκπαίδευσης που περιέχει μια ακολουθία από T λέξεις $w_1, w_{t2}, w_{t3}, \ldots, w_T$ οι οποίες ανήκουν σε ένα λεξιλόγιο V του οποίου το μέγεθος είναι $|V|$, κάθε λέξη w συσχετίζεται με μια αναπαράσταση εισόδου v_w και μια αναπαράσταση εξόδου v_w. Τελικά βελτιστοποιείται μια αντικειμενική συνάρτηση J_θ όσον αφορά τις παραμέτρους ενός μοντέλου θ το οποίο τελικά παράγει το διάνυσμα αναπαράστασης: $f_\theta(w_m)$. Η συνάρτηση J_θ είναι της μορφής:

$$J_\theta = \frac{1}{\mathrm{T}} \sum_{\mathrm{T}}^{t=1} \log p(w_t \mid w_{t-1}, \dots, w_{t-n+1})$$

, όπου n είναι ο αριθμός των προηγούμενων λέξεων που έχουν τροφοδοτηθεί στο μοντέλο.

Το μοντέλο τους που μεγιστοποιεί την συνάρτηση J_θ όπως προτάθηκε από το κλασικό νευρωνικό μοντέλο γλώσσας (παραλείπεται ο όρος κανονικοποίησης (regularization) για απλότητα) είναι:

$$J_\theta = \frac{1}{\mathrm{T}} \sum_{\mathrm{T}}^{t=1} \log f(w_t, w_{t-1}, \dots, w_{t-n+1})$$

, όπου $f(w_t, w_{t-1}, \dots, w_{t-n+1})$ είναι η έξοδος του μοντέλου, δηλαδή η πιθανότητα $p(w_t \mid w_{t-1}, \dots, w_{t-n+1})$ όπως έχει υπολογιστεί από την συνάρτηση softmax:

$$p(w_t \mid w_{t-1}, \dots, w_{t-n+1}) = \frac{\exp(h^\top v'_{w_t})}{\sum_{w_i \varepsilon V} \exp(h^\top v'_{w_i})}$$

όπου h είναι το διάνυσμα εξόδου του προτελευταίου στρώματος δικτύου και $h^\top v'_{w_t}$ είναι το εσωτερικό γινόμενο που υπολογίζει την (μη κανονικοποιημένη) log-πιθανότητα της λέξης w_t.

Τα γενικά δομικά στοιχεία του μοντέλου τους, τα οποία εξακολουθούν να βρίσκονται σε όλα τα τρέχοντα νευρωνικά μοντέλα γλώσσας και διανυσματικών αναπαραστάσεων λέξεων, είναι:

- Στρώμα Εισόδου: ένα στρώμα που δημιουργεί αναπαραστάσεις λέξεων πολλαπλασιάζοντας ένα διάνυσμα index με έναν πίνακα διανυσματικής αναπαράστασης λέξης.
- Ενδιάμεσα Στρώματα: ένα ή περισσότερα στρώματα που παράγουν μια ενδιάμεση αναπαράσταση της εισόδου.
- Στρώμα softmax: το τελικό στρώμα που παράγει μια κατανομή πιθανότητας σε λέξεις στο λεξιλόγιο V.

Επιπλέον, εντόπισαν δύο ζητήματα που βρίσκονται στο επίκεντρο των σημερινών μοντέλων τελευταίας τεχνολογίας:

Παρατήρησαν ότι το ενδιάμεσο κρυφό στρώμα μπορεί να αντικατασταθεί από ένα δίκτυο Μακράς βραχυχρόνιας μνήμης (Long Short Term Memory Neural Networks - LSTM), το οποίο χρησιμοποιείται από μοντέλα νευρωνικών γλωσσών τελευταίας τεχνολογίας. Προσδιόρισαν το τελικό στρώμα softmax (ακριβέστερα: τον όρο

κανονικοποίησης) ως το κύριο σημείο συμφόρησης του δικτύου, καθώς το κόστος υπολογισμού της softmax είναι ανάλογο με τον αριθμό των λέξεων στο λεξιλόγιο V, το οποίο είναι συνήθως της τάξης των εκατοντάδων χιλιάδων ή εκατομμυρίων.

Η εύρεση τρόπων για τον μετριασμό του υπολογιστικού κόστους που σχετίζεται με τον υπολογισμό του softmax σε ένα μεγάλο λεξιλόγιο είναι επομένως μία από τις βασικές προκλήσεις τόσο στα νευρωνικά μοντέλα γλώσσας όσο και στα μοντέλα διανυσματικής αναπαράστασης λέξεων.

C&W model

Μετά τα πρώτα βήματα των Bengio κ.ά. [98] στα νευρωνικά μοντέλα γλώσσας, η έρευνα πάνω στην διανυσματική αναπαράσταση λέξεων παρέμεινε στάσιμη καθώς η υπολογιστική ισχύς και οι αλγόριθμοι δεν επέτρεπαν ακόμη την εκπαίδευση ενός μεγάλου λεξιλογίου.

Οι Collobert και Weston [99] σημειώνουν ότι οι διανυσματικές αναπαραστάσεις λέξεων που έχουν εκπαιδευτεί σε ένα αρκετά μεγάλο σύνολο δεδομένων έχουν συντακτικό και σημασιολογικό νόημα και βελτιώνουν την απόδοση σε 999 downstream εργασίες. Αναλύουν αυτό στην εργασία τους το 2011 [103].

Η λύση τους για να αποφύγουν τον υπολογισμό της υπολογιστικά ακριβής softmax είναι να χρησιμοποιήσουν μια διαφορετική αντικειμενική συνάρτηση. Αντί για το κριτήριο της διασταυρούμενης εντροπίας (cross-entropy criterion) των Bengio κ.ά. [98], το οποίο μεγιστοποιεί την πιθανότητα της επόμενης λέξης με βάση τις προηγούμενες λέξεις, οι Collobert και Weston εκπαιδεύουν ένα δίκτυο να υπολογίζει ως έξοδο την υψηλότερη βαθμολογία f_θ για μια σωστή ακολουθία λέξεων (μια πιθανή ακολουθία λέξεων στο μοντέλο του Bengio) παρά για μια λανθασμένη. Για το σκοπό αυτό, χρησιμοποιούν ένα κριτήριο κατάταξης κατά ζεύγη, όπως:

$$J_\theta = \sum_{x \varepsilon X} \sum_{w \varepsilon V} max\{0{,}1 - f_{\theta(x)} + f_\theta(x^{(w)})\}$$

Δείχνουν τα σωστά παράθυρα x που περιέχουν n λέξεις από το σύνολο όλων των πιθανών παραθύρων X στο σώμα κειμένου (corpus). Για κάθε παράθυρο x , στη συνέχεια παράγουν μια κατεστραμμένη (corrupted), εσφαλμένη έκδοση $x(w)$ αντικαθιστώντας την κεντρική λέξη του x με μια άλλη λέξη w από το λεξιλόγιο V.

Ελαχιστοποιώντας τον στόχο, το μοντέλο θα μάθει τώρα να εκχωρεί μια βαθμολογία για το σωστό παράθυρο που είναι υψηλότερη από τη βαθμολογία για το λανθασμένο παράθυρο κατά τουλάχιστον ένα περιθώριο 1. Η αρχιτεκτονική του μοντέλου τους χωρίς τον στόχο κατάταξης, είναι ανάλογη με το μοντέλο των Bengio κ.ά.

Το προκύπτον γλωσσικό μοντέλο παράγει διανύσματα που ήδη διαθέτουν πολλές από τις σχέσεις για τις οποίες έχουν γίνει γνωστές οι διανυσματικές αναπαραστάσεις λέξεων, π.χ. οι χώρες συγκεντρώνονται κοντά μεταξύ τους και συντακτικά παρόμοιες λέξεις καταλαμβάνουν παρόμοιες θέσεις στον διανυσματικό χώρο. Ενώ ο στόχος κατάταξής τους εξαλείφει την πολυπλοκότητα της συνάρτησης softmax, διατηρούν το ενδιάμεσο πλήρως συνδεδεμένο κρυφό στρώμα (fully-connected hidden layer) των Bengio κ.ά., το οποίο αποτελεί μια άλλη πηγή δαπανηρών υπολογισμών. Εν μέρει λόγω αυτού, το

πλήρες μοντέλο τους εκπαιδεύεται για επτά εβδομάδες συνολικά με μέγεθος λεξιλογίου $|V|$ = 130000.

Word2Vec

Στην πρώτη τους εργασία, οι Mikolov κ.ά. [100] προτείνουν δύο αρχιτεκτονικές για την εκμάθηση διανυσματικών αναπαραστάσεων λέξεων που είναι υπολογιστικά λιγότερο δαπανηρές από τα προηγούμενα μοντέλα. Στη δεύτερη εργασία τους [104], βελτιώνουν αυτά τα μοντέλα χρησιμοποιώντας πρόσθετες στρατηγικές επιτυγχάνοντας καλύτερη ταχύτητα εκπαίδευσης και μεγαλύτερη ακρίβεια.

Αυτές οι αρχιτεκτονικές προσφέρουν δύο κύρια πλεονεκτήματα σε σχέση με το μοντέλο C&W και το μοντέλο γλώσσας Bengio:

- Καταργούν το ακριβό κρυφό στρώμα.
- Επιτρέπουν στο γλωσσικό μοντέλο να λάβει υπόψιν πρόσθετο συγκείμενο πλαίσιο.

Οι δύο αυτές αρχιτεκτονικές είναι:

Continuous Bag-of-Words (CBOW)

Ενώ τα παραπάνω γλωσσικά μοντέλα λαμβάνουν υπόψιν μόνο τις προηγούμενες λέξεις για τις προβλέψεις τους, καθώς αξιολογούνται με βάση την ικανότητά τους να προβλέπουν κάθε επόμενη λέξη στο σώμα κειμένου, ένα μοντέλο που στοχεύει απλώς να δημιουργήσει ακριβείς αναπαραστάσεις λέξεων δεν “υποφέρει” από αυτόν τον περιορισμό. Οι Mikolov κ.ά. [100] χρησιμοποίησαν έτσι και τις n λέξεις πριν και μετά τη λέξη-στόχο w_t για να την προβλέψει όπως απεικονίζεται στην εικόνα 7. Αυτό το ονόμασαν Continuous Bag-of-Words (CBOW), καθώς χρησιμοποιεί συνεχείς αναπαραστάσεις λέξεων των οποίων η σειρά δεν έχει σημασία.

Η αντικειμενική συνάρτηση του CBOW με τη σειρά της είναι ελαφρώς διαφορετική από το μοντέλο γλώσσας:

$$J_\theta = \frac{1}{T}\sum_{t=1}^{T} \log p(w_t \mid w_{t-n}, \ldots, w_{t-1}, w_{t+1}, \ldots, w_{t+n})$$

Αντί να τροφοδοτήσει n προηγούμενες λέξεις στο μοντέλο, το μοντέλο λαμβάνει ένα παράθυρο n λέξεων γύρω από τη λέξη-στόχο w_t σε κάθε χρονικό βήμα t.

Skip-gram

Ενώ το CBOW μπορεί να θεωρηθεί ως προγνωστικό γλωσσικό μοντέλο, το skip-gram ανατρέπει τον στόχο του γλωσσικού μοντέλου: Αντί να χρησιμοποιεί τις γύρω λέξεις για να προβλέψει την κεντρική λέξη όπως με το CBOW, το skip-gram χρησιμοποιεί την κεντρική λέξη για να προβλέψει τις γύρω λέξεις όπως φαίνεται στην εικόνα 7.

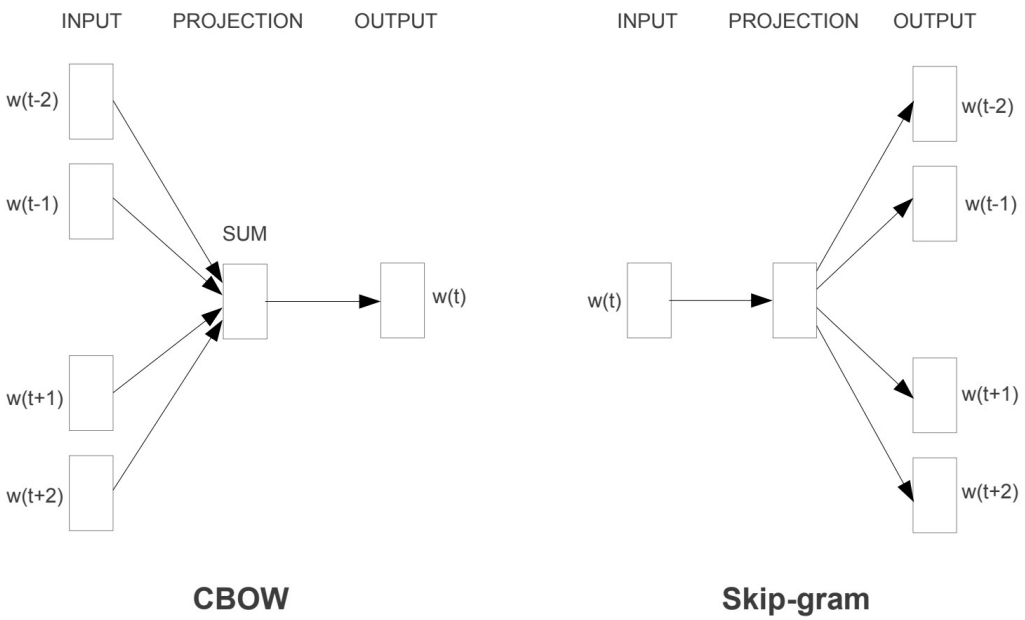
Ο στόχος skip-gram αθροίζει έτσι τις πιθανότητες καταγραφής των γύρω n λέξεων στα αριστερά και στα δεξιά της λέξης στόχου w_t για να παράγει την ακόλουθη αντικειμενική συνάρτηση ως στόχο:

$$J_\theta = \frac{1}{T}\sum_{t=1}^{T} \sum_{-n \le j \le n, \neq 0} \log p\left(w_{t+j} \mid w_t\right)$$

Οπότε η softmax συνάρτηση γίνεται:

$$p\left(w_{t+j} \mid w_t\right) = \frac{\exp\left(v_{w_t}^{\top} v'_{w_{t+j}}\right)}{\sum_{w_i \varepsilon V} \exp\left(v_{w_t}^{\top} 1'_{w_i}\right)}$$

καθώς η h είναι απλώς η αναπαράσταση v_{w_t} της λέξης εισόδου w_t.



Εικόνα 7: Αρχιτεκτονικές μοντέλων CBOW και Skip-gram

GloVe

Το GloVe προέρχεται από τις λέξεις “Global Vectors”. Εν συντομία, το GloVe δημιουργεί αναπαραστάσεις λέξεων με τρόπο ώστε ένας συνδυασμός διανυσμάτων λέξεων να σχετίζεται άμεσα με την πιθανότητα συν-εμφάνισης αυτών των λέξεων στο σώμα κειμένου. Συγκεκριμένα, οι συγγραφείς του GloVe δείχνουν ότι η αναλογία των πιθανοτήτων συν-εμφάνισης δύο λέξεων (και όχι οι ίδιες οι πιθανότητες συν-εμφάνισής τους) είναι αυτό που περιέχει αρκετές πληροφορίες και στοχεύει να κωδικοποιήσει αυτές τις πληροφορίες ως διανυσματικές διαφορές.

Για να το πετύχουν αυτό, προτείνουν έναν σταθμισμένο στόχο ελαχίστων τετραγώνων J που στοχεύει άμεσα στην ελαχιστοποίηση της διαφοράς μεταξύ του εσωτερικού γινομένου των διανυσμάτων δύο λέξεων και του λογαρίθμου του αριθμού των συν-εμφανίσεών τους:

$$J_\theta = \sum_{V}^{i,j=1} f(X_{ij})\left(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij}\right)^2$$

Όπου w_i και b_i είναι η λέξη διάνυσμα και η μεροληψία (bias) αντίστοιχα της λέξης i, $\tilde{w}_j$ και $\tilde{b}_j$ είναι το διάνυσμα λέξεων πλαισίου και η μεροληψία της λέξης j, X_{ij} είναι ο αριθμός των φορών που εμφανίζεται η λέξη i στο πλαίσιο της λέξης j και f είναι μια συνάρτηση στάθμισης που εκχωρεί σχετικά χαμηλότερο βάρος σε σπάνιες και συχνές συν-εμφανιζόμενες λέξεις.

FastText

Η ιδέα είναι πολύ παρόμοια με το Word2Vec αλλά με μια σημαντική διαφορά. Αντί να χρησιμοποιεί λέξεις για τη δημιουργία διανυσματικών αναπαραστάσεων λέξεων, το FastText πηγαίνει ένα επίπεδο βαθύτερα. Αυτό το βαθύτερο επίπεδο αποτελείται από μέρος λέξεων και χαρακτήρων. Κατά μία έννοια, μια λέξη γίνεται το πλαίσιό της. Το μοντέλο λοιπόν χρησιμοποιεί χαρακτήρες αντί για λέξεις.

Οι αναπαραστάσεις λέξεων που παράγονται από το FastText μοιάζουν πολύ με αυτές που παρέχονται από το Word2Vec. Ωστόσο, δεν υπολογίζονται άμεσα. Αντίθετα, είναι ένας συνδυασμός αναπαραστάσεων χαμηλότερου επιπέδου.

Υπάρχουν δύο βασικά πλεονεκτήματα σε αυτή την προσέγγιση. Πρώτον, η γενίκευση είναι δυνατή εφόσον στην περίπτωση που εμφανιστούν νέες λέξεις, αυτές έχουν τους ίδιους χαρακτήρες με τις ήδη υπάρχουσες. Δεύτερον, χρειάζονται λιγότερα δεδομένα εκπαίδευσης, καθώς μπορούν να εξαχθούν πολύ περισσότερες πληροφορίες από κάθε κομμάτι κειμένου. Γι' αυτό υπάρχουν προ-εκπαιδευμένα μοντέλα FastText για πολύ περισσότερες γλώσσες από ό,τι για κάθε άλλο αλγόριθμο διανυσματικής αναπαράστασης λέξεων.

3.4.4 Τεχνικές Διανυσματικής Αναπαράστασης Λέξεων με βάση το συγκείμενο πλαίσιο (Contextual / Dynamic Embeddings)

Για την διανυσματική αναπαράσταση λέξεων με χρήση μεθόδων όπως το Word2Vec, το GloVe και το FastText, υπάρχουν δύο βασικά προβλήματα. Πρώτον, υπάρχει πάντα η ίδια αναπαράσταση για έναν τύπο λέξης, ανεξάρτητα από το συγκείμενο πλαίσιο στο οποίο εμφανίζεται μια λέξη που λαμβάνεται, και μπορεί να χρειάζεται μια πολύ λεπτή αποσαφήνιση της έννοιας της κάθε λέξης.

Δεύτερον, έχουμε μόνο μία αναπαράσταση για μια λέξη, αλλά οι λέξεις έχουν διαφορετικές πτυχές, συμπεριλαμβανομένης της σημασιολογίας, της συντακτικής συμπεριφοράς και της συνδήλωσης. Για παράδειγμα, κατά κάποια έννοια, η σημασιολογία των λέξεων: «φθάνω» και «άφιξη» είναι σχεδόν ίδια, αλλά είναι διαφορετικά μέρη του λόγου: το «φθάνω» είναι ρήμα και το «άφιξη» είναι ουσιαστικό,

επομένως μπορούν να εμφανίζονται σε αρκετά διαφορετικά σημεία στον διανυσματικό χώρο. Υπάρχει η ανάγκη να διακρίνονται οι λέξεις και στις παραπάνω βάσεις. Αυτά είναι τα είδη προβλημάτων τα οποία τεχνικές Διανυσματικής Αναπαράστασης Λέξεων με βάση το συγκείμενο πλαίσιο επιχειρούν να επιλύσουν.

Οι Διανυσματικές Αναπαραστάσεις Λέξεων με βάση το συγκείμενο πλαίσιο έχουν πετύχει πρωτοποριακή απόδοση σε ένα ευρύ φάσμα εργασιών επεξεργασίας φυσικής γλώσσας. Οι αναπαραστάσεις με βάση τα συμφραζόμενα εκχωρούν σε κάθε λέξη μια αναπαράσταση με βάση το πλαίσιό της, χρησιμοποιώντας ακολουθίες λέξεων, αποτυπώνοντας έτσι τις χρήσεις λέξεων σε διάφορα περιβάλλοντα και δύνανται να κωδικοποιούν την γνώση αυτή με τέτοιο τρόπο ώστε να μπορεί να μεταφέρεται σε διάφορες γλώσσες.

Η εργασία των Dai και Le [105] είναι η πρώτη που γνωρίζουμε ότι χρησιμοποιεί μοντελοποίηση γλώσσας μαζί με έναν αυτόματο κωδικοποιητή ακολουθίας λέξεων για τη βελτίωση της εκμάθησης ακολουθιών με επαναλαμβανόμενα δίκτυα. Έτσι, μπορεί να θεωρηθεί ως πρόδρομος των σύγχρονων μεθόδων διανυσματικής αναπαράστασης γλώσσας με βάση τα συμφραζόμενα.

TagLM

Μία από τις πρώτες προσπάθειες για αναπαράσταση λέξεων με βάση τα συγκείμενο πλαίσιο ήταν μια εργασία των Peters κ.ά.[106], του Allen Institute, όπου δημιουργήθηκε το TagLM το οποίο είναι ο προκάτοχος των σύγχρονων εκδοχών αναπαραστάσεων λέξεων. Σε αυτό εκπαιδεύτηκε ένα συμβατικό μοντέλο διανυσματικής αναπαράστασης λέξεων όπως το Word2Vec ταυτόχρονα με ένα Αμφίδρομο Δίκτυο Μακράς Βραχύχρονης Μνήμης (BiLSTM). Έγινε προσθήκη ετικετών ακολουθίας που στοχεύει στην πρόβλεψη μιας γλωσσικής ετικέτας για κάθε λέξη, όπως η επισήμανση μέρους του λόγου (Part of Speech Tagging - POS) και η τμηματοποίηση κειμένου και η Αναγνώριση Ονομαστικών Οντοτήτων (Name Entity Recognition - NER).

ELMo

Το ELMo (Embeddings from Language Models) όμοια με το TagLM, παρουσιάστηκε από τους Peters κ.ά. [82], του Allen Institute. Η διαφορά μεταξύ TagLM και ELMo είναι πολύ μικρή και βρίσκεται στις λεπτομέρειες. Το ELMo είναι ένα μοντέλο βαθιάς διανυσματικής αναπαράστασης λέξεων το οποίο για να δημιουργήσει την κωδικοποίηση χρησιμοποιεί εκτός από την σύνταξη (POS) και την σημασιολογία (NER) της λέξης και όλα τα γλωσσικά περιβάλλοντα (έτσι ώστε να λαμβάνεται υπόψιν και η πολυσημία).

Τα διανύσματα προέρχονται από BiLSTM μοντέλο, όπως στο TagLM, το οποίο όμως έχει εκπαιδευτεί σε τεράστια δεδομένα κειμένου σε μια προσπάθεια να βρεθεί ένα συμπαγές μοντέλο γλώσσας που θα ήταν εύκολο να χρησιμοποιηθεί σε άλλες εργασίες, ακόμα κι αν δεν διατίθενται ισχυροί υπολογιστές.

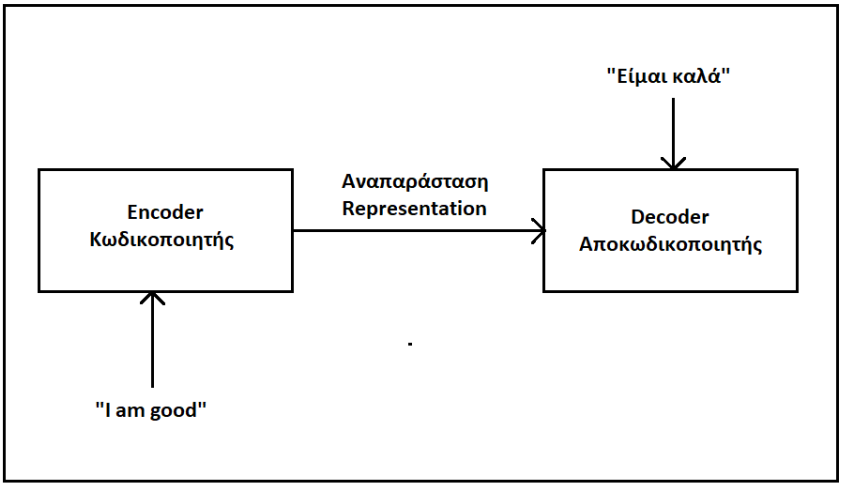
Σε αντίθεση με προηγούμενες προσεγγίσεις διανυσμάτων με συμφραζόμενα, το μοντέλο ELMo είναι βαθύ με την έννοια ότι οι αναπαραστάσεις είναι συναρτήσεις όλων των επιπέδων του γλωσσικού μοντέλου. Πιο αναλυτικά, το ELMo μαθαίνει ένα γραμμικό συνδυασμό των διανυσμάτων της κάθε λέξης που δίνεται σαν είσοδος κάτι που βελτιστοποιεί την απόδοση σε σχέση με τη χρήση μόνο του τελευταίου επιπέδου του LSTM. Συνδυάζοντας έτσι τις εσωτερικές καταστάσεις με τον παραπάνω τρόπο, το μοντέλο EMLo δημιουργεί πολύ πλούσιες λεξικές αναπαραστάσεις. Επίσης, τα ανώτερα επίπεδα του LSTM συγκρατούν πληροφορία σε σχέση με τα συμφραζόμενα, ενώ

χαμηλότερα επίπεδα μοντελοποιούν συντακτικά στοιχεία των λέξεων. Αυτή η διαπίστωση, η οποία έχει τεκμηριωθεί μέσω πειραμάτων είναι ιδιαιτέρως χρήσιμη γιατί επιτρέπει στα ήδη προ-εκπαιδευμένα μοντέλα να εστιάσουν στα επίπεδα του μοντέλου για το ειδικό πρόβλημα που πρέπει να επιλύσουν την εκάστοτε φορά.

BERT

Το μοντέλο BERT, “Bidirectional Encoder Representations from Transformers”, δημιουργήθηκε και δημοσιεύτηκε από την Google και τους Devlin κ.ά. [83], και αποτέλεσε αμέσως την state-of-the-art τεχνική στην ΕΦΓ για την δημιουργία embeddings. Ένα εκπαιδευμένο BERT μοντέλο παίρνει ως είσοδο μία πρόταση και δίνει ως έξοδο διανύσματα για κάθε λέξη της πρότασης, που περιλαμβάνουν πληροφορία για την θέση τους και τις περιβάλλουσες λέξεις. Το μοντέλο BERT κάνει χρήση του μοντέλου Transformer για την δημιουργία διανυσματικών αναπαραστάσεων λέξεων που λαμβάνουν υπόψη ολόκληρο το συγκείμενο πλαίσιο μιας λέξης.

Το μοντέλο Transformer εισήχθη από την Google στην εργασία “Attention is All You Need” το 2017 [107]. Είναι μία καινοτόμα αρχιτεκτονική που στοχεύει στο να λύσει sequence-to-sequence εργασίες, ενώ χειρίζεται με ευκολία τις εξαρτήσεις μεγάλης εμβέλειας. Τα μοντέλα sequence-tosequence (Seq2Seq) αφορούν την μετατροπή ακολουθιών (sequences) από έναν τομέα (π.χ. προτάσεις στα Αγγλικά) σε ακολουθίες σε έναν άλλο τομέα (π.χ. οι ίδιες προτάσεις μεταφρασμένες στα Ελληνικά). Ο Transformer περιλαμβάνει δύο διαφορετικούς μηχανισμούς: έναν κωδικοποιητή που “διαβάζει” το κείμενο εισόδου και μαθαίνει μια αναπαράσταση και έναν αποκωδικοποιητή που λαμβάνει ως είσοδο την αναπαράσταση που στέλνει ο κωδικοποιητής και παράγει μία πρόβλεψη για το task. Το μοντέλο Transformer, βασίζεται στην ιδέα του self-attention, δηλαδή δεν επεξεργάζεται μία ακολουθία εισόδου λέξη-λέξη, αλλά λαμβάνει ως είσοδο ολόκληρη την ακολουθία, και έτσι επεξεργάζεται τις λέξεις σε σχέση με όλες τις υπόλοιπες λέξεις σε μία πρόταση. Η λειτουργία του συνοψίζεται στο ακόλουθο σχήμα:



Εικόνα 8: Κωδικοποιητής και Αποκωδικοποιητής των Transformer

Το BERT φτιάχτηκε για να καταλαβαίνει την πρόθεση πίσω από τα αιτήματα αναζήτησης. Αυτό που κάνει το BERT καινοτόμο, είναι ότι πρόκειται για την πρώτη βαθιά αμφίδρομη, μη επιβλεπόμενη γλωσσική αναπαράσταση, προ-εκπαιδευμένη χρησιμοποιώντας μόνο απλό κείμενο (συγκεκριμένα εκπαιδεύτηκε στην Wikipedia). Μπορούμε να χρησιμοποιήσουμε το μοντέλο BERT για να εξάγουμε υψηλής ποιότητας

γλωσσικά χαρακτηριστικά από τα δεδομένα μας, ή διαφορετικά μπορούμε να το προσαρμόσουμε με ακρίβεια σε συγκεκριμένα tasks με δικά μας δεδομένα.

Συνθέσεις BERT

Οι ερευνητές του BERT παρουσίασαν το μοντέλο σε δύο τυπικές συνθέσεις:

- **BERT-base**: Αποτελείται από 12 κωδικοποιητές με τον καθένα να στοιβάζεται πάνω στον άλλο. Οι κωδικοποιητές συνήθως δηλώνονται με το αγγλικό γράμμα L. Όλοι οι κωδικοποιητές χρησιμοποιούν 12 attention heads, όπου καθένα αποτελείται από ένα σύνολο μητρώων. Τα attention heads δηλώνονται με το γράμμα A. Έρευνες έχουν δείξει ότι πολλά attention heads στους Transformers κωδικοποιούν σχέσεις συνάφειας που μπορούν να ερμηνευτούν από τον άνθρωπο (Clark κ.ά. [108]). Το πρόσθιας τροφοδότησης δίκτυο του κωδικοποιητή αποτελείται από 768 κρυφές μονάδες, οι οποίες δηλώνονται ως H. Συνεπώς, το μέγεθος της αναπαράστασης που λαμβάνεται από το BERT-base είναι 768. Το μοντέλο αυτό συνοπτικά χαρακτηρίζεται από τα ακόλουθα στοιχεία: L = 12, A = 12 και H = 768.
- **BERT-large**: Αυτή η σύνθεση αποτελείται από 24 κωδικοποιητές, οι οποίοι στοιβάζονται ο ένας πάνω στον άλλο. Παράλληλα, όλοι οι κωδικοποιητές χρησιμοποιούν 16 attention heads, ενώ το πρόσθιας τροφοδότησης δίκτυο του κωδικοποιητή αποτελείται από 1024 κρυφές μονάδες. Τέλος, το μέγεθος της αναπαράστασης που λαμβάνεται από το BERT-large είναι 1024. Τα χαρακτηριστικά του μοντέλου αυτού συνοψίζονται ως εξής: L = 24, A = 16 και H = 1024.

Εκτός από τις προηγούμενες δύο τυπικές συνθέσεις, μπορούμε επίσης να δημιουργήσουμε το μοντέλο BERT με άλλα διαφορετικά χαρακτηριστικά. Κάποια από αυτά εμφανίζονται ακολούθως:

Προ-εκπαιδεύοντας ένα μοντέλο BERT

Έστω ότι έχουμε ένα μοντέλο. Πρώτον, εκπαιδεύουμε το μοντέλο με ένα τεράστιο σύνολο δεδομένων για μια συγκεκριμένη εργασία και αποθηκεύουμε το εκπαιδευμένο μοντέλο. Τώρα, για μια νέα εργασία, αντί να ξεκινήσουμε ένα νέο μοντέλο με τυχαία βάρη, θα ξεκινήσουμε το μοντέλο με τα βάρη του ήδη εκπαιδευμένου μοντέλου μας, (προ-εκπαιδευμένο μοντέλο). Δηλαδή, δεδομένου ότι το μοντέλο είναι ήδη εκπαιδευμένο σε ένα τεράστιο σύνολο δεδομένων, αντί να εκπαιδεύουμε ένα νέο μοντέλο από το μηδέν για μια νέα εργασία, χρησιμοποιούμε το προ-εκπαιδευμένο μοντέλο και προσαρμόζουμε (βελτιώνουμε) τα βάρη του σύμφωνα με τη νέα εργασία. Αυτός είναι ένας τύπος μάθησης μεταφοράς.

Το μοντέλο BERT είναι προ-εκπαιδευμένο σε ένα τεράστιο σώμα κειμένου χρησιμοποιώντας δύο ενδιαφέρουσες τεχνικές, αυτή του Γλωσσικού Μοντέλου Απόκρυψης (Masked Language Model) και αυτή της Πρόβλεψης της Επόμενης Πρότασης (Next Sentence Prediction - NSP). Μετά την προ-εκπαίδευση, αποθηκεύουμε το προ-εκπαιδευμένο μοντέλο BERT. Για μια νέα εργασία, ας πούμε την απάντηση ερωτήσεων, αντί να εκπαιδεύσουμε το BERT από το μηδέν, θα χρησιμοποιήσουμε το προ-εκπαιδευμένο μοντέλο BERT και θα προσαρμόσουμε (βελτιώσουμε) τα βάρη του για τη νέα εργασία.

Το BERT είναι ένα γλωσσικό μοντέλο αυτόματης κωδικοποίησης, πράγμα που σημαίνει ότι διαβάζει την πρόταση και προς τις δύο κατευθύνσεις για να κάνει μια πρόβλεψη. Μέσω της διεργασίας του Γλωσσικού Μοντέλου Απόκρυψης, σε μια δεδομένη πρόταση,

αποκρύπτεται τυχαία το 15% των λέξεων και εκπαιδεύεται το δίκτυο για να προβλέψει τις κρυμμένες λέξεις. Για την πρόβλεψη αυτών, το μοντέλο διαβάζει την πρόταση και προς τις δύο κατευθύνσεις και προσπαθεί να προβλέψει τις κρυμμένες λέξεις.

Η Πρόβλεψη της Επόμενης Πρότασης (NSP) είναι μια ενδιαφέρουσα στρατηγική που χρησιμοποιείται για την εκπαίδευση του μοντέλου BERT. Το NSP είναι μια διεργασία δυαδικής ταξινόμησης. Στην διεργασία αυτή, τροφοδοτούμε δύο προτάσεις στο BERT μοντέλο και αυτό πρέπει να προβλέψει εάν η δεύτερη πρόταση είναι η συνέχεια (επόμενη πρόταση) της πρώτης πρότασης. Η εκτέλεση της διεργασίας Πρόβλεψης της Επόμενης Πρότασης, μπορεί να βοηθήσει το μοντέλο να κατανοήσει τη σχέση μεταξύ των δύο προτάσεων. Η κατανόηση της σχέσης μεταξύ δύο προτάσεων είναι χρήσιμη στην περίπτωση πολλών εργασιών, όπως η απάντηση ερωτήσεων και η δημιουργία κειμένου.

Βέβαια, η διαδικασία προ-εκπαίδευσης του BERT από το μηδέν είναι υπολογιστικά ακριβή. Για το λόγο αυτό δίνεται η δυνατότητα χρήσης ενός ήδη προ-εκπαιδευμένου μοντέλου. Η Google παρέχει στους χρήστες ένα σύνολο από τέτοια μοντέλα, τα οποία μπορούν να βρεθούν μέσω του GitHub της Google Research στη σελίδα: “https://github.com/google-research/bert”. Τα προ-εκπαιδευμένα αυτά μοντέλα μπορούν να χρησιμοποιηθούν με τους ακόλουθους δύο τρόπους:

- Ως εργαλείο εξαγωγής διανυσματικών αναπαραστάσεων των λέξεων.
- Ρυθμίζοντας το προ-εκπαιδευμένο μοντέλο BERT σε μεταγενέστερες εργασίες, όπως η ταξινόμηση κειμένου, η απάντηση ερωτήσεων και άλλα.

Εξαγωγή διανυσματικών αναπαραστάσεων των λέξεων από προ-εκπαιδευμένο μοντέλο BERT

Στην παρούσα έρευνα ο τρόπος που μας ενδιαφέρει και, συνεπώς, θα αναλύσουμε είναι αυτός της εξαγωγής διανυσματικών αναπαραστάσεων των λέξεων. Για να γίνει αυτό εφικτό, γίνεται αρχικά διαχωρισμός σε tokens των φράσεων του κειμένου, τα οποία, στην συνέχεια, τροφοδοτούν το προ-εκπαιδευμένο BERT μοντέλο. Το αποτέλεσμα που επιστρέφεται είναι οι διανυσματικές αναπαραστάσεις που προέκυψαν για κάθε token. Εκτός από τη λήψη της διανυσματικής αναπαράστασης σε επίπεδο token (επίπεδο λέξης), μπορούμε επίσης να αποκτήσουμε την αναπαράσταση σε επίπεδο φράσης.

Αρχικά, πρέπει να μετατρέψουμε το κείμενο σε διάνυσμα. Μπορούν να χρησιμοποιηθούν μέθοδοι όπως TF-IDF, Word2Vec και άλλες, αλλά στην συγκεκριμένη περίπτωση μελετάται αυτή του BERT. Συγκεκριμένα, πριν από την τροφοδοσία της εισόδου στο προ-εκπαιδευμένο μοντέλο BERT, μετατρέπουμε την είσοδο σε τριών ειδών διαφορετικές αναπαραστάσεις:

- Αναπαράσταση των token
- Αναπαράσταση των προτάσεων
- Αναπαράσταση της θέσης

Αναπαράσταση των token

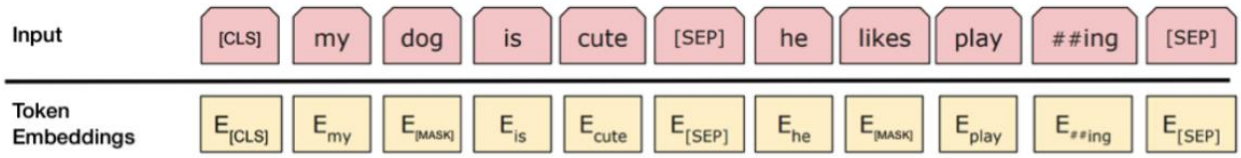
Το BERT χρησιμοποιεί έναν ειδικό τύπο tokenizer που ονομάζεται WordPiece. Ο WordPiece tokenizer ελέγχει κατά πόσο η λέξη υπάρχει στο λεξιλόγιό του BERT. Εάν η λέξη υπάρχει στο λεξιλόγιο, τότε αντιμετωπίζεται ως token. Εάν η λέξη δεν υπάρχει στο λεξιλόγιο, τότε χωρίζεται σε υπο-λέξεις και ελέγχεται αν η υπο-λέξη υπάρχει στο λεξιλόγιο. Εάν αυτή υπάρχει στο λεξιλόγιο, τότε χρησιμοποιείται ως token. Αλλά αν ούτε

η υπο-λέξη υπάρχει στο λεξιλόγιο, τότε πάλι ξαναχωρίζεται και ελέγχεται εάν υπάρχει στο λεξιλόγιο. Εάν υπάρχει στο λεξιλόγιο, τότε χρησιμοποιείται ως token, διαφορετικά χωρίζεται ξανά. Με αυτόν τον τρόπο, συνεχίζεται ο διαχωρισμός, ελέγχοντας την ύπαρξη της λέξης στο λεξιλόγιο μέχρι την κατάληξη σε μεμονωμένους χαρακτήρες αν αυτή δεν υπάρχει. Αυτή η τεχνική είναι αποτελεσματική στο χειρισμό των λέξεων εκτός λεξιλογίου (στα αγγλικά out-of-vocabulary ή OOV).

Το μέγεθος του λεξιλογίου BERT είναι 30K tokens. Εάν μια λέξη ανήκει σε αυτά τα 30K tokens, τότε τη χρησιμοποιούμε ως token. Διαφορετικά, χωρίζουμε τη λέξη σε δευτερεύουσες λέξεις και ελέγχεται αν η υπο-λέξη ανήκει στο λεξιλόγιο κ.ο.κ, όπως αναφέρθηκε παραπάνω. Παράδειγμα μίας τέτοιας περίπτωσης είναι αυτή που ακολουθεί. Στο συγκεκριμένο παράδειγμα η λέξη pre-training δεν ανήκει στο λεξιλόγιο του BERT και τα tokens που προκύπτουν είναι όπως φαίνονται παρακάτω:

Ταυτόχρονα, στην περίπτωση ύπαρξης περισσότερων από μία προτάσεων σε ένα κείμενο εισάγεται το token [SEP], ενώ κάθε φορά το πρώτο token της πρώτης πρότασης του κειμένου είναι το [CLS]. Το token [CLS] χρησιμοποιείται σε προβλήματα ταξινόμησης, ενώ και το token [SEP] χρησιμοποιείται για να δείξει το τέλος κάθε πρότασης. Παράδειγμα μίας τέτοιας περίπτωσης είναι το ακόλουθο.

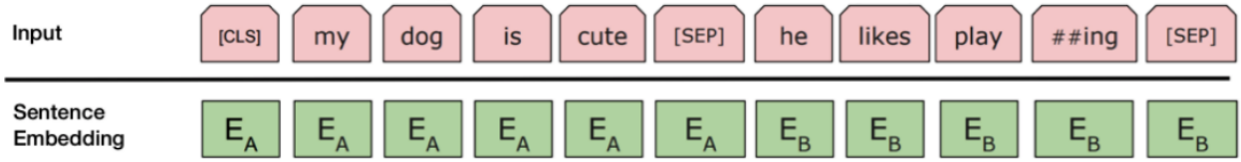
Πριν την εισαγωγή των token στο BERT, γίνεται μετατροπή αυτών σε διανυσματικές αναπαραστάσεις. Οι αναπαραστάσεις αυτές αποτελούν τις αναπαραστάσεις των token. Σημειώνεται ότι η τιμή των διανυσμάτων αυτών ρυθμίζεται κατά τη διάρκεια της εκπαίδευσης. Όπως φαίνεται στην παρακάτω εικόνα, έχουμε αναπαράσταση για όλα τα token, δηλαδή, το E[CLS] υποδηλώνει την αναπαράσταση του token [CLS], όπως και το E_dog που υποδηλώνει την αναπαράσταση της λέξης dog κ.ο.κ.:



Εικόνα 9: Αναπαράσταση των token

Αναπαράσταση των προτάσεων

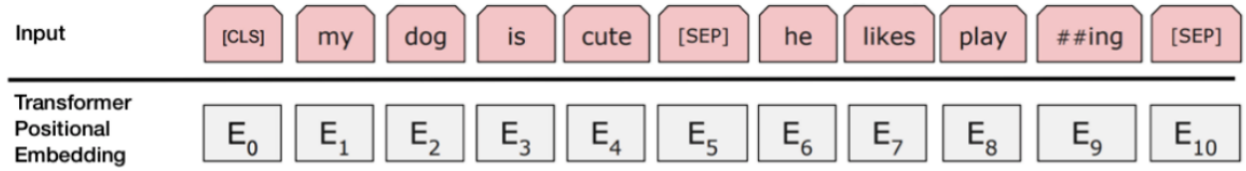
Στην συνέχεια ακολουθεί η αναπαράσταση των προτάσεων. Η συγκεκριμένη αναπαράσταση χρησιμοποιείται για τη διάκριση μεταξύ δύο δεδομένων προτάσεων. Αν πάρουμε ως παράδειγμα το τελευταίο που αναφέρθηκε παραπάνω, εκτός από το token [SEP], πρέπει να δοθεί ένα είδος δείκτη στο μοντέλο μας για να γίνεται διάκριση μεταξύ των δύο προτάσεων. Για να γίνει αυτό, τροφοδοτούμε τα tokens εισόδου στο επίπεδο αναπαράστασης των προτάσεων. Το επίπεδο αυτό επιστρέφει μόνο μία από τις δύο αναπαραστάσεις, E_A ή E_B, ως έξοδο. Αυτό σημαίνει ότι εάν το token εισόδου ανήκει στην πρώτη πρόταση, τότε αυτό θα αντιστοιχιστεί στην αναπαράσταση E_A και εάν το token ανήκει στη δεύτερη πρόταση, τότε θα αντιστοιχιστεί στην αναπαράσταση E_B. Όπως φαίνεται στο παρακάτω σχήμα, όλα τα tokens από την πρώτη πρόταση αντιστοιχίζονται στην αναπαράσταση E_A και αυτά της δεύτερης αντιστοιχίζονται στην αναπαράσταση E_B:



Εικόνα 10: Αναπαράσταση των προτάσεων

Αναπαράσταση της θέσης

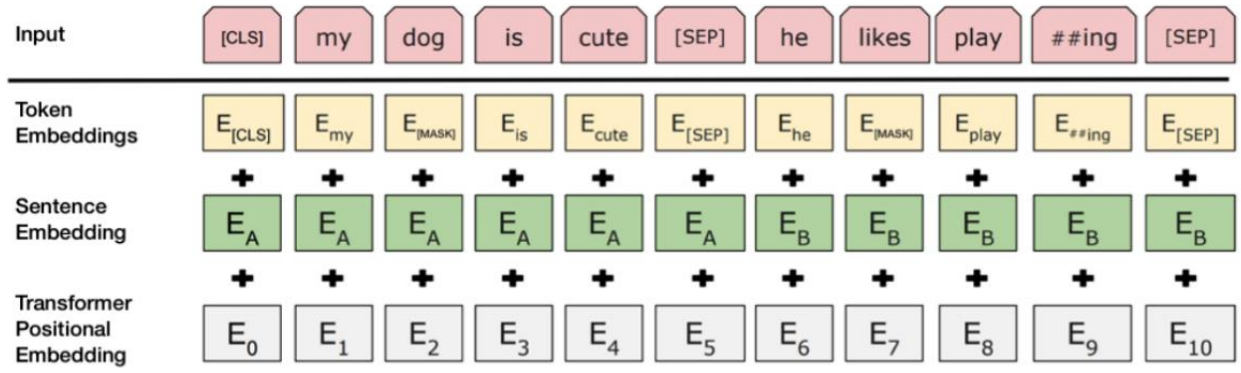
Η αναπαράσταση αυτή είναι απαραίτητη λόγω της ιδιότητας των Transformer να επεξεργάζεται όλες τις λέξεις παράλληλα, επομένως καθίσταται αναγκαίο να παρέχεται η γνώση σχετικά με τη σειρά των λέξεων και, συνεπώς, η κωδικοποίηση της θέσης των λέξεων. Η πληροφορία της θέσης των token σε μία φράση, δημιουργεί μια αναπαράσταση η οποία τροφοδοτεί το μοντέλο BERT με το επίπεδο αυτό να ονομάζεται αναπαράσταση της θέσης. Στο παράδειγμα που ακολουθεί το E_0 υποδηλώνει την αναπαράσταση της θέσης του token [CLS], το E_1 την αναπαράσταση της θέσης του token dog, κ.ο.κ.:



Εικόνα 11: Αναπαράσταση της θέσης

Τελική Αναπαράσταση

Για τη παραγωγή των διανυσματικών αναπαραστάσεων των λέξεων από το μοντέλο BERT, έγινε μετατροπή των φράσεων εισόδου σε tokens και εισαγωγή αυτών στα επίπεδα αναπαράστασης των tokens, των προτάσεων και της θέσης. Οι αναπαραστάσεις που δημιουργήθηκαν από τα τρία επίπεδα προστίθενται και τροφοδοτούν το BERT μοντέλο, όπως φαίνεται ακολούθως:



Εικόνα 12: Τελική Αναπαράσταση

4. ΜΕΘΟΔΟΛΟΓΙΑ ΚΑΙ ΥΛΟΠΟΙΗΣΗ ΣΥΣΤΗΜΑΤΟΣ

Για την υλοποίηση της έρευνας έγινε χρήση πλήθους βιβλιοθηκών της **Python**, εκ των οποίων οι σημαντικότερες αναφέρονται παρακάτω.

Αρχικά, το σύνολο των δεδομένων που χρησιμοποιήθηκε δημιουργήθηκε με χρήση του Twitter API μέσω της βιβλιοθήκης **Tweepy** της γλώσσας **Python**.

Επίσης, γίνεται λόγος για τη βιβλιοθήκη **Pandas**, η οποία χρησιμοποιείται ευρέως για "καθαρισμό", μετασχηματισμό και ανάλυση των δεδομένων. Παράλληλα, η βιβλιοθήκη **NumPy** η οποία παρέχει μαθηματικές συναρτήσεις υψηλού επιπέδου και χρήσιμες λειτουργίες σε πράξεις που αφορούν συστοιχίες και μητρώα.

Ταυτόχρονα, ακολουθεί μίας από τις σημαντικότερες βιβλιοθήκες της παρούσας έρευνας, η οποία είναι η **NLTK** (Natural Language ToolKit) η οποία χρησιμοποιείται για προ-επεξεργασία δεδομένων που περιέχουν κείμενο. Υποστηρίζει διάφορες λειτουργίες όπως: κατηγοριοποίηση, αναγνώριση λέξεων, αποκοπή καταλήξεων, απόδοση εννοιολογικής ετικέτας κ.α.

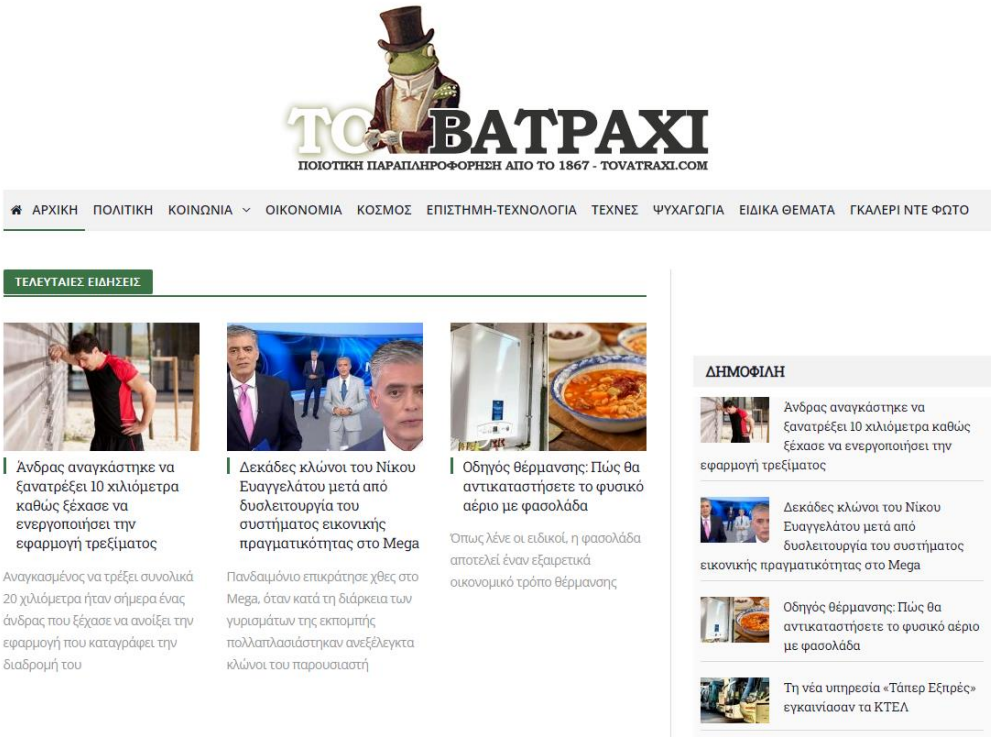
Τέλος, η λίστα με τις πιο σημαντικές βιβλιοθήκες ολοκληρώνεται με το **TensorFlow**. Το TensorFlow είναι μία βιβλιοθήκη ανοιχτού κώδικα που χρησιμοποιείται για τη δημιουργία μοντέλων μηχανικής μάθησης σε μεγάλη κλίμακα. Παρέχει την απαραίτητη υποδομή και υλικό, καθιστώντας την μία από τις κυρίαρχες βιβλιοθήκες στον τομέα της βαθιάς μάθησης. Χρησιμοποιείται κυρίως σε προβλήματα Κατηγοριοποίησης και Παλινδρόμησης. Μία άλλη βιβλιοθήκη για δημιουργία υψηλού επιπέδου νευρωνικών δικτύων είναι το **Keras**. Έχει φτιαχτεί πάνω στην βιβλιοθήκη TensorFlow και θεωρείται περισσότερο φιλική προς το χρήστη.

Αξίζει να σημειωθεί πως για τη δημιουργία των γραφημάτων χρησιμοποιήθηκε κατά κόρον η βιβλιοθήκη **Plotly**, ενώ σε μικρότερο βαθμό η **Seaborn**.

4.1 Δημιουργία συνόλου δεδομένων (Dataset Construction)

Στην προκειμένη περίπτωση, συλλέξαμε με το χέρι, μέσω του Twitter API, τους τίτλους ειδήσεων / tweets από 4 διαφορετικούς ελληνικούς ειδησεογραφικούς ιστότοπους: HuffPost Greece, CNN Greece, Το Κουλούρι, Το Βατράχι, από τους οποίους οι δύο τελευταίοι επικεντρώνονται σε σατιρικές ειδήσεις. Δεν κατέστη δυνατό να βρούμε περισσότερους ελληνικούς ιστότοπους σατιρικών ειδήσεων με αρκετά δείγματα δεδομένων, οπότε αποφασίσαμε να χρησιμοποιήσουμε τον ίδιο αριθμό ιστότοπων κανονικών ειδήσεων, προκειμένου να έχουμε ομοιόμορφη κατανομή των δεδομένων.

Εικόνα 13: Αρχική σελίδα της ιστοσελίδας "Το Κουλούρι"



Εικόνα 14: Αρχική σελίδα της ιστοσελίδας "Το Βατράχι"

Λόγω του περιορισμού που έχει υποβάλει το *Twitter* στα 3250 tweets ανά id, συγκεντρώσαμε 6500 δείγματα κανονικών ειδήσεων και 6500 δείγματα σατιρικών ειδήσεων. Το νούμερο αυτό είναι αρκετά μικρό, αλλά ένα τέτοιο ελληνικό σύνολο δεδομένων που να περιλαμβάνει πολλές χιλιάδες tweets, πιθανότατα, θα δημιουργηθεί και θα γίνει διαθέσιμο σε βάθος χρόνου από τώρα.

Για την δημιουργία του dataset, η λίστα με τα tweets προστέθηκε στην στήλη headline και στην στήλη is_sarcastic προστέθηκε το γνώρισμα που απασχολεί την παρούσα προσέγγιση. Οι τιμές που μπορεί να πάρει είναι οι ακόλουθες: 0 για μη σαρκαστικά tweets και 1 για σαρκαστικά αντίστοιχα.

Στην παρακάτω εικόνα φαίνεται μέρος του σετ δεδομένων που χρησιμοποιήθηκε:

	headline	is_sarcastic
0	Πέντε μήνες νωρίτερα δόθηκε σε κυκλοφορία η παράκαμψη Αμφιλοχίας	0
1	Είδηση σοκ: Με ΕΝΦΙΑ χρεώθηκε ηλικιωμένος που έμεινε πολλή ώρα ακίνητος	1
2	Ενήμερη η χώρα για τη βροχή μετά από ανάρτηση χρήστη του Facebook	1
3	Μάνος Λοΐζος - ΜΕΤΑ: Μια συναυλία-γιορτή για τα 35 χρόνια του Αθήνα 9.84	0
4	Η Warner ξόδεψε 100 εκατ. για την ταινία «Batgirl» που τελικά δεν θα δούμε	0

Εικόνα 15: Στιγμιότυπο από σετ δεδομένων

Ακολουθούν παραδείγματα σαρκαστικών τίτλων ειδήσεων του Twitter και από τις δύο πηγές σατιρικών ειδήσεων:

“Το Κουλούρι”

- «*Group chat με όλες τις περιοχές που καίγονται αυτή τη στιγμή έφτιαξε το 112.*»
- «*Νούμερο ένα αιτία θανάτου των υπαλλήλων της Adobe είναι οι εικόνες με καλημέρες στο Facebook.*»
- «*Ακόμη πιο δυστυχισμένος αλλά μαυρισμένος επέστρεψε σήμερα στη δουλειά 37χρονος ιδιωτικός υπάλληλος.*»
- «*Σούπερ μάρκετ βράβευσε ηλικιωμένο που δεν ζήτησε να ανοίξει άλλο ένα ταμείο.*»
- «*Ενθουσιασμένος με τη ρύθμιση για τις 120 δόσεις, οφειλέτης που θα «πιστολιάσει» ήδη από τη δεύτερη δόση.*»
- «*Ντυμένος μετροπόντικας θα δώσει το «παρών» στα εγκαίνια του μετρό στον Πειραιά ο Κυριάκος Μητσοτάκης.*»
- «*Πιο σιχαμεροί και από μας αυτοί που χτυπάνε γυναίκες, επιβεβαιώνουν οι κατσαρίδες.*»
- «*Τι ώρα θα συμβεί το τρακάρισμα σήμερα στον Κηφισό. Μείνετε ενημερωμένοι.*»
- «*Και επίσημα αθάνατη μετά το θάνατο της βασίλισσας Ελισάβετ η Ζωζώ Σαπουντζάκη.*»
- «*Στην ύλη των Μαθηματικών Κατεύθυνσης θα προστεθεί ο τρόπος ανακοίνωσης των κρουσμάτων στην Ελλάδα.*»

“Το Βατράχι”

- «*30η Ιανουαρίου 2019: «Χρόνια Πολλά, πολύχρονος», η αγαπημένη ευχή των ψηφοφόρων ΚΚΕ (μ-λ), Μ-Λ ΚΚΕ.*»
- «*Από το 2014: Τουρκία: Την πρόσβαση στον Ερντογάν «μπλοκάρουν» Twitter, YouTube και Facebook.*»

- «*Ούτε κλαίγοντας ούτε γελώντας πέρασε και τη φετινή επέτειο της 11ης Σεπτεμβρίου η Αλέκα Παπαρήγα*.»
- «*Χιλιάδες Έλληνες θα χρειαστεί να αφαιρεθούν χειρουργικά από τον καναπέ τους μετά τη λήξη της καραντίνας, λένε οι ειδικοί.*»
- «*Παρουσία του ίδιου του Θεού μετά από αίτημα της Νίκης Κεραμέως πραγματοποιήθηκε ο αγιασμός στα σχολεία*.»
- «*Ένα χρόνο πριν: Πολιτισμός: Αντάλλαξαν μάσκες οι Ευρωπαίοι ηγέτες μετά την ολοκλήρωση της Συνόδου Κορυφής*.»
- «*Ιστορική συμφωνία Αθήνας και Θεσσαλονίκης για το πώς θα λέγονται τα σουβλάκια*.»
- «*Οδηγός θέρμανσης: Πώς θα αντικαταστήσετε το φυσικό αέριο με φασολάδα*.»
- «*Πιο ευχάριστη η αφαίρεση φρονιμίτη χωρίς αναισθητικό από το να μπεις σε Τζάμπο λίγο πριν την έναρξη της σχολικής χρονιάς, αναφέρει έρευνα*.»
- «*Αυτοκίνητο με μηδενική κατανάλωση βενζίνης κατασκεύασαν Έλληνες ερευνητές*.»

Ακολουθούν παραδείγματα μη σαρκαστικών τίτλων ειδήσεων του Twitter και από τις δύο πηγές κανονικών ειδήσεων:

“HuffPost Greece”

- «*Βασίλισσα Μαργαρίτα Β’: Η μακροβιότερη μονάρχης του κόσμου μετά τον θάνατο της Ελισάβετ*.»
- «*Παρέμβαση υπουργείου Ανάπτυξης για την τιμή σε καυσόξυλα και πέλετ*.»
- «*Ουκρανία: Η Ρωσία χτυπά σταθμούς παραγωγής ενέργειας μετά την αντεπίθεση του Κιέβου*.»
- «*Μέτρα στήριξης: Ενίσχυση σε έξι πυλώνες – Ποιοι είναι οι μεγάλοι κερδισμένοι*.»
- «*Αντίθετη η Ελλάδα για πλαφόν στο φυσικό αέριο, επιφύλαξη για μείωση της ζήτησης ηλεκτρικής ενέργειας*.»
- «*Οικονόμου: Η ΝΔ θα αναζητήσει την εμπιστοσύνη των πολιτών για τον σχηματισμό μίας ισχυρής κυβέρνησης*.»

“CNN Greece”

- «*ΟΗΕ: Πενήντα εκατομμύρια άνθρωποι εξαναγκάζονται σε εργασία ή σε γάμο*.»
- «*“Νιώθω το βάρος της ιστορίας”: Πρώτη ομιλία του βασιλιά Καρόλου στο βρετανικό Κοινοβούλιο.*»
- «*Φεστιβάλ Ρεματιάς: Συνεχίζεται στο Ευριπίδειο Θέατρο Χαλανδρίου*.»
- «*Ο Π. Γεωργιόπουλος στο συνέδριο του Economist: η NOVA χτίζει ιδιόκτητο δίκτυο οπτικών ινών*.»
- «*Κικίλιας: Πρώτη σε όλη την Ευρώπη η Ελλάδα και στις πτήσεις Αυγούστου*.»
- «*Βενετία 2022: Η χρονιά ανήκει στις γυναίκες, την Κέιτ Μπλάνσετ και τον Κόλιν Φάρελ - Όλα τα βραβεία*.»

4.2 Προ-επεξεργασία κειμένου (Text Preprocessing)

4.2.1 Ανάλυση των βημάτων της προ-επεξεργασίας των δεδομένων

Εντοπισμός διπλότυπων εγγραφών

Ως πρώτο βήμα είναι αναγκαίο να εντοπιστούν οι διπλότυπες εγγραφές από το σετ δεδομένων και να διαγραφούν, κρατώντας μόνο μία από αυτές. Οι διπλότυπες εγγραφές δεν συνεισφέρουν με κάποιο τρόπο στην εξόρυξη γνώσης καθώς δύνανται να παράγουν λανθασμένα ή παραπλανητικά συμπεράσματα και για το λόγο αυτό συνήθως απορρίπτονται σε προβλήματα μηχανικής μάθησης. Παρατηρήθηκε τελικά, ότι κάθε τίτλος είναι μοναδικός και συνεπώς δεν απαιτείται η αφαίρεση κάποιου.

Αφαίρεση ορθογραφικών λαθών

Με μια πρώτη ματιά στο σύνολο δεδομένων δεν παρατηρήθηκαν ορθογραφικά λάθη, πράγμα λογικό καθώς πρόκειται για επίσημες πηγές ειδησεογραφικών ειδήσεων.

Αφαίρεση των τόνων

Μόνο για το μοντέλο με τις προ-εκπαιδευμένες αναπαραστάσεις Greek-Bert, αφαιρέθηκαν οι τόνοι από το σύνολο δεδομένων όπως καταδείκνυαν οι οδηγίες χρήσης των δημιουργών.

Διαχείριση σημείων στίξης και συμβόλων

Επόμενο βήμα στην προ-επεξεργασία των δεδομένων είναι η διαχείριση των σημείων στίξης. Κατά το “καθάρισμα” των δεδομένων θεωρείται βασική τεχνική η απομάκρυνση των σημείων στίξης. Εδώ, κάποιες κλασικές μέθοδοι προ-επεξεργασίας δεν εφαρμόσθηκαν ώστε να μην χαλάσει η δομή του κειμένου που θα χρησιμοποιηθεί για τα βαθιάς μάθησης μοντέλα που θα υλοποιηθούν. Τα σημεία στίξης αφαιρέθηκαν μόνο για τα παραδοσιακά μοντέλα που υλοποιήθηκαν. Τα σύμβολα που αφαιρέθηκαν σε αυτήν την περίπτωση είναι τα παρακάτω: “/-’‘”?!.,#$%*+-/:;<=>@[]{|} «»”.

Αντικατάσταση των mentions, hashtags και URLs

Συνήθης τεχνική σε προβλήματα επεξεργασίας κειμένου είναι η αφαίρεση των mentions, hashtags, URLs και των αριθμών, καθώς θεωρείται ότι δεν εμπεριέχουν χρήσιμη πληροφορία και αντιμετωπίζονται σαν θόρυβος στα δεδομένα. Αυτή η τεχνική θεωρείται χρήσιμη σε μοντέλα μηχανικής μάθησης όπως τα SVM, Logistic Regression, Naive Bayes κ.α., για αυτό το λόγο και για τα παραδοσιακά μοντέλα που εξετάζουμε εδώ αυτά πράγματι αφαιρέθηκαν.

Η ίδια τεχνική αντίθετα, δεν χρησιμοποιήθηκε στην περίπτωση των μοντέλων μας όπου γίνεται χρήση τεχνικών βαθιάς μάθησης. Τεχνικές βαθιάς μάθησης θεωρείται ότι ανταποκρίνονται καλύτερα με όση περισσότερη πληροφορία υπάρχει διαθέσιμη, ώστε να μην χάνεται το νόημα της πρότασης, εφόσον έχουμε να κάνουμε με δεδομένα κειμένου. Στόχος αυτών των τεχνικών είναι η εκμάθηση για το πώς λειτουργεί η γλώσσα από ολόκληρο το σύνολο δεδομένων ως ακολουθία, και αυτό περιλαμβάνει την παρακολούθηση του τρόπου με τον οποίο κάθε λέξη συνδέεται με τις γύρω λέξεις με μη

δομημένο τρόπο. Η αφαίρεση αυτών που θεωρούνται λιγότερο σημαντικές μπορεί να κάνει το πρόβλημα δυσκολότερο απ΄ ότι είναι ήδη.

Με την τεχνική των “REGular EXpressions” έγινε αντικατάσταση των mentions, hashtags και URLs των tweets. Συγκεκριμένα, λέξεις οι οποίες ξεκινούσαν με τον ειδικό χαρακτήρα “@” και συνεπώς, αναφέρονταν σε κάποιον εγγεγραμμένο χρήστη, αντικαταστάθηκαν με τη λέξη “MENTION”. Με τον ίδιο τρόπο, η λέξη “HASHTAG” πήρε τη θέση εκείνων των λέξεων που στην αρχή τους είχαν το ειδικό σύμβολο “#”. Ταυτόχρονα, οι υπερσύνδεσμοι οι οποίοι υπήρχαν στις καταγραφές αντικαταστάθηκαν με τη λέξη “URL”. Ακολουθώντας την προαναφερθείσα τεχνική, και οι αριθμοί που εμφανίζονται στα tweets του dataset αντικαταστάθηκαν με τη λέξη “NUMBER”.

Μετατροπή σε πεζούς χαρακτήρες

Όλα τα tweets του συνόλου δεδομένων μετατράπηκαν σε πεζούς χαρακτήρες. Αυτό συνέβη καθώς ίδιες λέξεις οι οποίες μπορεί να διαφέρουν σε κάποιο γράμμα που μπορεί να είναι κεφαλαίο στη μία περίπτωση και πεζό στην άλλη αντιμετωπίζονται ως διαφορετικές. Αυτό δεν συνεισφέρει στον στόχο της παρούσας προσέγγισης και γι΄ αυτό εφαρμόζεται η αντικατάσταση όλων των κεφαλαίων χαρακτήρων σε πεζούς.

Αφαίρεση stopwords

Μια συνήθης τεχνική στην Επεξεργασία Φυσικής Γλώσσας είναι και η αφαίρεση των stopwords. Αυτή η τεχνική, όπως αναφέρθηκε και προηγουμένως, δεν είναι χρήσιμη σε μοντέλα που χρησιμοποιούν βαθιά μάθηση παρά μόνο σε παραδοσιακά μοντέλα. Στην παρούσα έρευνα, αφαίρεση των stopwords πραγματοποιήθηκε μόνο για τα παραδοσιακά μοντέλα.

Για την αφαίρεση των stopwords χρησιμοποιήθηκε η βιβλιοθήκη NLTK και το σύνολο των ελληνικών stopwords. Στο σύνολο αυτό προστέθηκαν και οι παρακάτω λέξεις, οι οποίες συγκεντρώθηκαν μετά από παρατήρηση του συνόλου δεδομένων: “‘τη’, ‘της’, ‘τις’, ‘από’, ‘μια’, ‘τους’, ‘στις’, ‘ότι’, ‘σαν’, ‘σήμερα’, ‘RT’, ‘number’, ‘χρονος’, ‘στα’, ‘χρόνια’, ‘πριν’, ‘numberχρονος’, ‘numberχρονη’, ‘μας’, ‘έχει’ ”.

Αφαίρεση περιττών κενών

Στο βήμα αυτό έγινε αφαίρεση όλων των περιττών κενών, αφήνοντας μόνο ένα καθώς είτε στα αρχικά δεδομένα μπορεί να υπήρχαν επιπλέον κενά λόγω σφάλματος είτε μέσω των βημάτων που έχουν εφαρμοστεί μέχρι στιγμής να έχουν εισαχθεί περιττά κενά. Η συγκεκριμένη διαδικασία υλοποιήθηκε με χρήση κανονικών εκφράσεων.

Tokenization

Στο βήμα αυτό μελετάται μία πολύ κοινή διαδικασία που εφαρμόζεται στην επεξεργασία της φυσικής γλώσσας. Αυτή είναι η τεχνική του tokenization. Το tokenization είναι ένας τρόπος διαχωρισμού του κειμένου σε μικρότερες ενότητες που ονομάζονται “tokens”. Η πιο συνηθισμένη εκδοχή των tokens είναι οι λέξεις, η οποία και εφαρμόστηκε. Συνεπώς κάθε καταγραφή του συνόλου δεδομένων χωρίστηκε στις επιμέρους λέξεις αυτής.

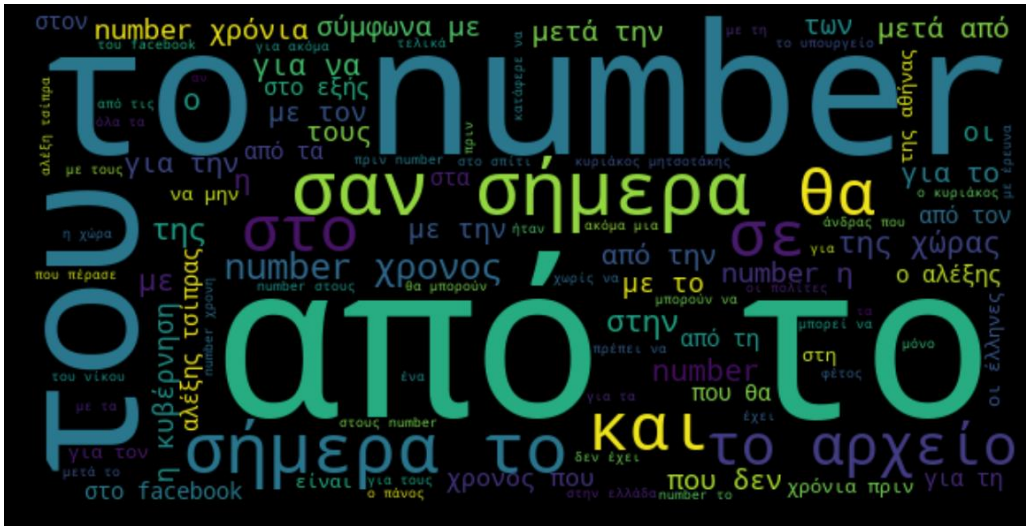
Για την υλοποίηση αυτού χρησιμοποιήθηκε από τη βιβλιοθήκη Keras ο tokenizer: Tokenizer. Ο συνολικός αριθμός των tokens υπολογίστηκε στα 24619.

Αφαίρεση κενών εγγραφών

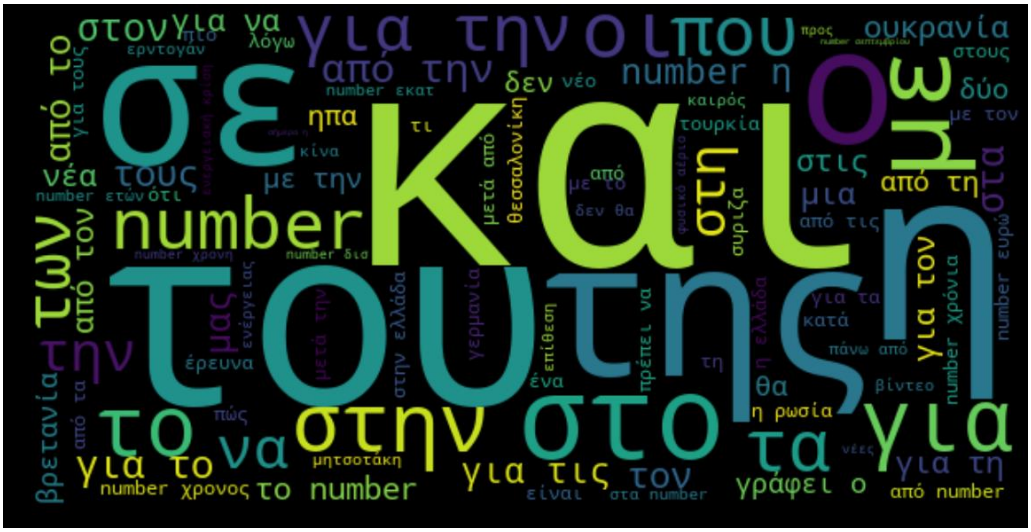
Τέλος, μετά την ολοκλήρωση όλων των απαραίτητων βημάτων για την προ-επεξεργασία των δεδομένων κειμένου, αφαιρέθηκαν όλες εκείνες οι καταγραφές που δεν περιείχαν γράμματα. Αυτό συνέβη καθώς ύστερα από όλη αυτή τη διαδικασία μπορεί να υπάρχουν καταγραφές χωρίς κείμενο, το οποίο δεν είναι χρήσιμο για τον απώτερο σκοπό της παρούσας έρευνας.

4.2.2 Οπτική παρατήρηση των δεδομένων

Πριν την προ-επεξεργασία των δεδομένων, δημιουργήθηκαν τα WordClouds για τις δύο κλάσεις (σαρκαστικές, μη σαρκαστικές ειδήσεις).



Εικόνα 16: WordCloud μη σαρκαστικών τίτλων ειδήσεων πριν την προ-επεξεργασία

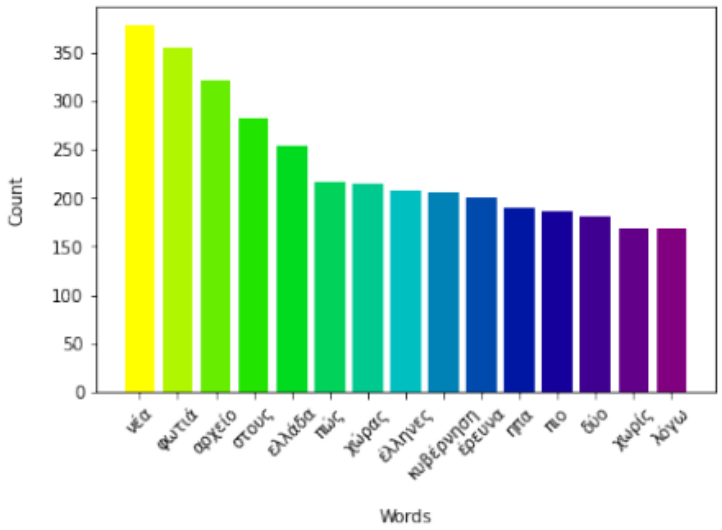


Εικόνα 17: WordCloud σαρκαστικών τίτλων ειδήσεων πριν την προ-επεξεργασία

Μετά την ολοκλήρωση της προ-επεξεργασίας των δεδομένων δημιουργήθηκαν κάποια γραφήματα που αφορούν τις λέξεις που κυριαρχούν συνολικά στο dataset. Αξίζει να σημειωθεί πως για τη δημιουργία αυτών δεν ελήφθησαν υπόψιν τα stopwords και τα σημεία στίξης. Παράλληλα, αγνοήθηκαν και οι λέξεις “HASHTAG”, “NUMBER”, “URL”

και “MENTION” που δημιουργήθηκαν κατά την προ-επεξεργασία των κειμένων για τις παραδοσιακές τεχνικές. Αυτό συνέβη καθώς το σύνολο των προαναφερθέντων λέξεων εμφανίζονται υπερβολικά συχνά και δεν προσφέρουν κάποια χρήσιμη πληροφορία σχετικά με τις κυρίαρχες λέξεις του αντικειμένου μελέτης μας.

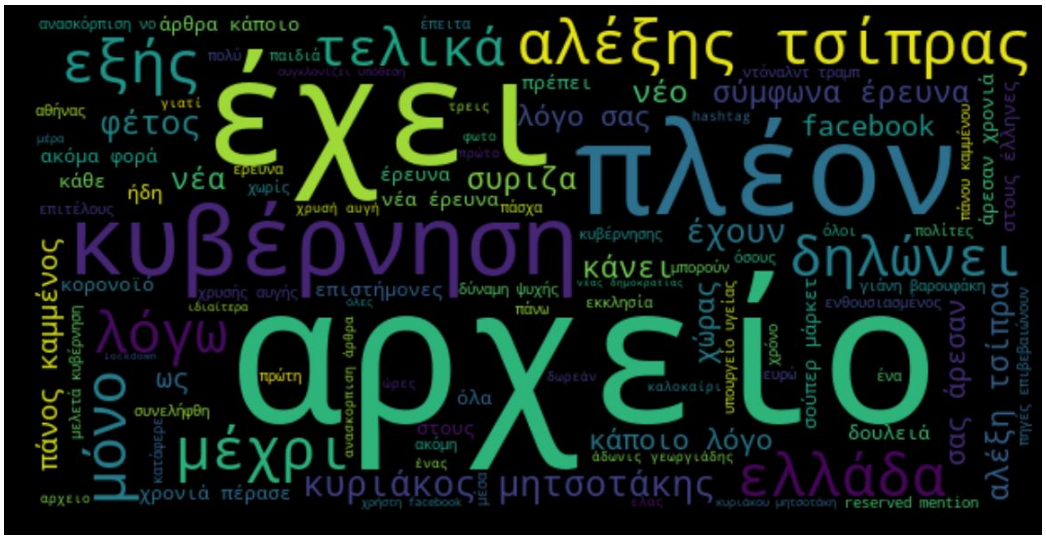
Επιπλέον, για τη δημιουργία όλων των γραφημάτων που θα παρουσιαστούν στο κεφάλαιο αυτό, έγινε συνένωση των προ-επεξεργασμένων tweets που ανήκουν και στις δύο κατηγορίες (σαρκαστικοί και μη σαρκαστικοί τίτλοι ειδήσεων), ώστε το αποτέλεσμα που θα προκύψει να αφορά τη συνολική εικόνα των διαθέσιμων δεδομένων. Στα γραφήματα του παρακάτω σχήματος παρουσιάζονται τα πιο κοινά unigrams:



Εικόνα 18: Κυρίαρχα 1-grams μετά την προ-επεξεργασία

Παρακάτω παρουσιάζονται δύο γραφήματα που δείχνουν τις κυρίαρχες 100 λέξεις που επικρατούν στα tweets ανάλογα το αν είναι σαρκαστικά ή όχι σε μορφή WordCloud. Όπως και στο παρακάτω σχήμα, έτσι και εδώ, έγινε η ίδια παραδοχή που σχετίζεται με την απουσία συγκεκριμένου συνόλου λέξεων για τη δημιουργία των ακόλουθων γραφημάτων:

Εικόνα 19: WordCloud μη σαρκαστικών τίτλων ειδήσεων



Εικόνα 20: WordCloud σαρκαστικών τίτλων ειδήσεων

4.3 Προ-εκπαιδευμένες ενσωματώσεις λέξεων (Pre-trained Word / Embeddings)

Στην προσέγγιση χρησιμοποιήθηκαν προ-εκπαιδευμένες αναπαραστάσεις λέξεων από τα μοντέλα Word2Vec, FastText και BERT για την ελληνική γλώσσα. Για κάθε λέξη δημιουργείται ένας πίνακας χαρακτηριστικών ο οποίος διαμορφώνεται ανάλογα με το ποια προ-εκπαιδευμένα διανύσματα λέξεων (word embeddings) χρησιμοποιούνται.

4.3.1 Word2Vec

Για την πρώτη προσέγγιση χρησιμοποιήθηκαν Word2Vec διανύσματα λέξεων. Το αρχείο με τα διανύσματα ('model.txt') βρίσκεται στον σύνδεσμο http://vectors.nlpl.eu/repository/# με id 46 το οποίο δημιουργήθηκε από το Language Technology Group του Πανεπιστημίου του Όσλο. Το Word2Vec μοντέλο εκπαιδεύτηκε χρησιμοποιώντας την CBOW αρχιτεκτονική σε 770.507.143 λέξεις του Greek CoNLL17 Corpus [109], με διάσταση 100. Το λεξιλόγιο που προκύπτει περιέχει 1.183.175 λέξεις με πεζά γράμματα.

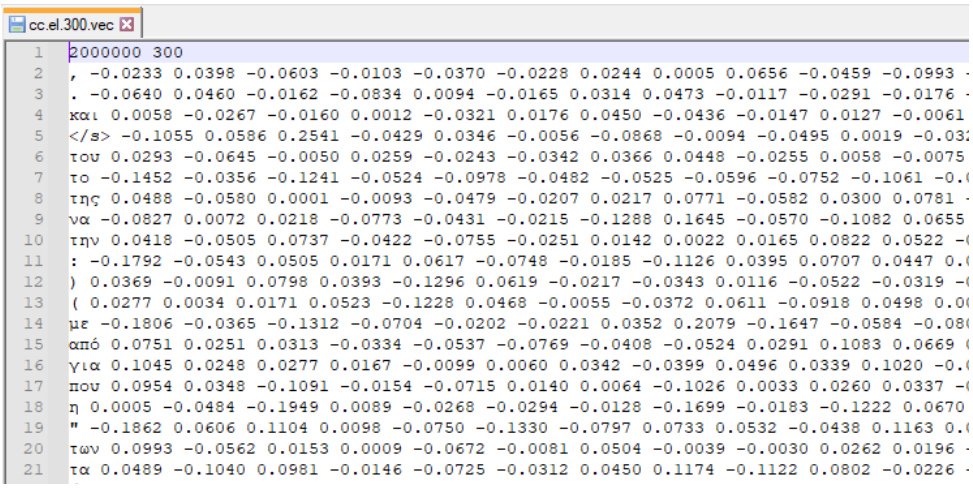
model2.txt

```
1  </s> 0.004003 0.004419 -0.003830 -0.003278 0.001367 0.003021 0.000941 0.000211 -0.003604 0.0
2  , -0.060499 -0.309158 0.050901 0.365739 -0.149589 -0.021702 -0.019373 0.027111 -0.303348 0.1
3  . -0.479390 -0.374027 -0.002442 0.158683 -0.212178 0.327398 0.178842 -0.375386 -0.172361 0.4
4  και -0.214778 -0.183442 0.157424 0.076186 0.040821 0.300206 -0.150424 -0.243173 -0.220828 0.
5  το -0.416298 -0.294495 0.289683 -0.000489 0.070684 0.204443 0.344132 -0.293695 -0.153062 0.5
6  : -0.385377 -0.198262 -0.181989 -0.269782 0.045979 0.207939 -0.312035 -0.106637 -0.296941 -0
7  από 0.269430 -0.196774 -0.159567 -0.308239 0.219025 -0.037669 0.404726 -0.589747 -0.039246 0
8  για 0.089024 0.048537 -0.330082 -0.131278 -0.114770 0.342854 -0.193557 -0.196195 -0.364376 0
9  του -0.370063 -0.651975 0.429695 0.060951 0.203414 -0.257936 0.255521 -0.383163 0.091272 0.3
10 ( 0.314283 -0.175481 0.424579 0.059685 0.173555 -0.213540 -0.049689 -0.031390 -0.094085 0.10
11 να -0.280827 -0.507761 -0.059257 0.110743 -0.312804 0.588456 0.287200 0.018265 0.191431 0.84
12 σε -0.335356 -0.220343 0.052549 -0.200304 0.113050 0.359509 0.617504 0.354451 -0.130005 0.30
13 - -0.053728 0.100391 0.450564 0.123237 0.193375 -0.710339 0.158967 -0.679327 -0.649015 0.226
14 με -0.253704 -0.104495 -0.250260 -0.157127 -0.058417 0.118014 -0.195012 -0.040589 -0.423305
15 ) 0.277721 -0.217080 0.345550 0.093594 0.175157 -0.254106 0.019955 0.049727 -0.193041 0.0643
16 της 0.207986 0.008972 0.484279 0.037356 -0.135328 0.017259 -0.060739 -0.077231 0.113185 0.20
17 την -0.284799 -0.291918 -0.026537 -0.298358 -0.157026 0.365852 0.230428 -0.483047 0.104473 0
18 τα -0.533573 -0.243430 0.256513 0.328281 0.462190 0.069918 0.453453 0.049374 -0.231523 0.724
19 η 0.027723 -0.462671 -0.138202 -0.240922 -0.042695 0.171110 0.125637 -0.296826 0.006392 0.13
20 ξενοδοχεία 0.106777 -0.610204 0.635607 -1.052009 -0.215624 -0.177671 0.720352 0.701773 0.258
21 στο -0.067719 -0.270510 -0.029586 0.128571 0.391803 0.158293 0.149723 -0.338312 -0.358916 0.
```

Εικόνα 21: Μορφή του αρχείου 'model.txt' που έχει παραχθεί από το Word2Vec μοντέλο

4.3.2 FastText

Στην δεύτερη προσέγγιση, χρησιμοποιήθηκαν FastText διανύσματα λέξεων για να αρχικοποιηθεί το embedding layer του δικτύου. Το FastText μοντέλο εκπαιδεύτηκε χρησιμοποιώντας την CBOW αρχιτεκτονική Common Crawl και Wikipedia, με αρνητική δειγματοληψία (negative sampling) ίση με 5, ελάχιστο πλήθος λέξεων 20 και διάσταση 300. Το λεξιλόγιο που προκύπτει περιέχει 2.000.000 λέξεις. Το αρχείο που χρησιμοποιήθηκε ('cc.el.300.vec') είναι από τον σύνδεσμο https://fasttext.cc/docs/en/crawl-vectors.html και δημιουργήθηκε από τους Grave κ.ά. [110].



Εικόνα 22: Μορφή του αρχείου 'cc.el.300.vec' που έχει παραχθεί από το FastText μοντέλο

4.3.3 BERT

Για την τρίτη προσέγγιση, χρησιμοποιήθηκε το μοντέλο 'bert-base-greek-uncased-v1' από τον σύνδεσμο huggingface.co, το οποίο κυκλοφόρησε επίσημα με το άρθρο GREEK-BERT: The Greeks Visiting Sesame Street από τους Koutsikakis κ.ά. [111]. Σύμφωνα με τις οδηγίες των δημιουργών, για να χρησιμοποιήσουμε το 'bert-base-greek-uncased-v1', πρέπει να προ-επεξεργαστούμε τα κείμενα μετατρέποντας τα γράμματα σε πεζά και να αφαιρέσουμε όλα τα ελληνικά διακριτικά (τόνους, κ.λπ.).

Τα προ-εκπαιδευτικά σώματα του bert-base-greek-uncased-v1 περιλαμβάνουν:

- Το ελληνικό μέρος της Wikipedia,
- Το ελληνικό μέρος του European Parliament Proceedings Parallel Corpus και
- Το ελληνικό μέρος του OSCAR, μια καθαρή έκδοση του Common Crawl.

Οι λεπτομέρειες για την εκπαίδευση του μοντέλου παρουσιάζονται παρακάτω:

- Το μοντέλο εκπαιδεύτηκε χρησιμοποιώντας τον επίσημο κώδικα που παρέχεται στο GitHub του Google BERT (https://github.com/google-research/bert).
- Στη συνέχεια χρησιμοποιήθηκε το σενάριο μετατροπής Hugging Face's Transformers για να μετατραπεί το σημείο ελέγχου TF και το λεξιλόγιο στην επιθυμητή μορφή ώστε να είναι δυνατή η φόρτωση του μοντέλου σε δύο γραμμές κώδικα για χρήστες PyTorch και TF2.

- Είναι ένα μοντέλο παρόμοιο με το αγγλικό μοντέλο ‘bert-base-uncased’ (12-layer, 768-hidden, 12-heads, 110M παράμετροι).
- Επιλέχθηκε να ακολουθηθεί το ίδιο πρόγραμμα εκπαίδευσης: 1 εκατομμύριο βήματα εκπαίδευσης με παρτίδες 256 ακολουθιών μήκους 512 με αρχικό ρυθμό εκμάθησης 1e-4.

4.4 Αρχιτεκτονικές Μοντέλων Ανίχνευσης Σαρκασμού

Το κεφάλαιο αυτό ασχολείται με την λεπτομερή περιγραφή των κατηγοριοποιητών που υλοποιήθηκαν για την ανίχνευση του σαρκασμού σε tweets ειδησεογραφικών πηγών. Οι κατηγοριοποιητές που θα παρουσιαστούν στην συνέχεια βασίζονται σε μοντέλα βαθιάς μάθησης και συγκεκριμένα σε LSTM μοντέλα. Θα γίνει λεπτομερής περιγραφή των διάφορων κατηγοριοποιητών που υλοποιήθηκαν και του τρόπου με τον οποίο αξιοποιούν τα δεδομένα εισόδου. Μετά την εκπαίδευση των μοντέλων, θα είναι ικανά να ανιχνεύσουν τον σαρκασμό. Συγκεκριμένα δημιουργήθηκαν τα διαφορετικά LSTM μοντέλα στα οποία έχουν δοθεί οι ακόλουθες ονομασίες και θα περιγραφούν παρακάτω:

1. Simple LSTM
2. BiLSTM
3. Word2Vec LSTM
4. FastText LSTM
5. BERT BiLSTM

4.4.1 Simple LSTM

Το μοντέλο αυτό θα παίρνει στην είσοδο του το κείμενο από κάποιο tweet και θα καλείται να εντοπίζει αν είναι σαρκαστικό ή όχι. Η ακριβής αρχιτεκτονική του Simple LSTM μοντέλου και το πως υλοποιήθηκε περιγράφεται αναλυτικά ακολούθως:

Αρχικά, χρησιμοποιήθηκε η συνάρτηση `train_test_split()` της βιβλιοθήκης sklearn της Python. Μέσω αυτής έγινε διαχωρισμός των προ-επεξεργασμένων δεδομένων σε δεδομένα τύπου train και validation. Ο διαχωρισμός πραγματοποιήθηκε με τυχαίο τρόπο, ενώ επιλέχθηκε το ποσοστό των δεδομένων που ανήκουν στο train set να είναι 80% με το υπόλοιπο 20% να αντιστοιχεί στο validation set. Το σύνολο των δεδομένων που ανήκει στο train set χρησιμοποιείται για την εκπαίδευση του μοντέλου μέσω της ρύθμισης των βαρών μεταξύ των νευρώνων και της πόλωσης κάθε νευρώνα. Αντίθετα, το validation set έχει ως στόχο τη ρύθμιση των υπερπαραμέτρων του μοντέλου έτσι ώστε να αποφευχθεί η υπερ-προσαρμογή (over-fitting), πρόβλημα το οποίο οδηγεί στην δημιουργία ενός μοντέλου το οποίο δεν μπορεί να λειτουργήσει αποτελεσματικά σε δεδομένα που δεν έχει ξαναδεί.

Στην συνέχεια, αξιοποιώντας τη βιβλιοθήκη Keras, έγινε χρήση της κλάσης Tokenizer(), η οποία χρησιμοποιείται ευρέως για την προετοιμασία κειμενικών δεδομένων που εισάγονται σε μοντέλα βαθιάς μηχανικής μάθησης. Εφαρμόζοντας την συνάρτηση `fit_on_texts()` της προαναφερθείσας κλάσης, δημιουργείται ένα λεξικό, όπου για κάθε λέξη του συνόλου των tweets αντιστοιχεί ένας μοναδικός αριθμός. Ο αριθμός αυτός που παράγεται είναι ανάλογος της συχνότητας εμφάνισης της λέξης ενδιαφέροντος, με αποτέλεσμα σε λέξεις που εμφανίζονται συχνά να αντιστοιχίζονται μικρότεροι αριθμοί, έναντι των σπανιότερα εμφανιζόμενων. Η συγκεκριμένη συνάρτηση εφαρμόστηκε στα δεδομένα που περιέχονται στα train και validation σύνολα. Ο αριθμός

0 δεν χρησιμοποιείται καθώς είναι δεσμευμένος και χρησιμοποιείται για την εφαρμογή του padding, που θα αναλυθεί στην συνέχεια. Ταυτόχρονα, ο αριθμός 1 χρησιμοποιείται για λέξεις που δεν έχει ξαναδεί το μοντέλο. Συνεπώς, ο αριθμός 1 στο λεξικό που δημιουργήθηκε δεν εμφανίζεται, εφόσον οι λέξεις του train και validation set αποτελούν το λεξιλόγιο πάνω στο οποίο θα εκπαιδευτούν τα μοντέλα που θα υλοποιηθούν. Το 1 θα εμφανιστεί στα δεδομένα του συνόλου test.

Μετά την `fit_on_texts()` του Tokenizer() API, ακολουθεί η εφαρμογή της συνάρτησης `texts_to_sequences`(). Αυτή, αντικαθιστά τις λέξεις από κάθε προ-επεξεργασμένο tweet με τους αριθμούς που ορίστηκαν από το λεξικό προηγουμένως. Συνεπώς οι προτάσεις που υπήρχαν προηγουμένως, αντικαταστάθηκαν από μία σειρά από αριθμούς. Ένα παράδειγμα ακολουθεί στον ακόλουθο πίνακα:

Πίνακας 3: Παράδειγμα εφαρμογής της συνάρτησης `fit_on_texts()` σε πραγματικά δεδομένα

"προαιρετική και όχι υποχρεωτική η αναγραφή του ζωδίου στις νέες ταυτότητες" $\xrightarrow{fit\_on\_texts()}$
[51 2134 6 1 118 3 3065 2518 4802 15 461 10375 1184 8 267 10376]

Σε αντίθεση με την `texts_to_sequences()`, η συνάρτηση `fit_on_texts()` εφαρμόζεται μόνο μία φορά. Η πρώτη μπορεί να καλεστεί περισσότερες φορές καθώς εφαρμόζεται για την μετατροπή δεδομένων που δεν έχει ξαναδεί το μοντέλο, ενώ η τελευταία χρησιμοποιείται για τη δημιουργία του λεξικού βάσει του οποίου θα υλοποιηθούν τα μοντέλα, συνεπώς υλοποιείται μία φορά. Το λεξικό που δημιουργήθηκε από την `fit_on_texts()` είναι το ίδιο που θα χρησιμοποιηθεί και για την αξιολόγηση του μοντέλου. Αυτό συμβαίνει καθώς αν δημιουργηθεί νέο λεξικό πάνω στα δεδομένα του test συνόλου, οι αριθμοί που θα αντικαταστήσουν κάθε λέξη του κειμένου θα είναι διαφορετικοί από αυτούς του αρχικού λεξικού, εφόσον αυτοί αποφασίζονται από την συχνότητα εμφάνισης των λέξεων στα tweets.

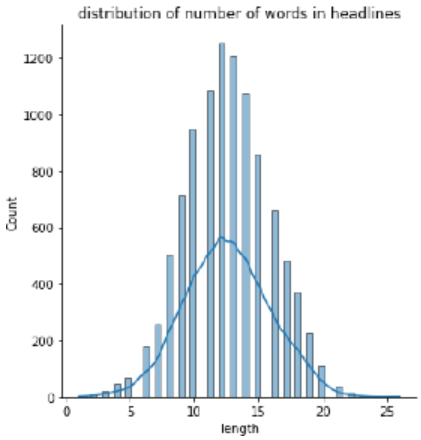
Ακολούθως υπολογίσθηκε το μέγιστο πλήθος λέξεων που εντοπίζεται στα tweets των train και validation συνόλων. Η πληροφορία αυτή είναι χρήσιμη καθώς, όπως όλα τα νευρωνικά δίκτυα, έτσι και τα LSTM, πρέπει να έχουν στην είσοδό τους κάθε φορά δεδομένα που να έχουν το ίδιο σχήμα και μέγεθος. Όταν, όμως, έχουμε να κάνουμε με δεδομένα κειμένου, δεν είναι δυνατό όλες οι προτάσεις να έχουν τον ίδιο αριθμό λέξεων, καθώς άλλες θα είναι μεγαλύτερες και άλλες μικρότερες. Συνεπώς, είναι απαραίτητο όλες οι προτάσεις του συνόλου δεδομένων μας να αποκτήσουν το ίδιο μέγεθος. Αυτό είναι εφικτό με την εφαρμογή του padding.

Για την εφαρμογή της ανωτέρω διαδικασίας πρέπει αρχικά να οριστεί ο μέγιστος αριθμός λέξεων που επιτρέπεται να έχει κάθε καταγραφή. Αν κάποια καταγραφή έχει περισσότερες λέξεις από τον μέγιστο αριθμό που ορίστηκε, οι επιπλέον λέξεις αφαιρούνται. Ως λέξεις, πλέον, ορίζονται οι αριθμητικές τιμές με τις οποίες έχουν αντικατασταθεί οι προ-επεξεργασμένες λέξεις του συνόλου δεδομένων. Στην περίπτωση όπου οι καταγραφές έχουν μικρότερο αριθμό από το μέγιστο, το πλήθος αυτών που υπολείπονται συμπληρώνονται με μηδενικά. Τα μηδενικά αυτά μπορούν να τοποθετηθούν είτε στην αρχή της καταγραφής, είτε στο τέλος. Στα πλαίσια της παρούσας μελέτης επιλέχθηκε να τοποθετούνται στο τέλος, το οποίο είναι γνωστό και ως post_padding. Με παρόμοιο τρόπο, καταγραφές που ξεπερνούν το μέγιστο αριθμό λέξεων, επιλέχθηκε να κόβονται οι επιπλέον λέξεις που βρίσκονται στο τέλος της φράσης. Ο υπολογισμός που αναφέρθηκε προηγουμένως και σχετίζεται με το μέγιστο πλήθος λέξεων που εντοπίζεται στα tweets, έδειξε ότι ο αριθμός αυτός δεν είναι τόσο μεγάλος, 26 συγκεκριμένα (Εικόνα 22), και συνεπώς επιλέχθηκε ως μέγιστο πλήθος

λέξεων για το padding να είναι ο αριθμός αυτός. Ως εκ τούτου δεν χρειάστηκε να σβηστούν λέξεις από τα tweets. Μετά την εφαρμογή του padding στα train και validation σύνολα, τα tweets που εμπεριέχονται σε αυτά απέκτησαν όλα το ίδιο μέγεθος, ώστε να εισαχθούν αργότερα στην είσοδο του Simple LSTM μοντέλου. Το παράδειγμα του πίνακα, μετά την εφαρμογή του padding παίρνει την ακόλουθη μορφή:

Πίνακας 4: Παράδειγμα εφαρμογής του padding με μέγιστο μήκος λέξεων=26

"προαιρετική και όχι υποχρεωτική η αναγραφή του ζωδίου στις νέες ταυτότητες" $\xrightarrow{padding}$
[51 2134 6 1 118 3 3065 2518 4802 15 461 10375 1184 8 267 10376 0 0 0 0 0 0 0 0 0 0]



Εικόνα 23: Κατανομή του αριθμού των λέξεων στους τίτλους ειδήσεων

Στο σημείο αυτό, το κείμενο από τα tweets έχει πάρει την κατάλληλη μορφή ώστε να εισαχθεί στο LSTM μοντέλο. Συνεπώς το μόνο που υπολείπεται είναι η περιγραφή της αρχιτεκτονικής πάνω στο οποίο χτίστηκε ο Simple LSTM κατηγοριοποιητής. Για την υλοποίηση αυτού του μοντέλου έγινε χρήση της βιβλιοθηκών TensorFlow και Keras. Αρχικά, ορίστηκε ότι ο κατηγοριοποιητής θα δομηθεί με χρήση του Sequential μοντέλου των βιβλιοθηκών. Το Sequential API είναι ο ευκολότερος τρόπος για κατασκευή και εκτέλεση μοντέλων στο Keras. Μέσω αυτού επιτρέπεται να δημιουργούνται μοντέλα με προσθήκη διαδοχικών επιπέδων βήμα βήμα. Το Sequential μοντέλο μπορεί να χρησιμοποιηθεί σε περιπτώσεις multi-class classification και πρέπει να υπάρχει μία είσοδος (τα tweets). Εφόσον ικανοποιούνται όλες οι συνθήκες, το Sequential API ήταν αυτό που ανέλαβε την διαδικασία για την υλοποίηση του Simple LSTM κατηγοριοποιητή. Τα επίπεδα τα οποία προστέθηκαν για την σχεδίαση αυτού του μοντέλου είναι τα ακόλουθα:

1. Το πρώτο επίπεδο του κατηγοριοποιητή αποτέλεσε το Keras Embedding layer, το οποίο χρησιμοποιείται κατά κόρον για δεδομένα κειμένου. Αυτό απαιτεί τα δεδομένα που εισάγονται να έχουν κωδικοποιηθεί σε μορφή ακεραίων, έτσι ώστε κάθε λέξη να αντιπροσωπεύεται από έναν μοναδικό ακέραιο. Η απαίτηση αυτή ικανοποιείται εφόσον αυτό το βήμα προετοιμασίας δεδομένων έχει εφαρμοστεί με χρήση του Tokenizer() API. Το Embedding επίπεδο αρχικοποιείται με τυχαία βάρη και κατά τη διάρκεια της εκπαίδευσης δημιουργείται μία διανυσματική αναπαράσταση κάθε λέξης του συνόλου δεδομένων.

Το συγκεκριμένο επίπεδο έχει τη δυνατότητα να χρησιμοποιηθεί με διάφορους τρόπους. Οιδιαφορετικοί αυτοί τρόποι εξετάζονται στην παρούσα προσέγγιση, δημιουργώντας διαφορετικά μοντέλα για τον καθένα με στόχο την εύρεση του καλύτερου τρόπου για βέλτιστη ανίχνευση του σαρκασμού. Στον Simple LSTM κατηγοριοποιητή το Embedding

επίπεδο χρησιμοποιείται ως μέρος του μοντέλου βαθιάς μάθησης, όπου η διανυσματική αναπαράσταση κάθε λέξης μαθαίνεται μαζί με το ίδιο το μοντέλο. Συνεπώς το πρώτο κρυφό επίπεδο του κατηγοριοποιητή που υλοποιείται, αποτελείται από το Embedding επίπεδο, το οποίο για να οριστεί απαιτείται ο προσδιορισμός του μεγέθους του λεξιλογίου των δεδομένων μας, το μέγεθος του διανυσματικού χώρου που επιθυμούμε να αναπαρασταθούν οι λέξεις, καθώς και το μήκος των δεδομένων εισόδου που θα δέχεται το μοντέλο στην είσοδό του. Στην περίπτωσή μας, το μέγεθος του λεξιλογίου βρέθηκε να είναι 22168, όσο και το πλήθος των διαφορετικών λέξεων που εντοπίστηκαν μετά την προ-επεξεργασία. Το μέγεθος του διανυσματικού χώρου για την αναπαράσταση των λέξεων επιλέχθηκε να είναι 200. Συνεπώς, κάθε λέξη των δεδομένων μας αναπαρίσταται από ένα διάνυσμα διάστασης 200. Τέλος, το μήκος των δεδομένων εισόδου τέθηκε όσο ο μέγιστος αριθμός λέξεων των καταγραφών. Στην έξοδο του Embedding επιπέδου που υλοποιήθηκε παράγεται ένα διάνυσμα δύο διαστάσεων (μήκος δεδομένων εισόδου x μέγεθος διανυσματικού χώρου για αναπαράσταση των λέξεων), το οποίο περιέχει την αναπαράσταση για κάθε λέξη του κειμένου που δέχεται στην είσοδο.

2. Η έξοδος του προηγούμενου επιπέδου εισάγεται στο SpatialDropout1D επίπεδο. Το επίπεδο SpatialDropout1D κατά τη διάρκεια της εκπαίδευσης του μοντέλου ρυθμίζει τυχαία κάποιες στήλες των δεδομένων εισόδου σε 0. Όταν μία στήλη επιλεγεί, τότε την τιμή 0 παίρνουν και οι υπόλοιπες είσοδοι (λέξεις) της συγκεκριμένης στήλης. Με αυτό τον τρόπο είναι σαν να αφαιρείται το ίδιο χαρακτηριστικό από τη διανυσματική αναπαράσταση κάθε λέξης του tweet. Το πλήθος των στηλών που θα πάρουν αυτή την τιμή ορίζεται από την συχνότητα που θα επιλέξει ο σχεδιαστής, με την τιμή αυτή να έχει τεθεί στο 25%. Η διαδικασία αυτή πραγματοποιείται προς αποτροπή του προβλήματος της υπερπροσαρμογής. Για την αποφυγή της υπερπροσαρμογής συνήθως χρησιμοποιείται το επίπεδο Dropout, το οποίο ρυθμίζει σε 0 διάφορα στοιχεία της κάθε εισόδου του με κάποια συχνότητα, χωρίς να απαιτεί να συμβεί αυτό και στα υπόλοιπα στοιχεία της ίδιας στήλης. Στην περίπτωση της παρούσας έρευνας έχουμε να κάνουμε με δεδομένα κειμένου και συνεπώς, προτιμάται το SpatialDropout1D έναντι του Dropout, καθώς οι γειτονικές στήλες στη διανυσματική αναπαράσταση των λέξεων έχουν συνήθως κοντινό νόημα, με αποτέλεσμα η πλήρης αφαίρεση της στήλης να βοηθάει καλύτερα το μοντέλο να γενικεύει την πληροφορία.

3. Μετά την εφαρμογή του SpatialDropout1D ακολουθεί το πρώτο LSTM επίπεδο, με την έξοδο του προηγούμενου επιπέδου να εισάγεται στο τελευταίο. Το νευρωνικό δίκτυο μακράς βραχυπρόθεσμης μνήμης αποτελείται από 180 νευρώνες, ενώ η τιμή του ορίσματος του κρυφού dropout τέθηκε στο 50%. Το συγκεκριμένο dropout είναι διαφορετικό από το Dropout επίπεδο που αναφέρθηκε η προηγουμένως. Η διαφορά του έγκειται στο γεγονός ότι το επίπεδο Dropout επηρεάζει τις τιμές που θα μπουν στην είσοδο του LSTM επιπέδου, ενώ το όρισμα του dropout κατά την σχεδίαση του LSTM σχετίζεται με τον τρόπο εκμάθησης των βαρών κατά τη διάρκεια της εκπαίδευσης. Συγκεκριμένα, δεν γίνεται εκμάθηση όλων των βαρών μαζί, αλλά εκμάθηση ενός κλάσματος αυτών στο δίκτυο σε κάθε επανάληψη της εκπαίδευσης. Συνεπώς, ένα μέρος των κόμβων ενεργοποιείται σε κάθε επανάληψη.

4. Στο επόμενο επίπεδο του Simple LSTM μοντέλου συναντάται το επίπεδο BatchNormalization. Το επίπεδο έχει τη δυνατότητα να ομαλοποιεί τα δεδομένα που λαμβάνει στην είσοδό του. Συγκεκριμένα, εφαρμόζει έναν μετασχηματισμό που διατηρεί τη μέση έξοδο κοντά στο 0 και την τυπική απόκλιση κοντά στο 1.

5. Τα δεδομένα που προκύπτουν από την έξοδο του BatchNormalization επιπέδου, εισάγονται σε ένα νέο LSTM νευρωνικό. Αυτό το επίπεδο αποτελείται από 90 νευρώνες, ενώ το όρισμα dropout τέθηκε και πάλι στο 50%.

6. Στην συνέχεια, εισάγεται ένα ακόμη BatchNormalization επίπεδο. Με τη χρήση αυτού του επιπέδου δίνεται η δυνατότητα της ταχύτερης εκπαίδευσης του νευρωνικού δικτύου, ενώ μπορεί να βελτιώσει την απόδοση του μοντέλου.

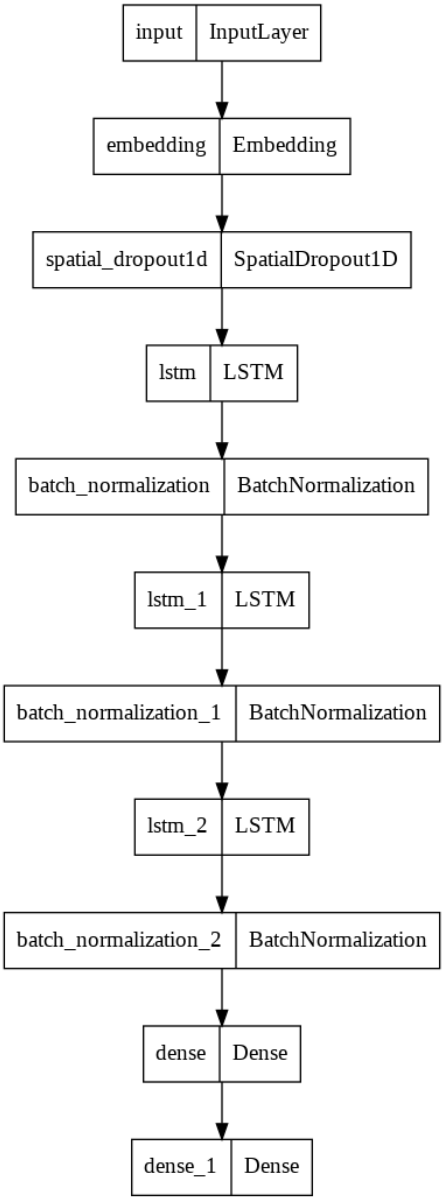
7. Ακολούθως προστίθεται ένα τελευταίο LSTM νευρωνικό με 40 νευρώνες, ενώ το dropout τέθηκε και πάλι στο 50%.

8. Τα δεδομένα που προέκυψαν από την έξοδο του τελευταίου LSTM επιπέδου, εισήχθησαν σε ένα ακόμη BatchNormalization επίπεδο.

9. Το επόμενο επίπεδο είναι ένα απλό πλήρως διασυνδεδεμένο νευρωνικό (Dense) με 20 νευρώνες. Η συνάρτηση ενεργοποίησης που επιλέχθηκε είναι η “relu”.

10. Τέλος, προστέθηκε και το τελευταίο νευρωνικό που αποτελείται από 2 νευρώνες, όσους και οι κλάσεις. Σε αυτό το επίπεδο η συνάρτηση ενεργοποίησης ήταν η “sigmoid”. Η επιλογή αυτής δικαιολογείται από το γεγονός ότι έχουμε να κάνουμε με δυαδική ταξινόμηση.

Το γράφημα που δείχνει τη δομή του Simple LSTM μοντέλου που σχεδιάστηκε φαίνεται ακολούθως στην εικόνα:

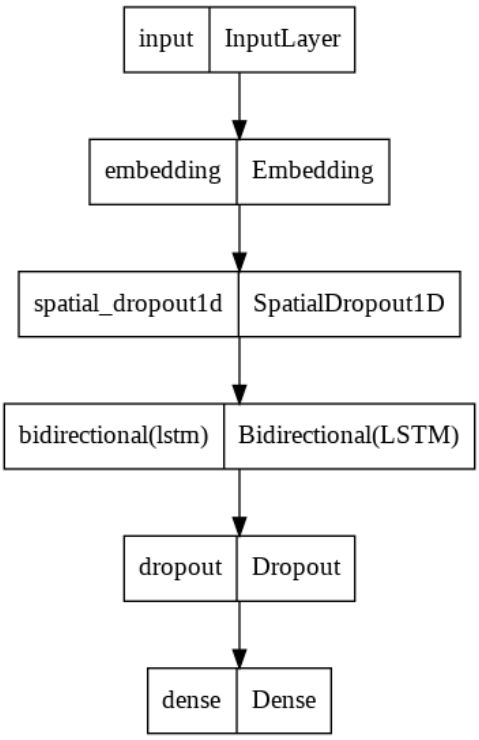


Εικόνα 24: Γράφημα Simple LSTM μοντέλου

4.4.2 BiLSTM

Το επόμενο μοντέλο που εξετάζεται είναι αυτό με τίτλο BiLSTM. Η αρχιτεκτονική αυτού του μοντέλου σε ένα βαθμό θυμίζει αυτή του Simple LSTM. Αυτό εξηγείται καθώς το Embedding επίπεδο χρησιμοποιείται ως μέρος του μοντέλου βαθιάς μάθησης, και συνεπώς, η διανυσματική αναπαράσταση κάθε λέξης μαθαίνεται μαζί με το ίδιο το μοντέλο. Μπορεί τα χαρακτηριστικά του Embedding επιπέδου να είναι ίδια με αυτά του Simple LSTM κατηγοριοποιητή, όμως η διαφοροποίηση του συγκεκριμένου μοντέλου σχετίζεται με την επιλογή του είδους του LSTM νευρωνικού, όπου στην συγκεκριμένη περίπτωση επιλέχθηκε το BiLSTM. Το BiLSTM νευρωνικό αποτελείται από 64 νευρώνες, ενώ η τιμή του dropout ορίστηκε στο 50%. Επιπλέον, το συγκεκριμένο μοντέλο δεν απαιτεί την εισαγωγή και άλλων LSTM/Dropout επιπέδων, καθώς λόγω των επιπλέον δυνατοτήτων που διαθέτει το BiLSTM δεν καθίσταται απαραίτητη η ένταξη άλλων επιπέδων για την επίτευξη του στόχου αυτού του κατηγοριοποιητή.

Κατά τ΄ άλλα οι τεχνικές που ακολουθήθηκαν για την προετοιμασία των δεδομένων ώστε να εισαχθούν στο πρώτο επίπεδο είναι όμοιες με αυτές των άλλων ταξινομητών που έχουν παρουσιαστεί. Το γράφημα που δείχνει τη δομή του BiLSTM μοντέλου που σχεδιάστηκε είναι το ακόλουθο:



Εικόνα 25: Γράφημα BiLSTM μοντέλου

4.4.3 Word2Vec LSTM

Το τρίτο μοντέλο που σχεδιάστηκε και αναλύεται σε αυτό το υποκεφάλαιο είναι ο Word2Vec LSTM κατηγοριοποιητής. Το μοντέλο αυτό έχει πολλά κοινά στοιχεία με το προηγούμενο, με τη μεγαλύτερη διαφορά να εντοπίζεται στον τρόπο σχεδίασης του Embedding επιπέδου. Προτού ξεκινήσει η περιγραφή της αρχιτεκτονικής του, αξίζει να σημειωθεί πως, όπως και προηγουμένως, διαχωρίστηκαν τα δεδομένα σε train και validation σετ. Παράλληλα, έγινε χρήση του Tokenizer API (μέσω των συναρτήσεων `texts_to_sequences()` και `fit_on_texts()`) και ακολούθησε η διεργασία του padding. Με το σύνολο αυτών των διαδικασιών τα δεδομένα έχουν πάρει την κατάλληλη

μορφή για την εισαγωγή τους στο κατηγοριοποιητή. Το Sequential API χρησιμοποιήθηκε και πάλι για την μοντελοποίηση. Οι λεπτομέρειες και η ανάλυση αυτού περιγράφονται παρακάτω:

1. Το πρώτο επίπεδο, όπως στην προηγούμενη περίπτωση, είναι το Embedding επίπεδο. Η διαφορά σε αυτήν, όμως, είναι ότι εισάγεται μία ήδη προ-εκπαιδευμένη διανυσματική αναπαράσταση των λέξεων, με αποτέλεσμα να μεταφέρεται η υπάρχουσα γνώση στο μοντέλο μας.

Με αυτό τον τρόπο, δεν χρειάζεται εκπαίδευση του Embedding επιπέδου με τον αριθμό των παραμέτρων που χρειάζονται εκμάθηση να μειώνεται δραματικά. Συγκρίνοντας τις εκπαιδεύσιμες παραμέτρους μεταξύ του Simple LSTM και του κατηγοριοποιητή που εξετάζεται η διαφορά είναι μεγαλύτερη από 4.000.000, με τις συνολικές παραμέτρους να είναι παρόμοιες και στα δύο μοντέλα. Μέσω αυτού, γίνεται και εξοικονόμηση χρόνου για την εκπαίδευση του Word2Vec LSTM κατηγοριοποιητή.

Η προ-εκπαιδευμένη διανυσματική αναπαράσταση του Word2Vec όπως περιγράφηκε σε προηγούμενο υποκεφάλαιο είναι διάστασης 100.

Μετά το κατέβασμα αυτού του αρχείου είναι απαραίτητη η φόρτωση αυτού στον κώδικα. Για να γίνει εφικτό αυτό, είναι αναγκαία η δημιουργία ενός λεξικού που θα διατηρεί τις αντιστοιχίσεις μεταξύ των λέξεων και των διανυσματικών τους αναπαραστάσεων που υπάρχουν στο αρχείο. Μετά τη υλοποίηση του λεξικού, γίνεται κατασκευή ενός πίνακα που περιέχει τις αναπαραστάσεις για εκείνες μόνο τις λέξεις που υπάρχουν στο σύνολο δεδομένων εκπαίδευσης του μοντέλου. Από τις συνολικά 1.183.175 λέξεις του αρχείου ταυτοποιήθηκαν 20724 από τις 26.995 του συνόλου δεδομένων.

Εφόσον ολοκληρώθηκε αυτή η διαδικασία, οι διανυσματικές αναπαραστάσεις μπορούν να εισαχθούν στο Embedding επίπεδο. Δοκιμάστηκαν 2 προσεγγίσεις όσον αφορά στις διανυσματικές αναπαραστάσεις του Embedding επιπέδου. Στην πρώτη, κατά τον σχεδιασμό αυτού του επιπέδου τέθηκε η παράμετρος trainable = False και ο πίνακας που δημιουργήθηκε προηγουμένως αποτέλεσε τα βάρη αυτού του layer τα οποία και μένουν παγωμένα. Στην δεύτερη προσέγγιση τα βάρη μπορούν να αλλάξουν κατά την διάρκεια της εκπαίδευσης και τίθεται το όρισμα trainable = True.

2. Η έξοδος των δεδομένων από το Embedding επίπεδο εισήχθησαν στο SpatialDropout1D, με την συχνότητα μηδενισμού των τιμών να ορίζεται σε ποσοστό 25% για κάθε λέξη.

3. Επόμενο επίπεδο είναι ένα LSTM νευρωνικό. Αυτό αποτελείται από 80 νευρώνες, ενώ η τιμή του dropout τέθηκε στο 50%.

4. Στο σημείο αυτό εισάγεται ένα Dropout layer. Το συγκεκριμένο επίπεδο είναι αυτό που περιγράφηκε στο βήμα 2 του υποκεφαλαίου 4.4.1 όπου περιγράφεται η δομή του Simple LSTM μοντέλου. Η χρήση ενός SpatialDropout1D δεν θεωρείται γόνιμη σε αυτό το σημείο του μοντέλου, καθώς η έξοδος του προηγούμενου LSTM επιπέδου είναι αριθμητικές τιμές που δεν περιέχουν χαρακτηριστικά για την κάθε λέξη του κειμένου. Ταυτόχρονα, η συχνότητα του Dropout τέθηκε στο 20%. Με χρήση αυτού του επιπέδου προλαμβάνεται το πρόβλημα της υπερ-προσαρμογής (over-fitting) των δεδομένων στα LSTM νευρωνικά.

5. Ακολούθως, τα δεδομένα από την έξοδο του Dropout layer εισάγονται σ΄ένα LSTM νευρωνικό αποτελούμενο από 40 νευρώνες και dropout rate στο 50%.

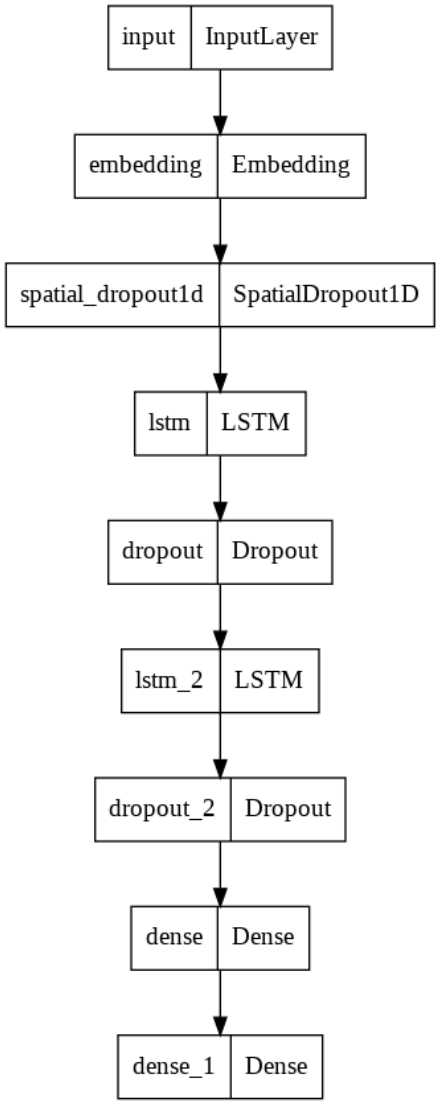
6. Εν συνεχεία, στον κατηγοριοποιητή εισάγεται ένα ακόμη Dropout με τα ίδια χαρακτηριστικά όπως προηγουμένως.

7. Στο βήμα αυτό, τα δεδομένα από την έξοδο του προηγούμενου επιπέδου εισάγονται σε ένα LSTM με τα ίδια στοιχεία με το βήμα 5, με τη μόνη διαφορά ότι ο αριθμός των νευρώνων είναι πλέον 20.

8. Στο σημείο αυτό γίνεται επανάληψη του βήματος 6, με την ενσωμάτωση ενός Dropout επιπέδου.

9. Ως τελευταίο επίπεδο εισάγεται ένα πλήρως διασυνδεδεμένο νευρωνικό δίκτυο με 2 νευρώνες εξόδου, όσες δηλαδή και οι κλάσεις που εξετάζονται. Με την ίδια λογική με προηγουμένως, ορίστηκε ως συνάρτηση ενεργοποίησης η “sigmoid”.

Η σχεδίαση του Word2Vec LSTM μοντέλου που περιγράφηκε παραπάνω απεικονίζεται στο ακόλουθο γράφημα.



Εικόνα 26: Γράφημα Word2Vec LSTM μοντέλου

4.4.4 FastText LSTM

Το τέταρτο μοντέλο που σχεδιάστηκε και αναλύεται σε αυτό το υποκεφάλαιο είναι ο FastText LSTM κατηγοριοποιητής. Η μόνη διαφορά με το προηγούμενο μοντέλο (Word2Vec LSTM) είναι η επιλογή της προ-εκπαιδευμένης διανυσματικής αναπαράστασης. Όμοια δοκιμάστηκαν 2 προσεγγίσεις με την αλλαγή του ορίσματος trainable = False ή trainable = True. Αυτή τη φορά, το αρχείο που χρησιμοποιήθηκε έχει διάσταση 300.

Όμοια με προηγουμένως, κατέβηκε το αρχείο και κατασκευάστηκε ένας πίνακας που περιέχει τις αναπαραστάσεις για εκείνες μόνο τις λέξεις που υπάρχουν στο σύνολο δεδομένων εκπαίδευσης του μοντέλου. Από τις συνολικά 2.000.000 λέξεις του αρχείου ταυτοποιήθηκαν 19146 από τις 26.995 του συνόλου δεδομένων.

4.4.5 BERT LSTM

Το επόμενο μοντέλο είναι ο κατηγοριοποιητής BERT BiLSTM. Ο συγκεκριμένος κατηγοριοποιητής βασίζει τη λειτουργία του στο νέο μοντέλο αναπαράστασης της γλώσσας BERT. Το μοντέλο BERT θα χρησιμοποιηθεί για την δημιουργία διανυσματικών αναπαραστάσεων για τις λέξεις των δεδομένων μας. Για να συμβεί αυτό, χρησιμοποιήθηκε το μοντέλο 'bert-base-greekuncased-v1' από τον σύνδεσμο huggingface.co, όπως αναφέρθηκε προηγουμένως. Η κλάση αυτή αρχικοποιείται ορίζοντας το URL που περιέχει τα στοιχεία του προ-εκπαιδευμένου κωδικοποιητή, συμπεριλαμβανομένου και τα βάρη του. Το embedding layer θα δημιουργηθεί από το μοντέλο TFBertModel, αφότου πάρει τις κατάλληλες εισόδους που θα δούμε στην συνέχεια και εκπαιδευτεί πλήρως, και θα αποτελέσει το πρώτο επίπεδο του μοντέλου μας το οποίο μπορεί να συνδυαστεί με άλλα επίπεδα και να διαμορφώσει ένα μοντέλο βαθιάς μάθησης.

Για την ολοκληρωμένη δημιουργία του embedding layer, ορίστηκε trainable = True, έτσι ώστε να περάσουμε στην δεύτερη φάση εκπαίδευσης της BERT αναπαράστασης, που ονομάζεται φάση βέλτιστης ρύθμισης παραμέτρων ("fine-tuning phase"). Στην συγκεκριμένη περίπτωση που εξετάζεται, τα δεδομένα που δόθηκαν στο BERT μοντέλο είναι τα δεδομένα πριν την προεπεξεργασία, με τον λόγο να εξηγείται στην αμέσως επόμενη παράγραφο.

Παράλληλα, τα δεδομένα αυτά δεν εισάγονται απευθείας στο μοντέλο, αλλά πρέπει πρώτα να πάρουν την κατάλληλη μορφή ώστε να εισαχθούν σε αυτό. Συγκεκριμένα, θα δημιουργηθούν 2 διαφορετικές είσοδοι από αυτά τα δεδομένα. Οι είσοδοι αυτές είναι: input_ids_in, input_masks_in και segment_ids, και θα περιγραφούν ακολούθως. Για την δημιουργία αυτών των εισόδων έγινε εφαρμογή κάποιων τεχνικών επεξεργασίας των δεδομένων, παρόμοιες με αυτές που αναφέρθηκαν στα προηγούμενα μοντέλα.

Αρχικά, έγινε χρήση του "FullTokenizer", δηλαδή του Tokenizer που έχει φτιαχτεί ειδικά για τις BERT αναπαραστάσεις και δεν λειτουργεί ακριβώς με τον ίδιο τρόπο με αυτόν του Tokenizer API της βιβλιοθήκης Keras. Λεπτομέρειες για τρόπο που υλοποιείται το tokenization στο BERT αναφέρονται στο κεφάλαιο 3.4.4. Αυτός, όμως, είναι και ο λόγος για τον οποίο στο συγκεκριμένο μοντέλο τα δεδομένα που χρησιμοποιήθηκαν ήταν αυτά πριν από την προ-επεξεργασία, λόγω του τελείως διαφορετικού τρόπου που πραγματοποιείται το tokenization με τον FullTokenizer. Για την εφαρμογή αυτού δίνεται πρόσβαση στο λεξιλόγιο του μοντέλου που κατεβάσαμε νωρίτερα, ενώ εισάγεται και η συνάρτηση εκείνη που μετατρέπει όλους τους χαρακτήρες από τους οποίους αποτελείται το dataset σε πεζούς. Εν συνεχεία, ο FullTokenizer εφαρμόζεται σε κάθε μη προ-επεξεργασμένο tweet.

Παράλληλα, ορίζεται το μέγιστο πλήθος λέξεων που αποτελούνται τα tweets. Ο αριθμός αυτός είναι τέτοιος ώστε να μην χρειάζεται να επιλεχθεί ένας μικρότερος αριθμός και να αφαιρεθούν λέξεις από το κείμενο του κάθε tweet. Όπως έχει αναφερθεί και προηγουμένως ο αριθμός αυτός είναι 26.

Στην συνέχεια, στην αρχή κάθε καταγραφής εισάγεται το token "[CLS]", ενώ στο τέλος το token "[SEP]" και ακολουθεί η εφαρμογή της συνάρτησης

`convert_tokens_to_ids()`, η αντίστοιχη συνάρτηση της `texts_to_sequences()` στη BERT αναπαράσταση. Με την εφαρμογή των ανωτέρω διαδικασιών δημιουργείται η πρώτη είσοδος input_ids_in. Συγκεκριμένα, η είσοδος αυτή αποτελείται από ένα σύνολο από αριθμούς που αντικαθιστούν κάθε λέξη του εκάστοτε tweet. Στην περίπτωση όπου ο αριθμός των λέξεων της καταγραφής είναι μικρότερος από το μέγιστο αριθμό που ορίστηκε προηγουμένως, για να συμπληρωθεί ο μέγιστος αριθμός εισάγονται στο τέλος της φράσης ένα σύνολο από τιμές “0” για εκείνες τις λέξεις που υπολείπονται. Η διαδικασία αυτή είναι ίδια με το post_padding που εφαρμόστηκε και στα προηγούμενα μοντέλα.

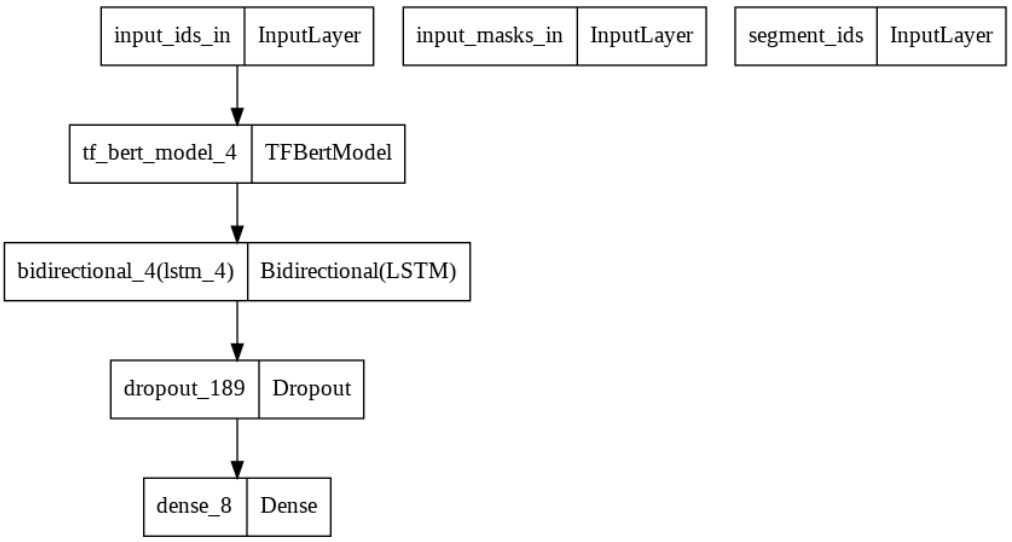
Για την δημιουργία της δεύτερης εισόδου input_masks_in, το array που υλοποιείται, αποτελείται απλά από σειρά από τις τιμές “1”, οι οποίες είναι τόσες όσες και ο αριθμός των λέξεων που αποτελείται το tweet. Ο αριθμός των λέξεων που υπολείπονται για την συμπλήρωση του μέγιστου αριθμού λέξεων κάθε καταγραφής συμπληρώνονται με “0”. Η συγκεκριμένη είσοδος έχει ως στόχο να υποδείξει ποια είναι εκείνα τα στοιχεία που αποτελούν τα tokens του tweet και ποια εκείνα που σχετίζονται με το padding.

Τέλος, η τρίτη είσοδος στο embedding layer είναι η segment_ids, η οποία χρησιμοποιείται για τη διάκριση μεταξύ διαφορετικών προτάσεων. Στην περίπτωση μας, κάθε tweet θεωρείται ως μία πρόταση και συνεπώς, για την δημιουργία αυτής της εισόδου το array αποτελείται από ένα σύνολο από “0”, με την τιμή αυτή να εμφανίζεται τόσες φορές όσες και ο μέγιστος αριθμός λέξεων. Με την ανωτέρω διαδικασία ολοκληρώνεται η προετοιμασία των δεδομένων για εισαγωγή στο embedding layer παράγοντας την BERT αναπαράσταση των λέξεων.

Το embedding layer παράγει, όμως, δύο εξόδους: την pooled_output και την sequence_output. Η πρώτη έξοδος περιέχει αναπαράσταση για ολόκληρη την φράση εισόδου, ενώ η δεύτερη την αναπαράσταση για κάθε token. Βάσει του σκοπού της παρούσας έρευνας, η έξοδος που μας ενδιαφέρει είναι η sequence_output. Συνεπώς, τα δεδομένα που εξάγονται από το sequence_output του προηγουμένου επιπέδου, εισάγονται σε ένα BiLSTM νευρωνικό. Συγκεκριμένα αποτελείται από 50 νευρώνες και το dropout rate τέθηκε στο 50%, όπως και στα προηγούμενα μοντέλα. Ακολούθως, η έξοδος του BiLSTM εισάγεται σε ένα Dropout επίπεδο, με τη συχνότητα να ορίζεται στο 50%, εξίσου.

Ως τελευταίο επίπεδο εισάγεται ένα πλήρως διασυνδεδεμένο νευρωνικό δίκτυο (Dense) με 2 νευρώνες εξόδου, όσες δηλαδή και οι κλάσεις που εξετάζονται. Με τον ίδια λογική με προηγουμένως, ορίστηκε ως συνάρτηση ενεργοποίησης η “sigmoid”.

Με την ολοκλήρωση της περιγραφής του BERT BiLSTM κατηγοριοποιητή, ακολουθεί το γράφημα που δείχνει την αρχιτεκτονική που περιγράφηκε, όπως την παρήγαγε η βιβλιοθήκη Keras:



Εικόνα 27: Γράφημα BERT BiLSTM μοντέλου

4.5 Παραδοσιακά μοντέλα

Στα πλαίσια ανάδειξης της υψηλής ικανότητας ανίχνευσης του συναισθήματος από τα LSTM μοντέλα σε συνδυασμό με την εισαγωγή προ-εκπαιδευμένων αναπαραστάσεων λέξεων, υλοποιήθηκε ένα σύνολο από κατηγοριοποιητές οι οποίοι βασίζονται σε πιο παραδοσιακούς αλγορίθμους μηχανικής μάθησης. Απώτερος σκοπός είναι η σύγκριση των αποτελεσμάτων και η εύρεση των καταλληλότερων τεχνικών σε θέματα επεξεργασίας φυσικής γλώσσας και την ανάλυσης συναισθήματος. Συγκεκριμένα οι κατηγοριοποιητές που υλοποιήθηκαν και θα αναλυθούν στην συνέχεια είναι οι εξής και έχουν περιγραφεί στο υποκεφάλαιο 3.2:

1. TF-IDF & Multinomial NaiveBayes
2. TF-IDF & Logistic Regression
3. TF-IDF & Support Vector Machines
4. BoW & Multinomial NaiveBayes
5. BoW & Logistic Regression
6. BoW & Support Vector Machines

4.5.1 TF-IDF & Multinomial NaiveBayes

Για την υλοποίηση της τεχνικής του TF-IDF στην έρευνα χρησιμοποιήθηκε η συνάρτηση TfidfVectorizer της βιβλιοθήκης sklearn. Με τη χρήση αυτής της συνάρτησης γίνεται μετατροπή μιας συλλογής πρωτοτύπων εγγράφων σε έναν πίνακα τιμών TF-IDF.

Στα ορίσματα της συγκεκριμένης συνάρτησης τέθηκαν οι τιμές: min_df = 2, max_df = 0.5 και ngram_range = (1, 1). Η παράμετρος max_df αποτελεί το κατώφλι που σύμφωνα με αυτό κατά τη δημιουργία του λεξιλογίου αγνοούνται όροι που έχουν συχνότητα εγγράφου αυστηρά υψηλότερη από το δεδομένο κατώφλι. Η παράμετρος min_df αγνοεί όρους που έχουν συχνότητα εγγράφου αυστηρά υψηλότερη από το δεδομένο κατώφλι. Τέλος, η παράμετρος ngram_range περιλαμβάνει το κατώτερο και ανώτερο όριο του εύρους των τιμών n για διαφορετικά n-gram που πρέπει να εξαχθούν. Όλες οι τιμές του n είναι τέτοιες έτσι ώστε να χρησιμοποιούνται min_n ≤ n ≤ max_n.

Στην συγκεκριμένη περίπτωση min_n = max_n = 1, δηλαδή δημιουργούνται μόνο uni-grams λόγω των μεγάλων απαιτήσεων μνήμης που δημιουργούνται όσο αυξάνονται τα n-grams. Τέλος, χρησιμοποιήθηκε η παράμετρος stopwords με όρισμα τις stopwords που έχουν βρεθεί από προηγούμενο βήμα.

Με τη δημιουργία του πίνακα που ανέθεσε σε κάθε λέξη του εκάστοτε tweet την τιμή TF-IDF που της αντιστοιχεί, τα δεδομένα αυτά εισήχθησαν σε ένα μοντέλο Multinomial NaiveBayes και χρησιμοποιήθηκε η συνάρτηση `MultinomialNB()` της βιβλιοθήκης sklearn.

4.5.2 TF-IDF & Logistic Regression

Στο μοντέλο αυτό, ακολουθήθηκε η ίδια τεχνική TF-IDF και οι ίδιες παράμετροι με προηγουμένως. Το μητρώο που δημιουργήθηκε εισήχθη στον ταξινομητή Logistic Regression και η συνάρτηση που χρησιμοποιήθηκε είναι η `LogisticRegression()` της βιβλιοθήκης sklearn.

4.5.3 TF-IDF & Support Vector Machines

Όπως και στα προηγούμενα δύο μοντέλα, έτσι και σε αυτό, η τεχνική μετατροπής των λέξεων σε αριθμούς είναι αυτή του TF-IDF, διατηρώντας ίδιες τις παραμέτρους για την δημιουργία του μητρώου. Η διαφορά σε αυτό το μοντέλο είναι ότι ο ταξινομητής που επιλέχθηκε είναι οι Μηχανές Διανυσμάτων Υποστήριξης - SVM. Η συνάρτηση που χρησιμοποιήθηκε είναι η `SVC()` της βιβλιοθήκης sklearn.

4.5.4 BoW & Multinomial NaiveBayes

Χρησιμοποιήθηκε η συνάρτηση CountVectorizer της βιβλιοθήκης sklearn για την υλοποίηση της τεχνικής του BoW. Με τη χρήση αυτής της συνάρτησης γίνεται μετατροπή μιας συλλογής πρωτοτύπων εγγράφων σε έναν πίνακα τιμών BoW.

Όμοια με την τεχνική TF-IDF, στα ορίσματα της συνάρτησης τέθηκαν οι τιμές: min_df = 2, max_df = 0.5 και ngram_range = (1, 1). Η παράμετρος max_df αποτελεί το κατώφλι που σύμφωνα με αυτό κατά τη δημιουργία του λεξιλογίου αγνοούνται όροι που έχουν συχνότητα εγγράφου αυστηρά υψηλότερη από το δεδομένο κατώφλι. Η παράμετρος min_df αγνοεί όρους που έχουν συχνότητα εγγράφου αυστηρά υψηλότερη από το δεδομένο κατώφλι. Τέλος, η παράμετρος ngram_range περιλαμβάνει το κατώτερο και ανώτερο όριο του εύρους των τιμών n για διαφορετικά n-gram που πρέπει να εξαχθούν. Όλες οι τιμές του n είναι τέτοιες έτσι ώστε να χρησιμοποιούνται min_n ≤ n ≤ max_n. Στην συγκεκριμένη περίπτωση min_n = max_n = 1, δηλαδή δημιουργούνται μόνο uni-grams λόγω των μεγάλων απαιτήσεων μνήμης που δημιουργούνται όσο αυξάνονται τα n-grams. Τέλος, χρησιμοποιήθηκε η παράμετρος stopwords με όρισμα τις stopwords που έχουν βρεθεί από προηγούμενο βήμα.

Τα δεδομένα αυτά εισήχθησαν σε ένα μοντέλο Multinomial NaiveBayes και χρησιμοποιήθηκε η συνάρτηση `MultinomialNB()`.

4.5.5 BoW & Logistic Regression

Στο μοντέλο αυτό, ακολουθήθηκε η ίδια τεχνική BoW και το μητρώο που δημιουργήθηκε εισήχθη στον ταξινομητή Logistic Regression και όμοια με προηγουμένως χρησιμοποιήθηκε η συνάρτηση `LogisticRegression()`.

4.5.6 BoW & Support Vector Machines

Όμοια, ακολουθήθηκε η ίδια τεχνική BoW και το μητρώο που δημιουργήθηκε εισήχθη στον ταξινομητή BoW και όμοια χρησιμοποιήθηκε η συνάρτηση `SVC()`.

4.6 Διαδικασία εκπαίδευσης μοντέλων

4.6.1 Συνάρτηση compile()

Μετά τον πλήρη σχεδιασμό όλων των LSTM μοντέλων της έρευνας, καλείται η συνάρτηση `compile()` της βιβλιοθήκης Keras, η οποία αναλαμβάνει τη διαμόρφωση του μοντέλου για εκπαίδευση.

Στα ορίσματα της συγκεκριμένης συνάρτησης τέθηκε ως συνάρτηση απωλειών (loss function) η binary crossentropy. Η συγκεκριμένη συνάρτηση χρησιμοποιείται για την δημιουργία κατηγοριοποιητών με δύο κλάσεις. Παράλληλα, ως βελτιστοποιητής (optimizer) του μοντέλου ορίστηκε ο adam, με τον ρυθμό εκμάθησης (learning rate) να διαφέρει ανά μοντέλο. Ο adam βελτιστοποιητής χρησιμοποιείται ευρέως σε προβλήματα βαθιάς μάθησης. Ταυτόχρονα, ως όρισμα της μετρικής που αξιολογείται κατά τη διάρκεια της εκπαίδευσης και της αξιολόγησης του μοντέλου επιλέχθηκε να είναι το accuracy. Η μετρική accuracy, ή αλλιώς ορθότητα, υπολογίζει πόσο συχνά οι προβλέψεις ισούνται με την πραγματική τιμή. Οι ανωτέρω πληροφορίες συνοψίζονται στον ακόλουθο πίνακα:

Πίνακας 5: Τελική διαμόρφωση μοντέλων για την εκπαίδευση

Συνάρτηση απωλειών	*binary crossentropy*
Βελτιστοποιητής	*adam*
Μετρική Αξιολόγησης	*accuracy*

4.6.2 Συνάρτηση fit()

Η συνάρτηση `fit()` είναι εκείνη η οποία αναλαμβάνει να εκπαιδεύσει το εκάστοτε μοντέλο. Για να συμβεί αυτό είναι απαραίτητη η ρύθμιση κάποιων παραμέτρων που οδηγούν στη δημιουργία του βέλτιστου κατηγοριοποιητή.

Αρχικά, πρέπει να εισαχθούν τα δεδομένα εισόδου. Για τα μοντέλα Simple LSTM, Word2Vec LSTM, FastText LSTM και BiLSTM τα δεδομένα που εισάγονται είναι τα προ-επεξεργασμένα δεδομένα κειμένου μετά την εφαρμογή των διεργασιών του Tokenizer API και του padding. Αντίστοιχα για τον BERT BiLSTM κατηγοριοποιητή είναι εκείνα τα δεδομένα που προέκυψαν με χρήση του BertTokenizer και της διαδικασίας του padding στα προ-επεξεργασμένα tweets. Τέλος, για τον BERT BiLSTM κατηγοριοποιητή τα δεδομένα εισόδου αποτέλεσαν τα input_word_ids και input_mask.

Στην συνέχεια, ορίστηκαν τα δεδομένα στόχου, όπου στην περίπτωση μας είναι το αν το tweet που εξετάζεται είναι σαρκαστικό ή όχι και το καθένα από αυτά μετατρέπεται σε μία αναπαράσταση από 1 ψηφίο (1 ή 0 αντίστοιχα).

Επόμενη παράμετρος που ρυθμίζεται είναι αυτή των validation data. Τα validation δεδομένα πρέπει να περιέχουν ένα μέρος των δεδομένων εκπαίδευσης, μαζί με την μεταβλητή του σαρκασμού (τιμή στόχος). Η συγκεκριμένη παράμετρος έχει

προσδιοριστεί με χρήση της συνάρτησης `train_test_split()`, όπου τα δεδομένα διαχωρίστηκαν στα δεδομένα train και validation data. Σε όλες τις περιπτώσεις, το 20% των δεδομένων αποτέλεσε το σετ των validation δεδομένων και το υπόλοιπο 80% τα train δεδομένα.

Επιπλέον, η μεταβλητή των epochs (εποχών) ρυθμίστηκε σε αυτό το σημείο. Ο αριθμός των εποχών που ορίζονται στο σύστημα σχετίζεται με τον αριθμό των επαναλήψεων πάνω στο σύνολο δεδομένων εκπαίδευσης του μοντέλου. Στα μοντέλα Simple LSTM, BiLSTM, Word2Vec LSTM και FastText LSTM πήραν ως αριθμό εποχών την τιμή 100, εφόσον χρειάζονται πολλές επαναλήψεις πάνω στα δεδομένα για να εκπαιδευτεί το μοντέλο. Αντίθετα, στο μοντέλο BERT BiLSTM, λόγω της πολυπλοκότητας και της υπολογιστικής ισχύς που απαιτείται, οι 8 εποχές είναι αρκετές για την εκμάθηση των παραμέτρων του κατηγοριοποιητή. Σημειώνεται, βέβαια, πως δεν είναι αναγκαίο το μοντέλο να εξαντλήσει τον αριθμό των εποχών ώστε να εκπαιδευτεί βέλτιστα. Για το λόγο αυτό, η συγκεκριμένη μεταβλητή ελέγχεται και από ένα callback που θα εξηγηθεί στην συνέχεια, δίνοντας τη δυνατότητα πρόωρης διακοπής της εκπαιδευτικής διαδικασίας πριν τον προβλεπόμενο αριθμό εποχών.

Παράλληλα, μία παράμετρος απαραίτητη για την εκπαίδευση του νευρωνικού είναι το batchsize. Η παράμετρος αυτή είναι ίση με το πλήθος των δειγμάτων που εισάγονται κάθε φορά στο νευρωνικό για τη ρύθμιση των βαρών στους νευρώνες. Με βάση το πλήθος των δεδομένων εκπαίδευσης, μία εποχή αποτελείται από ένα σύνολο από batch size αριθμό δειγμάτων. Η συγκεκριμένη παράμετρος σε όλα τα μοντέλα που παρουσιάστηκαν έχει πάρει την τιμή 32.

Στο σημείο αυτό εξετάζεται η παράμετρος callback που μπορεί να εφαρμοστεί στην συνάρτηση `fit()`. Τα callbacks είναι ένα σύνολο συναρτήσεων που δύνανται να εφαρμοστούν σε διάφορα στάδια κατά τη διάρκεια της εκπαίδευσης και μέσω αυτών είναι εφικτό να αυτοματοποιούνται κάποιες διαδικασίες. Σε όλα τα μοντέλα της παρούσας έρευνας χρησιμοποιήθηκαν 2 τέτοιες συναρτήσεις: το EarlyStopping και το ModelCheckpoint.

Το EarlyStopping είναι ένα callback που εφαρμόζεται για την αποφυγή της υπερπροσαρμογής και δίνει την δυνατότητα να σταματάει η διαδικασία της εκμάθησης των βαρών πριν φτάσει το μοντέλο το μέγιστο αριθμό εποχών που θέσαμε νωρίτερα. Δίνοντας στην EarlyStopping συνάρτηση ένα σύνολο από ορίσματα αποφασίζεται βάσει ποιων συνθηκών η εκπαίδευση του μοντέλου πρέπει να σταματήσει.

Στην περίπτωση της παρούσας έρευνας επιθυμείται η διακοπή της ανωτέρω διαδικασίας όταν οι απώλειες στα validation δεδομένα έχουν σταματήσει να μειώνονται. Συγκεκριμένα, αν παρατηρηθεί ότι οι απώλειες ενδιαφέροντος εξακολουθούν να μην μειώνονται σε βάθος 7 εποχών, τότε η διαδικασία εκπαίδευσης τερματίζεται. Είναι απαραίτητη η εφαρμογή του συγκεκριμένου callback, καθώς από ένα σημείο και μετά παρατηρείται το φαινόμενο της υπερπροσαρμογής, όπου οι απώλειες στα train δεδομένα μειώνονται ενώ στα validation παραμένει σταθερό. Αυτό συνεπάγεται ότι το μοντέλο μαθαίνει “απ΄ έξω” την πληροφορία που περιέχεται στα δεδομένα εκπαίδευσης, χωρίς να έχει τη δυνατότητα να τα γενικεύει.

Το callback ModelCheckpoint σχετίζεται με την αποθήκευση του μοντέλου και των βαρών του βάσει συγκεκριμένων συνθηκών που ορίζει ο χρήστης ανάλογα τις ανάγκες. Στην συγκεκριμένη περίπτωση γίνεται αποθήκευση του μοντέλου κάθε φορά που παρατηρείται αύξηση της ακρίβειας στα validation δεδομένα και όχι σε κάθε εποχή εκπαίδευσης. Με αυτό τον τρόπο, κάθε φορά που αυξάνεται η ακρίβεια γίνεται αντικατάσταση των χαρακτηριστικών του μοντέλου με αυτά που έχει όταν αυξήθηκε η ακρίβεια.

Για τα παραδοσιακά μοντέλα χρησιμοποιήθηκε όμοια η συνάρτηση `fit()` του εκάστοτε αλγορίθμου χωρίς ορίσματα.

4.6.3 Συνάρτηση predict()

Μετά την ολοκλήρωση της διαδικασίας εκμάθησης, ακολουθεί η αξιολόγηση των μοντέλων που δημιουργήθηκαν παραπάνω. Μέσω αυτής της διαδικασίας γίνεται δοκιμή των κατηγοριοποιητών, καθώς καλούνται να ανιχνεύσουν τον σαρκασμό σε δεδομένα που δεν έχουν ξαναδεί. Για την εφαρμογή αυτής της διεργασίας είναι απαραίτητη η κλήση της συνάρτησης `predict()`.

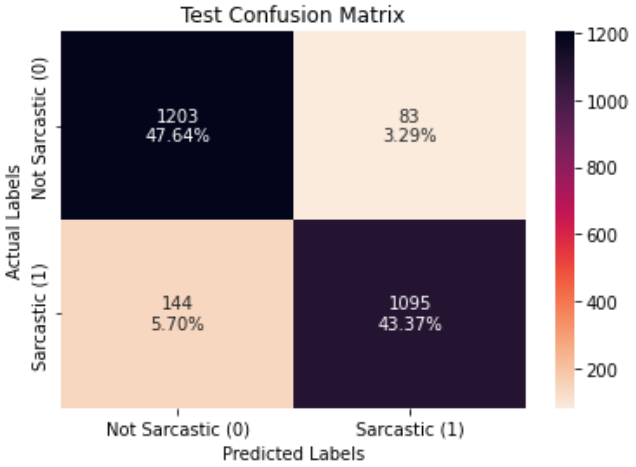
Η συγκεκριμένη συνάρτηση ανήκει στα Model training APIs της βιβλιοθήκης Keras, μαζί με τις `compile()` και `fit()`. Η `predict()` λαμβάνει στην είσοδό της ένα σύνολο δεδομένων και καλείται να εντοπίσει την ύπαρξη σαρκασμού σε αυτά. ΄Οπως και στην `fit()`, έτσι και εδώ, πρέπει να πραγματοποιηθούν κάποιες αλλαγές στην μορφή των δεδομένων test έτσι ώστε να μπορέσει να κληθεί η προαναφερθείσα συνάρτηση. Για όλα τα μοντέλα γίνεται εφαρμογή όλων των βημάτων προ-επεξεργασίας των δεδομένων όπως περιγράφηκε αναλυτικά στο κεφάλαιο 4.2.1. Αντίστοιχα με τα δεδομένα train, στο μοντέλο BERT BiLSTM στα tweets έγινε επιπλέον αφαίρεση όλων των διακριτικών.

Παράλληλα, σε όλα τα μοντέλα ακολούθησε η διεργασία μετατροπής των λέξεων σε μία σειρά από ακέραιους αριθμούς (είτε με χρήση του Tokenizer API είτε του BertTokenizer, ανάλογα το μοντέλο), καθώς και αυτή του padding. Με τις διεργασίες αυτές τα test δεδομένα έχουν πάρει την κατάλληλη μορφή ώστε να γίνει κλήση της συνάρτησης `predict()` και να γίνει αξιολόγηση των αποτελεσμάτων των μοντέλων που υλοποιήθηκαν. Τα αποτελέσματα που προέκυψαν παρουσιάζονται στο επόμενο κεφάλαιο.

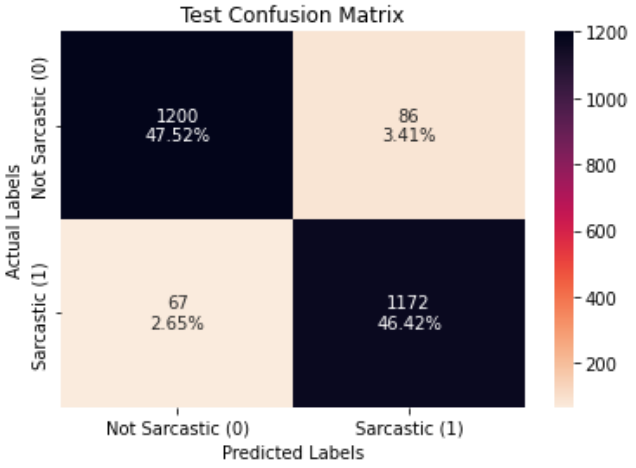
5. ΑΠΟΤΕΛΕΣΜΑΤΑ & ΑΞΙΟΛΟΓΗΣΗ ΜΟΝΤΕΛΩΝ

5.1 Πίνακες Σύγχυσης

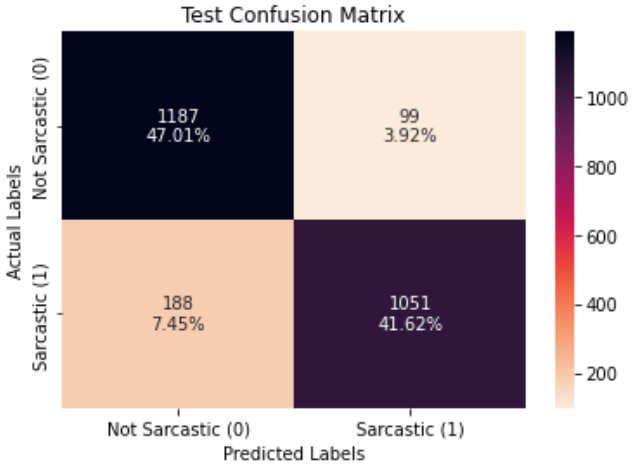
Ακολουθεί η διαδικασία αξιολόγησης των μοντέλων που παρουσιάστηκαν σε test δεδομένα ως προς την ικανότητά τους στην ανίχνευση του σαρκασμού. Παρακάτω, παρουσιάζονται οι πίνακες σύγχυσης από το test set και αφορούν στον ποσοστιαίο διαχωρισμό 80% train - 20% test:



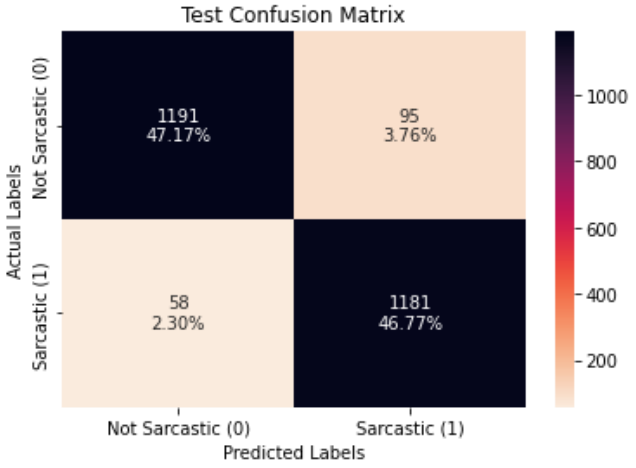
Εικόνα 28: Confusion matrix Simple LSTM μοντέλου



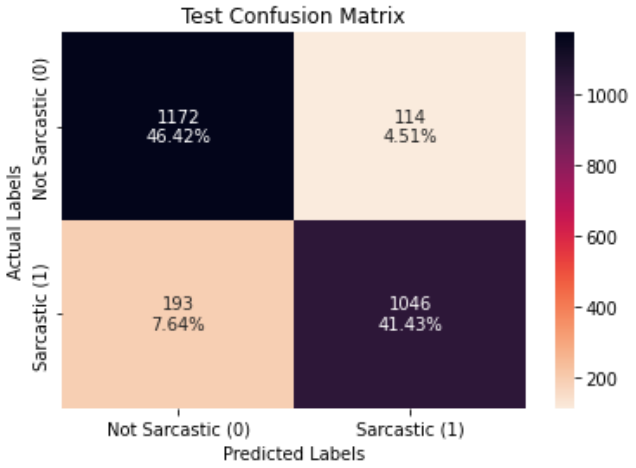
Εικόνα 29: Confusion matrix BiLSTM μοντέλου



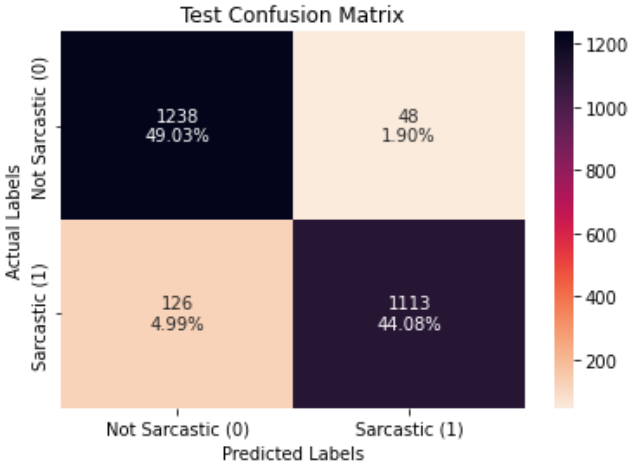
Εικόνα 30: Confusion matrix Word2Vec LSTM μοντέλου (trainable = False)



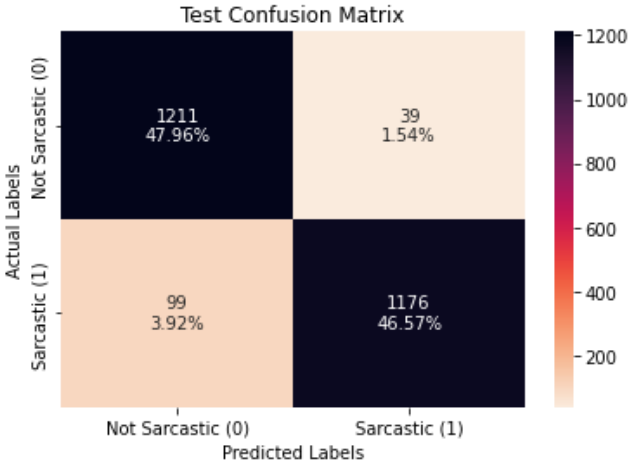
Εικόνα 31: Confusion matrix Word2Vec LSTM μοντέλου (trainable = True)



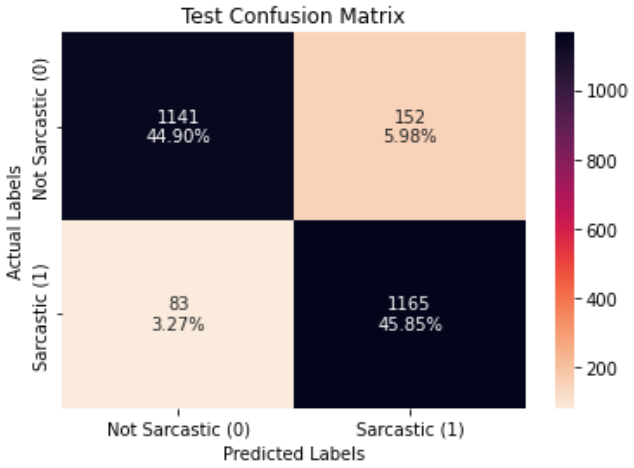
Εικόνα 32: Confusion matrix Word2Vec LSTM μοντέλου (trainable = True)



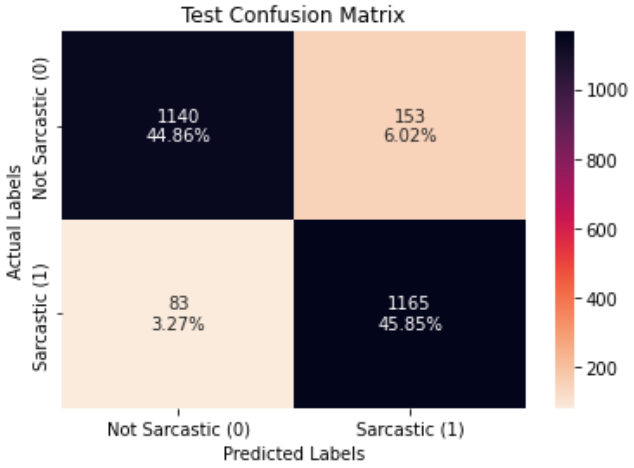
Εικόνα 33: Confusion matrix FastText LSTM μοντέλου (trainable = True)



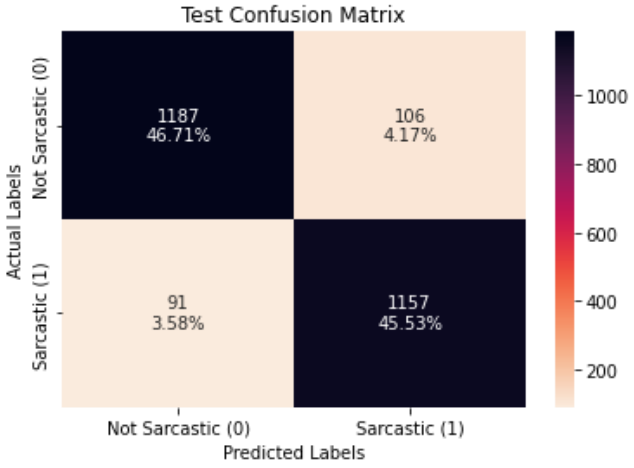
Εικόνα 34: Confusion matrix BERT BiLSTM μοντέλου



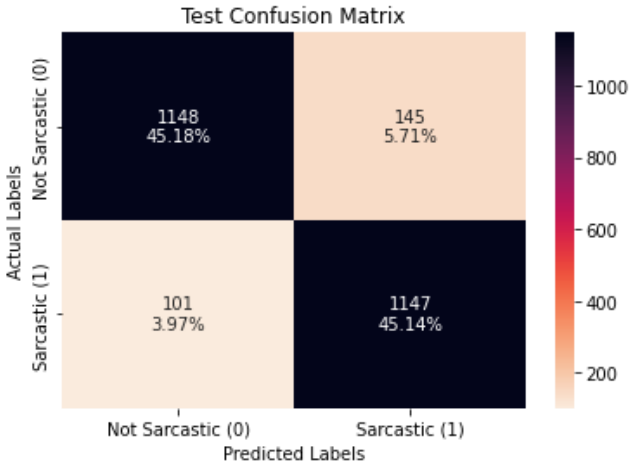
Εικόνα 35: Confusion matrix TF-IDF & Naive Bayes μοντέλου



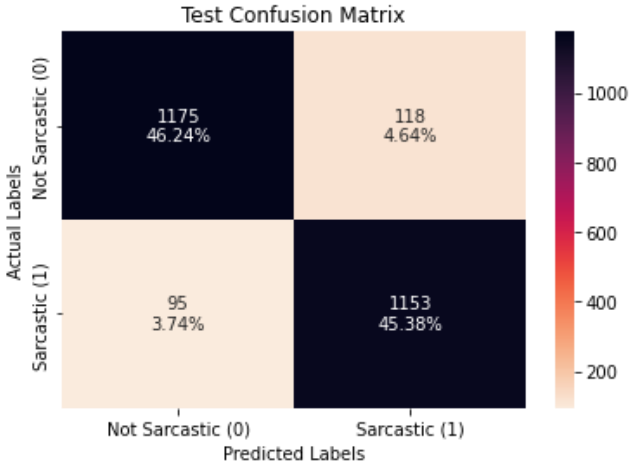
Εικόνα 36: Confusion matrix TF-IDF & Logistic Regression μοντέλου



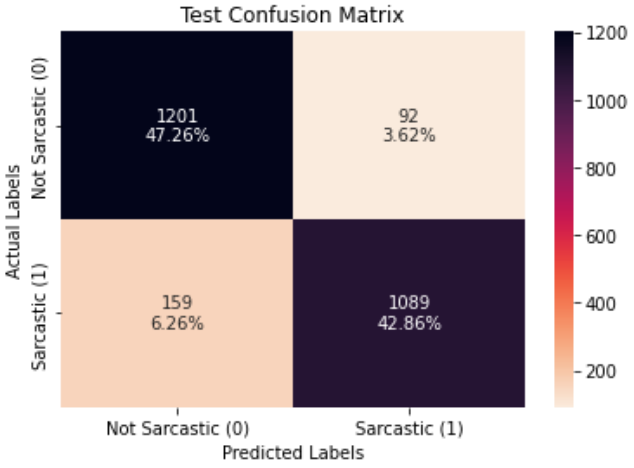
Εικόνα 37: Confusion matrix TF-IDF & SVM μοντέλου



Εικόνα 38: Confusion matrix BoW & Naive Bayes μοντέλου



Εικόνα 39: Confusion matrix BoW & Logistic Regression μοντέλου

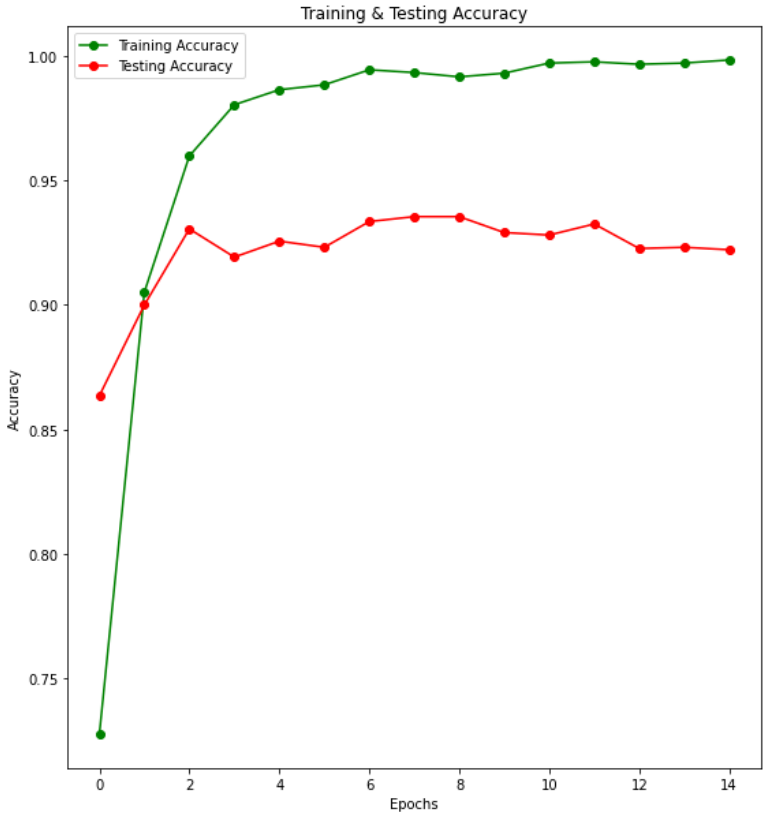


Εικόνα 40: Confusion matrix BoW & SVM μοντέλου

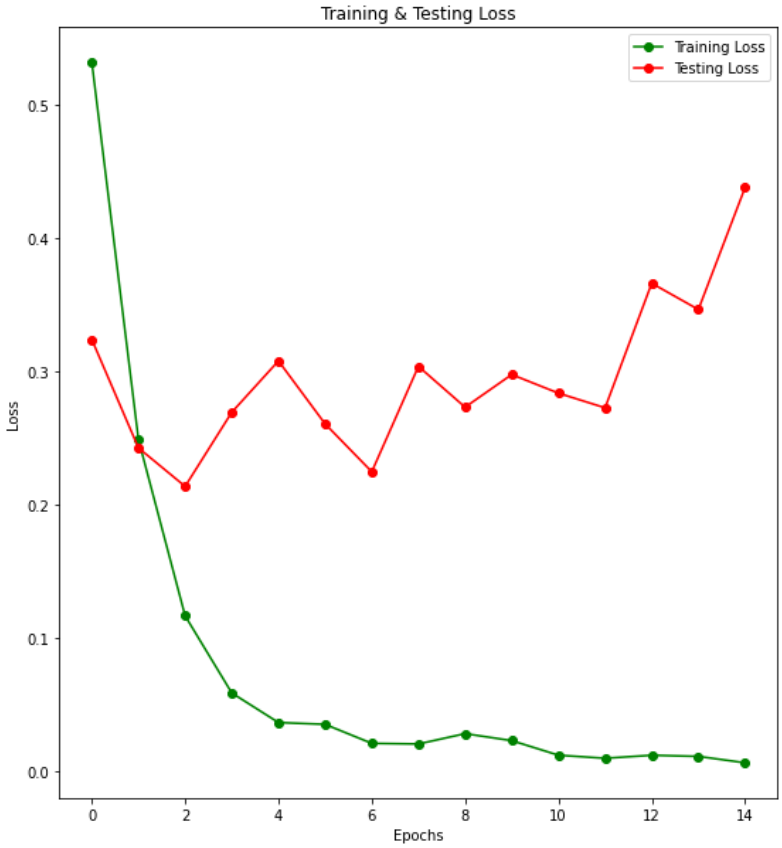
Από τα ανωτέρω σχήματα, το πλήθος των δειγμάτων που κατηγοριοποιήθηκαν σωστά ως σαρκαστικά (θετικά) είναι αυτά που ανήκουν στην πάνω αριστερά γωνία του πίνακα σύγχυσης (confusion matrix). Το σύνολο αυτών των δειγμάτων ονομάζονται True Positive (TP) (Actual Label = Predicted Label = Sarcastic). Αντίστοιχα το πλήθος των δειγμάτων που κατηγοριοποιήθηκαν σωστά ως μη σαρκαστικά (αρνητικά) ονομάζονται True Negative (TN) (Actual Label = Predicted Label = Not Sarcastic).

Αντίθετα, το σύνολο εκείνων των δειγμάτων που θεωρήθηκε από τον κατηγοριοποιητή ότι είναι σαρκαστικά, ενώ δεν είναι πραγματικά, ονομάζονται False Positive (FP) (Actual Label- Not Sarcastic ≠ Predicted Label-Sarcastic). Παράλληλα, εκείνα τα στοιχεία που ταξινομήθηκαν ως μη σαρκαστικά ενώ είναι στην πραγματικότητα, ονομάζονται False Negative (FN) (Actual Label-Sarcastic ≠ Predicted Label-Not Sarcastic).

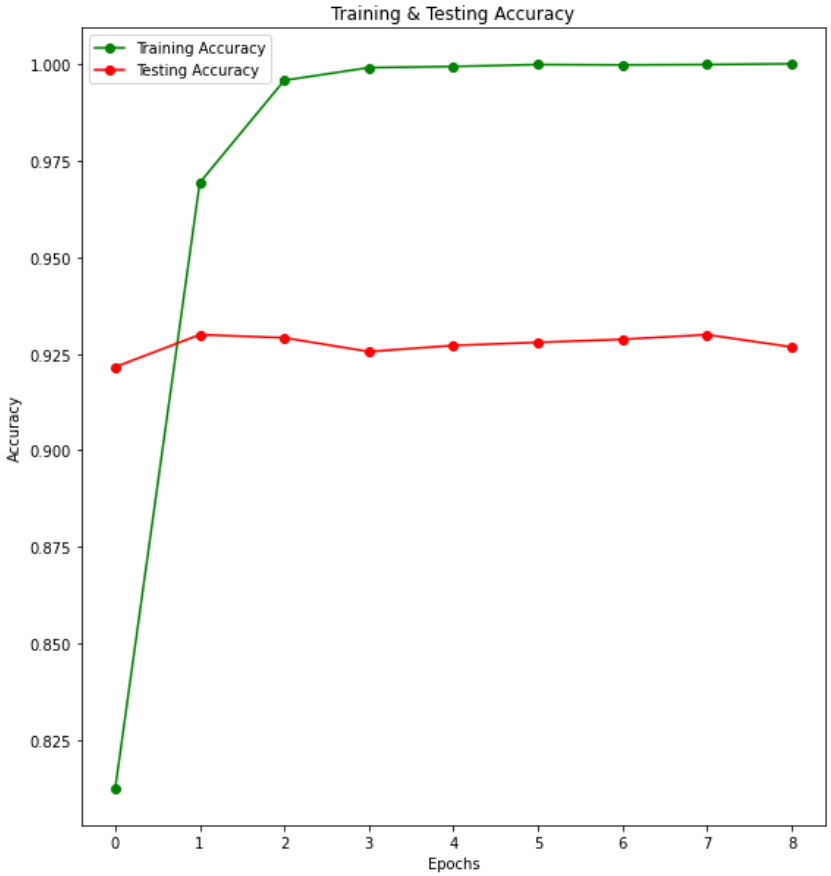
5.2 Γραφήματα εκπαίδευσης των μοντέλων



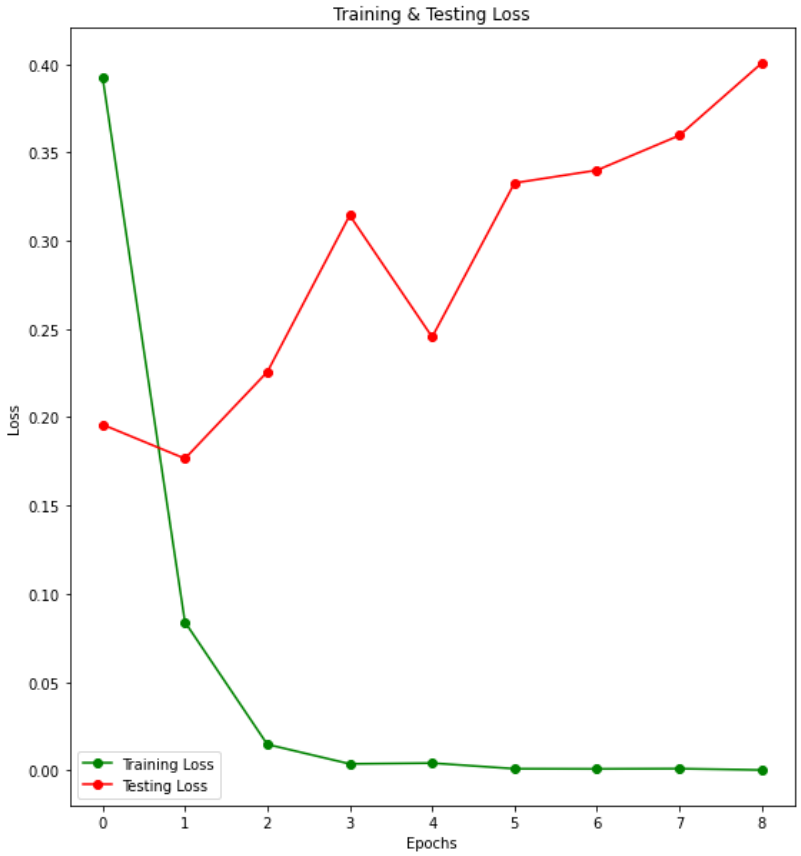
Εικόνα 41: Απεικόνιση της ακρίβειας του μοντέλου LSTM σε συνάρτηση με το epoch κατά την διάρκεια εκπαίδευσης



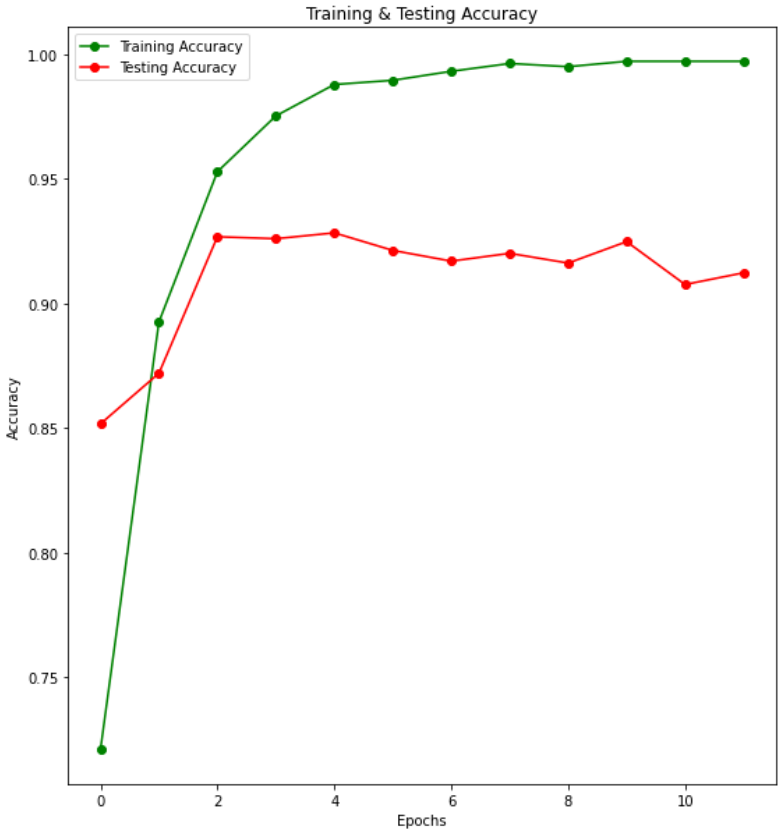
Εικόνα 42: Απεικόνιση της απώλειας του μοντέλου LSTM σε συνάρτηση με το epoch κατά την διάρκεια εκπαίδευσης



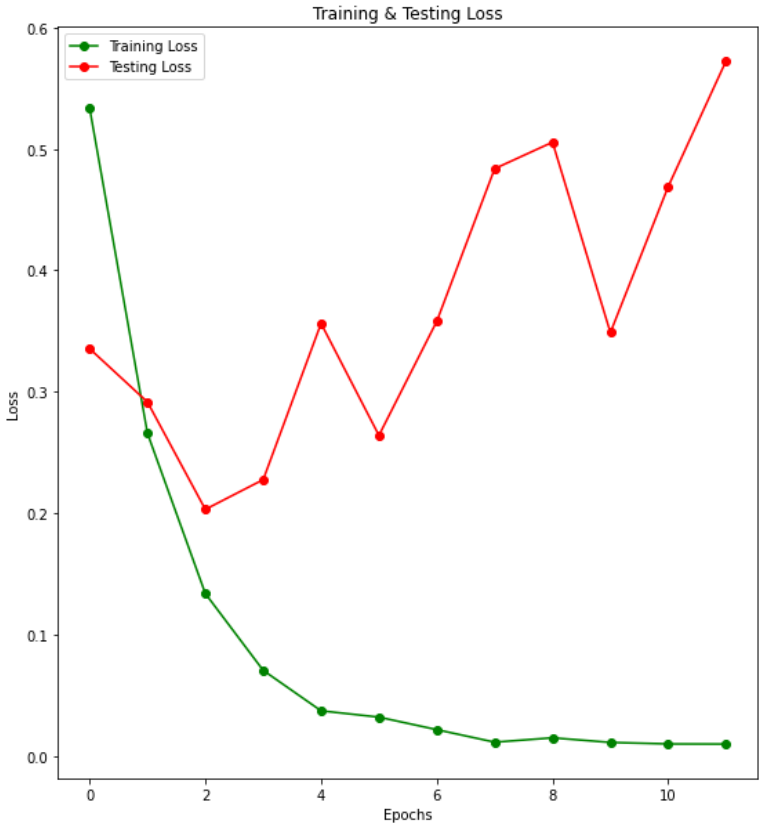
Εικόνα 43: Απεικόνιση της ακρίβειας του μοντέλου BiLSTM σε συνάρτηση με το epoch κατά την διάρκεια εκπαίδευσης



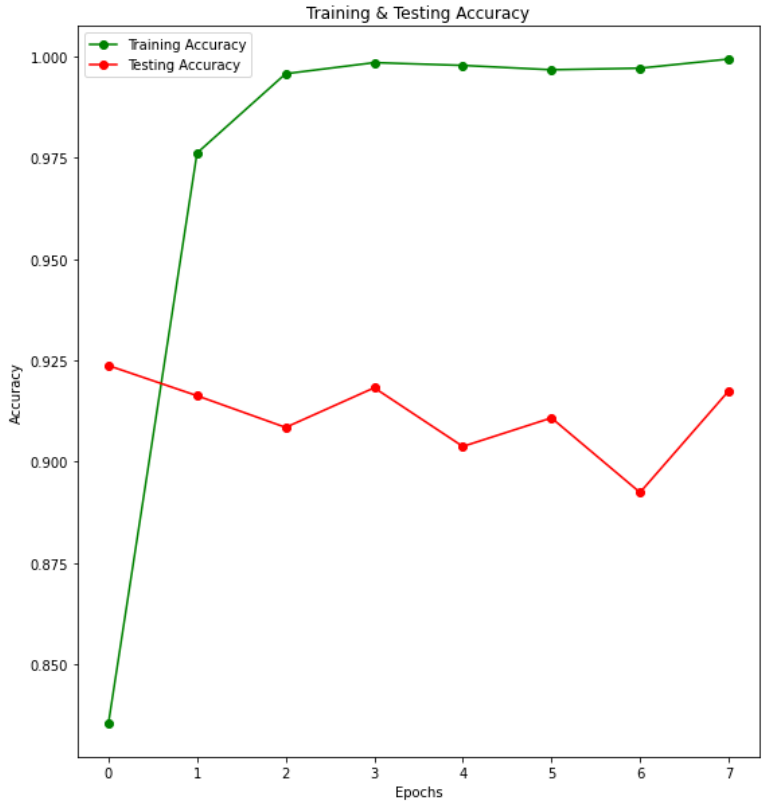
Εικόνα 44: Απεικόνιση της απώλειας του μοντέλου BiLSTM σε συνάρτηση με το epoch κατά την διάρκεια εκπαίδευσης



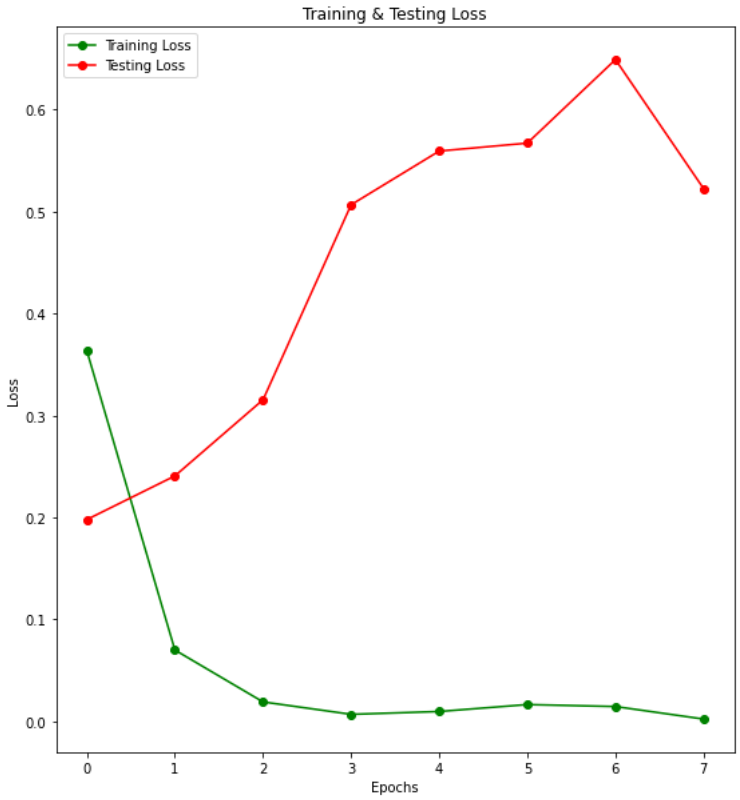
Εικόνα 45: Απεικόνιση της ακρίβειας του μοντέλου Word2Vec LSTM σε συνάρτηση με το epoch κατά την διάρκεια εκπαίδευσης trainable = True



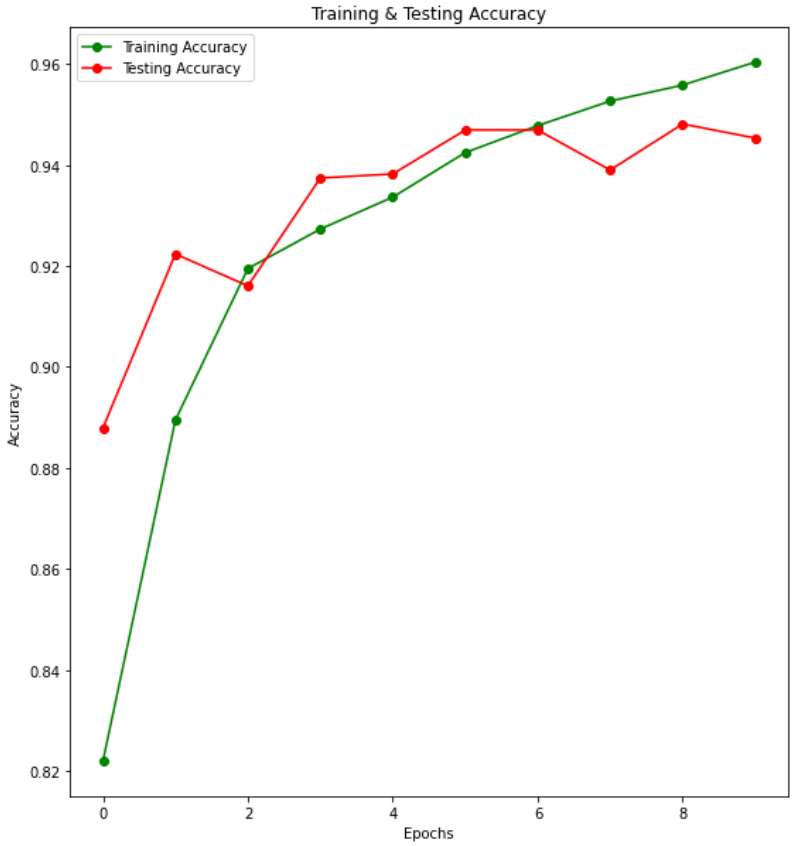
Εικόνα 46: Απεικόνιση της απώλειας του μοντέλου Word2Vec LSTM σε συνάρτηση με το epoch κατά την διάρκεια εκπαίδευσης trainable = True



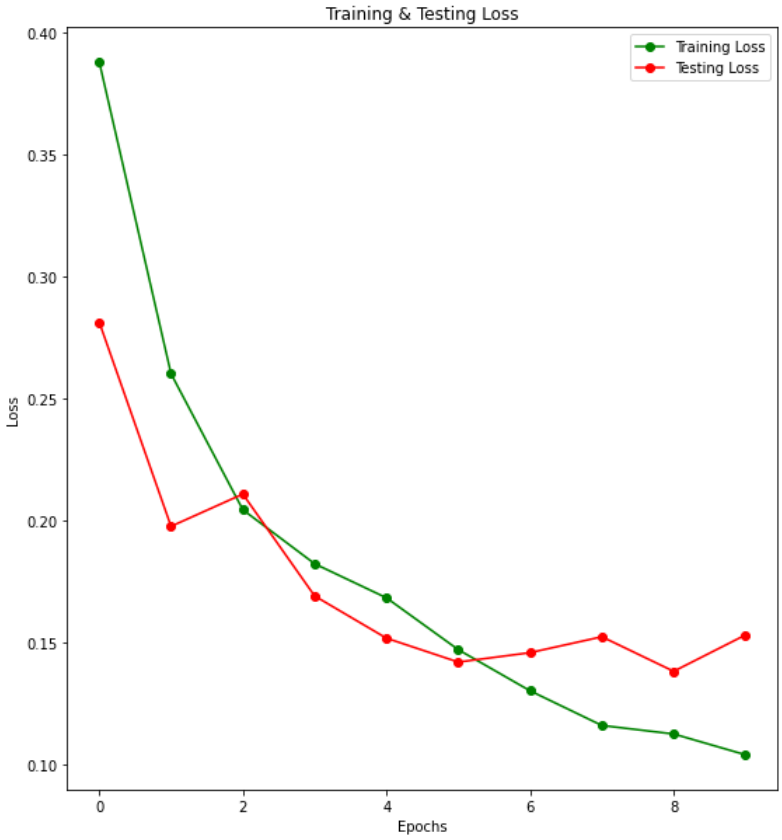
Εικόνα 47: Απεικόνιση της ακρίβειας του μοντέλου FastText LSTM σε συνάρτηση με το epoch κατά την διάρκεια εκπαίδευσης trainable = True



Εικόνα 48: Απεικόνιση της απώλειας του μοντέλου FastText LSTM σε συνάρτηση με το epoch κατά την διάρκεια εκπαίδευσης trainable = True



Εικόνα 49: Απεικόνιση της ακρίβειας του μοντέλου BERT BiLSTM σε συνάρτηση με το epoch κατά την διάρκεια εκπαίδευσης



Εικόνα 50: Απεικόνιση της απώλειας του μοντέλου BERT BiLSTM σε συνάρτηση με το epoch κατά την διάρκεια εκπαίδευσης

5.3 Μετρικές Αξιολόγησης

Οι μετρικές που επιλέχθηκαν για να γίνει αξιολόγηση της απόδοσης του συστήματος αποτελούν σημαντικό κομμάτι της διαδικασίας καθώς επηρεάζουν άμεσα τα συμπεράσματά που εξάγονται για κάποιο μοντέλο και μπορεί να διαφέρουν ανάλογα με την εφαρμογή και τον σκοπό της έρευνας.

Αυτές που χρησιμοποιήθηκαν για την παρούσα διπλωματική, όπως περιγράφονται από τις Vijayarani και Janani [112], είναι οι ακόλουθες:

- **Ακρίβεια - Ορθότητα (Accuracy)**: Ο λόγος του αριθμού των σωστών προβλέψεων του μοντέλου (True Positive, True Negative) με τον συνολικό αριθμό δειγμάτων εισόδου (True Positive, True Negative, False Positive, False Negative). Όσο καλύτερα είναι διανεμημένα τα δεδομένα στις κλάσεις τόσο πιο αξιόπιστη είναι.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

- **Πιστότητα (Precision)**: Φανερώνει το ποσοστό των δειγμάτων που ταξινομήθηκαν σωστά από όλα τα θετικά δείγματα (True Positive). Για παράδειγμα σε μια πρόβλεψη για ασθενείς με καρκίνο, το μοντέλο ταξινομεί κάποιους ασθενείς σαν παθόντες σωστά και κάποιους λανθασμένα. Ο λόγος του αριθμού των παθόντων που προέβλεψε το μοντέλο προς τον αριθμό των πραγματικά παθόντων (True Positive, False Positive) δίνει την ακρίβεια του. Η μετρική αυτή εστιάζει στα ψευδή θετικά (False Positive), δηλαδή στα δείγματα που ταξινομεί λανθασμένα ως θετικά.

$$\text{Precision} = \frac{TP}{TP + FP}$$

- **Ανάκληση (Recall)**: Με βάση το προηγούμενο παράδειγμα, για την ανάκληση υπολογίζουμε το λόγο του αριθμού των ατόμων που έχουν καρκίνο (True Positive) προς τον συνολικό αριθμό των ατόμων που κατέληξε το μοντέλο ότι έχουν (True Positive, False Negative). Αντίστοιχα, αυτή η μετρική δίνει έμφαση στα δείγματα που απέτυχε να ταξινομήσει σωστά ο αλγόριθμος.

$$\text{Recall} = \frac{TP}{TP + FN}$$

- **F1-Score**: Μετρική που συνδυάζει τις ακρίβεια και ανάκληση υπολογίζοντας τον σταθμισμένο μέσο. Ο μαθηματικός τύπος της εκφράζεται ως εξής:

$$F1 = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{TP}{TP + \frac{1}{2}(FP + FN)}$$

5.4 Αποτελέσματα

Στην παρούσα έρευνα η μετρική που μας ενδιαφέρει περισσότερο είναι αυτή της ακρίβειας (accuracy). Ο πίνακας που ακολουθεί παρουσιάζει τα αποτελέσματα που προέκυψαν ως προς αυτή την μετρική για το κάθε μοντέλο για διαφορετικά ποσοστά δεδομένων στα σύνολα train/validation. Συνολικά, παρατίθενται οι εξής ποσοστιαίοι διαχωρισμοί:

- 70% train/validation - 30% test
- 80% train/validation - 20% test
- 90% train/validation - 10% test

Πίνακας 6: Ακρίβεια κατηγοριοποίησης των μοντέλων για όλους τους ποσοστιαίους διαχωρισμούς

Μοντέλο	Accuracy (70-30)	Accuracy (80-20)	Accuracy (90-10)
Simple LSTM	90.05%	91.09%	91.61%
BiLSTM	93.13%	93.58%	93.99%
Word2Vec LSTM (trainable=False)	87.06%	88.63%	87.63%
Word2Vec LSTM (trainable=True)	92.80%	93.03%	92.78%
FastText LSTM (trainable=False)	85.83%	87.84%	86.79%
FastText LSTM (trainable=True)	92.53%	92.88%	92.59%
BERT BiLSTM	94.18%	94.53%	94.59%
TF-IDF & Naive Bayes	89.56%	90.75%	90.62%
TF-IDF & Logistic Regression	90.16%	90.71%	91.02%
TF-IDF & SVM	89.29%	90.24%	91.26%
BoW & Naive Bayes	90.18%	90.33%	91.10%
BoW & Logistic Regression	90.03%	90.65%	91.37%
TF-IDF & Logistic Regression	89.72%	90.12%	89.95%

Βάσει των αποτελεσμάτων που προέκυψαν, αρχικά παρατηρούμε ότι ο ποσοστιαίος διαχωρισμός 90-10 παρουσιάζει, σε γενικές γραμμές, τα καλύτερα αποτελέσματα. Στη συνέχεια, παρατίθενται τα αποτελέσματα που προέκυψαν από αυτόν τον διαχωρισμό.

Ο κατηγοριοποιητής με την μεγαλύτερη ακρίβεια είναι ο BERT BiLSTM, με ποσοστό που άγγιξε το 94.59%, έχοντας μικρή απόσταση σε σχέση με τον δεύτερο καλύτερο κατηγοριοποιητή, τον BiLSTM με ποσοστό 93.99%. Μετά ακολουθούν τα μοντέλα Word2Vec LSTM και FastText LSTM με παράμετρο trainable = True με ποσοστά ακρίβειας 92.78 % και 92.59% αντίστοιχα. Με πολύ μικρή διαφορά, ακολουθεί ο Simple LSTM κατηγοριοποιητής με ακρίβειας 91.61%.

Τα παραδοσιακά μοντέλα BoW & Naive Bayes, BoW & Logistic Regression και BoW & SVM, πέτυχαν ποσοστά ακρίβειας 91.10%, 91.37% και 89.95% αντίστοιχα, και τα μοντέλα TFIDF & Naive Bayes, TF-IDF & Logistic Regression και TF-IDF & SVM πέτυχαν ποσοστά 90.62%, 91.02% και 91.26% αντίστοιχα. Δεν παρατηρείται να συνεισέφερε θετικά στα αποτελέσματα κάποιος από τους δύο τρόπους αναπαράστασης λέξεων (BoW ή TF-IDF), όπως επίσης και καμία τεχνική μηχανικής μάθησης δεν φαίνεται να πέτυχε καλύτερα αποτελέσματα από τις άλλες (Logistic Regression, Naive Bayes ή SVM).

Τα μοντέλα με την μικρότερη επιτυχία ως προς την μετρική ενδιαφέροντος είναι τα μοντέλα Word2Vec LSTM και FastText LSTM με παράμετρο trainable = False με ποσοστά: 87.63% και 86.79%, αντίστοιχα.

Οι πίνακες που ακολουθούν παρουσιάζουν τις μέσες τιμές των μετρικών Precision, Recall και F1 Score για το κάθε μοντέλο για τους παραπάνω ποσοστιαίους διαχωρισμούς (70/30, 80/20 και 90/10). Όπως και προηγουμένως, έτσι και εδώ, φαίνεται ότι ο BERT BiLSTM κατηγοριοποιητής βγάζει τα καλύτερα αποτελέσματα.

Πίνακας 7: Αξιολόγηση μοντέλων με ποσοστό διαχωρισμού 70/30

Μοντέλο	Precision	Recall	F1 Score
Simple LSTM	0.92	0.92	0.92
BiLSTM	0.93	0.93	0.93
Word2Vec LSTM (trainable=False)	0.87	0.87	0.87
Word2Vec LSTM (trainable=True)	0.93	0.93	0.93
FastText LSTM (trainable=False)	0.86	0.86	0.86
FastText LSTM (trainable=True)	0.93	0.93	0.93
BERT BiLSTM	0.94	0.94	0.94
TF-IDF & Naive Bayes	0.90	0.89	0.90
TF-IDF & Logistic Regression	0.90	0.90	0.90
TF-IDF & SVM	0.89	0.89	0.89
BoW & Naive Bayes	0.91	0.91	0.91
BoW & Logistic Regression	0.90	0.90	0.90
TF-IDF & Logistic Regression	0.90	0.90	0.90

Πίνακας 8: Αξιολόγηση μοντέλων με ποσοστό διαχωρισμού 80/20

Μοντέλο	Precision	Recall	F1 Score
Simple LSTM	0.91	0.92	0.92
BiLSTM	0.93	0.93	0.93
Word2Vec LSTM (trainable=False)	0.89	0.89	0.89
Word2Vec LSTM (trainable=True)	0.93	0.93	0.93
FastText LSTM (trainable=False)	0.88	0.88	0.88
FastText LSTM (trainable=True)	0.91	0.92	0.92
BERT BiLSTM	0.94	0.94	0.94
TF-IDF & Naive Bayes	0.91	0.91	0.91
TF-IDF & Logistic Regression	0.91	0.91	0.91
TF-IDF & SVM	0.92	0.92	0.92
BoW & Naive Bayes	0.90	0.90	0.90
BoW & Logistic Regression	0.90	0.91	0.90
TF-IDF & Logistic Regression	0.90	0.90	0.90

Πίνακας 9: Αξιολόγηση μοντέλων με ποσοστό διαχωρισμού 90/10

Μοντέλο	Precision	Recall	F1 Score
Simple LSTM	0.92	0.92	0.92
BiLSTM	0.93	0.93	0.93
Word2Vec LSTM (trainable=False)	0.88	0.87	0.87

Word2Vec LSTM (trainable=True)	0.92	0.92	0.92
FastText LSTM (trainable=False)	0.87	0.87	0.87
FastText LSTM (trainable=True)	0.94	0.94	0.94
BERT BiLSTM	0.94	0.94	0.94
TF-IDF & Naive Bayes	0.91	0.91	0.91
TF-IDF & Logistic Regression	0.91	0.91	0.91
TF-IDF & SVM	0.91	0.91	0.91
BoW & Naive Bayes	0.92	0.92	0.92
BoW & Logistic Regression	0.92	0.92	0.92
TF-IDF & Logistic Regression	0.91	0.91	0.91

5.4.1 Μελέτη παραδειγμάτων

Οι παρακάτω Πίνακες 11 και 10 παρουσιάζουν το σύνολο των σωστά και λανθασμένα ταξινομημένων τίτλων ειδήσεων, αντίστοιχα, από τα πέντε καλύτερα μοντέλα και επιβεβαιώνουν τα παραπάνω αποτελέσματα. Ο καλύτερος ταξινομητής είναι ο BERT BiLSTM και ακολουθεί με μικρή διαφορά ο BiLSTM. Ωστόσο, παρατηρούμε πως ο BERT BiLSTM είναι καλύτερος στο να ταξινομεί σωστά τους σαρκαστικούς τίτλους ειδήσεων, υστερεί λίγο στο να ταξινομεί σωστά τους μη σαρκαστικούς τίτλους ειδήσεων.

Πίνακας 10: Αληθώς θετικά και αληθώς αρνητικά των πέντε καλύτερων μοντέλων

Μοντέλο	True Positives	True Negatives	Total
LSTM	1095	1203	2298
BiLSTM	1172	1200	2372
Word2Vec LSTM (trainable=True)	1181	1191	2372
FastText LSTM (trainable=True)	1113	1238	2351
BERT BiLSTM	1176	1211	2387

Πίνακας 11: Ψευδώς θετικά και ψευδώς αρνητικά των πέντε καλύτερων μοντέλων

Μοντέλο	False Positives	False Negatives	Total
LSTM	83	144	227
BiLSTM	86	67	153
Word2Vec LSTM (trainable=True)	95	58	153
FastText LSTM (trainable=True)	48	126	174
BERT BiLSTM	39	99	138

Μελετήσαμε στην συνέχεια ένα μικρό δείγμα ψευδώς θετικών (False Positives) και ψευδώς αρνητικών (False Negatives) τίτλων ειδήσεων που παράγονται από τα μοντέλα για να εντοπίσουμε τα κύρια αδύναμα σημεία στα οποία αυτά τα μοντέλα αποτυγχάνουν.

Ο Πίνακας 12 δείχνει ορισμένους τίτλους που λανθασμένα χαρακτηρίστηκαν ως σαρκαστικά από όλα τα μοντέλα.

Ρίχνοντας μια ματιά σε αυτά τα ψευδώς θετικά, μπορεί να υποτεθεί ότι τα μοντέλα φαίνεται να θεωρούν αυτές τις περιπτώσεις που περιέχουν χιουμοριστικές ή πολύ σουρεαλιστικές προτάσεις ως σαρκασμό. Για παράδειγμα, όλα τα δύο μοντέλα συμφωνούν ότι η πρόταση «*Κουτσούμπας: είστε βαθιά γελασμένοι αν νομίζετε πως θα ξεφύγετε από τις ευθύνες σας» είναι σαρκαστικός. Αυτό θα μπορούσε να οφείλεται στην έκφραση «βαθιά γελασμένοι*», κάτι που μπορεί να έχει ως αποτέλεσμα τα μοντέλα να το θεωρούν ως αστείο. Η ίδια κατάσταση συμβαίνει για προτάσεις όπως «*Ζαχαριάδης για Γεωργιάδη στο Twitter: τρομερή παιχτούρα ο Άδωνις, όχι ο Μαρσέλο.*», που μπορεί να ερμηνευθεί ως αστεία λόγω της έκφρασης «τρομερή παιχτούρα». Αυτό οφείλεται στην παρουσία στόμφου με χρήση επιθέτων και αργκό ορολογίας, που έχει συσχετιστεί με το χιούμορ και τον σαρκασμό. Δίνοντας προσοχή σε μια άλλη φράση, την οποία όλα τα μοντέλα θεώρησαν σαρκαστική: «*Σαν σήμερα: number Σεπτεμβρίου - Κώστας Γεωργάκης, ο άνθρωπος που αυτοπυρπολήθηκε για να φύγει η χούντα.*», φαίνεται ότι τα μοντέλα μπορεί να μάθουν να βλέπουν το χιούμορ σε καταστάσεις που δεν είναι χιουμοριστικές όπως η λέξη «*αυτοπυρπολήθηκε*» που σπάνια (ευτυχώς) συναντάται στην ειδησεογραφία, ενώ πιο συχνά απαντάται σε χιουμοριστικά κείμενα.

Πίνακας 12: Παραδείγματα ψευδώς θετικών τίτλων ειδήσεων

	Headline
1	«*Πρόκληση "Fast and Furious" για γέλια: οδηγός BMW έκανε σμπαράλια το αυτοκίνητό του.*»
2	«*Ζαχαριάδης για Γεωργιάδη στο Twitter: τρομερή παιχτούρα ο Άδωνις, όχι ο Μαρσέλο.* »
3	«*Κουτσούμπας: είστε βαθιά γελασμένοι αν νομίζετε πως θα ξεφύγετε από τις ευθύνες σας.*»
4	«*Μυστηριώδης μπάλα φωτιάς στον ουρανό μπερδεύει τους επιστήμονες.*»
5	«*"Winnie the pooh, blood and honey": ο Γουίνι το αρκουδάκι πρωταγωνιστής σε ταινία τρόμου.*»
6	«*Σαν σήμερα: number Σεπτεμβρίου - Κώστας Γεωργάκης, ο άνθρωπος που αυτοπυρπολήθηκε για να φύγει η χούντα.*»
7	«*Ο κροκόδειλος κατάφερε να ξεφύγει. Κολύμπησε μακριά και τα λιοντάρια απομακρύνθηκαν σαν να μην είχε συμβεί τίποτα.*»
8	«*Εφαρμογή ανιχνεύει τον κορωνοϊό από τη φωνή, με τη βοήθεια τεχνητής νοημοσύνης.*»
9	«*Θαύμα στην Βόρεια Κορέα: όλοι οι "ασθενείς με πυρετό", δηλ. κορονοϊό, έχουν "ιαθεί".*»
10	«*Έφυγε από τη ζωή ο Πειραιώτης επιχειρηματίας Δημήτρης Γλου*.»

Αντίστοιχα μελετήσαμε ένα μικρό δείγμα ψευδώς αρνητικών (False Negatives) που παράγονται από τα μοντέλα. Ο Πίνακας 13 δείχνει ορισμένους τίτλους ειδήσεων που λανθασμένα χαρακτηρίστηκαν ως μη σαρκαστικά από όλα τα μοντέλα.

Για παράδειγμα, για τον τίτλο «*Βιογραφικό στο Μπάκιγχαμ έστειλε ο Πρίγκιπας Παύλος.*», όλα τα μοντέλα συμφωνούν ότι αυτό δεν είναι σαρκαστικό, ακόμα κι αν είναι όντως σαρκαστικό. Χρειάζεται να γνωρίσουμε ποιός είναι ο Πρίγκιπας Παύλος και τι σημαίνει η λέξη «*βιογραφικό*». Η ίδια κατάσταση συμβαίνει με τον σαρκαστικό τίτλο «*Ένταση στο λιμάνι του Πειραιά από πλήθος κόσμου που θέλει να ακολουθήσει την Μαριανίνα Κριεζή στη Λιλιπούπολη.*», ο οποίος εσφαλμένα χαρακτηρίστηκε ως μη σαρκαστικός από όλα τα μοντέλα. Χρειάζεται να γνωρίζουμε ότι η Μαριανίνα Κριεζή

είναι η στιχουργός της «*Λιλιπούπολης*» για να κατανοήσουμε την πρόταση. Θα μπορούσε να είναι χρήσιμο στον εντοπισμό σημαντικών θεμάτων στις προτάσεις και στη λήψη κάποιου πλαισίου από αυτά, προκειμένου να βοηθηθούν τα μοντέλα να κάνουν πιο ακριβείς προβλέψεις.

Υπάρχουν ορισμένοι τίτλοι που είναι ιδιαίτερα δύσκολο να εντοπιστούν, καθώς θα μπορούσαν να είναι πραγματικά γεγονότα, ανεξάρτητα από το πλαίσιο που δίνεται στο μοντέλο. Για παράδειγμα, ο τίτλος «*Στο ψηφοδέλτιο επικρατείας του ΚΚΕ η Τζούλια Νόβα στις επόμενες εκλογές.*», ακόμα κι αν ο τίτλος δόθηκε σε έναν άνθρωπο που γνωρίζει ποια είναι Τζούλια Νόβα και είχε το πλαίσιο για να κατανοήσει τον τίτλο, το άτομο θα μπορούσε να σκεφτεί ότι θα μπορούσε να είναι αληθινή πρόταση καθώς είναι μια πιθανότητα, και επομένως, το πλαίσιο από μόνο του, δεν θα έπαιζε μεγάλο ρόλο για αυτού του είδους τις προτάσεις. Εάν τα μοντέλα ήταν σε θέση να γνωρίζουν την «απόλυτη αλήθεια» γύρω από τους προτεινόμενους τίτλους, θα μπορούσαν εύκολα να τα ταξινομήσουν ως σαρκαστικά/μη σαρκαστικά. Ωστόσο, η παροχή πλαισίου στο μοντέλο είναι μια πρόκληση). Ωστόσο, αυτό είναι ένα δύσκολο έργο, το οποίο προτείνεται να αντιμετωπιστεί σε μελλοντικές έρευνες.

Πίνακας 13: Παραδείγματα ψευδώς αρνητικών τίτλων ειδήσεων

	Headline
0	«*....., του Κώστα Καραμανλή.*»
1	«*Το παν στο σκάκι είναι η τριγωνική επίθεση, του Λευτέρη Αυγενάκη*»
2	«*Η κηδεία της κυρίας Γεννηματά, ιδανική ευκαιρία για συνέδριο ΠΑΣΟΚ όπως θα είναι μαζεμένοι, του Γιώργου Κύρτσου.*»
3	«*Έξαλλος ο Ντόναλντ Τραμπ, κατηγορεί τους Κινέζους και για τον ρατσισμό στις ΗΠΑ.*»
4	«*Νέα πρόκληση Ερντογάν: "Ο ελληνικός καφές είναι τούρκικος".*»
5	«*Κρίση στα νότια προάστια: ξεράθηκαν οι φοίνικες στη Γλυφάδα.* »
6	«*Ένταση στο λιμάνι του Πειραιά από πλήθος κόσμου που θέλει να ακολουθήσει την Μαριανίνα Κριεζή στη Λιλιπούπολη.*»
7	«*Έντονη η οσμή τιρινίνης στην ατμόσφαιρα, καθησυχαστικοί οι επιστήμονες.*»
8	«*Βιογραφικό στο Μπάκιγχαμ έστειλε ο Πρίγκιπας Παύλος.*»
9	«*Στο ψηφοδέλτιο επικρατείας του ΚΚΕ η Τζούλια Νόβα στις επόμενες εκλογές.*»
10	«*Στο μηδέν παραμένουν τα κρούσματα κορωναϊού σε Τουρκία και Βόρεια Κορέα.*»

Στην τρέχουσα μελέτη, ο ταξινομητής BERT BiLSTM ήταν σε θέση να ταξινομήσει σωστά διάφορες περιπτώσεις που τα άλλα μοντέλα δεν κατάφεραν. Για τους τίτλους «*Ξεχωριστά σποτ για γυναίκες με ροζ ζωγραφιές και κούκλες θα δημιουργήσει η Πολιτική Προστασία.*» και «*Ο κορονοϊός είναι από την Κίνα, περιμένουμε σύντομα να χαλάσει μόνος του, του Βασίλη Κικίλια*.» μόνο ο BERT ήταν σε θέση να αναγνωρίσει ότι υπήρχε σαρκασμός στις προτάσεις. Από την άλλη πλευρά, οι τίτλοι «*Μπάιντεν: Δεν έχω αποφασίσει αν θα διεκδικήσω την επανεκλογή μου.*» και «*Πιτσιόρλας: Δεν θέλω να*

πιστέψω ότι υπήρξε εντολή παρακολούθησής μου από τον Τσίπρα.» σωστά ταξινομήθηκαν ως μη σαρκαστικά από το BERT BiLSTM, αλλά λανθασμένα ως σαρκαστικά από τα υπόλοιπα μοντέλα.

6. ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην παρούσα έρευνα υλοποιήθηκε ένα σύνολο από μοντέλα που στόχο έχουν να ανιχνεύσουν τον σαρκασμό σε ελληνικούς ειδησεογραφικούς τίτλους στο *Twitter*. Τα δεδομένα που συγκεντρώθηκαν ήταν περίπου 13000 και αυτό οφείλεται στον περιορισμό του Twitter API.

Για την ανίχνευση του σαρκασμού στα tweets δημιουργήθηκαν 7 διαφορετικά μοντέλα βαθιάς μάθησης που αποτελούνταν από νευρωνικά δίκτυα μακράς βραχυπρόθεσμης μνήμης (LSTM) και 6 μοντέλα βασιζόμενα σε παραδοσιακές τεχνικές μηχανικής μάθησης.

Οι LSTM κατηγοριοποιητές διαφέρουν μεταξύ τους ως προς το είδος του LSTM νευρωνικού που χρησιμοποιήθηκε ή ως προς το είδος της τεχνικής που χρησιμοποιήθηκε για την αναπαράσταση των λέξεων που υπάρχουν στα tweets. Παράλληλα, έγινε προ-επεξεργασία των δεδομένων κειμένου προτού εισαχθούν στα μοντέλα. Για το μοντέλο BERT BiLSTM αφαιρέθηκαν επιπρόσθετα και όλα τα διακριτικά, σύμφωνα με τις οδηγίες των δημιουργών. Αφαίρεση των stopwords και των σημείων στίξης πραγματοποιήθηκε μόνο για τα παραδοσιακά μοντέλα.

Όπως αναφέραμε και στο προηγούμενο κεφάλαιο, η δοκιμή διαφορετικών ποσοστών όσον αφορά στο διαχωρισμό των train/validation και test συνόλων απέδειξε ότι όσο μεγαλύτερο το σύνολο δεδομένων που χρησιμοποιείται για την εκπαίδευση των μοντέλων τόσο επιτυγχάνεται καλύτερη ανίχνευση του σαρκασμού στα tweets. Συγκεκριμένα, παρατηρήθηκε ότι στο διαχωρισμό 90/10 τα μοντέλα έχουν υψηλότερα ποσοστά σωστής ανίχνευσης σαρκασμού σε σχέση με το διαχωρισμό 80/20 και 70/30. Η παρακάτω ανάλυση βασίζεται σε αυτόν τον ποσοστιαίο διαχωρισμό.

Από τα 7 μοντέλα βαθιάς μάθησης, ο ταξινομητής που είχε τη δυνατότητα να ανιχνεύσει ορθότερα τον σαρκασμό στο εκάστοτε tweet ήταν ο BERT BiLSTM με ακρίβεια (accuracy) 94.59%. Το συγκεκριμένο μοντέλο βασίζεται στη δημιουργία BERT αναπαραστάσεων για τις λέξεις, σε συνδυασμό με ένα BiLSTM νευρωνικό δίκτυο.

Παράλληλα, η χρήση έτοιμων διανυσματικών αναπαραστάσεων για τις λέξεις φάνηκε να συνεισφέρει θετικά στην ανίχνευση του σαρκασμού από το μοντέλο μόνο στην περίπτωση που είχε οριστεί να μπορεί να εκπαιδευτεί το embedding επίπεδο (trainable=True), όπως αποδείχθηκε από τους Word2Vec LSTM και FastText LSTM ταξινομητές που πέτυχαν ακρίβεια σε ποσοστό 92.78% και 92.59% αντίστοιχα σε σχέση με το μοντέλο Simple LSTM με ακρίβεια 91.61%. Στην περίπτωση που είχε οριστεί να μην μπορεί να εκπαιδευτεί το embedding επίπεδο (trainable=False), η δυνατότητα του μοντέλου να ανιχνεύει τον σαρκασμό επηρεάστηκε αρνητικά, με το μοντέλο Word2Vec LSTM να δίνει ακρίβεια 87.63% και το μοντέλο FastText LSTM 86.79% αντίστοιχα. Τα μοντέλα αυτά έχουν και τα χειρότερα αποτελέσματα. Το γεγονός ότι τα embeddings αυτά δημιουργήθηκαν πριν την εκπαίδευση του μοντέλου και όχι μαζί με αυτό, σίγουρα συντέλεσε αρνητικά. Αυτό επιβεβαιώνεται από το γεγονός ότι στο Simple LSTM μοντέλο που δημιουργήθηκε η αναπαράσταση ταυτόχρονα με την εκπαίδευση του κατηγοριοποιητή, η ακρίβεια άγγιξε, όπως αναφέρθηκε προηγουμένως, το 91.61%.

Όσον αφορά στην μικρή διαφορά μεταξύ των μοντέλων Word2Vec LSTM και FastText LSTM, με το Word2Vec να έχει τα καλύτερα αποτελέσματα, υποθέτουμε ότι αιτία είναι η διαφορά στο πλήθος των λέξεων του συνόλου δεδομένων που αναγνωρίστηκαν κατά την προ-επεξεργασία. Από το σύνολο των λέξεων του συνόλου δεδομένων (26995) αναγνωρίστηκαν 20724 λέξεις για το μοντέλο Word2Vec και 19146 λέξεις για το FastText Μοντέλο.

Ταυτόχρονα, φάνηκε πως οι κατηγοριοποιητές που έχουν σχεδιαστεί με χρήση BiLSTM νευρωνικών έχουν και καλύτερη ακρίβεια (accuracy), ενώ αποφεύγεται και το πρόβλημα της αναπαραγωγής των αποτελεσμάτων. Η δυσκολία με την αναπαραγωγή αντιμετωπίστηκε, εν μέρει, στο μοντέλο Simple LSTM με χρήση 3 επιπέδων, στα μοντέλα Word2Vec LSTM και FastText LSTM με χρήση 3 επιπέδων LSTM, ενώ με τη χρήση BiLSTM νευρωνικών, ακόμη και ενός επιπέδου, το πρόβλημα αντιμετωπίζεται πλήρως. Οι ταξινομητές BiLSTM και BERT BiLSTM που υλοποιήθηκαν με BiLSTM νευρωνικά δείχνουν τις δυνατότητες που έχουν αυτά ως προς την διαχείριση δεδομένων κειμένου, καθώς πέτυχαν τα καλύτερα αποτελέσματα με ακρίβεια 94.59% και 93.99%, αντίστοιχα.

Όσον αφορά στις παραδοσιακές τεχνικές που υλοποιήθηκαν, η TF-IDF αναπαράσταση φαίνεται ότι επηρέασε θετικά τα παραδοσιακά μοντέλα. Αλλά ο κατηγοριοποιητής BoW & Logistic Regression φαίνεται να είναι εκείνος ο οποίος έχει τη δυνατότητα να ανιχνεύει καλύτερα τον σαρκασμό στο σύνολο δεδομένων με ακρίβεια 91.37%.

Τα μοντέλα όπου χρησιμοποιήθηκε βαθιά μάθηση με LSTM νευρωνικά δίκτυα ξεπερνούν αρκετά σε επιδόσεις τα απλά μοντέλα μηχανικής μάθησης στην περίπτωση που αφήνουμε το embedding layer να είναι εκπαιδεύσιμο.

Τέλος, πρέπει να τονιστεί ότι η ανάλυση και υλοποίηση αφορά τα συγκεκριμένα δεδομένα που συλλέξαμε και επεξεργαστήκαμε.

7. ΠΡΟΤΑΣΕΙΣ ΓΙΑ ΜΕΛΛΟΝΤΙΚΗ ΕΡΕΥΝΑ

Στα πλαίσια της παρούσας έρευνας μελετήθηκαν διαφορετικές προσεγγίσεις για την επίλυση του προβλήματος της Ανίχνευσης Σαρκασμού στην ελληνική γλώσσα. Παρακάτω παραθέτουμε επιπλέον προτάσεις για μελλοντική έρευνα τόσο στο συγκεκριμένο πρόβλημα με στόχο την επίτευξη καλύτερων αποτελεσμάτων όσο και στο ευρύτερο πεδίο της ανίχνευσης μεταφορικού λόγου.

Τα αποτελέσματα που προέκυψαν με την εφαρμογή των μοντέλων για ανίχνευση του σαρκασμού σε τίτλους ειδήσεων στο Twitter της έρευνας ποικίλλουν ανάλογα με τον κατηγοριοποιητή που σχεδιάστηκε. Κάποια από τα μοντέλα που πέτυχαν χαμηλά ποσοστά επιτυχίας, χρήζουν βελτίωσης, ενώ οι ταξινομητές με καλά αποτελέσματα αξίζει να δοκιμαστούν με διάφορες παραλλαγές για να εξεταστεί αν μπορούν να επιτύχουν ακόμη καλύτερη ακρίβεια. Οι βελτιώσεις που θα μπορούσαν να εφαρμοστούν σε μία μελλοντική έρευνα παρουσιάζονται στην συνέχεια.

Αρχικά, αξίζει να σημειωθεί ότι για την εξαγωγή πιο αξιόπιστων αποτελεσμάτων θα πρέπει να δημιουργηθεί μεγαλύτερο σύνολο δεδομένων στην ελληνική γλώσσα. Αυτό θα μπορούσε να πραγματοποιηθεί με δύο τρόπους, είτε με την συγκέντρωση παλαιότερων τίτλων ειδήσεων από τα διαδικτυακά ειδησεογραφικά sites, είτε με την δημιουργία συνόλου δεδομένων από διάφορες πλατφόρμες κοινωνικής δικτύωσης με την βοήθεια σχολιαστών οι οποίοι θα προσθέτουν ετικέτα ύπαρξης ή μη σαρκασμού σε δημοσιεύσεις χρηστών.

Επίσης, αξίζει να δοκιμαστούν παραλλαγές και συνδυασμοί των κατηγοριοποιητών που παρουσιάστηκαν στην παρούσα έρευνα, ώστε να μελετηθεί κατά πόσο είναι εφικτό να βελτιωθεί η ακρίβεια που επιτυγχάνουν. Παράδειγμα αποτελεί η περίπτωση των Simple LSTM, Word2Vec LSTM και FastText LSTM μοντέλων που δεν δοκιμάστηκαν με ένα BiLSTM νευρωνικό.

Παράλληλα, μία σημαντική δοκιμή που αξίζει να πραγματοποιηθεί είναι η εισαγωγή προ-επεξεργασμένων δεδομένων στο μοντέλο BERT BiLSTM. Τα προ-επεξεργασμένα δεδομένα που εφαρμόσθηκαν στα άλλα μοντέλα, θεωρήθηκε ότι δεν ταιριάζουν στον τρόπο με τον οποίο λειτουργεί η διαδικασία του tokenization στον BERT κωδικοποιητή. Η εφαρμογή τεχνικών προεπεξεργασίας, διαφορετικών από αυτές που έχουν ήδη υλοποιηθεί στην παρούσα έρευνα, αξίζει να εφαρμοστούν ώστε να δοκιμαστεί αυτό το μοντέλο σε άλλες συνθήκες.

Επιπρόσθετα, τα ήδη υπάρχοντα μοντέλα αξίζει να δοκιμαστούν σε ελληνικά σύνολα δεδομένων, μεγαλύτερου όγκου ώστε να επαληθευτεί η υψηλή ακρίβεια στην ανίχνευση του σαρκασμού. Πέραν του όγκου, είναι σημαντικό για το σετ δεδομένων, πέραν του επεξεργασμένου κειμένου, να δημιουργηθεί μια σειρά από αριθμητικά δεδομένα που θα περιγράφουν τα γνωρίσματα κάθε τίτλου ειδήσεων. Παραδείγματα τέτοιων αριθμητικών δεδομένων είναι: το πλήθος λέξεων, το πλήθος μοναδικών λέξεων, πλήθος γραμμάτων, πλήθος stopwords, πλήθος σημείων στίξης, πλήθος mentions, hashtags, URLs, πλήθος ρημάτων, ουσιαστικών, επιθέτων κ.λπ.

Μία πρόταση υψίστης σημασίας είναι να γίνουν ακόμα περισσότερα πειράματα για κάθε μοντέλο, ώστε να γίνει εφικτή μία στατιστική μελέτη που θα αποτυπώνει πιο ολοκληρωμένα την απόδοση του κάθε μοντέλου, με την καταγραφή της διακύμανσης της ακρίβειας και άλλων στατιστικών μετρικών.

Εν κατακλείδι, οι τρέχουσες προσεγγίσεις για τον εντοπισμό και επεξεργασία του σαρκασμού έχουν ακόμη πολλά περιθώρια βελτίωσης, δεδομένου ότι ο σαρκασμός αποτελεί ένα πολύπλοκο φαινόμενο με κοινωνιο-γλωσσολογικές προεκτάσεις και με άμεση εξάρτηση από παραμέτρους κάθε φυσικής γλώσσας.

ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ

ΑΣ	Ανάλυση Συναισθήματος
ΕΦΓ	Επεξεργασία Φυσικής Γλώσσας
ELMo	Embeddings from Language Models
BERT	Bidirectional Encoder Representations from Transformers
BiLSTM	Bidirectional Long Short – Term Memory
BoW	Bag of Words
CBOW	Continuous Bag-of-Words
GloVe	Global Vectors
LSTM	Long Short – Term Memory
NB	Naïve Bayes
SVM	Support Vector Machines
TF-IDF	Term Frequency – Inverse Document Frequency

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] Cambria, E., Poria, S., Bajpai, R. & Schuller, B. (2016). SenticNet 4: A semantic resource for sentiment analysis based on conceptual primitives. COLING 2016 - 26th International Conference on Computational Linguistics, Proceedings of COLING 2016: Technical Papers.

[2] Littman, D. C. & Mey, J. L. (1991). The nature of irony: Toward a computational model of irony. Journal of Pragmatics, 15(2). https://doi.org/10.1016/0378-2166(91)90057-5

[3] Wilson, D. (2006). The pragmatics of verbal irony: Echo or pretence? Lingua, 116(10). https://doi.org/10.1016/j.lingua.2006.05.001

[4] Bryant, G. A. & Fox Tree, J. E. (2005). Is there an Ironic Tone of Voice? Language and Speech, 48(3). https://doi.org/10.1177/00238309050480030101

[5] Gibbs, R. W. & O'Brien, J. (1991). Psychological aspects of irony understanding. Journal of Pragmatics, 16(6). https://doi.org/10.1016/0378-2166(91)90101-3

[6] Attardo, S. (2000). Irony as relevant inappropriateness. Journal of Pragmatics, 32(6). https://doi.org/10.1016/S0378-2166(99)00070-3

[7] Partington, A. (2011). Phrasal irony: Its form, function and exploitation. Journal of Pragmatics, 43(6). https://doi.org/10.1016/j.pragma.2010.11.001

[8] Colston, H. L. (1997). Salting a wound or sugaring a pill: The pragmatic functions of ironic criticism. Discourse Processes, 23(1). https://doi.org/10.1080/01638539709544980

[9] Ι. Φαρακλού, «Ειρωνεία και πραγματολογική θεωρία» Διδακτορική διατριβή, Τμήμα Φιλολογίας, Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης, 2000 https://doi.org/10.12681/eadd/5805

[10] Lee, C. J. & Katz, A. N. (1998). The Differential Role of Ridicule in Sarcasm and Irony. Metaphor and Symbol, 13(1). https://doi.org/10.1207/s15327868ms1301{_}1

[11] Gaudio, R. P. (2000). John Haiman , Talk is cheap: Sarcasm, alienation, and the evolution of language . Oxford & New York: Oxford University Press, 1998. Pp. ix, 220. Hb $40.00, pb $18.95. Language in Society, 29(1). https://doi.org/10.1017/s0047404500211032

[12] *Recommendation ITU-R BT.601, Encoding Parameters of Digital Television for Studios,* Int'l Telecommunications Union, 1992. Rockwell, P. (2000). Lower, slower, louder: Vocal cues of sarcasm. Journal of Psycholinguistic Research, 29(5). https://doi.org/10.1023/A:1005120109296

[13] Jorgensen, J. (1996). The functions of sarcastic irony in speech. Journal of Pragmatics, 26(5). https://doi.org/10.1016/0378-2166(95)00067-4

[14] Toplak, M. & Katz, A. N. (2000). On the uses of sarcastic irony. Journal of Pragmatics, 32(10). https://doi.org/10.1016/S0378-2166(99)00101-0

[15] Liu, B. (2010). Sentiment analysis and subjectivity. Handbook of Natural Language Processing, Second Edition.

[16] Maynard, D. & Greenwood, M. A. (2014). Who cares about sarcastic tweets? Investigating the impact of sarcasm on sentiment analysis. Proceedings of the 9th International Conference on Language Resources and Evaluation, LREC 2014.

[17] Barbieri, F., Ronzano, F. & Saggion, H. (2015). Is this tweet satirical? A computational approach for satire detection in Spanish. Procesamiento de Lenguaje Natural, 55.

[18] Barbieri, F., Saggion, H. & Ronzano, F. (2015). Modelling Sarcasm in Twitter, a Novel Approach. https://doi.org/10.3115/v1/w14-2609

[19] Cignarella, A. T., Frenda, S., Basile, V., Bosco, C., Patti, V. & Rosso, P. (2018). Overview of the EVALita 2018 task on irony detection in Italian tweets (IRONITA). CEUR Workshop Proceedings, 2263. https://doi.org/10.4000/books.aaccademia.4470

[20] Liebrecht, C., Kunneman, F. & Van den Bosch, A. (2013). The perfect solution for detecting sarcasm in tweets #not.

[21] Charalampakis, B., Spathis, D., Kouslis, E. & Kermanidis, K. (2016). A comparison between semi-supervised and supervised text mining techniques on detecting irony in Greek political tweets. Engineering Applications of Artificial Intelligence, 51. https://doi.org/10.1016/j.engappai.2016.01.007

[22] Tsakalidis, A., Papadopoulos, S., Voskaki, R., Ioannidou, K., Boididou, C., Cristea, A. I., Liakata, M. & Kompatsiaris, Y. (2018). Building and evaluating resources for sentiment analysis in the Greek language. Language Resources and Evaluation, 52(4). https://doi.org/10.1007/s10579-018-9420-4

[23] Ranti, K. S. & Girsang, A. S. (2020). Indonesian Sarcasm Detection Using Convolutional Neural Network. International Journal of Emerging Trends in Engineering Research.

[24] Lunando, E. & Purwarianti, A. (2013a). Indonesian Social Media Sentiment Analysis With Sarcasm Detection, 195–198. https://doi.org/10.1109/ICACSIS.2013.6761575

[25] Abercrombie, G. & Hovy, D. (2016). Putting sarcasm detection into context: The effects of class imbalance and manual labelling on supervised machine classification of twitter conversations. 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016 – Student Research Workshop. https://doi.org/10.18653/v1/p16-3016

[26] Karoui, J., Benamara, F., Moriceau, V., Patti, V., Bosco, C. & Aussenac-Gilles, N. (2017). Exploring the impact of pragmatic phenomena on irony detection in tweets: A multilingual corpus study. 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017 - Proceedings of Conference, 1. https://doi.org/10.18653/v1/e17-1025
[27] Ghanem, B., Karoui, J., Benamara, F., Rosso, P. & Moriceau, V. (2020). Irony detection in a multilingual context. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 12036 LNCS. https://doi.org/10.1007/978-3-030-45442-5{_}18
[28] El Mahdaouy, A., El Mekki, A., Essefar, K., El Mamoun, N., Berrada, I. & Khoumsi, A. (2021). Deep Multi-Task Model for Sarcasm Detection and Sentiment Analysis in Arabic Language. Association for Computational Linguistics.
[29] Abu-Farha, I. & Magdy, W. (2020). From Arabic Sentiment Analysis to Sarcasm Detection: The ArSarcasm Dataset. Aclweb.Org, European L(May).
[30] Ionescu, R. T. & Chifu, A. G. (2021). FreSaDa: A French Satire Data Set for Cross-Domain Satire Detection. Proceedings of the International Joint Conference on Neural Networks, 2021-July. https://doi.org/10.1109/IJCNN52387.2021.9533661
[31] Salas-Zárate, M. d. P., Paredes-Valverde, M. A., Rodriguez-García, M. Á., Valencia-García, R. & Alor-Hernández, G. (2017). Automatic detection of satire in Twitter: A psycholinguistic-based approach. Knowledge-Based Systems, 128. https://doi.org/10.1016/j.knosys.2017.04.009
[32] García-Díaz, J. A. & Valencia-García, R. (2022). Compilation and evaluation of the Spanish SatiCorpus 2021 for satire identification using linguistic features and transformers. Complex & Intelligent Systems. https://doi.org/10.1007/s40747-021-00625-1
[33] Ortega-Bueno, R., Rosso, P. & Medina Pagola, J. E. (2022). Multi-view informed attention-based model for Irony and Satire detection in Spanish variants. Knowledge-Based Systems, 235. https://doi.org/10.1016/j.knosys.2021.107597
[34] Schubert, G. & Freitas, L. (2020). The Construction of a Corpus for Detecting Irony and Sarcasm in Portuguese.
[35] McHardy, R., Adel, H. & Klinger, R. (2019). Adversarial training for satire detection: Controlling for confounding variables. NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference, 1. https://doi.org/10.18653/v1/n19-1069
[36] Nguyen, Q. T., Nguyen, T. L., Luong, N. H. & Ngo, Q. H. (2020). Fine-Tuning BERT for Sentiment Analysis of Vietnamese Reviews. Proceedings - 2020 7th NAFOSTED Conference on Information and Computer Science, NICS 2020. https://doi.org/10.1109/NICS51282.2020.9335899
[37] Swami, S., Khandelwal, A., Singh, V., Akhtar, S. S. & Shrivastava, M. (2018). A Corpus of English-Hindi Code-Mixed Tweets for Sarcasm Detection. ArXiv, abs/1805.11869.
[38] Ptáček, T., Habernal, I. & Hong, J. (2014). Sarcasm detection on Czech and English twitter. COLING 2014 - 25th International Conference on Computational Linguistics, Proceedings of COLING 2014: Technical Papers.
[39] Tang, Y. J. & Chen, H. H. (2014). Chinese irony corpus construction and ironic structure analysis. COLING 2014 - 25th International Conference on Computational Linguistics, Proceedings of COLING 2014: Technical Papers.
[40] Davidov, D., Tsur, O. & Rappoport, A. (2010). Semi-Supervised Recognition of Sarcastic Sentences in Twitter and Amazon. Proceedings of the Fourteenth Conference on Computational Natural Language Learning, 107–116.
[41] Barbieri, F. & Saggion, H. (2014). Automatic detection of irony and humour in Twitter. Proceedings of the 5th International Conference on Computational Creativity, ICCC 2014.
[42] Joshi, A., Sharma, V. & Bhattacharyya, P. (2015). Harnessing context incongruity for sarcasm detection. ACL-IJCNLP 2015 - 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, Proceedings of the Conference, 2. https://doi.org/10.3115/v1/p15-2124
[43] Ghosh, A. & Veale, T. (2017). Magnets for sarcasm: Making sarcasm detection timely, contextual and very personal. EMNLP 2017 - Conference on Empirical Methods in Natural Language Processing, Proceedings. https://doi.org/10.18653/v1/d17-1050
[44] Ahuja, R., Bansal, S., Prakash, S., Venkataraman, K. & Banga, A. (2018). Comparative study of different sarcasm detection algorithms based on behavioral approach. Procedia Computer Science, 143. https://doi.org/10.1016/j.procs.2018.10.412
[45] Khodak, M., Saunshi, N. & Vodrahalli, K. (2019). A large self-annotated corpus for sarcasm. LREC 2018 - 11th International Conference on Language Resources and Evaluation.
[46] Barbieri, F., Saggion, H. & Ronzano, F. (2014). Modelling Sarcasm in Twitter, a Novel Approach. WASSA@ACL.

[47] Sykora, M., Elayan, S. & Jackson, T. W. (2020). A qualitative analysis of sarcasm, irony and related #hashtags on Twitter. Big Data and Society, 7(2). https://doi.org/1 .1177/2053951720972735
[48] Oprea, S. & Magdy, W. (2020). iSarcasm: A Dataset of Intended Sarcasm. https://doi.org/10.18653/v1/2020.acl-main.118
[49] Filatova, E. (2012). Irony and sarcasm: Corpus generation and analysis using crowdsourcing. Proceedings of the 8th International Conference on Language Resources and Evaluation, LREC 2012.
[50] Riloff, E., Qadir, A., Surve, P., De Silva, L., Gilbert, N. & Huang, R. (2013). Sarcasm as contrast between a positive sentiment and negative situation. EMNLP 2013 - 2013 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference.
[51] Van Hee, C., Lefever, E. & Hoste, V. (2018a). SemEval-2018 Task 3: Irony Detection in English Tweets, 39–50. https://doi.org/10.18653/v1/S18-1005
[52] Sarsam, S. M., Al-Samarraie, H., Alzahrani, A. I. & Wright, B. (2020). Sarcasm detection using machine learning algorithms in Twitter: A systematic review. International Journal of Market Research, 62(5). https://doi.org/10.1177/1470785320921779
[53] Yaghoobian, H., Arabnia, H. & Rasheed, K. (2021). Sarcasm Detection: A Comparative Study.
[54] Veale, T. & Hao, Y. (2010). Detecting ironic intent in creative comparisons. Frontiers in Artificial Intelligence and Applications, 215. https://doi.org/10.3233/978-1-60750-606-5-765
[55] Bharti, S. K., Babu, K. S. & Jena, S. K. (2015). Parsing-based sarcasm sentiment recognition in Twitter data. Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2015. https://doi.org/10.1145/2808797.2808910
[56] Van Hee, C., Lefever, E. & Hoste, V. (2018b). We Usually Don't Like Going to the Dentist: Using Common Sense to Detect Irony on Twitter. Computational Linguistics, 44(4), 793–832. https://doi.org/10.1162/coli{_}a{_}00337
[57] Greene, S. & Resnik, P. (2009). More than words: Syntactic packaging and implicit sentiment. NAACL HLT 2009 - Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Proceedings of the Conference.
[58] Tepperman, J., Traum, D. & Narayanan, S. (2006). "Yeah right": Sarcasm recognition for spoken dialogue systems. INTERSPEECH 2006 and 9th International Conference on Spoken Language Processing, INTERSPEECH 2006 - ICSLP, 4. https://doi.org/10.21437/interspeech.2006-507
[59] Carvalho, P., Sarmento, L., Silva, M. & Oliveira, E. (2009). Clues for Detecting Irony in User-Generated Contents: Oh...!! It's "so easy" ;-). International Conference on Information and Knowledge Management, Proceedings. https://doi.org/10.1145/1651461.1651471
[60] Tsur, O., Davidov, D. & Rappoport, A. (2010). ICWSM - A great catchy name: Semi-supervised recognition of sarcastic sentences in online product reviews. ICWSM 2010 – Proceedings of the 4th International AAAI Conference on Weblogs and Social Media.
[61] Reyes, A., Rosso, P. & Buscaldi, D. (2012). From humor recognition to irony detection: The figurative language of social media. Data and Knowledge Engineering, 74. https://doi.org/10.1016/j.datak.2012.02.005
[62] Reyes, A., Rosso, P. & Veale, T. (2013). A multidimensional approach for detecting irony in Twitter. Language Resources and Evaluation, 47(1). https://doi.org/10.1007/s10579-012-9196-x
[63] Buschmeier, K., Cimiano, P. & Klinger, R. (2015). An Impact Analysis of Features in a Classification Approach to Irony Detection in Product Reviews. https://doi.org/10.3115/v1/w14-2608
[64] Mishra, A., Kanojia, D., Nagar, S., Dey, K. & Bhattacharyya, P. (2016). Harnessing cognitive features for sarcasm detection. 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016 - Long Papers, 2. https://doi.org/10.18653/v1/p16-1104
[65] Joshi, A., Tripathi, V., Patel, K., Bhattacharyya, P. & Carman, M. (2016). Are word embedding-based features useful for sarcasm detection? EMNLP 2016 - Conference on Empirical Methods in Natural Language Processing, Proceedings. https://doi.org/10.18653/v1/d16-1104
[66] Kreuz, R. & Caucci, G. (2007). Lexical Influences on the Perception of Sarcasm. Proceedings of the Workshop on Computational Approaches to Figurative Language. https://doi.org/10.3115/1611528.1611529
[67] Reyes, A. & Rosso, P. (2012). Making objective decisions from subjective data: Detecting irony in customer reviews. Decision Support Systems, 53(4). https://doi.org/10.1016/j.dss.2012.05.027
[68] González-Ibáñez, R., Muresan, S. & Wacholder, N. (2011). Identifying sarcasm in Twitter: A closer look. ACL-HLT 2011 - Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, 2.
[69] Mukherjee, S. & Bala, P. K. (2017). Sarcasm detection in microblogs using Naïve Bayes and fuzzy clustering. Technology in Society, 48. https://doi.org/10.1016/j.techsoc.2016.10.003
[70] Bamman, D. & Smith, N. A. (2015). Contextualized sarcasm detection on twitter. Proceedings of the 9th International Conference on Web and Social Media, ICWSM 2015.

[71] Liu, P., Chen, W., Ou, G., Wang, T., Yang, D. & Lei, K. (2014). Sarcasm detection in social media based on imbalanced classification. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 8485 LNCS. https://doi.org/10.1007/978-3-319-08010-9{_}49
[72] Ghosh, A. & Veale, T. (2016). Fracking Sarcasm using Neural Network. https://doi.org/10.13140/RG.2.2.16560.15363
[73] Nozza, D., Fersini, E. & Messina, V. (2016). Unsupervised Irony Detection: A Probabilistic Model with Word Embeddings. https://doi.org/10.5220/0006052000680076
[74] Blei, D. M., Ng, A. Y. & Jordan, M. I. (2003). Latent Dirichlet allocation. Journal of Machine Learning Research, 3(4-5). https://doi.org/10.1016/b978-0-12-411519-4.00006-9
[75] Wallace, B. C., Choe, D. K., Kertz, L. & Charniak, E. (2014). Humans require context to infer ironic intent (so computers probably do, too). 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014 - Proceedings of the Conference, 2. https://doi.org/10.3115/v1/p14-2084
[76] Hazarika, D., Poria, S., Gorantla, S., Cambria, E., Zimmermann, R. & Mihalcea, R. (2018). CASCADE: Contextual sarcasm detection in online discussion forums. COLING 2018 - 27th International Conference on Computational Linguistics, Proceedings.
[77] Wang, Z., Wu, Z., Wang, R. & Ren, Y. (2015). Twitter sarcasm detection exploiting a context-based model. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9418. https://doi.org/10.1007/978-3-319-26190-4{_}6
[78] Agrawal, A., An, A. & Papagelis, M. (2020). Leveraging Transitions of Emotions for Sarcasm Detection. SIGIR 2020 - Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. https://doi.org/10.1145/3397271.3401183
[79] Li, M., Lang, C., Yu, M., Lu, Y., Liu, C., Jiang, J. & Huang, W. (2020). SCX-SD: Semi-supervised Method for Contextual Sarcasm Detection. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 12275 LNAI. https://doi.org/10.1007/978-3-030-55393-7{_}26
[80] Nimala, K., Jebakumar, R. & Saravanan, M. (2021). Sentiment topic sarcasm mixture model to distinguish sarcasm prevalent topics based on the sentiment bearing words in the tweets. Journal of Ambient Intelligence and Humanized Computing, 12(6). https://doi.org/10.1007/s12652-020-02315-1
[81] Pennington, J., Socher, R. & Manning, C. D. (2014). GloVe: Global vectors for word representation. EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference. https://doi.org/10.3115/v1/d14-1162
[82] Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K. & Zettlemoyer, L. (2018). Deep contextualized word representations. NAACL HLT 2018 - 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference, 1. https://doi.org/10.18653/v1/n18-1202
[83] Devlin, J., Chang, M. W., Lee, K. & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference, 1.
[84] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. & Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. ArXiv, abs/1907.11692.
[85] Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. & Le, Q. V. (2019). XLNet: Generalized autoregressive pretraining for language understanding. Advances in Neural Information Processing Systems, 32.
[86] Potamias, R. A., Siolas, G. & Stafylopatis, A. G. (2020). A transformer-based approach to irony and sarcasm detection. Neural Computing and Applications, 32(23). https://doi.org/10.1007/s00521-020-05102-3
[87] Dadu, T. & Pant, K. (2020). Sarcasm Detection using Context Separators in Online Discourse. https://doi.org/10.18653/v1/2020.figlang-1.6
[88] Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P. & Soricut, R. (2020). ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. International Conference on Learning Representations. https://openreview.net/forum?id=H1eA7AEtvS
[89] Jaidka, K., Singh, I., Lu, J., Chhaya, N. & Ungar, L. (2020). A report of the CL-Aff OffMyChest shared task: Modeling supportiveness and disclosure. CEUR Workshop Proceedings, 2614.
[90] Shangipour ataei, T., Javdan, S. & Minaei-Bidgoli, B. (2020). Applying Transformers and Aspect-based Sentiment Analysis approaches on Sarcasm Detection. https://doi.org/10.18653/v1/2020.figlang-1.9
[91] Sahami, M., Dumais, S., Heckerman, D. & Horvitz, E. (1998). A Bayesian approach to filtering junk e-mail. Learning for Text Categorization: Papers from the AAAI Workshop, WS-98-05(Cohen).

[92] Androutsopoulos, I., Koutsias, J., Chandrinos, K. V., Paliouras, G. & Spyropoulos, C. D. (2000). An Evaluation of Naive Bayesian Anti-Spam Filtering. Machine Learning in the New Information Age.
[93] Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. Psychological Review, 65(6). https://doi.org/10.1037/h0042519
[94] Hornik, K., Stinchcombe, M. & White, H. (1989). Multilayer feedforward networks are universal approximators. Neural Networks, 2(5). https://doi.org/10.1016/0893-6080(89)90020-8
[95] Lipton, Z. (2015). A Critical Review of Recurrent Neural Networks for Sequence Learning.
[96] Staudemeyer, R. & Morris, E. (2019). Understanding LSTM – a tutorial into Long Short-Term Memory Recurrent Neural Networks.
[97] Karpathy, A., Johnson, J. & Li, F.-F. (2015). Visualizing and Understanding Recurrent Networks. Cornell Univ. Lab.
[98] Bengio, Y., Ducharme, R., Vincent, P. & Jauvin, C. (2003). A Neural Probabilistic Language Model. Journal of Machine Learning Research, 3(6). https://doi.org/10.1162/153244303322533223
[99] Collobert, R. & Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. Proceedings of the 25th International Conference on Machine Learning.
[100] Mikolov, T., Chen, K., Corrado, G. & Dean, J. (2013a). [Word2Vec] Distributed Representations of Words and Phrases and their Compositionality (Google 2013). CrossRef Listing of Deleted DOIs, 1.
[101] Bojanowski, P., Grave, E., Joulin, A. & Mikolov, T. (2017). Enriching Word Vectors with Subword Information. Transactions of the Association for Computational Linguistics, 5. https://doi.org/10.1162/tacl{_}a{_}00051
[102] Ruder, S. (2016). On word embeddings - Part 1.
[103] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K. & Kuksa, P. (2011). Natural language processing (almost) from scratch. Journal of Machine Learning Research, 12.
[104] Mikolov, T., Chen, K., Corrado, G. & Dean, J. (2013b). Efficient estimation of word representations in vector space. 1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings.
[105] Dai, A. M. & Le, Q. V. (2015). Semi-supervised sequence learning. Advances in Neural Information Processing Systems, 2015-January.
[106] Peters, M. E., Ammar, W., Bhagavatula, C. & Power, R. (2017). Semi-supervised sequence tagging with bidirectional language models. ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers), 1. https://doi.org/10.18653/v1/P17-1161
[107] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 2017-December.
[108] Clark, K., Khandelwal, U., Levy, O. & Manning, C. D. (2019). What Does BERT Look at? An Analysis of BERT's Attention. https://doi.org/10.18653/v1/w19-4828
[109] Zeman, D., Popel, M., Straka, M., Hajič, J., Nivre, J., Ginter, F., Luotolahti, J., Pyysalo, S., Petrov, S., Potthast, M., Tyers, F., Badmaeva, E., Gökırmak, M., Nedoluzhko, A., Cinková, S., Hajič, J., Hlaváčová, J., Kettnerová, V., Urešová, Z., ... Li, J. (2017). CoNLL 2017 shared task: Multilingual parsing from raw text to universal dependencies. CoNLL 2017 – SIGNLL Conference on Computational Natural Language Learning, Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. https://doi.org/10.18653/v1/k17-3001
[110] Grave, E., Bojanowski, P., Gupta, P., Joulin, A. & Mikolov, T. (2019). Learning word vectors for 157 languages. LREC 2018 - 11th International Conference on Language Resources and Evaluation.
[111] Koutsikakis, J., Chalkidis, I., Malakasiotis, P. & Androutsopoulos, I. (2020a). GREEK-BERT: The Greeks Visiting Sesame Street. 11th Hellenic Conference on Artificial Intelligence, 110–117. https://doi.org/10.1145/3411408.3411440
[112] S, V. & R, J. (2016). Text Mining: open Source Tokenization Tools – An Analysis. Advanced Computational Intelligence: An International Journal (ACII), 3(1). https://doi.org/10.5121/acii.2016.3104