ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΕΦΑΡΜΟΣΜΕΝΩΝ ΜΑΘΗΜΑΤΙΚΩΝ ΚΑΙ ΦΥΣΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΔΙΑΤΜΗΜΑΤΙΚΟ ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

«ΜΑΘΗΜΑΤΙΚΗ ΠΡΟΤΥΠΟΠΟΙΗΣΗ σε ΣΥΓΧΡΟΝΕΣ ΤΕΧΝΟΛΟΓΙΕΣ

και την ΟΙΚΟΝΟΜΙΑ»

Εφαρμογές αλγορίθμων μηχανικής μάθησης στην ανίχνευση ψευδών ειδήσεων και διερεύνηση χρήσης τους ως εργαλεία των ευρωπαϊκών πολιτικών καταπολέμησης του φαινομένου της διαδικτυακής παραπληροφόρησης

ΘΕΟΔΟΣΙΟΣ ΚΟΡΑΛΗΣ

ΑΡΙΘΜΟΣ ΜΗΤΡΩΟΥ: 09317009

ΕΠΙΒΛΕΠΟΝΤΕΣ ΚΑΘΗΓΗΤΕΣ: Δρ. Π. Στεφανέας, Δρ. Π. Τσιαβός

ΑΘΗΝΑ, ΣΕΠΤΕΜΒΡΙΟΣ 2022

Πίνακας περιεχομένων

Εισαγωγή – Το φαινόμενο της παραπληροφόρησης στην ψηφιακή Ευρωπαϊκή Ένωση 6

Περίληψη 7

Summary 8

Κεφάλαιο 1 - Ορισμοί και έννοιες 9

1.1 Πλαίσιο και ευρωπαϊκές στρατηγικές προσδιορισμού και περιορισμού της παραπληροφόρησης μέσω της αναγνώρισης των ψευδών ειδήσεων 10

1.2 Μοντέλα περιγραφής και ταξινόμησης 12

1.3 Διαθέσιμα σύνολα δεδομένων 13

Κεφάλαιο 2 - Υπολογιστικές μέθοδοι αναγνώρισης ψευδών ειδήσεων 16

2.1 Επισκόπηση Μεθόδων ανάλυσης περιεχομένου 16

2.2 Μέθοδοι ανάλυσης περιεχομένου - Μοντέλα Επεξεργασίας Φυσικής Γλώσσας (NLP) 17

2.2.1 Δημιουργία / Εξαγωγή χαρακτηριστικών περιεχομένου 19

2.2.2 Μοντέλα ταξινόμησης Μηχανικής Μάθησης 31

2.3 Μέθοδοι ανάλυσης συμπεριφοράς χρήστη– Social Context based/ Network methods 36

2.4 Μέθοδοι βασισμένες σε βαθιά Μηχανική Μάθηση - Deep Learning Based 37

Κεφάλαιο 3 - Εφαρμογή αλγορίθμων μηχανικής μάθησης για την αναγνώριση ψευδών ειδήσεων 41

3.1 Αξιολόγηση και επιλογή μεθόδου ταξινόμησης 41

3.2 Επιλογή πλατφόρμας εφαρμογής 42

3.3 Εύρεση και επιλογή βάσης δεδομένων 43

3.4 Περιγραφή Συνόλου Δεδομένων 44

3.5 Ανάπτυξη εργαλείου ταξινόμησης 47

3.5.1 Προετοιμασία συνόλου δεδομένων και διακριτοποίηση 47

3.5.2 Επιλογή Χαρακτηριστικών....49

3.5.3 Εκπαίδευση μοντέλων μηχανικής μάθησης....50

3.6 Παρουσίαση και αξιολόγηση αποτελεσμάτων αλγορίθμων....51

Κεφάλαιο 4 - Συμπεράσματα & μελλοντική έρευνα....62

4.1 Συμπεράσματα....62

4.2 Προτάσεις για μελλοντική έρευνα....63

4.2.1 Ευρωπαϊκές Ερευνητικές Πρωτοβουλίες....64

4.2.2 Συνδυαστικές μορφές αξιολόγησης εγκυρότητας – Σύνδεση με ευρωπαϊκές στρατηγικές....65

4.2.3 Πλαίσιο τεχνορρύθμισης....66

Βιβλιογραφία....68

Παράρτημα....80

Εικόνες και Γραφήματα

Εικόνα 1 - Εξέλιξη αριθμού δημοσιεύσεων ανά έτος αναζητώντας την έκφραση "fake news detection" σύμφωνα με τους (Choras, και συν., 2021)....15

Εικόνα 2 - Πλήθος συνόλων δεδομένων που χρησιμοποιούνται σε επιστημονικές εκδόσεις ανά έτος (D'Ulizia, Cascherra, Ferri, & Grifoni, 2021)....15

Εικόνα 3 - Ταξινόμηση μεθόδων ανάλυσης περιεχομένου (Varshney & Vishwakarma, 2021)....16

Εικόνα 4 - Απεικόνιση του τρόπου εκτέλεσης του αλγορίθμου CBOW και της εκπαίδευσης των πινάκων μεταφοράς, (Manning, et al., 2019)....27

Εικόνα 5 - Γραφική αναπαράσταση της λειτουργίας του αλγορίθμου Skip-gram και της εκπαίδευσης πινάκων μεταφοράς (Manning, και συν., 2019)....29

Εικόνα 6 -Ταξινόμηση μεθόδων βαθιάς μηχανικής μάθησης (Islam, Liu, Wang, & Xu, 2020)....39

Εικόνα 7 - Επισκόπηση των μεθόδων για την αναγνώριση ψευδών ειδήσεων (Guo, και συν., 2021) 40

Εικόνα 8 - Δείγμα εξαγωγής του ανοιχτού συνόλου δεδομένων Fake News Inference Dataset (Sadeghi, Bidgoly, & Amirkhani, 2020) 45

Εικόνα 9 - Κατανομή κλάσης αξιοπιστίας των ειδήσεων του FNID [0: TRUE, 1: FAKE] 46

Εικόνα 10 - Γραφικό των συχνότερα εμφανιζόμενων λέξεων στο πεδίο "Statement" του FNID. Το μέγεθος εξαρτάται από τη συχνότητα εμφάνισης της εκάστοτε λέξης 46

Εικόνα 11 - Διαδικασία Επεξεργασίας δεδομένων και διακριτοποίησης 48

Εικόνα 12 - Δείγμα python script για καθαρισμό και διακριτοποίηση του κειμένου των ειδήσεων 49

Εικόνα 13 - Διάγραμμα ροής εφαρμογής εκπαίδευσης και πρόβλεψης μοντέλων ΜΜ μέσω διακριτοποίησης με αθροιστές λέξεων και TF-IDF 50

Εικόνα 14 - Διάγραμμα ροής εφαρμογής εκπαίδευσης και πρόβλεψης μοντέλων ΜΜ μέσω διακριτοποίησης και διανυσματικής αναπαράστασης μέσω GloVe 51

Εικόνα 15 - Λίστα των εφαρμοζόμενων μεθόδων της εργασίας, μαζί με τις αντίστοιχες παραμέτρους 52

Εικόνα 16 - Αποτελέσματα εφαρμογής μεθόδων μετά τη διακριτοποίηση κειμένου βάσει της μεθόδου CountVectorizer στο υποσύνολο εκπαίδευσης 53

Εικόνα 17 - Γραφήματα αξιολόγησης ταξινόμησης (Confusion Graphs) της μεθόδου CountVectorizer, στο υποσύνολο εκπαίδευσης 54

Εικόνα 18 - Αποτελέσματα εφαρμογής μεθόδων μετά τη διακριτοποίηση κειμένου βάσει της μεθόδου nGram , TF-IDF (3) 55

Εικόνα 19 - Γραφήματα αξιολόγησης ταξινόμησης (Confusion Graphs) της μεθόδου ngram(3) – TF-IDF στο υποσύνολο εκπαίδευσης. 56

Εικόνα 20 - GridSearch αναζήτηση των βέλτιστων παραμέτρων για τον αλγόριθμο Random Forest 57

Εικόνα 21 - GridSearch αναζήτηση βέλτιστων παραμέτρων για τον αλγόριθμο Logistic Regression 57

Εικόνα 22 - Απόδοση τελικών μοντέλων RF + LR μετά την εφαρμογή Gridsearch για εύρεση βέλτιστων παραμέτρων.....58

Εικόνα 23 - Γραφήματα αξιολόγησης ταξινόμησης (Confusion Graphs) των αλγορίθμων RF και LR,μετά την επιλογή των βέλτιστων παραμέτρων μέσω GridSearch, στο υποσύνολο εκπαίδευσης.....58

Εικόνα 24 - Γραφήματα εξέλιξης μάθησης των αλγορίθμους υπό εξέταση – Με κόκκινο η εξέλιξη του υποσυνόλου μάθησης και με πράσινο του υποσυνόλου validation. 59

Εικόνα 25 - Precision - Recall GRaph για τα βέλτιστα RF και LR 60

Εικόνα 26 - Στοιχεία του LSTM RNN μοντέλου ταξινόμησης.....61

Εικόνα 27 - RNN LSTM - Κείμενο 600 λέξεων 61

Εικόνα 28 - RNN LSTM - Περίληψη 30 λέξεων.....61

Εισαγωγή – Το φαινόμενο της παραπληροφόρησης στην ψηφιακή Ευρωπαϊκή Ένωση

Η μετάβαση σε ένα ολοένα και περισσότερο ψηφιοποιούμενο περιβάλλον λειτουργίας, εξοπλίζει τους ευρωπαίους πολίτες με πολλούς νέους τρόπους προσωπικής έκφρασης αλλά και πρόσβασης στην πληροφορία συνολικά, αυξάνοντας όμως παράλληλα τον κίνδυνο έκθεσής τους σε μεγάλο όγκο διακινούμενων μορφών παραπληροφόρησης.

Σύμφωνα με την υπάρχουσα, εκτενή βιβλιογραφία, η συστηματική έκθεση σε διαφόρων μορφών πηγές παραπληροφόρησης ενέχει κινδύνους εθνικής και συστημικής ασφαλείας, λειτουργίας των δημοκρατικών θεσμών, κοινωνικής συνοχής και μπορεί να υπονομεύσει την εμπιστοσύνη σε πολιτικές της Κοινωνίας της Πληροφορίας όπως επίσης και στη λειτουργία της ενιαίας Ψηφιακής Αγοράς.

Η Ευρωπαϊκή Ένωση και πολλά από τα κράτη μέλη της αντιμετωπίζουν τους ανωτέρω κινδύνους έχοντας ως σημαντικά εργαλεία την υψηλού επιπέδου μόρφωση, τα υψηλά ποσοστά ψηφιακά διασυνδεδεμένων πληθυσμών, τους ισχυρούς πολιτικούς θεσμούς, την πολυφωνία Μέσων Ενημέρωσης και τις ανταγωνιστικές κρατικές οικονομίες. (Gabriel & Buning, 2018)

Τα ενδεχόμενα προβλήματα της παραπληροφόρησης, και ειδικότερα μέσω της διασποράς ψευδών ειδήσεων, είναι άρρηκτα συνδεδεμένα με την εξέλιξη της χρήσης των μέσων κοινωνικής δικτύωσης καθώς και των ψηφιακών πλατφορμών επικοινωνίας, τα οποία και εξαρτώνται από τις ενέργειες των εμπλεκόμενων μερών (κρατικοί ή μη παράγοντες, εταιρείες που αποσκοπούν στο κέρδος, Μέσα Μαζικής Ενημέρωσης καθώς και πολίτες ως μονάδες ή συλλογικότητες. Την τελευταία δεκαετία, έχουν ενταθεί οι ερευνητικές προσπάθειες στο χώρο των ψηφιακών μέσων για την χαρτογράφηση και αποσαφήνιση των νέων μεθόδων που εφαρμόζονται μέσω των υποδομών επικοινωνίας (ΜΜΜ, πλατφόρμες διασύνδεσης, κοινωνικά δίκτυα κ.α.) με απώτερο σκοπό την παραγωγή, διακίνηση και μεγέθυνση φαινομένων παραπληροφόρησης σε μεγάλη κλίμακα.

Περίληψη

Η παρούσα εργασία στοχεύει τόσο στην περιγραφή και ταξινόμηση με βάση την αλγοριθμική προσέγγιση, των υπαρχόντων εργαλείων εύρεσης/ αναγνώρισης ψευδών ειδήσεων σε διαδικτυακές πλατφόρμες ενημέρωσης όσο και στην παρουσίαση των αποτελεσμάτων πέντε εξ' αυτών μετά την εφαρμογή τους σε συγκεκριμένο σύνολο ανοιχτών δεδομένων ειδήσεων (Fake News Inference Dataset – FNID). Μετά την παρουσίαση των ποσοτικών αποτελεσμάτων της εφαρμογής, θα συζητηθούν τόσο εναλλακτικές ή/και συμπληρωματικές μέθοδοι για την αναγνώριση ψευδών ειδήσεων αλλά θα τοποθετηθούν και στο ευρύτερο πλαίσιο συζήτησης προς εφαρμογή τους ως κοινοτικά (ευρωπαϊκά) εργαλεία. Επίσης, γίνεται αναφορά στην ποιότητα των δεδομένων και πως αυτή επηρεάζει την απόδοση των υπαρχόντων εργαλείων και μεθόδων αλλά και πως τα διαθέσιμα ανοιχτά και μη σύνολα δεδομένων μπορούν να εμπλουτιστούν για αυτό τον σκοπό.

Summary

The present work's main aim is to describe and classify based on their algorithmic approach, the already available tools of finding and identifying fake news as parts of online media and news platforms. Additionally, five of those algorithmic approaches are applied on an open News Dataset (Fake News Inference Dataset – FNID) and their results are hereby presented. Following the quantitative results' presentation, additional and/or alternative methods of fake news identification methods will be mentioned as potential candidates for inclusion in various European Union's current and future initiatives against disinformation. Furthermore, data quality and how it affects the efficiency and accuracy of the existing classification and identification tools is also briefly examined. We will finally mention ways based on referenced literature to enhance currently, open or not, news datasets in order to aid the fake news identification processes.

Κεφάλαιο 1 - Ορισμοί και έννοιες

Ο πολλαπλασιασμός του περιεχομένου, δημιουργούμενου απευθείας από διαδικτυακούς χρήστες όπως επίσης και η επικράτηση των πλατφορμών Επικοινωνίας μέσω Υπολογιστή (Computer Mediated Communication Platforms – CMC Platforms), έχουν μεγεθύνει πολλαπλά τη δυνατότητα και ταχύτητα διάδοσης ειδήσεων σε παγκόσμια κλίμακα (Conroy, Rubin, & Chen, 2015).

Η ευρωπαϊκή επιτροπή αναγνωρίζοντας τις ραγδαίες αλλαγές στους τρόπους επικοινωνίας μέσω Διαδικτύου έχει ενεργοποιήσει ποικίλες δράσεις οι οποίες αποσκοπούν στην αναγνώριση προσπαθειών αλλά και αντιμετώπιση της παραπληροφόρησης συνολικά για τους πολίτες της ΕΕ. (European Commission, 2018). Όλες οι δράσεις συντονίζονται από μια ειδική ομάδα εμπειρογνωμόνων η οποία λογοδοτεί στην Ευρωπαϊκή Επιτροπή και προτείνει πρωτοβουλίες διαμόρφωσης ευρωπαϊκών πολιτικών για την αντιμετώπιση της διάχυσης των ψευδών ειδήσεων και της παραπληροφόρησης μέσω διαδικτύου.

Η ειδική ομάδα εμπειρογνωμόνων (High Level Expert Group) συμπεριλαμβάνει μέσω της έκθεσής της στις μορφές παραπληροφόρησης, εκείνες τις μορφές λόγου που εμπίπτουν εκτός των κατηγοριών παράνομων εκφράσεων λόγου (προσβολή, δυσφήμιση κ.α.) αλλά μπορούν παρόλα αυτά να είναι επιβλαβείς. Η λανθασμένη πληροφόρηση (misinformation) ορίζεται από την ίδια επιστημονική επιτροπή ως παραπλανητικά ή ανακριβή σύνολα πληροφορίας τα οποία όμως έχουν ακούσια παραχθεί ή διαμοιραστεί.

Η παραπληροφόρηση (disinformation) ορίζεται ως ο σχεδιασμός, παρουσίαση ή προώθηση αναληθών, ανακριβών η παραπλανητικών συνόλων πληροφοριών σκόπιμα με στόχο το κέρδος συγκεκριμένων μερών ή την πρόκληση προβλημάτων στο δημόσιο βίο.

Η ίδια επιτροπή, αν και αποφεύγει να ορίσει σε αυστηρά νομικά πλαίσια την κατηγορία των ψευδών ειδήσεων, καθώς τη θεωρεί ως ένα μικρό υποσύνολο της ευρύτερης οικογένειας της παραπληροφόρησης αρκείται στον εξής προσδιορισμό: Ως ψευδείς ειδήσεις θεωρούνται εκείνα τα σύνολα πληροφοριών που διαχέονται μέσω πλατφορμών επικοινωνίας τα οποία εμπεριέχουν κομμάτι η αποτελούνται εξ ολοκλήρου από ανακριβείς και αναληθείς πληροφορίες, χωρίς

αντίστοιχη αναφορά στις ανακρίβειες αυτές. Η επιτροπή εμπειρογνωμόνων τονίζει επίσης πως στην κατηγορία αυτή εμπίπτουν ειδήσεις είτε των οποίων το περιεχόμενο είναι εσκεμμένα κατασκευασμένο με τέτοιο τρόπο που αναμιγνύει αληθινά με ψευδή στοιχεία είτε είναι αποτέλεσμα πρακτικών που ως στόχο δεν έχουν την ενημέρωση (διακίνηση μέσω ψευδών λογαριασμών, διαφήμιση κ.α.). Η χρήση του όρου «ψευδείς ειδήσεις» για πολιτικές σκοπιμότητες με σκοπό πάντα την προώθηση προσωπικών ή συλλογικών πολιτικών επιλογών είναι ένας επιπλέον λόγος για τον οποίο η επιτροπή δεν τον υιοθετεί ως τη μοναδική πηγή παραπληροφόρησης.

Ως (ψηφιακή/υπολογιστική) πλατφόρμα επικοινωνίας, ορίζεται εκείνο το φάσμα δραστηριοτήτων που περιλαμβάνει ανταλλαγή πληροφοριών μεταξύ χρηστών και μπορεί να εμπεριέχει τα μέσα κοινωνικής δικτύωσης, ψηφιακές αγορές, μηχανές αναζήτησης, καταστήματα εφαρμογών, συστήματα πληρωμής, πλατφόρμες συνεργασίας, ειδησεογραφικές ιστοσελίδες και πολλά άλλα. Χαρακτηρίζονται από κοινού από τη χρήση της πληροφορίας αλλά και των τεχνολογικών μέσων επικοινωνίας που χρησιμοποιούν έτσι ώστε να διευκολύνουν τις διεπαφές μεταξύ των χρηστών τους ως «μεσάζοντες» πληροφορίας, συλλέκτες και χρήστες των παραγόμενων πληροφοριών αλληλεπιδράσεων και διαχειριστές των δικτύων διεπαφών με απώτερο σκοπό τη μεγιστοποίηση της αξίας χρήσης της πλατφόρμας για τους χρήστες. Στο πλαίσιο του περιορισμού διασποράς των μορφών παραπληροφόρησης, οι πλατφόρμες θα μπορούσαν να αποτελέσουν σημαντικά σημεία στην αλυσίδα διάδοσης των ψευδών ειδήσεων και οι ενδεχόμενες προτάσεις ή εργαλεία αντιμετώπισης αυτής της διασποράς θα πρέπει να είναι σχεδιασμένα βάσει των χαρακτηριστικών της κάθε πλατφόρμας, ακολουθώντας βεβαίως σε υψηλό επίπεδο τις γενικές αρχές λειτουργίας ανεξάρτητες από την πλατφόρμα εφαρμογής. (European Commission, 2018)

1.1 Πλαίσιο και ευρωπαϊκές στρατηγικές προσδιορισμού και περιορισμού της παραπληροφόρησης μέσω της αναγνώρισης των ψευδών ειδήσεων

Το ευρύτερο πλαίσιο λειτουργίας των ευρωπαϊκών θεσμών ενάντια στην παραπληροφόρηση, περιλαμβάνει αρχικά την κατανόηση σε βάθος του μεγέθους,

των στόχων και της ακριβούς φύσης των προβλημάτων που δημιουργούνται με σκοπό το σχεδιασμό λύσεων που αντιμετωπίζουν το αντίστοιχο πρόβλημα χωρίς να δυσχεραίνουν την ομαλή λειτουργία του ευρωπαϊκού ψηφιακού οικοσυστήματος και των τεχνολογιών. Η συνεχής αξιολόγηση του φαινομένου κρίνεται απαραίτητη έτσι ώστε οι λύσεις που σχεδιάζονται να είναι μετρήσιμες, επιτεύξιμες, ρεαλιστικές και εντός αποδεκτών χρονοδιαγραμμάτων.

Η συνολική προσέγγιση της επιτροπής χωρίζεται σε τρεις βασικές κατηγορίες:

1. Στην αρχική αξιολόγηση των μέτρων ενάντια στη διασπορά ψευδών ειδήσεων την παραπληροφόρηση γενικότερα, τα οποία ήδη εφαρμόζουν οι διάφοροι εμπλεκόμενοι φορείς (χρήστες, πλατφόρμες, θεσμοί)
2. Στην αξιολόγηση του υφιστάμενοι νομικού πλαισίου και των αρχών στις οποίες αυτό έχει βασιστεί.
3. Στην αναγνώριση πιθανών στρατηγικών απέναντι στο συνολικό πρόβλημα οι οποίες θα πηγάζουν από τα 2 παραπάνω σημεία.

Αναφορικά με τις υφιστάμενες πρωτοβουλίες των εμπλεκόμενων φορέων, το πλαίσιο αποδοχής ορίζει την καταπολέμηση του φαινομένου της ψηφιακής παραπληροφόρησης μέσω βέλτιστων πρακτικών που θα σέβονται την ελευθερία του λόγου και όλα τα θεμελιώδη ανθρώπινα δικαιώματα. Ενδεικτικά, οι ενέργειες που ήδη εφαρμόζονται περιλαμβάνουν την ενίσχυση πρακτικών διαφάνειάς, την εφαρμογή και τροποποίηση των αλγορίθμων εύρεσης και αναγνώρισης ψευδών ειδήσεων και τέλος την ενίσχυση του ψηφιακού αλφαβητισμού.

Για το πρώτο σκέλος της ενίσχυσης των πρακτικών διαφάνειας και υπευθυνότητας, οι διαδικτυακές πλατφόρμες εστιάζουν τις προσπάθειές τους στην αναγνώριση και αφαίρεση χρηστών που χρησιμοποιούν αθέμιτους τρόπους επικοινωνίας, στην χρήση αλγορίθμων ταξινόμησης της αξιοπιστίας περιεχομένου, οι οποίοι δίνουν πρόσβαση σε αντίστοιχες με το περιεχόμενο ειδήσεις και στη συνεργασία με ανεξάρτητους οργανισμούς εμπειρογνωμόνων που βοηθούν στην αξιολόγηση της αυθεντικότητας και ακρίβειας των διακινούμενων ειδήσεων και πληροφοριών. Όλες αυτές οι προσπάθειες είναι ακόμη σε πολύ χαμηλά επίπεδα υιοθέτησης από τη συνολική βάση χρηστών της

κάθε πλατφόρμας, καθώς μέχρι στιγμής εστιάζονται σε συγκεκριμένες γεωγραφικές περιοχές αλλά και για συγκεκριμένες γλώσσες.

Αναφορικά με τις ανεξάρτητες πηγές και ομάδες αξιολόγησης αυθεντικότητας ειδήσεων και πληροφοριών, δεν υπάρχει μέχρι στιγμής κάποιο θεσμικό πλαίσιο μέσα στο οποίο η Ευρωπαϊκή Ένωση να προσπαθεί να συνεργαστεί με αυτούς τους φορείς.

Η δεύτερη κατηγορία στρατηγικών αντιμετώπισης βασίζεται στη χρήση εργαλείων τεχνητής νοημοσύνης και μηχανικής μάθησης ως μέτρα φιλτραρίσματος και ταξινόμησης της διακινούμενης πληροφορίας. Στο πλαίσιο αυτό, τοποθετείται και η εφαρμογή της παρούσας εργασίας. Μέσω της εφαρμογής των μοντέλων μηχανικής μάθησης και σε συνδυασμό με τους φορείς αξιολόγησης αυθεντικότητας (όπως για παράδειγμα ο IFCN – International Fact Checking Network), οι ενεργές πλατφόρμες μέχρι στιγμής προσπαθούν να αναγνωρίσουν και να περιορίσουν τη διασπορά ψευδών ειδήσεων.

1.2 Μοντέλα περιγραφής και ταξινόμησης

Η διαδικασία της αναγνώρισης ψευδών ειδήσεων ορίζεται ως η προσπάθεια κατάταξης και ταξινόμησης των ειδήσεων σε ένα φάσμα ή μέσω μια κλίμακας αξιοπιστίας, μαζί με ένα ποσοστό βεβαιότητας της ταξινόμησης αυτής. (Conroy, Rubin, & Chen, 2015)

Η αξιοπιστία των ειδήσεων δυνητικά μειώνεται με την παρουσία εσκεμμένων, παραπλανητικών χαρακτηριστικών. Η φύση της διαδικτυακής δημοσίευσης ειδήσεων έχει αλλάξει κατά τέτοιο τρόπο που η εφαρμογή παραδοσιακών μεθόδων αναγνώρισης ψευδών στοιχείων ή/και ολόκληρων ειδήσεων καθίσταται πολύ δύσκολη κυρίως λόγω του πλήθους των ειδών δημοσίευσης, δημιουργών περιεχομένου αλλά και πηγών δημοσίευσης. (Rubin V. L., Conroy, Chen, & Sarah, 2016)

Οι περισσότερες αυτόματες τεχνικές της πιο πρόσφατης βιβλιογραφίας χρησιμοποιούν μοντέλα μηχανικής μάθησης και δομούν το πρόβλημα ταξινόμησης των ειδήσεων σε ψευδείς η αληθείς ως ένα πρόβλημα δυαδικής

ταξινόμησης, μέσω μεθόδων εποπτευόμενης μάθησης (supervised learning). Κατά αυτό τον τρόπο, οι ειδήσεις κατηγοριοποιούνται συνολικά και ένα σύνολο από αυτές χρησιμοποιείται τόσο για την εκπαίδευση όσο και (μέρος αυτού) για τον έλεγχο της αξιοπιστίας και ακρίβειάς τους (D'Ulizia, Cascherra, Ferri, & Grifoni, 2021). Για αυτό το λόγο, η διαθεσιμότητα της συνολικής πληροφορίας της κάθε είδησης είναι άρρηκτα συνδεδεμένη με την ποιότητα πρόβλεψης των αυτοματοποιημένων μεθόδων ταξινόμησης.

1.3 Διαθέσιμα σύνολα δεδομένων

Η αυξημένη σημαντικότητα της έγκαιρης και έγκυρης εύρεσής ψευδών ειδήσεων μεταξύ της ερευνητικής κοινότητας οφείλεται σε παράγοντες που σχετίζονται τόσο με την εξελισσόμενη φύση των ειδήσεων μέσω των δικτυακών πλατφορμών όσο και με την ολοένα αυξανόμενη ταχύτητα διάδοσης τους μέσα από τα μέσα δικτυακής επικοινωνίας. Αρκετά σύνολα δεδομένων που περιέχουν χαρακτηρισμένες και ταξινομημένες ειδήσεις έχουν χρησιμοποιηθεί τα τελευταία χρόνια ως ερευνητικά εργαλεία για την ανάπτυξη νέων και βελτίωση υπαρχόντων αυτοματοποιημένων μεθόδων εύρεσης ψευδών ειδήσεων (D'Ulizia, Cascherra, Ferri, & Grifoni, 2021).

Η συλλογή ειδήσεων, η αξιολόγησή τους και η συνεχής δημιουργία ή ανάπτυξη των διαθέσιμων συνόλων δεδομένων για τις ανωτέρω ερευνητικές προσπάθειες είναι μια διαδικασία που αποτελείται από αρκετά στάδια επεξεργασίας της κάθε είδησης ξεχωριστά. Πρώτα από΄ όλα απαιτείται η διασταύρωση πληροφορίας (fact-checking) του περιεχομένου των ειδήσεων έτσι ώστε να χαρακτηριστούν οι ειδήσεις ψευδείς ή αληθείς. Οι διάφοροι τρόποι διασταύρωσης της αυθεντικότητας είναι η αξιολόγηση μέσω εμπειρογνωμόνων, μέσω αξιοποίησης υπολογιστικών εργαλείων (knowledge graphs ή ηλεκτρονικές πηγές ελεύθερης πρόσβασης), μέσω ευρείας αξιολόγησης από πλατφόρμες crowdsourcing και τέλος μέσω ιστοσελίδων αξιολόγησης αξιοπιστίας ειδήσεων. Η πρώτη από τις ανωτέρω μεθόδους απαιτεί πολύ χρόνο, καθώς περιλαμβάνουν αλληλεπίδραση μεταξύ ανθρώπου – περιεχομένου και τη χειρωνακτική αξιολόγηση της κάθε είδησης ξεχωριστά. Οι τελευταίες 3 μέθοδοι προσφέρονται για χρήση ψηφιακών

εργαλείων και συλλογικής ανάμιξης περισσότερων του ενός χρηστών για την αναζήτηση, ανάλυση και τελική αξιολόγηση του περιεχομένου των ειδήσεων.

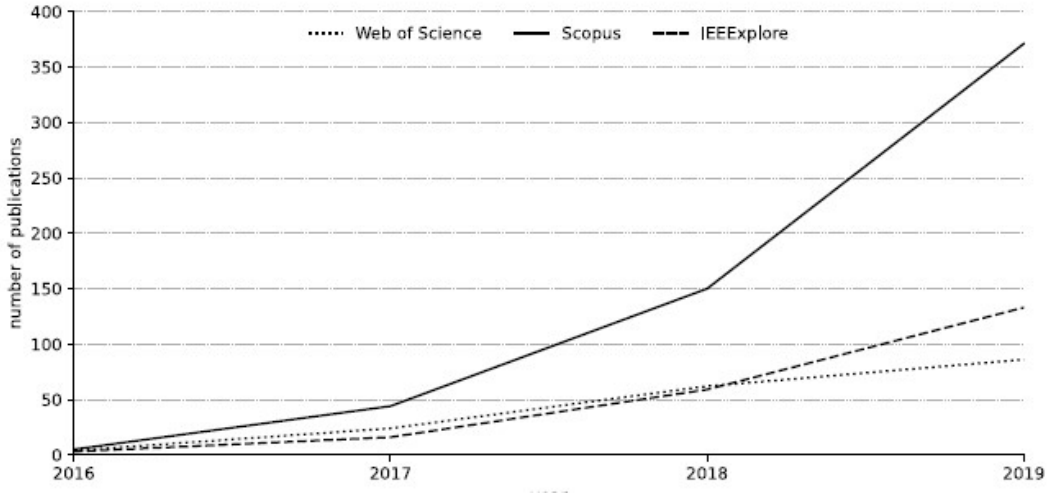
Σημαντικό περιορισμό για την πλήρως αυτοματοποιημένη αξιολόγηση των ειδήσεων, κυρίως μέσω του περιεχομένου τους, αποτελεί η εγγενής αοριστία της χρήσης της ανθρώπινης γλώσσας που δυσχεραίνει την ακρίβεια σε αυτοματοποιημένες μεθόδους πρόβλεψης αξιοπιστίας. Καθιστώντας έτσι τις συνολικά βασιζόμενες στη συλλογική αξιολόγηση (crowd assessment) λιγότερο αποδοτικές. Εξαιτίας του παραπάνω, η δημιουργία πλούσιων, αξιόπιστων συνόλων δεδομένων ειδήσεων καθίσταται χρονοβόρα και επίπονη διαδικασία, δημιουργώντας παράλληλα την ανάγκη για συμβιβασμό μεταξύ του μεγέθους, της αξιοπιστίας και του χρόνου εμπλουτισμού του κάθε συνόλου. (D'Ulizia, Cascherra, Ferri, & Grifoni, 2021)

Άλλες παράμετροι που λαμβάνονται υπόψη κατά την ανάπτυξη και εμπλουτισμό τέτοιων συνόλων δεδομένων ειδήσεων είναι η ποικιλία θεματολογίας (κοινωνικές, πολιτικές, οικονομικές, υγειονομικές και άλλες ειδήσεις), τα πολυμέσα και τα, επιπλέον του κειμένου, είδη περιεχομένου τα οποία αυτές περιέχουν (εικόνα, κινούμενη εικόνα, γραφήματα, πίνακες δεδομένων κ.α.) και τέλος η διαφορετική φύση είδους παραπληροφόρησης (φήμη, ψευδής είδηση κ.α.)

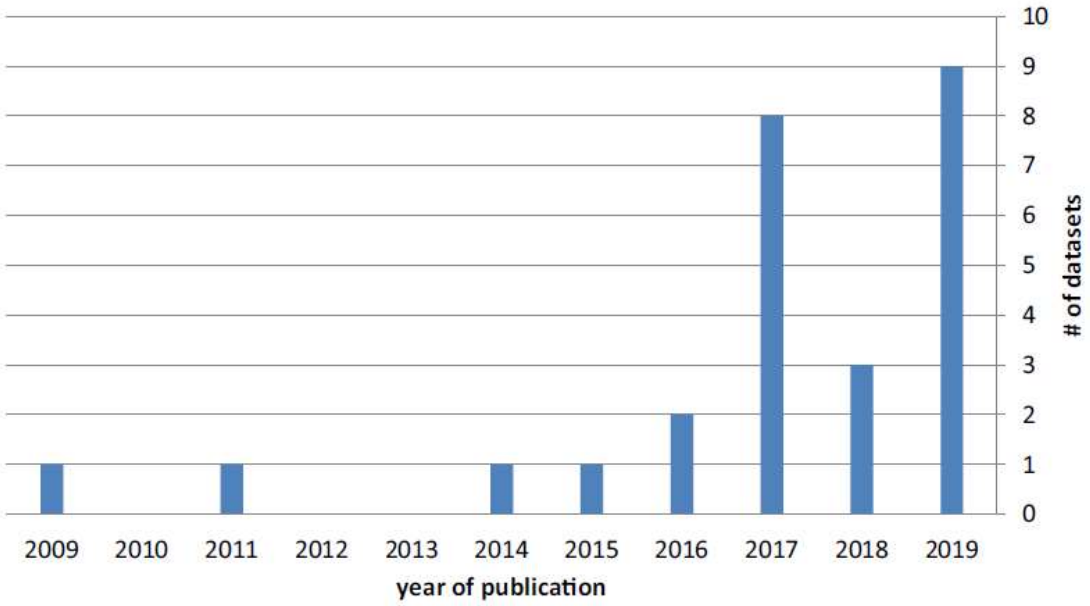
Πέραν όλων των ανωτέρω διαφοροποιήσεων, τα διαθέσιμα σύνολα δεδομένων μπορούν επίσης να ταξινομηθούν βάσει του καναλιού μέσω του οποίου δημοσιοποιούνται οι ειδήσεις οι οποίες περιέχουν, του σκοπού χρήσης του κάθε συνόλου (π.χ. εντοπισμός ψευδών ειδήσεων, εντοπισμός φημών, εντοπισμός clickbait κ.λπ.), της γλώσσας των ειδήσεων, του μεγέθους, του είδους περιεχομένου (μόνο κείμενο, κείμενο και εικόνα, κείμενο και βίντεο κ.α.), της αξιολόγησης από τους χρήστες και της περιόδου εξαγωγής των ειδήσεων.

Η εξέλιξη και ο εμπλουτισμός των διαθέσιμων συνόλων ταξινομημένων ειδήσεων συνάδει με την ραγδαία αύξηση των δημοσιευμένων μελετών γύρω από το αντικείμενο της ανίχνευσης ψευδών ειδήσεων (Εικόνα 4.) και αναμένεται να αυξηθεί (Cloudhary & Arora, 2021) εντός των επόμενων χρόνων καθώς το

ερευνητικό ενδιαφέρον γύρω από την αυτοματοποιημένη αναγνώριση ψευδών ειδήσεων ολοένα και αυξάνεται.



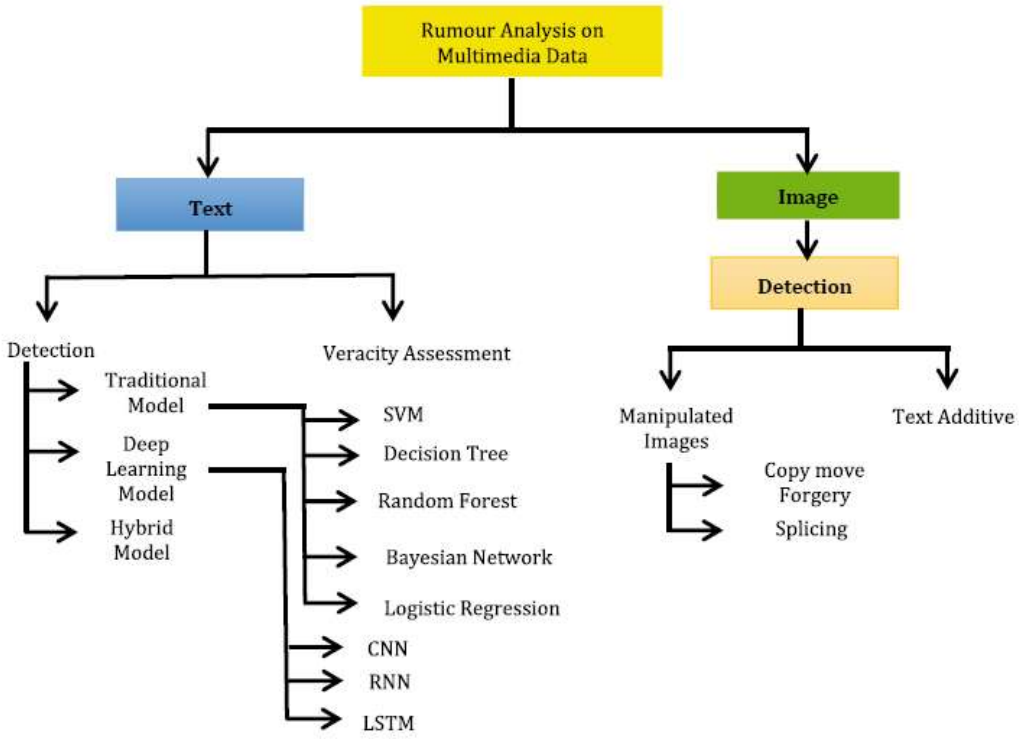
Εικόνα 1 - Εξέλιξη αριθμού δημοσιεύσεων ανά έτος αναζητώντας την έκφραση "fake news detection" σύμφωνα με τους (Choras, και συν., 2021)



Εικόνα 2 - Πλήθος συνόλων δεδομένων που χρησιμοποιούνται σε επιστημονικές εκδόσεις ανά έτος (D'Ulizia, Cascherra, Ferri, & Grifoni, 2021)

Κεφάλαιο 2 - Υπολογιστικές μέθοδοι αναγνώρισης ψευδών ειδήσεων

2.1 Επισκόπηση Μεθόδων ανάλυσης περιεχομένου



Εικόνα 3 - Ταξινόμηση μεθόδων ανάλυσης περιεχομένου (Varshney & Vishwakarma, 2021)

Οι (Varshney & Vishwakarma, 2021) πραγματοποίησαν μια ποσοτική και ποιοτική καταγραφή της βιβλιογραφίας που έχει εφαρμόσει μεθόδους ανάλυσης περιεχομένου και η αρχική κατάταξη έγινε ανάμεσα σε παραδοσιακές τεχνικές που βασίζονται σε επεξεργασία του λεκτικού μέρους των ειδήσεων και σε επεξεργασία και αξιολόγηση αξιοπιστίας και αυθεντικότητας των εικόνων και άλλων πολυμέσων που ενδεχομένως περιέχονται στην είδηση αυτή.

Η ακρίβεια των περισσότερων μεθόδων παραδοσιακής μηχανικής μάθησης που στηρίζονταν σε χαρακτηριστικά των κειμένων κυμαίνεται από 85-95%, ενώ για τις περιπτώσεις μεθόδων υβριδικών η βαθιάς μηχανικής μάθησης, τα ποσοστά ακρίβειας πρόβλεψης αυξάνονται στο 97%, ποσοστά εξαρτώμενα πάντα και από τα αντίστοιχα σύνολα δεδομένων στα οποία εφαρμόζονται οι συγκεκριμένες μέθοδοι.

2.2 Μέθοδοι ανάλυσης περιεχομένου - Μοντέλα Επεξεργασίας Φυσικής Γλώσσας (NLP)

Οι μέθοδοι αυτές βασίζονται κυρίως σε ανάλυση των λεκτικών, συντακτικών και εννοιολογικών χαρακτηριστικών του περιεχομένου των ειδήσεων. Οι αντίστοιχές εφαρμογές περιλαμβάνουν τον υπολογισμό και την ανάλυση γλωσσικών χαρακτηριστικών που εξάγονται από κείμενα ειδήσεων, με σκοπό τη συνολική αξιολόγηση της αξιοπιστίας τους, όπως για παράδειγμα ο μέσος όρος των λέξεων, το πλήθος των εξωτερικών συνδέσμων που περιέχονται στο κείμενο και ο αριθμός των θετικά διακείμενων λέξεων. (Castillo, Mendoza, & Poblete, 2011), (Guo, et al., 2021). Οι (Potthast, Kiesel, Reinartz, Janek, & Stein, 2017) χρησιμοποίησαν επίσης μοντέλα «μέτα-μάθησης» τα οποία συγκρίνουν και βρίσκουν διαφορές μεταξύ κειμένων αληθινών και ψευδών ειδήσεων. Επεκτάσεις των μεθόδων αναγνώρισης που βασίζονται σε περιεχόμενο περιλαμβάνουν εφαρμογές αναγνώρισης γεννητριών ανεπιθύμητης αλληλογραφίας (Huang & Chen, 2020) βασισμένες στην εννοιολογική ανάλυση του περιεχομένου.

Οι μέθοδοι ανάλυσης περιεχομένου είναι και οι πρώτες που επιστρατεύτηκαν για την αλγοριθμική αναγνώριση ψευδών ειδήσεων καθώς χαρακτηρίζονται ως οι πιο εύκολες στην υλοποίηση τους σε σχέση με τις υπόλοιπες κατηγορίες μεθόδων. Συνήθως χρησιμοποιούν χαρακτηριστικά που εξάγονται μέσω ταξινομητών (classifiers) οι οποίοι εκπαιδεύονται μέσω αλγορίθμων μηχανικής μάθησης όπως οι Support Vector Machines και οι Random Forest. Παρά το γεγονός ότι οι ανωτέρω μέθοδοι θεωρούνται τεχνικά επιτυχημένοι, καθώς βασίζονται αποκλειστικά σε δεδομένα σχετιζόμενα με το περιεχόμενο της εκάστοτε είδησης, θεωρούνται και οι πιο εύκολα προσπελάσιμοι. Ενδεικτικά παραδείγματα αδυναμιών των συγκεκριμένων αλγορίθμων ταξινόμησης είναι τα κάτωθι:

- Απαιτούν χρόνο και είναι υπολογιστικά απαιτητικές καθώς παράγουν τα χαρακτηριστικά με μη αυτόματες μεθόδους.
- Είναι ευάλωτες σε διάφορες μορφές ψηφιακής χειραγώγησης, μέσω άλλων αλγορίθμων καθώς βασίζονται αποκλειστικά σε χαρακτηριστικά κειμένου.
- Αναλύουν μεμονωμένα την αξιοπιστία της κάθε είδησης, αγνοώντας την πιθανή σχέση της με αντίστοιχες ειδήσεις. (Guo, et al., 2021)

Από τις βασικότερες μεθόδους αναπαράστασης κειμένου σε υπολογιστικούς αλγορίθμους είναι η μέθοδος των "bag of words", η οποία μεταχειρίζεται την κάθε λέξη ενός κειμένου ως μια ενιαία, εξίσου σημαντική μονάδα. Κατά αυτό τον τρόπο μετρούνται, αθροίζονται και αναλύονται τόσο μεμονωμένα όσο και σε n-επιμέρους συλλογές συνεχόμενων λέξεων (εφεξής n-grams) όλα τα στοιχεία ενός κειμένου, προσπαθώντας να αποκαλύψουν στοιχεία εξαπάτησης (Hancock, Curry, Goorha, & Woodworth, 2008).

Επιπλέον ανάλυση αναφορικά με θέση τους εντός του κειμένου αλλά και την περιπλοκότητα αυτών των n-grams των επιμέρους λέξεων μπορούν να οδηγήσουν σε στοιχεία (σύνολα συχνοτήτων) που εν τέλει εμφανίζουν γλωσσολογικά σημάδια εξαπάτησης.

Η απλότητα της συγκεκριμένης μεθόδου, όπως αναφέρθηκε και στην ανάλογη περιγραφική ενότητα των μεθόδων ανάλυσης περιεχομένου, οδηγεί όμως και στην πλήρη εξάρτησή της αποτελεσματικότητάς της από τα γλωσσικά χαρακτηριστικά , καθώς βασίζεται εξ ολοκλήρου στην ανάλυση των n-grams, χωρίς απαραίτητα αυτή να βασίζεται το εννοιολογικό περιεχόμενο αυτών. Μέσω αυτής της μεθόδου, δεν παρέχεται προστασία αναφορικά με το νόημά ή το εννοιολογικό περιεχόμενο των αναλύσεων, γεγονός που κατατάσσει τη μέθοδο (βιβλιογραφικά) ως συμπληρωματική σε περισσότερες και αυστηρότερες αναλύσεις. (Zhang, Fan, Zeng, & Liu, 2012)

Περιπλοκότερες αναλύσεις γλωσσικής δομής (ή αλλιώς σύνταξης κειμένου) χρησιμοποιούνται επίσης από τη βιβλιογραφία ως συμπληρώματα των απλούστερων "bag-of-words" στην προσπάθεια αναγνώρισης ψευδών ειδήσεων. Διαφορετικής υπολογιστικής προσέγγισης μέθοδοι, μετασχηματίζουν ολόκληρες προτάσεις σε ένα σύνολο κανόνων που αντικατοπτρίζει τη συντακτική δομή σε μιας μορφής διάγραμμα δέντρου μέσω ειδικών αλγορίθμων (Probability Context Free Grammars), απαλλαγμένων από τις συγκεκριμένες λέξεις, αποτυπώνοντας έτσι τη συντακτική δομή ολόκληρων κειμένων. (Feng, Banerjee, & Choi, 2012). Μέσω στατιστικής ανάλυσης της συνολικής συντακτικής δομής του κειμένου, εμφανίζονται μοτίβα που βοηθούν στην σύγκριση με αληθείς και τελικώς αναγνώριση ψευδών ειδήσεων. Οι μέθοδοι αυτές ονομάζονται μέθοδοι βαθιάς σύνταξης (Deep Syntax Methods).

Οι πιο σύγχρονες μέθοδοι ανάλυσης περιεχομένου περιλαμβάνουν τη μετάβαση από το χώρο κειμένου στο διανυσματικό χώρο. Μέσω της μεθόδου του "Word Embedding", η ανάλυση περιεχομένου δε βασίζεται πλέον στην απλουστευμένη σύγκριση μεταξύ απλών λέξεων αλλά λαμβάνει υπόψη συσχετισμούς στο νόημα μεταξύ συγγενικών λέξεων αξιοποιώντας μεθόδους ελαχιστοποίησης απόστασης στο διανυσματικό χώρο που δημιουργείται από τα διανύσματα των εκάστοτε λέξεων του κειμένου, που βοηθούν στον προσδιορισμό των εννοιών της ομοιότητας ή εννοιολογικής διαφοράς μεταξύ των λέξεων (Mikolov, Chen, Corrado, & Dean, 2013). Σαν πρώτο βήμα, η ανωτέρω χαρτογράφηση των λέξεων στον διανυσματικό πλέον χώρο δίνει τη δυνατότητα εφαρμογής δύο νέων μεθόδων προσδιορισμού ομοιότητας στο νόημα δύο διαφορετικών προτάσεων. Οι δημοφιλέστερες μέθοδοι είναι η ανάλυση των δημιουργούμενων από τις προτάσεις πινάκων ως σύνολα των επιμέρους διανυσμάτων στις ιδιάζουσες τιμές τους (Singular Value Decomposition - SVD), ως μιας μορφής παραγοντοποίηση και μοντελοποίηση ελαττωμένης τάξης, αποδίδοντας εν τέλει πραγματικές τιμές πιθανότητας στους συσχετισμούς των λέξεων μεταξύ τους. Έτσι τελικά, δίνεται η δυνατότητα ακόμη και με μικρές αλλαγές στο περιεχόμενο ειδήσεων, να γίνει σωστή σύγκριση και ταξινόμησή τους ως αληθείς ή ψευδείς.

Σε κείμενα μεγάλης έκτασης η ανωτέρω μεθοδολογία SVD απαιτεί αυξημένες υπολογιστικές δυνατότητες και αντί αυτής προτιμάται η μέθοδος «Continuous Bag of Words – CBOW», όπου μέσω τυπολογικής βελτιστοποίησης, συσχετίζει τις λέξεις μέσω της διανυσματικής μετρικής τους απόστασης στο διανυσματικό χώρο μέσω πινάκων μεταφοράς.

2.2.1 Δημιουργία / Εξαγωγή χαρακτηριστικών περιεχομένου

Μετά την αρχική επεξεργασία και καθαρισμό τους, τα σύνολα δεδομένων που εμπεριέχουν τα κείμενα των ειδήσεων που πρόκειται να ταξινομηθούν, οργανώνονται σε αριθμητικές αναπαραστάσεις με τη μορφή διανυσμάτων, που αναπαριστούν τη σύνταξη και το είδος των λέξεων που περιέχουν. Κάθε είδηση που αποτελεί κομμάτι του συνόλου δεδομένων, που αρχικά απαρτίζεται αποκλειστικά από μια σειρά χαρακτήρων, μετατρέπεται σε ένα σύνολο διακριτών

λέξεων και στη συνέχεια μετασχηματίζεται σε ένα διάνυσμα χαρακτηριστικών (feature vector). Το βήμα αυτό πραγματοποιείται μέσω έτοιμων αλγορίθμων προ-εκπαίδευσης (pre-processing, pre-training) των επακόλουθων μοντέλων μηχανικής μάθησης.

2.2.1.1 Count Vectorizer – Αλγόριθμοι Αθροιστών λέξεων

Ο απλούστερος τρόπος μετατροπής σε διανυσματικά μεγέθη του λεκτικού μέρους των ειδήσεων προς ταξινόμηση είναι η αναπαράστασή τους σε αραιούς πίνακες. Οι αλγόριθμοι CountVectorizer kai BagOfWords, προσφέρουν απλούς τρόπους διακριτοποίησης των κειμένων σε επιμέρους σύνολα λέξεων, αναπαριστώντας έτσι όλα τα επιμέρους κείμενα του συνόλου των ειδήσεων με τη μορφή συνόλου λέξεων, δημιουργώντας παράλληλα ένα ολικό σύνολο λέξεων που χρησιμοποιείται ως ο χάρτης λεξιλογίου για όλο τον πληθυσμό.

Ειδικότερα, για ένα σύνολο ειδήσεων που εμπεριέχουν κείμενο, *S,* ο αλγόριθμος δημιουργεί έναν αραιό πίνακα ***A***, διαστάσεων *m επί n,* όπου *m* = ο συνολικός αριθμός των κειμένων και *n* = συνολικός αριθμός μοναδικά εμφανιζόμενων λέξεων εντός του *S.*

$$A = \begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ am1 & \cdots & amn \end{bmatrix}$$

Ο πίνακας αυτός είναι η αναπαράσταση διαφορετικών λέξεων που εμπεριέχονται στα επιμέρους κείμενα των ειδήσεων. Το στοιχείο ***aij*** καταδεικνύει τον συνολικό αριθμό των φορών που εμφανίζεται η j-στη λέξη στο i-οστό κείμενο. Ο μετασχηματισμός των αραιών αυτών πινάκων σε πυκνότερους γίνεται αφότου αναγνωρίσουμε πως αρκετές από τις λέξεις του συνολικού λεξιλογίου απουσιάζουν από έναν μεγάλο αριθμό κειμένων και για αυτό το λόγο τα αντίστοιχα στοιχεία του πίνακα είναι μηδενικά. Ενδεικτικά η μετατροπή σε πυκνή αναπαράσταση είναι η κάτωθι:

Value = [a_{11}	…	a_{1n}	…	a_{m1}	…	a_{mn}]
COL_INDEX = [0	…	0	…	m-1	…	m-1]^T
ROW_INDEX = [0	…	n-1	…	0	…	n-1]^T

2.2.1.2 Συχνότητα Εμφάνισης Όρου – Αντίστροφη Συχνότητα Κειμένου Term Frequency – Inverse Document Frequency (TF-IDF)

Αν και η TF-IDF μέθοδος θεωρείται από τις παλαιότερες στον κλάδο της ανάλυσης περιεχομένου και Επεξεργασίας Φυσικής Γλώσσας (*Natural Language Processing – NLP*), δεν παύει να είναι ένας απλός και αποδοτικός τρόπος στην εκπαίδευση των αλγορίθμων μηχανικής μάθησης για την αναγνώριση ψευδών ειδήσεων. Το σύνολο των υπολογισμών του αλγορίθμου έχει ως απώτερο στόχο τον υπολογισμού του γινομένου δύο βασικών μεγεθών, της συχνότητας εμφάνισης όρου (*term frequency - TF*) και τις ανάστροφης συχνότητας κειμένου (*inverse document frequency - IDF*).

Η συχνότητα εμφάνισης όρου *tf (t,d)* υπολογίζει την αναλογία της εμφάνισης του κάθε όρου *t ϵ V (d)* στο κείμενο *d*.

Το λεξιλόγιο $V\ (d) = \sum_t n(t,d)$ κατασκευάζεται από το κείμενο *d.* Κατά παρόμοιο τρόπο, εάν μια λέξη *w'* δεν ανήκει σε ένα κείμενο *d'*, η συχνότητα εμφάνισης όρου *tf (t', d')* θα είναι 0. Κατά παρόμοιο τρόπο θα μπορούσε να γραφεί ως:

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

Όπου *fr (x, t) = 1, if x = t ή fr(x, t) =0* σε διαφορετική περίπτωση. Η ιδέα της συχνότητας εμφάνισης όρου είναι ουσιαστικά η ίδια με την περίπτωση των αθροιστών λέξεων της προηγούμενης μεθόδου. Καταλήγοντας, ορίζεται η συχνότητα εμφάνισης όρου ως

$$tf(t,d) = \frac{n(t,d)}{V(d)}$$

, όπου N(t, d) το πλήθος εμφάνισης της λέξης t στο έγγραφο t.

Σε ένα δεδομένο σύνολο κειμένων *D,* η αντίστροφη συχνότητα κειμένου (inverse Document Frequency) *idf(t, D),* είναι ο λογάριθμος του αριθμού ων κειμένων του συνόλου διαιρεμένος με τον όρο *df(t, D)*, που υποδηλώνει τον αριθμό των κειμένων $d \epsilon D$ που εμπεριέχουν τον όρο *t.* Σαν αποτέλεσμα αυτού, συνήθεις λέξεις στο *D,* θα έχουν χαμηλή συχνότητα, ενώ λιγότερο συνήθεις θα έχουν υψηλότερη συχνότητα. Έτσι η συχνότητα εμφάνισης όρου θα είναι πολύ πιο πιθανό να διακρίνει κείμενα ψευδών ειδήσεων που συνήθως θα έχουν λιγότερο κοινότυπες λέξεις από τις πραγματικά αληθινές ειδήσεις οι οποίες συνήθως αποτελούνται από συχνότερα χρησιμοποιούμενο λεξιλόγιο.

$$idf(t, D) = \log(\frac{N}{df(t, D)})$$

Ή πιο αναλυτικά

$$idf(t, D) = \log(\frac{N}{df(t,D)})$$

Συνοπτικά, το TF-IDF σκορ *w(t, d)* για μια λέξη αυξάνεται με το πλήθος εμφάνισής της, αλλά μειώνεται εάν η λέξη εμφανίζεται σε πολλά από τα κείμενα του συνόλου κειμένων.

$$w(t, d) = tf(t, d) \times idf(t, D)$$

Ομοίως με την προηγούμενη μέθοδο αθροιστών λέξεων, οι παραγόμενοι πίνακες αυτής της μεθόδου είναι αραιοί και για την ευκολία διαχείρισής τους, τους μετασχηματίζουμε σε απεικονίσεις πυκνών πινάκων.

2.2.1.3 Μέθοδοι αναπαράστασης σε διανυσματικό χώρο - Word Embedding

2.2.1.3.1 Διανύσματα αναπαράστασης λέξεων – Μέθοδοι επίλυσης πινάκων

Η συνολική ιδέα είναι η αναπαράσταση του συνολικού λεξιλογίου μιας γλώσσας (π.χ. Αγγλικής) σε έναν N-διάστατο χώρο όπου N << αριθμός συνολικών λέξεων ο οποίος αναπαριστά ικανοποιητικά τις εννοιολογικές σχέσεις των λέξεων συνολικά. Κάθε μια από τις διαστάσεις θα κωδικοποιούσε μιας μορφής νόημα που

εκφέρουμε και στον προφορικό λόγο. Για παράδειγμα μια από τις διαστάσεις θα μπορούσε να αναφέρεται στον χρόνο που τοποθετείται κάθε λέξη (παρελθόν, παρόν, μέλλον), στον αριθμό (ενικός ή πληθυντικός) ή/και στο γένος (αρσενικό, θηλυκό, ουδέτερο). Μιας τέτοιας μορφής αναπαράσταση, θα μπορούσε να είναι η παρακάτω, σύμφωνα με τους (Manning, και συν., 2019)

$$\text{waardvark} = \begin{bmatrix}1\\0\\0\\\vdots\\0\end{bmatrix}, \text{w}a = \begin{bmatrix}0\\1\\0\\\vdots\\0\end{bmatrix}, \text{w}at = \begin{bmatrix}0\\0\\1\\\vdots\\0\end{bmatrix}, \dots, \text{w}zebra = \begin{bmatrix}0\\0\\0\\\vdots\\1\end{bmatrix}$$

καθώς ο αριθμός των λέξεων είναι πολύ μεγάλος, η τελική απεικόνιση και δημιουργία διανυσμάτων ανά λέξη, θα μπορούσε να γίνει μόνο με την εννοιολογική συσχέτισή τους μέσα από προτάσεις. Χρησιμοποιώντας τεράστια σύνολα δεδομένων (όπως για παράδειγμα ΟΛΑ τα λήμματα της Wikipedia) με αρχικές επαναλήψεις από πρόταση σε πρόταση θα μπορούσε να κατασκευαστούν πίνακες ταυτόχρονης εμφάνισης (co-occurrence) λέξεων σε μιας μορφής πίνακα, όπως φαίνεται παρακάτω.

$$X = \begin{array}{c} \\ I \\ like \\ enjoy \\ deep \\ learning \\ NLP \\ flying \\ . \end{array} \begin{array}{c} \begin{array}{cccccccc} I & like & enjoy & deep & learning & NLP & flying & . \end{array} \\ \begin{bmatrix} 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix} \end{array}$$

Εάν στον προηγούμενο πίνακα εφαρμόζαμε μεθόδους ανάλυσής του σε ιδιάζουσες τιμές (Singular value Decomposition – SVD), για να πάρουμε μια μορφής $X = USV^T$ ανάλυση, θα μπορούσαμε να χρησιμοποιήσουμε το στοιχείο U ως τα διανύσματα λέξεων για ολόκληρο τον πληθυσμό λέξεων/λεξιλόγιο (V) της γλώσσας που εξετάζουμε.

Με τη χρήση των μέτρων διασποράς

$$\frac{\sum_{i=1}^{k} \sigma i}{\sum_{i=1}^{|V|} \sigma j}$$

Θα μπορούσαμε να γράψουμε:

$$\begin{array}{cc} & |V| \\ |V| & \left[\begin{array}{c} \\ X \\ \\ \end{array}\right] \end{array} = \begin{array}{cc} & |V| \\ |V| & \left[\begin{array}{ccc} | & | & \\ u_1 & u_2 & \cdots \\ | & | & \end{array}\right] \end{array} \begin{array}{cc} & |V| \\ |V| & \left[\begin{array}{ccc} \sigma_1 & 0 & \cdots \\ 0 & \sigma_2 & \cdots \\ \vdots & \vdots & \ddots \end{array}\right] \end{array} \begin{array}{cc} & |V| \\ |V| & \left[\begin{array}{ccc} - & v_1 & - \\ - & v_2 & - \\ & \vdots & \end{array}\right] \end{array}$$

και προσπαθώντας να μειώσουμε της διαστάσεις, επιλέγοντας τα k μοναδιαία διανύσματα:

$$\begin{array}{cc} & |V| \\ |V| & \left[\begin{array}{c} \\ \hat{X} \\ \\ \end{array}\right] \end{array} = \begin{array}{cc} & k \\ |V| & \left[\begin{array}{ccc} | & | & \\ u_1 & u_2 & \cdots \\ | & | & \end{array}\right] \end{array} \begin{array}{cc} & k \\ k & \left[\begin{array}{ccc} \sigma_1 & 0 & \cdots \\ 0 & \sigma_2 & \cdots \\ \vdots & \vdots & \ddots \end{array}\right] \end{array} \begin{array}{cc} & |V| \\ k & \left[\begin{array}{ccc} - & v_1 & - \\ - & v_2 & - \\ & \vdots & \end{array}\right] \end{array}$$

Κατά αυτό τον τρόπο, θα μπορούσαμε να εξαγάγουμε συνολικά τα μοναδιαία ανά λέξη διανύσματα θυσιάζοντας όμως εξαιρετικά υψηλούς υπολογιστικούς πόρους καθώς ο πίνακας θα ήταν εξαιρετικά πολλών διαστάσεων (για την Αγγλική γλώσσα της τάξης ~10^6 x 10^6.

2.2.1.3.2 Επαναληπτικές Μέθοδοι – Word2Vec

Εναλλακτική της ανωτέρω μεθόδου, είναι η δημιουργία ενός μοντέλου το οποίο θα εκπαιδεύεται όχι σε στατικά συνολικά δεδομένα (τα οποία ενδεχομένως να αποτελούνται από δισεκατομμύρια προτάσεων), αλλά θα ακολουθεί μια επαναληπτική διαδικασία μάθησης η οποία εν τέλει θα αποθηκεύει την πιθανότητα εμφάνισης μιας λέξης δεδομένου του «γειτονικού» νοήματος ανάλογα με της πρόταση στην οποία τοποθετείται. Η αρχική ιδέα μια τέτοιας προσπάθειας μοντελοποίησης είναι ο σχεδιασμός ενός εργαλείου του οποίου οι παράμετροι είναι τα διανύσματα λέξεων και η μετέπειτα εκπαίδευσή του προς ένα συγκεκριμένο στόχο. Σε κάθε επανάληψη, αφού υπολογίσουμε τις αποκλίσεις από έναν τέτοιο στόχο, ακολουθώντας ένα συγκεκριμένο κανόνα επικαιροποίησης των στοιχείων που θα επιδρούσε στις παραμέτρους του μοντέλου θα μπορούσαμε να βαθμονομήσουμε τα διανύσματα που οδηγούν στις αποκλίσεις. Κατά αυτό τον τρόπο, σε κάθε επαναληπτικό βήμα τα διανύσματα

λέξεων θα επικαιροποιούνταν κατάλληλα σε κάθε βήμα. Η μέθοδος της αξιολόγησης και βαθμονόμησης είναι η γνωστή μέθοδος της όπισθεν ανατροφοδότησης (back propagation).

Μέσω απλών νευρωνικών δικτύων, που ουσιαστικό στόχο έχουν την πρόβλεψη της επόμενης λέξης που ακολουθεί ένα δοθέν σύνολο λέξεων συγκεκριμένου νοήματος, δημιουργείται ένας διανυσματικός χώρος $N/\ \mathbb{R}^{|V|X1}$ διαστάσεων στον οποίο αναπαρίστανται όλες οι λέξεις με ένα συγκεκριμένο διάνυσμα που ανήκει στο χώρο αυτό. Το κάθε διάνυσμα αναπαριστά το ειδικό βάρος της κάθε σύνδεσης από τις N του στρώματος εισόδου (input layer) με κάθε ένα από τα κρυμμένα στρώματα υπολογισμού του νευρωνικού δικτύου. Η πληροφορία αυτή ουσιαστικά είναι η κωδικοποίηση της εννοιολογικής σημασίας της συγκεκριμένης λέξης με το σώμα (συλλογή κειμένων) με το οποίο εκπαιδεύεται το εκάστοτε μοντέλο αναπαράστασης.

Δύο βασικοί αλγόριθμοι, βάσει της σύγχρονης βιβλιογραφίας προσεγγίζουν το ανωτέρω πρόβλημα μέσω αντίστοιχων υπολογισμών:

Αλγόριθμος συνεχούς σύνολο λέξεων (Continuous Bag of Words - CBOW) - Στόχος του συγκεκριμένου αλγορίθμου είναι η πρόβλεψη μιας κεντρικής λέξης από το νόημα του συνόλου των λέξεων που την περιβάλλουν.

Αναλυτικότερα, ο αλγόριθμος ακολουθεί τα παρακάτω βήματα:

	Αλγόριθμος: Continuous Bag of Words
Αρχικοί Ορισμοί	Δημιουργούμε 2 πίνακες , $V \in \mathbb{R}^{n\times|V|}$ και $U \in \mathbb{R}^{|V|\times n}$, όπου n η διάσταση του διανυσματικού χώρου στον οποίο θέλουμε να αναπαραστήσουμε τα διανύσματα λέξεων
1:	Δημιουργούμε τα διανύσματα λέξεων για την εισαγωγή των δεδομένων, μεγέθους m, όπου $m : (x^{(c-m)}, \ldots, x^{(c-1)}, x^{(c+1)}, \ldots, x^{(c+m)} \in \mathbb{R}^{/V/})$.
2:	Υπολογίζουμε τα διανύσματα λέξεων για το περιεχόμενο $u_{(c-m)} = Vx^{(c-m)}, u_{(c-m+1)} = Vx^{(c-m+1)}, \ldots, u_{(c+m)} = Vx^{(c+m)}, \in \mathbb{R}^n$
3:	Υπολογίζοντας το διάνυσμα μέσου όρου: $avg(u) = (u_{(c-m)} + u_{(c-m+1)} + \ldots + u_{(c+m)}) /(2 *m), \in \mathbb{R}^n$

4:	Υπολογισμός και δημιουργία διανύσματος $z = U \cdot \text{avg}(u)$, $\epsilon\ \mathbb{R}^{\lvert V\rvert}$ Καθώς το εσωτερικό γινόμενο παρόμοιων διανυσμάτων είναι μεγαλύτερο, οι αντίστοιχες λέξεις που είναι σε μικρότερη απόσταση μεταξύ τους, παράγουν μεγαλύτερα σκορ στο z
5:	Με τη χρήση της softmax ως τελεστή δημιουργούμε ένα αντίστοιχο διάνυσμα πιθανοτήτων μέσω του z $\text{avg}(y) = \text{softmax}(z)\ \epsilon\ \mathbb{R}^{\lvert V\rvert}$
6.	Ο στόχος είναι η δημιουργία πιθανοτήτων τέτοιων ώστε το avg(y) να ταυτίζεται με το $y\ \epsilon\ \mathbb{R}^{\lvert V\rvert}$, διάνυσμα το οποίο εν τέλει αναπαριστά την επιθυμητή τελική λέξη.
Επεξηγήσεις	*V:* Ο πίνακας λέξεων που εισάγονται στο μοντέλο, τέτοιος ώστε η *ι-οστή* στήλη του είναι το v_i*:{1xn}* διαστάσεων διάνυσμα για τη λέξη w_i όταν αυτή εισάγεται στο μοντέλο *U*: Είναι ο πίνακας λέξης εξόδου (αποτελέσματος), του οποίου η *j-οστή* σειρά είναι το διάνυσμα u_j*:{1xn}* της λέξης w_j όπου αυτή θεωρείται το αποτέλεσμα υπολογισμού του μοντέλου Για κάθε λέξη w_i, παράγονται 2 διανύσματα , το v_i και το u_j Ο τελεστής softmax μετασχηματίζει ένα διάνυσμα σε ένα άλλο του οποίου με τις εξής ιδιότητες: $softmax(avg(y)) = \frac{(e^{avg(y)i})}{\sum_{k=1}^{\lvert V\rvert}(e^{avg(y)k})}$ Διαιρώντας με $\sum_{k=1}^{\lvert V\rvert}(e^{avg(y)k})$ Κανονικόποιείται το διάνυσμα $\sum_{k=1}^{n}(avg(y)k) = 1$

Με τη χρήση του ανωτέρω μοντέλου, αρκεί να κατασκευαστεί μια αντικειμενική συνάρτηση, που στην προκειμένη περίπτωση, καθώς θέλουμε να μάθουμε την απόσταση μιας υπολογιζόμενης πιθανότητας από την πραγματική πιθανότητα αναζητούμε μια μετρική τη απόστασης μεταξύ των δύο κατανομών. Χρησιμοποιώντας μία από τις πιο διαδεδομένες μετρικές απόστασης/απώλειας, της cross – entropy H(avg(y), y).

Για την ανωτέρω προσέγγιση αυτή απλοποιείται ως εξής:

$$H(avg(y), y) = -\sum_{k=1}^{\lvert V\rvert}(yj * \log(avg(y)j)))$$

Και καθώς το διάνυσμα y είναι μονοδιάστατο, η ανωτέρω απλοποιείται στην

$$H(avg(y), y) = -yi * \log(avg(y)j))$$

Έτσι, η τελική συνολική αντικειμενική συνάρτηση γράφεται:

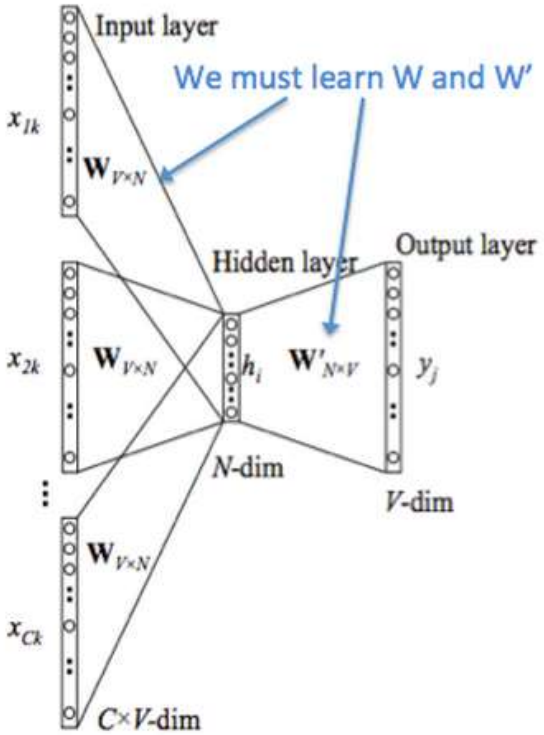
ελαχιστοποίηση J $= -\log P(w_c/w_{c-m}, \dots, w_{c-1}, w_{c+1}, \dots, w_{c+m})$

$= -\log P(u_c/\theta)$

$= -\log(\exp(u_c^T\theta)/\text{sum}(\exp(u_j^T\theta)), j \in [1,|V|]$

$= -u_c^T\theta + \text{sum}(\exp(u_j^T\theta)), j \in [1,|V|]$

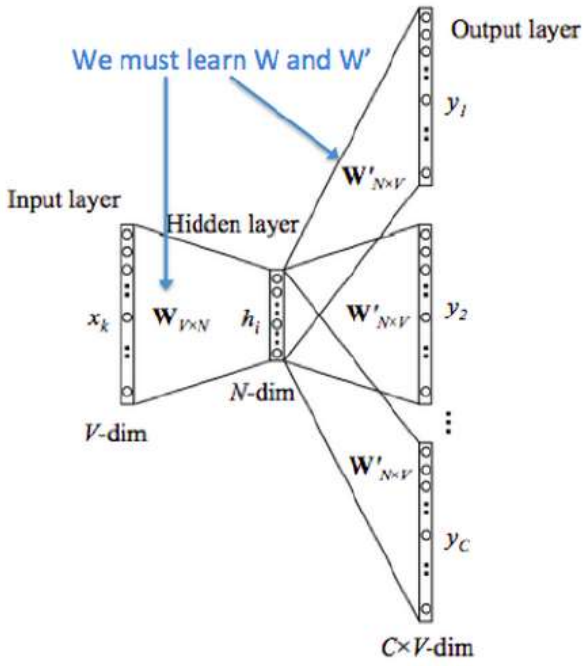
Τέλος με τη χρήση της μεθόδου SGD (Stochastic Gradient Descent), το μοντέλο καλείται να ενημερώσει όλα τα διανύσματα λέξεων u_c και v_j .



Εικόνα 4 - Απεικόνιση του τρόπου εκτέλεσης του αλγορίθμου CBOW και της εκπαίδευσης των πινάκων μεταφοράς, (Manning, et al., 2019)

Αλγόριθμος Skip-Gram – Στόχος του είναι η πρόβλεψη της κατανομής (πιθανότητας) του νοήματος των γειτονικών γύρω από μια δοθείσα λέξη.

	Αλγόριθμος: Skip-Gram
1:	Δημιουργείται ένας , $x \in \mathbb{R}^{\|V\|}$ που αναπαριστά το διάνυσμα της κεντρικής λέξης που εισάγεται.
2:	Υπολογίζεται το εμφωλευμένο διάνυσμα λέξης για την κεντρική λέξη $v_c = V \cdot x \in \mathbb{R}^n$
3:	Δημιουργείται το διάνυσμα βαθμολόγησης $z = U \cdot v_c \in \mathbb{R}^{n \times \|V\|}$
4:	Μετασχηματίζεται το διάνυσμα βαθμολόγησης μέσω του τελεστή softmax σε διάνυσμα πιθανοτήτων *avg(y) = softmax(z)*. Σημειώνεται πως τα $avg(y)_{c-m}$, $avg(y)_{c-1}$, $avg(y)_{c+1}$, …, $avg(y)_{c+m}$ είναι οι πιθανότητες του γειτονικού νοήματος της κάθε λέξης.
5:	Τελικώς είναι επιθυμητή η δημιουργία κατάλληλου διανύσματος πιθανοτήτων *avg(y)* ώστε να ταυτίζεται με τις πραγματικές πιθανότητες $y^{(c-m)}$, …, $y^{(c-1)}$, $y^{(c+1)}$, …, $y^{(c+m)}$, του διανύσματος εξόδου.
Επεξηγήσεις	$V \in \mathbb{R}^{n \times \|V\|}$: Ο πίνακας λέξεων που εισάγονται στο μοντέλο, τέτοιος ώστε η *ι-οστή* στήλη του είναι το v_i *:{1xn}* διαστάσεων διάνυσμα για τη λέξη w_i όταν αυτή εισάγεται στο μοντέλο $U \in \mathbb{R}^{n \times \|V\|}$: Είναι ο πίνακας λέξης εξόδου (αποτελέσματος), του οποίου η *j-οστή* σειρά είναι το διάνυσμα u_j*:{1xn}* της λέξης w_j όπου αυτή θεωρείται το αποτέλεσμα υπολογισμού του μοντέλου
ελαχιστοποίηση J	$= -\log P(w_{c-m}, \dots, w_{c-1}, w_{c+1}, \dots, w_{c+m} \| w_c)$ $= -\log(Product(P(w_{c-m+j} \| w_c))), j \in [0,2m], j \neq m$ $= -\log(Product(P(u_{c-m+j} \| v_c))), j \in [0,2m], j \neq m$ $= -\log(Product(\exp(u_{c-m+j}^T \| v_c)/sum(\exp(u_k^T v_c)))), j \in [0,2m], j \neq m, k \in [1,\|V\|]$ $= -sum(u_{c-m+j}^T \| v_c) + +2m\log(sum(\exp(u_k^T v_c)), j \in [0,2m], j \neq m, k \in [1,\|V\|]$



Εικόνα 5 - Γραφική αναπαράσταση της λειτουργίας του αλγορίθμου Skip-gram και της εκπαίδευσης πινάκων μεταφοράς (Manning, και συν., 2019).

Κατ' αντιστοιχία με τον ανωτέρω αλγόριθμο, σε κάθε βήμα χρησιμοποιείται η μέθοδος SGD για την ελαχιστοποίηση της αντικειμενικής συνάρτησης.

2.2.1.3.3 Επαναληπτικές μέθοδοι - Η μέθοδος GloVe (Global Vectors for Word Representation)

Σε αντίθεση με την αυστηρά περιορισμένη στο παράθυρο ορισμού της (μέσω συγκεκριμένων συνδυασμών και ακολουθιών λέξεων) μέθοδο Word2Vec, η μέθοδος GloVe αποτελείται στον πυρήνα της από ένα μοντέλο σταθμισμένων ελάχιστων τετραγώνων, που εκπαιδεύεται σε συνολικές εμφανίσεις ζευγών λέξη προς λέξη και κάνει αποδοτική χρήση της στατιστικής. Το μοντέλο παράγει ένα διάνυσμα-λέξη πλούσιο σε νόημα και πληροφορία για τη δομή από την οποία δημιουργήθηκε. Δίνει, σύμφωνα με τη βιβλιογραφία (Manning, et al., 2019), την πιο ακριβή συσχέτιση μεταξύ των λέξεων και είναι επαρκώς αποδοτικότερη από αντίστοιχες μεθόδους αναπαράστασης λέξεων στο διανυσματικό χώρο.

Αν *X* είναι ο πίνακας που υποδηλώνει το πλήθος των λέξεων μεταξύ κειμένων, όπου το στοιχείο *Xij* καταδεικνύει τον αριθμό των εμφανίσεων της λέξης j στο κείμενο I, ορίζεται το $X_i = \Sigma_k X_{ik}$ ως ο πλήθος των εμφανίσεων της κάθε λέξης k σε σχέση με τη λέξη i. Τέλος, ορίζεται η πιθανότητα *Pij = P(wj|wi) = Xij / Xi* ως η πιθανότητα της j να εμφανίζεται σε σχέση με την λέξη i.

Για την πλήρωση ενός τέτοιου πίνακα απαιτείται μόνο μια εμπρόσθια (forward pass) εκτέλεση από το σύνολο των προτάσεων για τη συλλογή στοιχείων.

Με τη χρήση της softmax, δημιουργείται ο αντίστοιχος πίνακας πιθανοτήτων από τον P_{ij}, Qij ως παρακάτω:

$$Q_{ij} = \frac{(\exp(u_j{}^T v_i)}{\sum_{w=1}^{W} \exp(u_w{}^T v_i)}$$

Και η αντίστοιχη αντικειμενική συνάρτηση για τη συγκεκριμένη μέθοδο διαμορφώνεται ως εξής:

$$J = \sum_{i\epsilon\ corpus} \sum_{j\epsilon\ context(i)} \log Q_{ij}$$

Και απλοποιείται τελικώς ως εξής:

$$J = sum(sum(f(X_{ij})(u_j{}^T v_i - \log X_{ij})^2)),\ \ I,j \in [1,W]$$

Στις ακόλουθες αναλύσεις και ανάπτυξη εργαλείου για τη δοκιμή με το ανοιχτό σύνολο δεδομένων, χρησιμοποιήθηκαν τα αποτελέσματα και η τελική χαρτογράφηση των λέξεων σε διανυσματικό χώρο 200 διαστάσεων, όπως αυτά προέκυψαν από τον αλγόριθμο GloVe. Περαιτέρω ανάλυση αλλά και παρουσίαση δείγματος των διανυσμάτων χαρτογράφησης ακολουθεί στην ενότητα της Ανάπτυξης του εργαλείου σε λογισμικό – Προετοιμασία Συνόλου Δεδομένων.

2.2.2 Μοντέλα ταξινόμησης Μηχανικής Μάθησης

2.2.2.1 Λογιστική Παλινδρόμηση – Logistic Regression (LR)

Πρόκειται για μια από τις πιο συνήθεις μεθόδους μηχανικής μάθησης που χρησιμοποιείται για την πρόβλεψη της τιμής της μεταβλητής $y \epsilon [0,1]$ ως δυαδικής ταξινόμησης. Η αρνητική κλάση υποδηλώνεται από το 0 ενώ η θετική κλάση από το 1. Έχοντας ω απώτερο στόχο την ταξινόμηση σε μια από τις 2 κατηγορίες 0 η 1, σχηματίζεται η υπόθεση $h(\theta) = \theta^T X$, με κατώφλι αποκοπής της συγκεκριμένης υπόθεσης να ενεργοποιείται όταν $h\theta(x) = 0.5$. Εάν η τιμή της υπόθεσης hθ(x) >= 0.5, τότε ο αλγόριθμος θα προβλέψει y=1, το οποίο και εάν εφαρμοστεί για την αναγνώριση ψευδών ειδήσεων, θα κατατάξει την εν λόγω είδηση ως αληθή, ενώ εάν η τιμή της hθ(x) < 0.5 τότε η είδηση χαρακτηρίζεται ως ψευδής.

Η πρόβλεψη της λογιστικής παλινδρόμησης γίνεται κάτω από την προϋπόθεση του $0 < h\theta(x) < 1$. Η σιγμοειδής συνάρτηση της λογιστικής παλινδρόμησης μπορεί να γραφτεί και ως κάτωθι:

$$h\theta(x) = g((\theta^T X))$$

όπου $g(z) = 1/(1+x^{-z})$ και $h\theta(x) = 1/(1+x^{-\theta TX})$ και η επακόλουθη συνάρτηση κόστους

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} \cos(h\theta(x^{(i)}),\ y^{(i)})$$

2.2.2.2 Δέντρα Αποφάσεων – Decision Trees (DT)

Τα δέντρα αποφάσεων είναι από τις σημαντικότερες και πιο διαδεδομένες μεθόδους μηχανικής μάθησης. Η βασική ιδέα των δέντρων απόφασης είναι η ανάπτυξη ενός μοντέλου που να προβλέπει την τιμή μιας εξαρτημένης μεταβλητής βασιζόμενο σε διάφορους κανόνες απόφασης που εξάγονται από το σύνολο των δεδομένων που το τροφοδοτούν. Οι αλγόριθμοι των δέντρων αποφάσεων έχουν αυστηρά ιεραρχική δομή και προσομοιάζουν σε σχήμα ένα δέντρο του οποίου οι κόμβοι μπορεί να είναι είτε μορφής φύλλου – ή ομάδας ταξινόμησης, είτε μορφής κόμβου απόφασης. Οι αλγόριθμοί DT χαρακτηρίζονται

από την ευκολία ερμηνείας τους από τους φυσικούς χρήστες, αναφορικά με τη διαδικασία της ταξινόμησης λόγω του σχήματος αλλά θεωρούνται αδύναμες μέθοδοι μάθησης, γεγονός που ενδεχομένως οδηγεί σε χαμηλές αποδόσεις προβλέψεις για μικρού πληθυσμού σύνολα δεδομένων.

Η βασική αρχή μάθησης της μεθόδου των δέντρων απόφασης είναι η επιλογή του κατάλληλου χαρακτηριστικού βάσει του οποίου θα γίνει η πρόβλεψη/ταξινόμηση. Για την επίλυση ενός τέτοιου προβλήματος, δημιουργούνται διαφορετικών μορφών δέντρα βάσει συγκεκριμένων μετρικών όπως τα information gain η και το gain ratio.

Ξεκινώντας με την υπόθεση ότι το διακριτό χαρακτηριστικό A έχει n διαφορετικές τιμές και D_i είναι το σύνολο το οποίο περιέχει όλα τα δείγματα τα οποία έχουν τιμή I στο σύνολο εκπαίδευσης D., ορίζονται τα gain ratio και information gain ως κάτωθι (Jiang, Haq, Saboor, & Ali, 2021):

$$Gain\ (A,\ D) = Entropy(D) - \sum_{i=1}^{n} \frac{|D_i|}{|D|} Entropy(Di)$$

$$GainRatio(A,\ D) = \frac{Gain(D, A)}{IV(A)}$$

$$IV(A) = Entropy(D) - \sum_{i=1}^{n} \frac{|D_i|}{|D|} log2 \frac{|D_i|}{|D|}$$

Μία συνοπτική απεικόνιση των βημάτων του αλγορίθμου για τα δέντρα απόφασης είναι η κάτωθι, όπου η **test_condition** υπορουτίνα εμπεριέχει τους ελέγχους της ανωτέρω Gain Μεθόδου.

	Αλγόριθμος Ταξινόμησης: Δέντρα Αποφάσεων
	Υπορουτίνα: **GenerateDecisionTrees**(Δείγματα s, Χαρακτηριστικά F)
1	Αν συνθήκες_τερματισμού(S,F) = TRUE τότε
	α. Leaf = **create_Node**()
	β. Ταξινόμηση Leaf = **Ταξινόμηση**(s)
	γ. **Return** Φύλλο
2	root = **create_Node**()
	root.**testcondition** = **find_bestSplit**(s,f)

	v = {v | v ένα πιθανό αποτέλεσμα της root.**testcondition}** Για κάθε τιμή v ϵ *V*: sv:= {s | root.**testcondition**(s)=v και s ϵ *S }* child = **Tree_Growth**(Sv, F) Ανάπτυξε το child ως παράγωγο της root και όρισέ το ως άκρο (root -> child) v **Return** root

2.2.2.3 k-Nearest Neighbour – kNN

Ο kNN είναι πασίγνωστος βιβλιογραφικά αλγόριθμος μηχανικής μάθησης, του οποίου οι διαδικασίες είναι πολύ απλές. Δοθέντος ενός δείγματος προς εξέταση, αρχικά βρίσκει τους k κοντινότερους γείτονες στο δείγμα εφαρμόζοντας ένα μέτρο απόστασης. Ακολούθως προβλέπει το χαρακτηρισμό της κλάσης του δείγματος βασιζόμενος στη στρατηγική κύρια ψήφου. Η απόδοση του αλγορίθμου kNN βασίζεται σε μεγάλο βαθμό στο πλήθος των διαστάσεων του προβλήματος και επίσης θεωρείται ένας αργός προς μάθηση αλγόριθμος, ξοδεύοντας αρκετό υπολογιστικό χρόνο για την τελική ταξινόμηση. Μια γενική απεικόνιση της διαδικασίας που ακολουθεί ο kNN είναι η κάτωθι:

	Αλγόριθμος: kNN
1:	Για όλα τα μη ταξινομημένα δεδομένα, του συνόλου, u
2:	Για όλα τα ταξινομημένα δεδομένα, του συνόλου, v
3:	Υπολογίζεται η απόσταση μεταξύ u και v
4:	Βρίσκονται οι κ μικρότερες αποστάσεις και εντοπίζονται οι αντίστοιχες ταμπέλες ταξινόμησης v1, …, vk
5:	Ανατίθενται τα μη ταξινομημένα δεδομένα u στην πιο συχνά εμφανιζόμενη κλάση από τις ήδη χρησιμοποιούμενες προς ταξινόμηση
6:	Τέλος επανάληψης βήματος 2
7:	Τέλος επανάληψης βήματος 1
8:	Τέλος αλγορίθμου

2.2.2.4 Αλγόριθμος Random Forest – RF

Οι αλγόριθμοι Random Forest είναι συλλογές αποτελούμενες από αλλεπάλληλα δέντρα αποφάσεων που έχουν προκύψει από τυχαία επιλογή χαρακτηριστικών σε κάθε επίπεδο τους. Κάθε μοναδιαίο δέντρο απόφασης στον random forest αλγόριθμο παράγει μια πρόβλεψη και συνολικά, βασιζόμενος σε ένα σύστημα

ψηφοφορίας, ο random forest παράγει την τελική του πρόβλεψη ταξινόμησης. Ο αλγόριθμος RF μπορεί να περιγραφεί ως εξής:

$$F(x) = argmax\{\sum_{i=1}^{z} T(A(B, \theta_k))\}$$

Όπου F(x) είναι το μοντέλο του αλγορίθμου, j είναι η επιθυμητή κατηγορική μεταβλητή και F η χαρακτηριστική συνάρτηση. Για να εξασφαλιστεί η ποικιλία (η τυχαιότητα δηλαδή με την οποία δημιουργούνται) των δέντρων απόφασης γίνεται δειγματοληπτική επιλογή μέρους των δέντρων απόφασης, βασισμένη σε τυχαίους αριθμούς.

2.2.2.5 Αλγόριθμος Support Vector Machine (SVM)

Για προβλήματα δυαδικής ή πολλαπλής ταξινόμησης, ο SVM είναι από τους πιο δημοφιλείς αλγορίθμους βάσει της βιβλιογραφίας. (Chen, Zhang, Trajevski, Zhong, & Zhang, 2019)

Τα στοιχεία διαχωρίζονται από ένα υπερεπίπεδο μέσω προβλήματος δυαδικής ταξινόμησης έτσι ώστε $w^T x + b = 0$, όπου w είναι ένας συντελεστής βαρύτητας του υπερεπιπέδου. Ο όρος b, αναπαριστά την απόκλιση (offset) των τιμών από την αρχή των αξόνων και τα σημεία των στοιχείων του συνόλου αναπαρίστανται από το x. Ο καθορισμός κατάλληλων τιμών για τα w και b είναι ο βασικός στόχος του SVM αλγορίθμου.

Στην απλή γραμμική περίπτωση, το w μπορεί να επιλυθεί χρησιμοποιώντας τη συνάρτηση Lagrange. Στο μέγιστο όριο, τα στοιχεία του συνόλου δεδομένων ονομάζονται support vectors. Σαν αποτέλεσμα, η λύση του w μπορεί να εκφραστεί μαθηματικά ως:

$$w = \sum_{i=1}^{n} a_i Y_i X_i$$

Όπου το n υποδηλώνει τα support vectors, και η επιθυμητή περιγραφή της κλάσης ταξινόμησης είναι Yi, η οποία αντιπροσωπεύει το δείγμα x. Ο όρος b μπορεί να υπολογιστεί από την yi($w^T x + b$) $-1 = 0$.

Για τις μη γραμμικές περιπτώσεις ο πυρήνας και η συνάρτηση απόφασης των n για τα w και b εκφράζονται μέσω της κάτωθι:

$$f(x) = sgn(\sum_{i=1}^{n} a_i y_i K(x_i, x) + b)$$

2.2.2.6 Αλγόριθμος Naïve-Bayes (NB)

Ο βασικός άξονας θεώρησης της μεθόδου είναι πως όλα τα χαρακτηριστικά (features) είναι μεταξύ τους ανεξάρτητα. Αν και, στις περιπτώσεις κειμένου όπου οι λέξεις πολύ σπάνια ακολουθούν μια τέτοια ισχυρή υπόθεση, ο αλγόριθμος φαίνεται να λειτουργεί εξαιρετικά αποδοτικά βάσει αυτής.

Πιο συγκεκριμένα, δοθέντος ενός στοιχείου της κλάσης y και ένα διάνυσμα χαρακτηριστικών $\boldsymbol{X} = (\chi 1, \ldots, \chi n)$, η πιθανότητα να ανήκει το X στην κλάση y, δίνεται από τον τύπο:

$$P(y|\mathbf{X}) = \frac{P(y) * P(\mathbf{X}|y)}{P(\mathbf{X})},$$

Και εξαιτίας της αρχικής υπόθεσης της γραμμικής ανεξαρτησίας των χαρακτηριστικών ισχύει:

$$P(x_i|y, x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) = P(x_i|y)$$

Χρησιμοποιώντας τον κανόνα του Bayes προκύπτει ότι

$$P(y|x_1, \ldots, x_n) = \frac{P(y) \prod_{i=1}^{n} P(x_i|y)}{P(x_1, \ldots, x_n)}$$

Και τέλος, καθώς $P(\chi 1, \ldots, \chi n)$ είναι σταθερό καταλήγουμε στον κανόνα ταξινόμησης:

$$\hat{y} = \underset{y}{argmax} P(y) \prod_{i=1}^{n} P(x_i|y)$$

2.3 Μέθοδοι ανάλυσης συμπεριφοράς χρήστη– Social Context based/ Network methods

Οι μέθοδοι ανάλυσης συμπεριφοράς χρήστη, σε αντίθεση με τις προηγουμένως αναφερθείσες μεθόδους ανάλυσης περιεχομένου, εστιάζουν κυρίως σε χαρακτηριστικά που εξάγονται από αλληλεπιδράσεις μεταξύ ανθρώπων-χρηστών μιας πλατφόρμας. Διάφορες ερευνητικές απόπειρες περιλαμβάνουν την αξιολόγηση της πιστότητας των δημοσιοποιούμενων ειδήσεων από προφίλ χρηστών κοινωνικών δικτύων αλλά και πλατφορμών ενημέρωσης που βασίζονται σε:

- Μοντελοποίηση της συμπεριφοράς των χρηστών: Το ιστορικό συμπεριφοράς του κάθε χρήστη είναι βασισμένο στο είδος και τον όγκο των ειδήσεων που ακολουθεί και επικροτεί ή απορρίπτει (Tacchini, Ballarin, Della Vedova, Moret, & de Alfaro, 2017), (Shu, Cui, Wang, Lee, & Liu, 2019).
- Χρονικά μοτίβα τα οποία συγκρίνουν την απόκριση του κάθε χρήστη σε ειδήσεις που σχετίζονται με την επικαιρότητα (Ma, Gao, Wei, Lu, & Wong, 2015)
- Μοντέλα διάδοσης αξιοπιστίας τα οποία εξάγουν υποστηρικτικές ή μη γνώμες μέσω διασυνδεδεμένων πλατφορμών. (Jin, Cao, Jiang, & Zhang, 2014)
- Αξιολόγηση της αυθεντικότητας των ειδήσεων, που βασίζεται στην αξιολόγηση της αξιοπιστίας εκείνων που τη δημοσιοποιούν και στις θέσεις των αναγνωστών πάνω στο θέμα της είδησης.

Συγκρινόμενη με τις μεθόδους ανάλυσης περιεχομένου, αυτή η ομάδα μεθόδων είναι δυνατόν να εξαγάγει πιο αποδοτικά χαρακτηριστικά εξαιτίας της ολοκληρωμένης προσέγγισης που βασίζεται σε γνωρίσματα των χρηστών αλλά και στις πληροφορίες που προκύπτουν από τη διάδραση μεταξύ αυτών (Jin, Cao, Jiang, & Zhang, 2014). Μέσω συστημάτων βαθμολόγησης επίδοσης των χρηστών με βάση τη συμβατότητα της ιδιότητας τους σχετικά με τη δημοσιοποιούμενη είδηση αλλά και με βάση την εγγύτητα χωρική ή χρονική σε σχέση με την είδηση, αξιολογείται συνολικά η αξιοπιστία του κάθε χρήστη και των δημοσιοποιούμενων, από τον ίδιο, ειδήσεων. Εγγενής, παρόλα αυτά, περιορισμός

της μεθόδου είναι η εξειδίκευση τέτοιου είδους βαθμολόγησης στην εκάστοτε πλατφόρμα την οποία χρησιμοποιεί ο κάθε χρήστης για τη δημοσιοποίηση ειδήσεων.

Μια επίσης κοινώς αποδεκτή πρακτική εφαρμογή τέτοιων μεθόδων βασίζεται στην αναγνώριση των μοτίβων διάδοσης των ψευδών ειδήσεων μέσω χρήσης δικτύων ή διαγραμμάτων δέντρων. Συνολικά οι μέθοδοι αυτές χαρακτηρίζονται από υψηλή δυνατότητα ερμηνείας των αποτελεσμάτων τους από ανθρώπινους χρήστες (σε αντίθεση με την κατηγορία των βαθιάς – deep learning – μηχανικής μάθησης μεθόδων). Ως αρνητικό στοιχείο τους, οι μέθοδοι αυτές είναι δύσκολες στην εφαρμογή καθώς βασίζονται στη χαρτογράφηση ολόκληρου του δικτύου μετάδοσης των ειδήσεων.

2.4 Μέθοδοι βασισμένες σε βαθιά Μηχανική Μάθηση - Deep Learning Based

Συνδυασμός των ανωτέρω δύο κατηγοριών είναι η κατηγορία των μεθόδων βαθιάς μηχανικής μάθησης στην οποία υπάγονται εφαρμογές που με τη χρήση Περιπεπλεγμένων Νευρωνικών Δικτύων (Convolutional Neural Networks), Επαναλαμβανόμενων Νευρωνικών Δικτύων (Recurrent Neural Networks) και άλλων αλγορίθμων (Π.χ. Gated Recurrent Unit - GRU), συμπεριλαμβάνουν στοιχεία και των προηγούμενων 2 κατηγοριών στην ανάλυση αξιοπιστίας της κάθε είδησης, όπως για παράδειγμα χαρακτηριστικά που εξάγονται από το περιεχόμενο της είδησης, χαρακτηριστικά που καταδεικνύουν τη συμπεριφορά των χρηστών οι οποίοι αλληλοεπιδρούν με την είδηση αλλά και συστήματα που αξιολογούν την εγκυρότητα της πηγής η οποία δημοσιεύει την είδηση). (Ma, Gao, Wei, Lu, & Wong, 2015)

Ειδικότερα, οι μέθοδοι βαθιάς μηχανικής μάθησης στοχεύουν στην αυτόματη δημιουργία και εξαγωγή αφηρημένων απεικονίσεων των χαρακτηριστικών του περιεχομένου των ειδήσεων (Guo, Ding, Yao, Liang, & Yu, 2020). Οι κυριότερες κατηγορίες τέτοιων μεθόδων είναι οι εξής:

- **Περιπεπλεγμένα Νευρωνικά Δίκτυα – Convoluted Neural Networks (CNNs)**

Πρόκειται περί νευρωνικών δικτύων πολλαπλών φίλτρων που περιπλέκονται με τα στρώματα εισόδου (input layers) για να δημιουργήσουν πλούσιους σε πληροφορία και μεγάλους σε διαστάσεις χάρτες χαρακτηριστικών (feature maps), οι οποίοι μετασχηματίζονται σε χαμηλότερων διαστάσεων πίνακες ώστε να επιταχυνθεί η διαδικασία μάθησης. Μέσω επαναλαμβανόμενων εκτελέσεων τα CNNs, μπορούν να αναγνωρίσουν τόσο τοπικά όσο και ολικά χαρακτηριστικά των εισόδων τόσο σε επίπεδο μεμονωμένων λέξεων όσο και φράσεων. (Jin, Cao, Jiang, & Zhang, 2014), (Wang, Wang, Yang, & Lian, 2021),

- **Γράφοι (Graphs) Περιπεπλεγμένων Νευρωνικών Δικτύων (GCNs)**
  Σύνθετη μορφή της ανωτέρω κατηγορίας ΝΔ, στην οποία εισάγονται δεδομένα se μορφή γράφων/γραφημάτων.

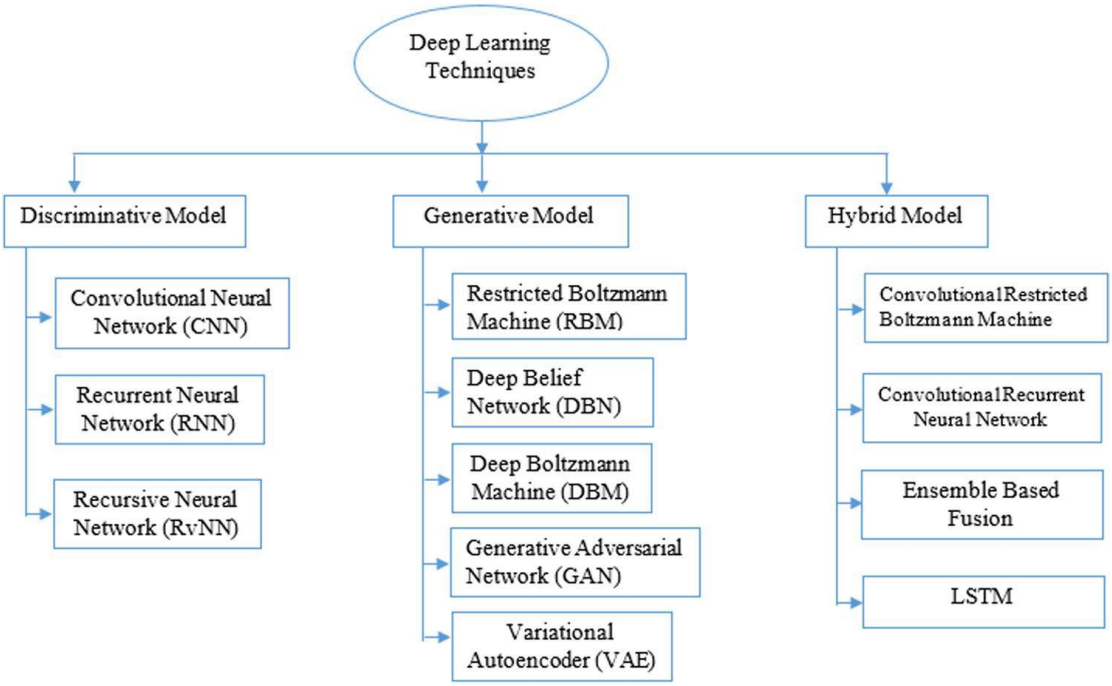
- **Επαναλαμβανόμενα Νευρωνικά Δίκτυα - Recurrent Neural Networks (RNNs)**
  Τα RNNs μπορούν να διαχειριστούν σειριακά δεδομένα, γεγονός που μειώνει τους απαιτούμενους υπολογιστικούς πόρους, καθώς η πληροφορία διανέμεται μεταξύ των νευρώνων του ίδιου κρυμμένου στρώματος. Η εισαγωγή επιπλέον μαθηματικών μεθόδων ενίσχυσης της «μακράς διάρκειας μνήμης» όπως οι LSTM (Long short-term memory) και GRU (Gated Recurrent Unit), ενισχύουν την απόδοση τέτοιων μεθόδων σε εφαρμογές ανάλυσης φυσικής γλώσσας (Natural Language Processing)

- **Παλινδρομικά Νευρωνικά Δίκτυα – Recursive Neural Networks (RcNNs)**
  Ομοίως με τα προαναφερθέντα RNNs αποσυνθέτουν το συνολικό κείμενο στα επιμέρους δομικά στοιχεία και δευτερογενώς αξιολογούν δομές (δέντρα) σύνταξης

- **Αυτόματοι Κωδικοποιητές – Automatic Encoders**
  Πρόκειται για μοντέλα μη επιβλεπόμενης μάθησης τα οποία περιλαμβάνουν στάδια κωδικοποίησης των εισόδων σε μορφές γνωστών

κατανομών (π.χ. gaussian), ανάλυσής τους σε ενδιάμεσα στάδια και τελικής αποκωδικοποίησης τους ώστε να μετατραπούν ξανά σε δεδομένα όμοια με τα αρχικά.

- **Generative Adversarial Networks (GANs)**
- **Μηχανισμοί προσοχής**



Εικόνα 6 -Ταξινόμηση μεθόδων βαθιάς μηχανικής μάθησης (Islam, Liu, Wang, & Xu, 2020)

Work	Method type	Detection model	Model input			
			News content	Propagation	User Response	User/Website profiles
Castillo et al. [98]	Content-based	J48 Decision Tree etc.	√			√
Potthast et al. [99]		Meta-learning Approach	√			
Hu et al. [100]		Matrix Factorization	√	√		
Qazvinian et al. [101]		Rumor Retrieval, Belief Classification	√	√		
Kwon et al. [102]		Decision Tree, Random Forest, SVM	√	√		
Horne et al. [103]		SVM etc.	√			
Tacchini et al. [104]	Social Context-based	Logistic Regression, Boolean Crowdsourcing Algorithms		√	√	
Ma et al. [105]		Dynamic Series-Time Structure Model	√	√		√
Jin et al. [106]		Credibility Propagation Network	√		√	
Yang et al. [107]		Probabilistic Graphical Model	√		√	√
Gupta et al. [108]		Credibility Propagation Network	√	√		
Jin et al. [109]		Hierarchical Content Network	√	√		
Shu et al. [110]		——				√
Wu et al. [111]		Hybrid SVM	√	√		√
Jin et al. [112]		Epidemiological Model		√		
Liu et al. [113]		Information Propagation Model		√		√
Kim et al. [114]		Homogeneity-Based Transmissive Process		√		√
Ma et al. [115]		Propagation Tree via Kernel Learning		√		
Yu et al. [116]	Deep learning-based	CNN	√		√	
Ma et al. [117]		RNN	√		√	
Li et al. [118]		GRU	√		√	
Liu et al. [119]		CNN+GRU		√		
Runchansky et al. [120]		RNN	√		√	√
Jin et al. [121]		RNN	√		√	
Liu et al. [122]		Attention	√		√	
Guo et al. [123]		LSTM+Attention	√		√	√
Popat et al. [124]		LSTM+Attention	√			√

Εικόνα 7 - Επισκόπηση των μεθόδων για την αναγνώριση ψευδών ειδήσεων (Guo, και συν., 2021)

Συνολικά, οι μέχρι στιγμής βιβλιογραφικές συγκεντρωτικές μελέτες (Guo, Ding, Yao, Liang, & Yu, 2020), (Guo, και συν., 2021) καταδεικνύουν πως οι συνδυαστικές μέθοδοι παρουσιάζουν υψηλότερα ποσοστά επιτυχημένης αναγνώρισης ψευδών ειδήσεων σε δικτυακές πλατφόρμες από ότι οι πρώτες 2 ομάδες μεθόδων, συγκρινόμενες σε ομάδες ανοιχτών δεδομένων. Παρά το γεγονός ότι οι μέθοδοι αυτές θεωρούνται οι πιο αποτελεσματικές, υστερούν στην ευκολία εφαρμογής τους.

Κεφάλαιο 3 - Εφαρμογή αλγορίθμων μηχανικής μάθησης για την αναγνώριση ψευδών ειδήσεων

3.1 Αξιολόγηση και επιλογή μεθόδου ταξινόμησης

Για το επιλεγμένο σύνολο ειδήσεων, θα χρησιμοποιηθούν μέθοδοι ανάλυσης περιεχομένου που περιεγράφηκαν αναλυτικά στα προηγούμενα κεφάλαια. Πιο συγκεκριμένα, το σύνολο των ειδήσεων θα περάσει από λεκτική επεξεργασία μετασχηματίζοντας τις ακολουθίες χαρακτήρων τους σε διανύσματα προς ποσοτική ανάλυση.

Η διαδικασία εξαγωγής τέτοιων διανυσμάτων θα είναι 3 διαφορετικών ειδών και πιο συγκεκριμένα θα ακολουθηθεί αρχικά η μέθοδος Bag Of Words, δημιουργώντας δυαδικές ομάδες λέξεων οι οποίες και για τροφοδοτήσουν μοντέλα μηχανικής μάθησης με σκοπό την εκπαίδευσή τους. Τα μοντέλα μηχανικής μάθησης που θα τροφοδοτηθούν από τα χαρακτηριστικά που θα προκύψουν από τη μέθοδο Bag Of Words, θα είναι μοντέλο Λογιστικής Παλινδρόμησης (LR), μοντέλο ταξινόμησης Naïve-Bayes (NB), Support vector Machine ταξινομητή (SVM), Support Vector machine με SGD μέθοδο και τέλος αλγόριθμο Random Forest.

Πέραν της ανωτέρω μεθόδου προεπεξεργασίας των κειμένων, θα εφαρμοστεί και η μέθοδος υπολογισμού των χαρακτηριστικών TF-IDF, τα οποία με τη σειρά τους θα τροφοδοτήσουν αντίστοιχα τα ίδια μοντέλα μηχανικής μάθησης με αυτά που θα χρησιμοποιηθούν για τη μέθοδο Bag of Words.

Τέλος, χρησιμοποιώντας το σύνολο διανυσμάτων λέξεων της μεθόδου GloVe, θα γίνει μια τρίτη προεπεξεργασία των κειμένων του συνόλου των ειδήσεων προς εξέταση ώστε εν τέλει να τροφοδοτήσουν ένα επαναληπτικό νευρωνικό Δίκτυο για την ταξινόμησή τους σε αληθείς ή ψευδείς χρησιμοποιώντας τη μέθοδο LSTM (Long Short-Term Memory).

Η τελική ποσοτική σύγκριση των τριών διαφορετικών διαδικασιών προεπεξεργασίας των κειμένων των ειδήσεων θα βασιστεί στα εξής μέτρα:

$$Accuracy\ (Acc) = \frac{TP + TN}{TP + TN + FP + FN} * 100\%$$

$$\text{Recall (Re)} = \frac{TP}{TP + FN} * 100\%$$

$$\text{Precision (Pre)} = \frac{TN}{TN + FP} * 100\%$$

$$\text{f1} - \text{score} = 2 * \frac{(precision) * (recall)}{(precision) + (recall)} * 100\%$$

Σαν επιπλέον μεταβλητές επίδοσης θα παρουσιαστούν η πρόοδος εκπαίδευσης όλων των μεθόδων σε σχέση με την αλλαγή στην προβλεπτική δυνατότητα της κάθε μιας, όπως επίσης και η προοδευτική εξέλιξη

3.2 Επιλογή πλατφόρμας εφαρμογής

Οι ασκήσεις εφαρμογής των επιλεγμένων μεθόδων πραγματοποιήθηκαν μέσω της γλώσσας προγραμματισμού Python 3.7 και όλες οι υπολογιστικές δοκιμές που έγιναν καθώς και οι μέθοδοι που εξετάστηκαν αποτελούν μέρος διαφορετικών βιβλιοθηκών μεθόδων, οι οποίες αναπτύσσονται και χρησιμοποιούνται κατά βάση από χρήστες για τις πειραματικές ανάγκες επίλυσής προβλημάτων μηχανικής και βαθιάς μηχανικής μάθησης. Ενδεικτικά, οι σημαντικότερες βιβλιοθήκες εξ' αυτών είναι οι TensorFlow, Keras και Sklearn. Επίσης χρησιμοποιήθηκαν διάφορα περιβάλλοντα εργασίας με τα πιο σημαντικά εξ αυτών το Jupyter Notebook, το το οποίο και επιτρέπει την ταυτόχρονη εκτέλεση του κώδικα μαζί με την αξιολόγηση των αποτελεσμάτων όπως επίσης και το περιβάλλον Spyder, το οποίο και δίνει ευκολότερη πρόσβαση σε όλο το πλήθος των διαθέσιμων μεταβλητών που χρησιμοποιήθηκαν κατά την εκτέλεση του κώδικα.

3.3 Εύρεση και επιλογή βάσης δεδομένων

Οι ερευνητικές προσπάθειες αναφορικά με την αναγνώριση ψευδών ειδήσεων έχουν οδηγήσει στην ανάπτυξη μεγάλου αριθμού συνόλου δεδομένων κατάλληλων για αντίστοιχες εφαρμογές.

Οι (D'Ulizia, Cascherra, Ferri, & Grifoni, 2021), κατέγραψαν και ταξινόμησαν τις 27 διαθέσιμες, μεγαλύτερες βάσεις ανοιχτών δεδομένων βάσει διάφορων χαρακτηριστικών όπως για παράδειγμα η γλώσσα των ειδήσεων και η πλατφόρμα δημοσιοποίησής τους. Το μεγαλύτερο ποσοστό αντίστοιχων δημοσιεύσεων χρησιμοποιεί σύνολα δεδομένων ειδήσεων στην Αγγλική γλώσσα τα οποία κυρίως εμπεριέχουν ειδήσεις που δημοσιοποιούνται στα ψηφιακά μέσα κοινωνικής δικτύωσης. Καθώς ο σκοπός της παρούσας εργασίας είναι η παρουσίαση των συγκριτικών αποτελεσμάτων εφαρμογής μεθόδων μηχανικής μάθησης, το φάσμα των ειδήσεων που επιλέχθηκε είναι από το φάσμα των κοινωνικών και πολιτικών ειδήσεων και ως επί το πλείστο από τα συμβατικά Μέσα Μαζικής ενημέρωσης (ειδησεογραφικά πρακτορεία, διεθνείς οίκοι ειδήσεων, γραφεία τύπου κλπ). Επιπλέον χαρακτηριστικά του συνόλου δεδομένων που αναζητήθηκε για την παρούσα εργασία είναι ο διαχωρισμός των ειδήσεων σε αληθείς και ψευδείς, η γλώσσα του περιεχομένου τους να είναι η αγγλική, να έχουν το χαρακτήρα των ελεύθερων προς χρήση δεδομένων και τέλος να αποτελούνται τουλάχιστον από μια σύντομη περίληψη όπως επίσης και να συνοδεύονται από το πλήρους έκτασης γραπτό περιεχόμενό τους, όπως αυτό δημοσιεύθηκε από την αντίστοιχη πηγή.

Επιθυμητά στοιχεία του συνόλου δεδομένων είναι επίσης η χρονική διασπορά τους, ώστε να εξασφαλίζεται η ποικιλία στη θεματολογία όχι μόνον λόγω των διαφορετικών μέσων αλλά και λόγω διαφορετικών χρονικών συγκυριών κατά τις οποίες οι ειδήσεις αυτές δημοσιοποιήθηκαν.

Στο Παράρτημα, είναι διαθέσιμη η συγκεντρωτική λίστα των (D'Ulizia, Cascherra, Ferri, & Grifoni, 2021), από την οποία επιλέχθηκαν τα δεδομένα που χρησιμοποιήθηκαν στην εφαρμογή.

3.4 Περιγραφή Συνόλου Δεδομένων

Όπως παρατέθηκαν ανωτέρω, τα συγκεκριμένα επιθυμητά χαρακτηριστικά του συνόλου δεδομένων προς εξέταση, καλύπτονταν από περισσότερα του ενός ανοιχτά σύνολα δεδομένων, τα οποία και ήταν διαφορετικά μεταξύ τους, αναφορικά με την διάρθρωσή τους. Για της ανάγκες της εφαρμογής επιλέχθηκε το σύνολο δεδομένων για την τεκμηρίωση ψευδών ειδήσεων FNID (Fake News Inference Dataset), του ερευνητικού οργανισμού IEEE (Sadeghi, Bidgoly, & Amirkhani, 2020) που έχει σχεδιαστεί για την αποθήκευση δεδομένων δυο εκ των μεγαλύτερων διεθνών πλατφορμών/αποθετηρίων που φιλοξενούν ψευδείς ειδήσεις, των LIAR και FakeNewsNet (Politifact). Διατίθεται προς ανάγνωση και χρήση κάτω από την ομπρέλα των Creative Commons 4.0. Και οι δύο πλατφόρμες ψευδών ειδήσεων οι οποίες αποτελούν και την μήτρα του FNID έχουν χρησιμοποιηθεί εκτενώς από τη διεθνή βιβλιογραφία (Bajaj, 2017), (Girgis, Amer, & Gadallah, 2020), (Katarya & Massoudi, 2020) και έχουν κριθεί ως αξιόπιστες αναφορικά με την ταξινόμηση των ειδήσεων μέσω διαφόρων καναλιών ενημέρωσης (ΜΜΕ, μέσα κοινωνικής δικτύωσης, ειδησεογραφικά δίκτυα κ.α.) που εμπεριέχουν, μέσα από διασταύρωση πληροφοριών και ανάλυση των πολιτικών γεγονότων από ομάδες ειδικών επιστημόνων. Η κύρια θεματική εστίαση των δύο πλατφορμών ανοικτών δεδομένων είναι οι ειδήσεις πολιτικού και κοινωνικού περιεχομένου που διανέμονται από τα μεγαλύτερα ειδησεογραφικά πρακτορεία παγκοσμίως. Σαν περαιτέρω επεξεργασία των εμπεριεχόμενων ειδήσεων, έχουν αφαιρεθεί τα σημεία εντός του κυρίως κειμένου της κάθε είδησης που τελικά λήφθηκαν υπόψη από τους εμπειρογνώμονες για την τελική ταξινόμηση της κάθε είδησης.

Το σύνολο δεδομένων για την τεκμηρίωση ψευδών ειδήσεων (FNID) που χρησιμοποιήθηκε στην εφαρμογή αλγορίθμων μηχανικής μάθησης για τους σκοπούς της παρούσας εργασίας αποτελείται συγκεκριμένα από 15212 δείγματα ειδήσεων προς εκπαίδευση των αλγορίθμων, 1058 δείγματα (το 6.1% του συνολικού πληθυσμού) για την επαλήθευση τη εκπαίδευσης των αλγορίθμων και τέλος 1054 (6.1% του συνολικού πληθυσμού) δείγματα για τον έλεγχο προβλέψεων των αλγορίθμων. Το σύνολο των δεδομένων του FNID εκτείνεται χρονικά σε μεγάλο διάστημα περίπου 7 ετών από το 2013 όταν και

πραγματοποιήθηκε η αρχική ιδέα δημιουργίας τέτοιων συνόλων ανοιχτών δεδομένων έως και τα μέσα του 2020.

Οι στήλες που εμπεριέχουν τη συνολική πληροφορία ανά είδηση στο FNID είναι οι εξής:

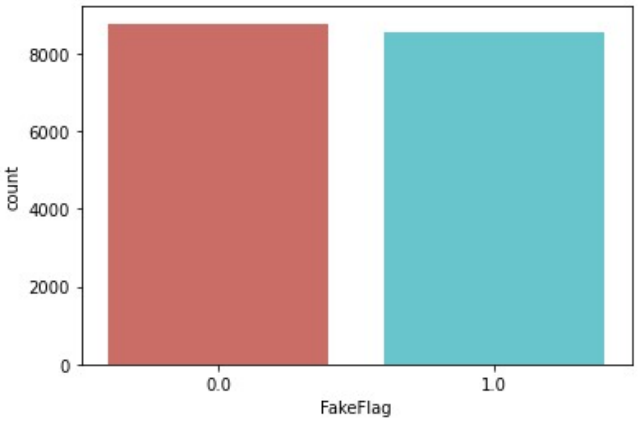
- ID: Ο κωδικός αναγνώρισης είδησης από το χρησιμοποιηθέν σύνολο δεδομένων (LIAR ή PolitiFact/FakeNewsNet αντίστοιχα)
- Date: Η ημερομηνία, κατά την οποία η συγκεκριμένη είδηση συμπεριλήφθηκε στο αντίστοιχο ανοιχτό σύνολο δεδομένων.
- Speaker: Όνομα προσώπου ή οργανισμού στο οποίο αναφέρεται η δήλωση
- Statement: Η περιληπτική περιγραφή της δήλωσης που δημοσιοποιήθηκε στα ΜΜΕ από ένα πρόσωπο ή οργανισμό και ερευνήθηκε από την ομάδα διαχείρισης.
- Sources: Η λίστα με τις πηγές στις οποίες βασίστηκαν οι ομάδες των ειδικών αναλυτών για να αντλήσουν πληροφορίες ώστε να αξιολογήσουν την αυθεντικότητα της εκάστοτε είδησης/ δήλωσης.
- Paragraph_based_content: Το περιεχόμενο της είδησης αποθηκευμένο όπως παρατέθηκε, σε μορφή λίστας.
- fullText_based_content: Παρατίθεται το πλήρες περιεχόμενο της είδησης, χωρίς επιπλέον επεξεργασία.
- Fake_fnn: Η τελική ταξινόμηση σε αληθείς ή ψευδείς ειδήσεις, όπως αυτή καταχωρήθηκε από την ομάδα ειδικών της εκάστοτε βάσης δεδομένων.

Παρακάτω ακολουθεί ενδεικτικό δείγμα εξαγωγής από τη βάση δεδομένων που χρησιμοποιήθηκε.

Index	id	date	speaker	statement	sources	paragraph_based_content	fullText_based_content	label_fnn
0	1636	2010-03-28T17:45:34-04:00	Charlie Crist	Rubio's tax swap proposal "…	['http://blogs.tampaba…	['Gov. Charlie Crist…	The renters A family of two with an ann…	fake
1	4352	2011-08-29T06:00:00-04:00	Bobby Scott	"The estimated savings of t…	['http://www.bobbyscot…	['U.S. Rep. Bobby Sc…	Norman Ornstein, a resident… The Bush tax cuts were enac…	real
2	16471	2019-02-12T17:35:38-05:00	Wisconsin Republican Legislative leaders	Foxconn has already "made a…	['https://www.wispolit…	["Amid reports quest…	When asked for backup to th… The letter said Foxconn "cr…	fake
3	1557	2010-03-05T18:24:02-05:00	Dave Aronberg	Says Gov. Charlie Crist has called him "a rock star."	['http://www.davearonb…	["State Sen. Dave Ar…	"To be honest, about anythi… Thanks to a tip from a…	real
4	12826	2016-07-29T18:09:31-04:00	Jeannette Vaught	"Only five Texas counties a…	['http://www.mystatesm…	['From the citrus of…	We sought to check on the d… To our inquiry, Michael Kla…	fake
5	5073	2012-01-06T13:29:58-05:00	Barack Obama	President Barack Obama "put…	['http://www.youtube.c…	['With all eyes on I…	Still, Obama's push for the… "There has to be some kind …	real
6	8954	2014-02-21T15:28:54-05:00	Dan Branch	"Worked with Ronald Reagan and helped re-elect him."	['https://docs.google.…	['Texas attorney gen…	We asked Branch's campaign … Asked for more detail, Marg…	real
7	6766	2012-09-25T11:52:41-04:00	Secure America Now	Says Obama adviser David Pl…	['http://www.youtube.c…	['There's the guilt-…	But he told us he's not an … The Post reported that a Tr…	real
8	11892	2016-02-11T14:21:04-05:00	Seminole Tribe of Florida	The Seminole gambling compa…	['https://youtu.be/6tr…	['The Seminole Tribe…	The Legislature is consider… Is that $3 billion in guara…	real
9	11287	2015-09-21T18:00:00-04:00	Viral image	Says Bernie Sanders opposes…	['http://www.motherjon…	['As the Democratic …	It is obvious that in the n… Another Mother Jones articl…	fake
10	6625	2012-09-04T22:45:17-04:00	Deval Patrick	"By the time (Mitt Romney) …	['http://www.cbsnews.c…	['Repeating a favori…	So the number is correct. But does Romney's scrutiny…	real

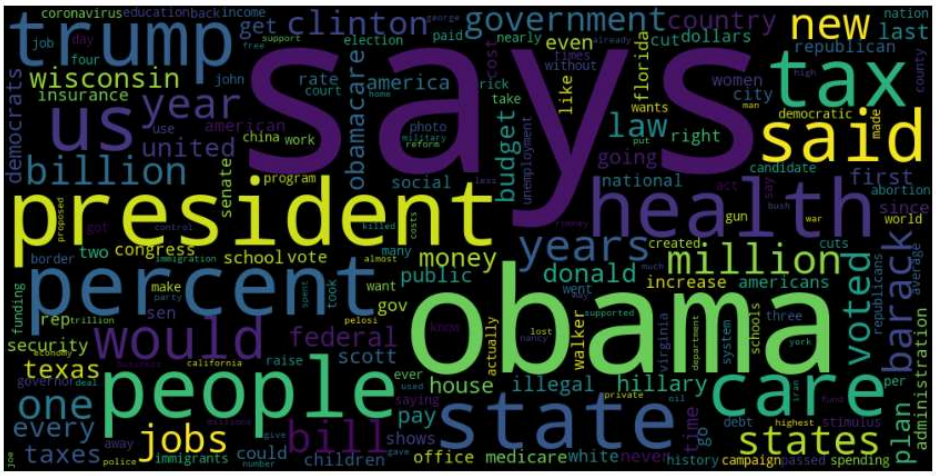
Εικόνα 8 - Δείγμα εξαγωγής του ανοιχτού συνόλου δεδομένων Fake News Inference Dataset (Sadeghi, Bidgoly, & Amirkhani, 2020)

Μια αρχική απεικόνιση της κατανομής της ταξινόμησης των ειδήσεων τού συνολικού πληθυσμού που χρησιμοποιήθηκε είναι η παρακάτω, στην οποία και παρατηρούμε σχεδόν ισόποση περιεκτικότητα σε αληθείς όσο και σε ψευδείς ειδήσεις, γεγονός που βοηθά στην επικείμενη εκπαίδευση των αλγορίθμων μηχανικής μάθησης, καθώς παρέχει και στις 2 πιθανές ταξινομήσεις σχεδόν ίσο αριθμό δεδομένων.



Εικόνα 9 - Κατανομή κλάσης αξιοπιστίας των ειδήσεων του FNID [0: TRUE, 1: FAKE]

Επιπλέον περιγραφικά στατιστικά του πληθυσμού είναι το συνολικά χρησιμοποιούμενο λεξιλόγιο της στήλης “Statement”, το οποίο και αποτελείται από περίπου 16000 διαφορετικές λέξεις, οι σημαντικότερες και πιο συχνά εμφανιζόμενες εκ των οποίων παρουσιάζονται στο παρακάτω γραφικό.



Εικόνα 10 - Γραφικό των συχνότερα εμφανιζόμενων λέξεων στο πεδίο "Statement" του FNID. Το μέγεθος εξαρτάται από τη συχνότητα εμφάνισης της εκάστοτε λέξης

3.5 Ανάπτυξη εργαλείου ταξινόμησης

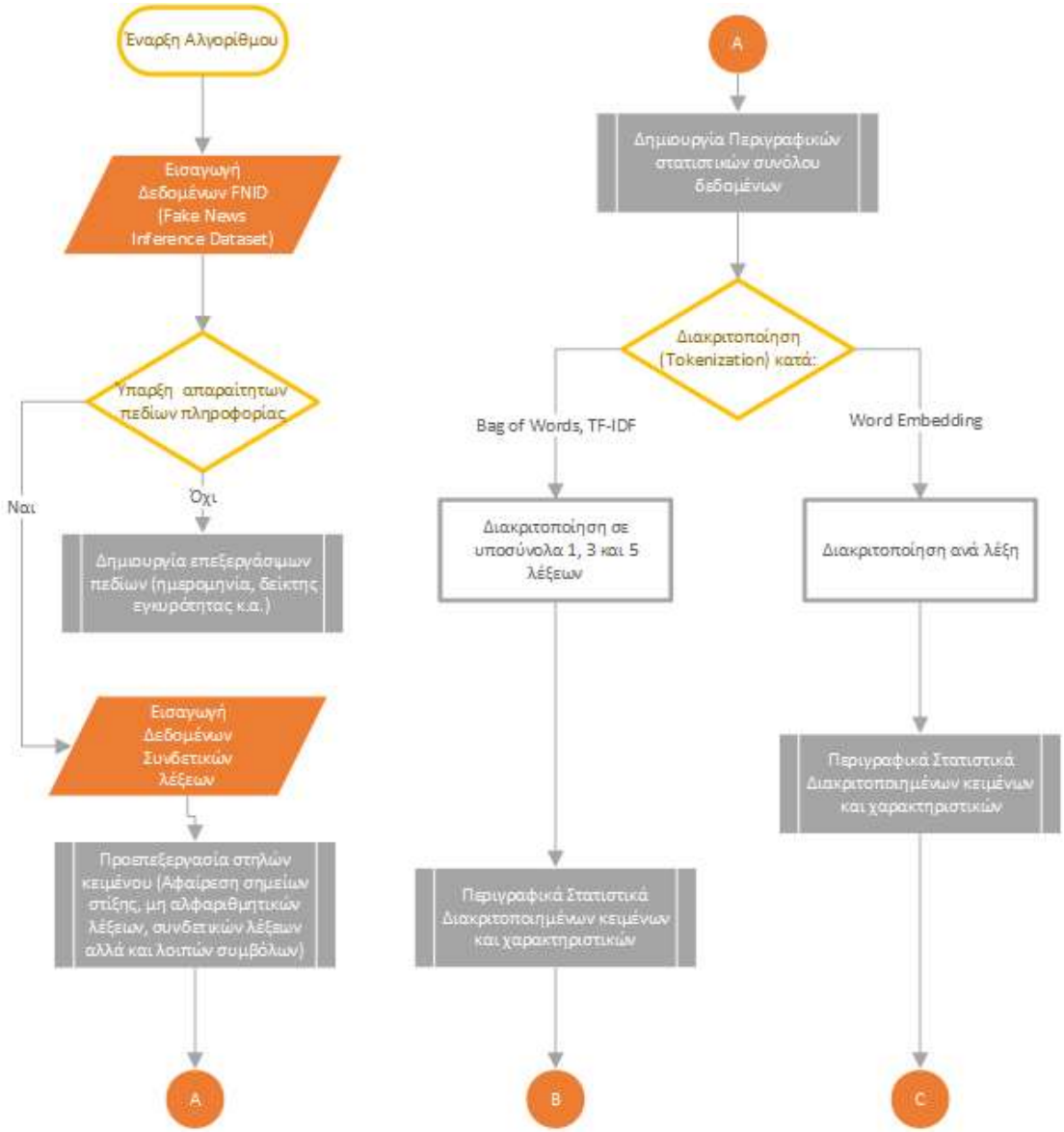
Η συνολική διαδικασία που ακολουθήθηκε μπορεί να περιγραφεί σε τρία διακριτά μεταξύ τους στάδια. Αρχικά περιλάμβανε την προετοιμασία του συνόλου των δεδομένων με αρχική αναγνώριση των πεδίων πληροφορίας από τα οποία αυτά συνοδεύονταν. Σαν επόμενο βήμα, έγινε η διακριτοποίηση (tokenization) των στοιχείων κειμένου (λέξεις, φράσεις, προτάσεις, n-grams) της κάθε είδησης και η δημιουργία ποσοτικής (διανυσματικής η βαθμωτής) αναπαράστασης για κάθε μια από τις ανωτέρω στρατηγικές διακριτοποίησης των στοιχείων του κάθε κειμένου. Τέλος, αξιοποιώντας τις ποσοτικές πλέον αναπαραστάσεις των κειμένων του συνόλου προς εξέταση, χρησιμοποιήθηκαν αλγόριθμοι και μοντέλα μηχανικής μάθησης για την ταξινόμηση υποσυνόλου των διαθέσιμων ειδήσεων σε ψευδείς ή αληθείς. Η διαδικασία της ταξινόμησης υποσυνόλου ειδήσεων, έγινε μετά και από την αρχική εκπαίδευση όλων αυτών των χρησιμοποιούμενων μοντέλων μηχανικής μάθησης από ένα διαφορετικού μεγέθους υποσύνολο, με στόχο την αρχική βαθμονόμησή τους.

Μετά και την αρχική ταξινόμηση της κάθε είδησης του υποσυνόλου δοκιμάστηκαν διαφορετικές προσεγγίσεις αναφορικά με την εκπαίδευση των μοντέλων αλλά και με την αρχική διακριτοποίηση των κειμένων.

3.5.1 Προετοιμασία συνόλου δεδομένων και διακριτοποίηση

Μετά την εισαγωγή των δεδομένων του ανοιχτού συνόλου FNID, πραγματοποιήθηκαν έλεγχοι αναφορικά με την εγκυρότητα και πληρότητα της περιεχόμενης πληροφορίας. Τα απαραίτητα για την άσκηση πεδία ήταν η περίληψη, το πλήρες κείμενο και η ταξινόμηση της είδησης ως ψευδής ή αληθής. Επιπλέον πεδία του συνόλου δεδομένων εμπεριέχουν το μέσο δημοσιοποίησης της είδησης, την ημερομηνία δημοσίευσης, το πρόσωπο ή οργανισμό στον οποίο πιστώνεται η εκάστοτε δήλωση της κάθε είδησης. Στόχος της προετοιμασίας των δεδομένων είναι επίσης η επεξεργασία του κειμένου με τέτοιο τρόπο ώστε να εισαχθεί σε μεθόδους διακριτοποίησης (tokenization techniques) με απώτερο σκοπό τη δημιουργία και εξαγωγή χαρακτηριστικών κειμένου (text/content features).

Το ακόλουθο διάγραμμα απεικονίζει τη διαδικασία εκτέλεσης των βημάτων του αλγορίθμου για τα προαναφερόμενα στάδια.



Εικόνα 11 - Διαδικασία Επεξεργασίας δεδομένων και διακριτοποίησης

Η δημιουργία και τελική επιλογή των χαρακτηριστικών που εξάγονται από το κείμενο της κάθε είδησης είναι η διαδικασία που θα καθορίσει την είσοδο της πληροφορίας στα μοντέλα μηχανικής μάθησης. Η ορθή διαμόρφωση των κειμένων προς διακριτοποίηση απαιτεί την απομάκρυνση των συνδετικών λέξεων, των λέξεων που δεν αποτελούν μέρος του λεξιλογίου της αγγλικής γλώσσας αλλά και των αριθμών ή συμβόλων.

```python
def CleanTokenize(df):
    head_lines = list()
    lines = df["statement"].values.tolist()

    for line in lines:
        line = clean_text(line)
        # Διακριτοποίηση κειμένου
        tokens = word_tokenize(line)
        # Αφαίρεση σημείων στίξης
        table = str.maketrans('', '', string.punctuation)
        stripped = [w.translate(table) for w in tokens]
        # Αφαίρεση όχι αλφαριθμητικών χαρακτήρων
        words = [word for word in stripped if word.isalpha()]
        stop_words = set(stopwords.words("english"))
        # Αφαίρεση "stopwords" - Κοινών συνδετικών λέξεων
        words = [w for w in words if not w in stop_words]
        head_lines.append(words)
    return head_lines

head_lines = CleanTokenize(data)
head_lines[0:10]
```

Εικόνα 12 - Δείγμα python script για καθαρισμό και διακριτοποίηση του κειμένου των ειδήσεων

Μετά τον τελικό καθαρισμό των κειμένων από τα ανωτέρω, πραγματοποιείται η διακριτοποίηση των κειμένων σε ngrams 1, 3 και 5 λέξεων, υπολογισμός TF-IDF, για κάθε έναν τύπο από αυτά τα n-grams. Για τη διακριτοποίηση ngram(1) ακολουθεί η αποτύπωση μέσω GloVE της κάθε λέξης στον αντίστοιχο διανυσματικό χώρο και τελικά ο συνολικός υπολογισμός του διανύσματος του συνολικά εξεταζόμενου κειμένου είτε 30 λέξεων (περίληψης) είτε του συνολικού 600 λέξεων.

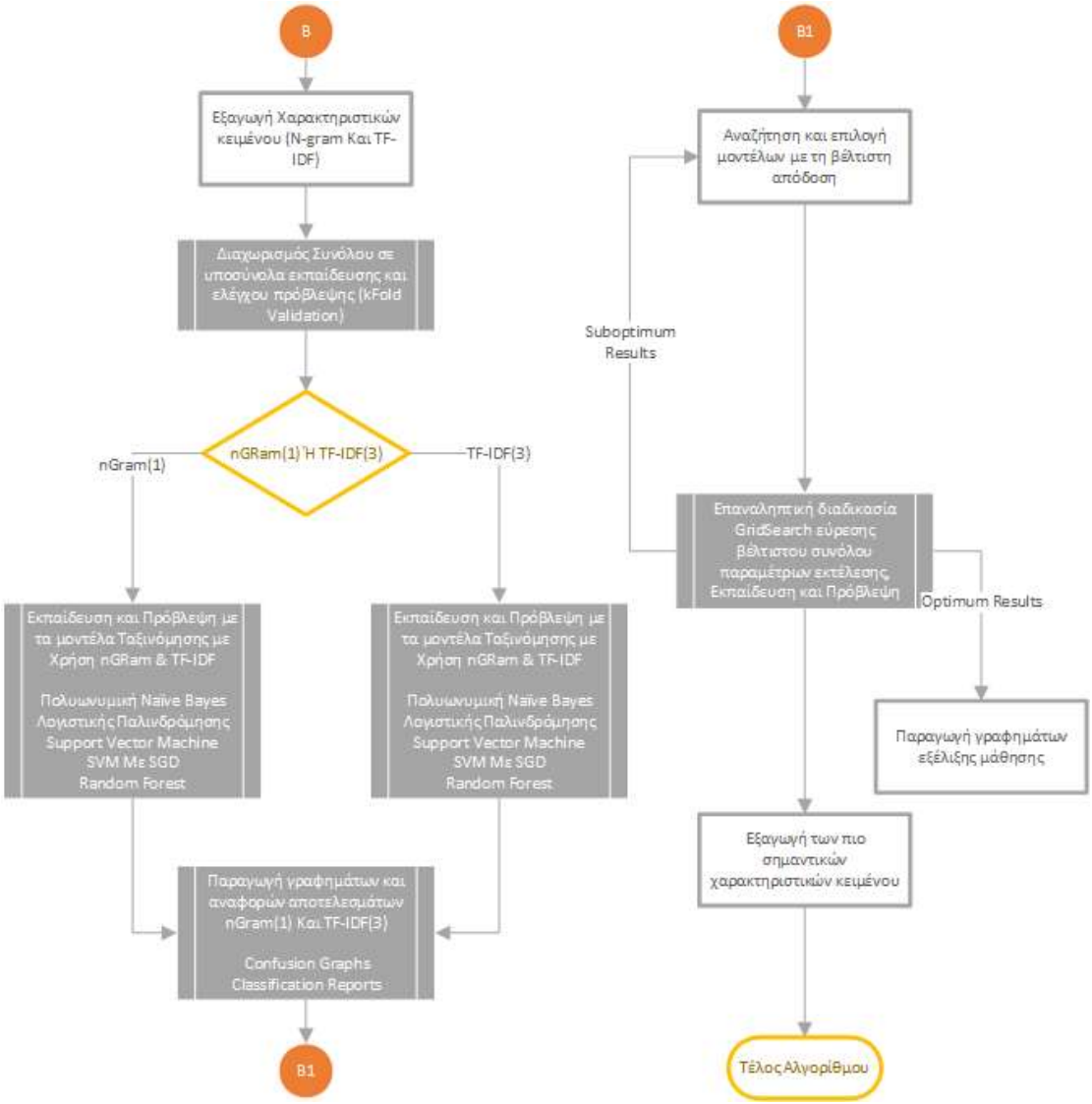
3.5.2 Επιλογή Χαρακτηριστικών

Το συνολικό πλήθος των αντίστοιχων n-grams, οι συχνότητες TF-IDF για κάθε ένα από αυτά και τέλος το συνολικό διάνυσμα αποτύπωσης του κειμένου, επιλέγονται ως χαρακτηριστικά και τροφοδοτούνται στα μοντέλα μηχανικής μάθησης για την αρχική εκπαίδευση και τέλος την απόπειρα ταξινόμησης ειδήσεων.

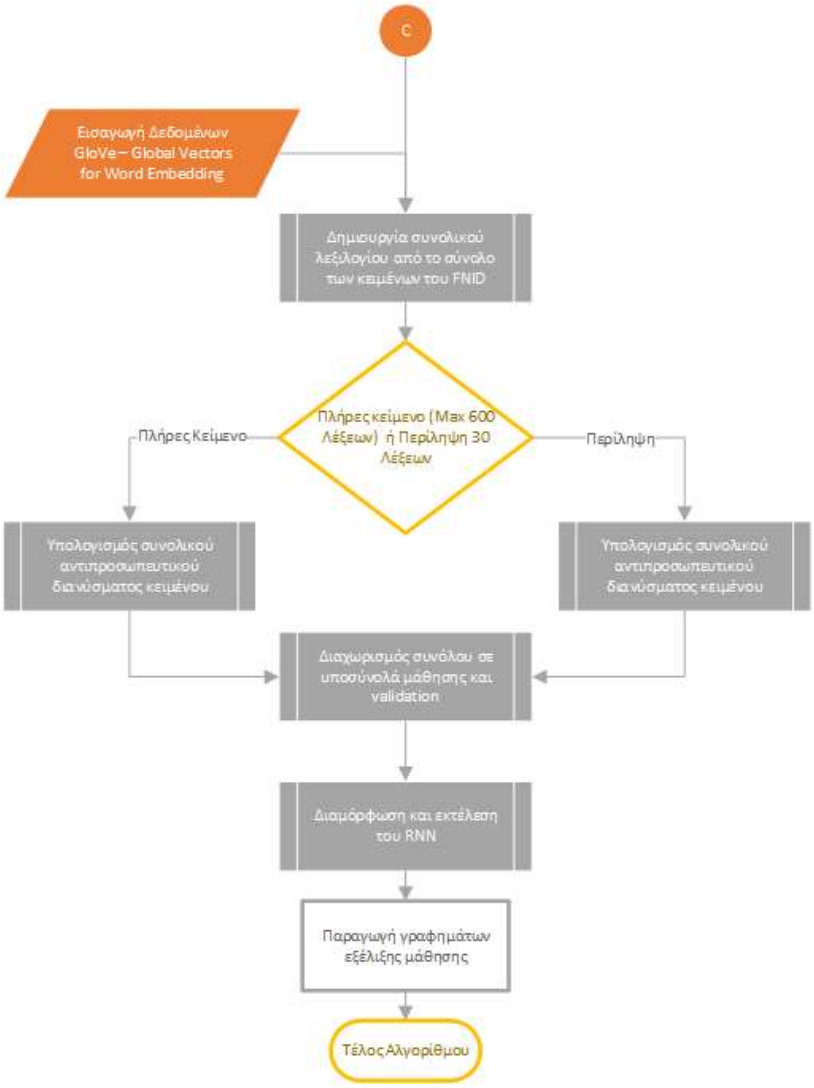
3.5.3 Εκπαίδευση μοντέλων μηχανικής μάθησης

Η διαδικασία εκπαίδευσης των μοντέλων μηχανικής μάθησης περιλαμβάνει την βαθμονόμηση των παραμέτρων της κάθε μεθόδου μετά και την επεξεργασία του υποσυνόλου με τα δεδομένα μάθησης με τα οποία τροφοδοτείται. Για τις ανάγκες της παρούσας εργασίας το υποσύνολο των 15212 ειδήσεων χρησιμοποιήθηκε ως δείγμα εκπαίδευσης των μοντέλων, ακολουθώντας την εκάστοτε διαδικασία

Ως αποτελέσματα της εκπαίδευσης, οι βελτιστοποιημένες παράμετροι πρόβλεψης χρησιμοποιήθηκαν με σκοπό τον έλεγχο της εγκυρότητας ειδήσεων για το δείγμα των 1054 ειδήσεων. Στα παρακάτω γραφήματα παρουσιάζονται οι 2 λογικές χρήσης των μοντέλων μηχανικής μάθησης που εφαρμόστηκαν.



Εικόνα 13 - Διάγραμμα ροής εφαρμογής εκπαίδευσης και πρόβλεψης μοντέλων ΜΜ μέσω διακριτοποίησης με αθροιστές λέξεων και TF-IDF



Εικόνα 14 - Διάγραμμα ροής εφαρμογής εκπαίδευσης και πρόβλεψης μοντέλων ΜΜ μέσω διακριτοποίησης και διανυσματικής αναπαράστασης μέσω GloVe

3.6 Παρουσίαση και αξιολόγηση αποτελεσμάτων αλγορίθμων

Η τελικά επιλεγμένη λίστα των μεθόδων μηχανικής μάθησης για την εύρεση και ταξινόμηση ψευδών ειδήσεων που εφαρμόστηκαν για τους σκοπούς της παρούσας εργασίας παρουσιάζεται παρακάτω:

Αλγόριθμος Ταξινόμησης	**Παράμετροι**	**Μέθοδος Διακριτοποίησης**
Naïve Bayes	Πολυωνυμική κατανομή	Count Vectorizer TF-IDF (3 -word)
Logistic Regression	Μέγιστος αριθμός επαναλήψεων = 1000, Penalty = 12	Count Vectorizer TF-IDF (3-word) TF-IDF (5) + GridSearch
Support Vector Machine	Μέγιστος αριθμός επαναλήψεων = 10000	Count Vectorizer TF-IDF
SVM με Stohastic Gradient Descent	Μέγιστος αριθμός επαναλήψεων = 1000, a=0.001	Count Vectorizer TF-IDF
Random Forest	Estimators (εκτιμητές) = 300, Max depth =15 *Max depth = 10 μέσω GridSearch	Count Vectorizer TF-IDF (3-word) TF-IDF (3) + GridSearch
LSTM - RNN	64 κόμβοι, dropout = 0.2, reccurent_dropout = 0.25, epochs = 20, verbose = 2, batch size = 32	Word Embedding, Full Text 600 words Word Embedding, Short Text, 30 words

Εικόνα 15 - Λίστα των εφαρμοζόμενων μεθόδων της εργασίας, μαζί με τις αντίστοιχες παραμέτρους

Η ομαδοποίηση τής παρουσίασης των αποτελεσμάτων έγινε με βάση την εκάστοτε μέθοδο διακριτοποίησης των κειμένων.

3.6.1 Αποτελέσματα ταξινόμησης μετά τη διακριτοποίηση μέσω CountVectorizer

Τα αποτελέσματα των 5 μεθόδων μετά τη διακριτοποίηση των κειμένων περίληψης βάσει του Count Vectorizer. Παρατηρείται εύκολα ότι με την απλούστερη μέθοδο διακριτοποίησης κειμένου, η ακρίβεια όλων των μεθόδων κυμαίνεται σε αρκετά χαμηλά επίπεδα. Επίσης τα συνολικά γραφήματα ταξινόμησης καταδεικνύουν τη χαμηλή προβλεπτική ικανότητα της μεθόδου αναφορικά με το εξεταζόμενο σύνολο δεδομένων. Η ακρίβεια πρόβλεψης τόσο στο δείγμα ελέγχου δεν ξεπερνά το 72%, ενώ στην εφαρμογή του στο συνολικό πληθυσμό μειώνεται στο ~63%.

Naïve Bayes

	precision	recall	f1-score	support
0.0	0.89	0.60	0.72	636
1.0	0.59	0.89	0.71	418
accuracy			0.71	1054
macro avg	0.74	0.74	0.71	1054
weighted avg	0.77	0.71	0.71	1054

Logistic Regression

	precision	recall	f1-score	support
0.0	0.85	0.57	0.68	636
1.0	0.56	0.84	0.68	418
accuracy			0.68	1054
macro avg	0.71	0.71	0.68	1054
weighted avg	0.73	0.68	0.68	1054

Support Vector Machine

	precision	recall	f1-score	support
0.0	0.72	0.54	0.62	636
1.0	0.49	0.68	0.57	418
accuracy			0.59	1054
macro avg	0.60	0.61	0.59	1054
weighted avg	0.63	0.59	0.60	1054

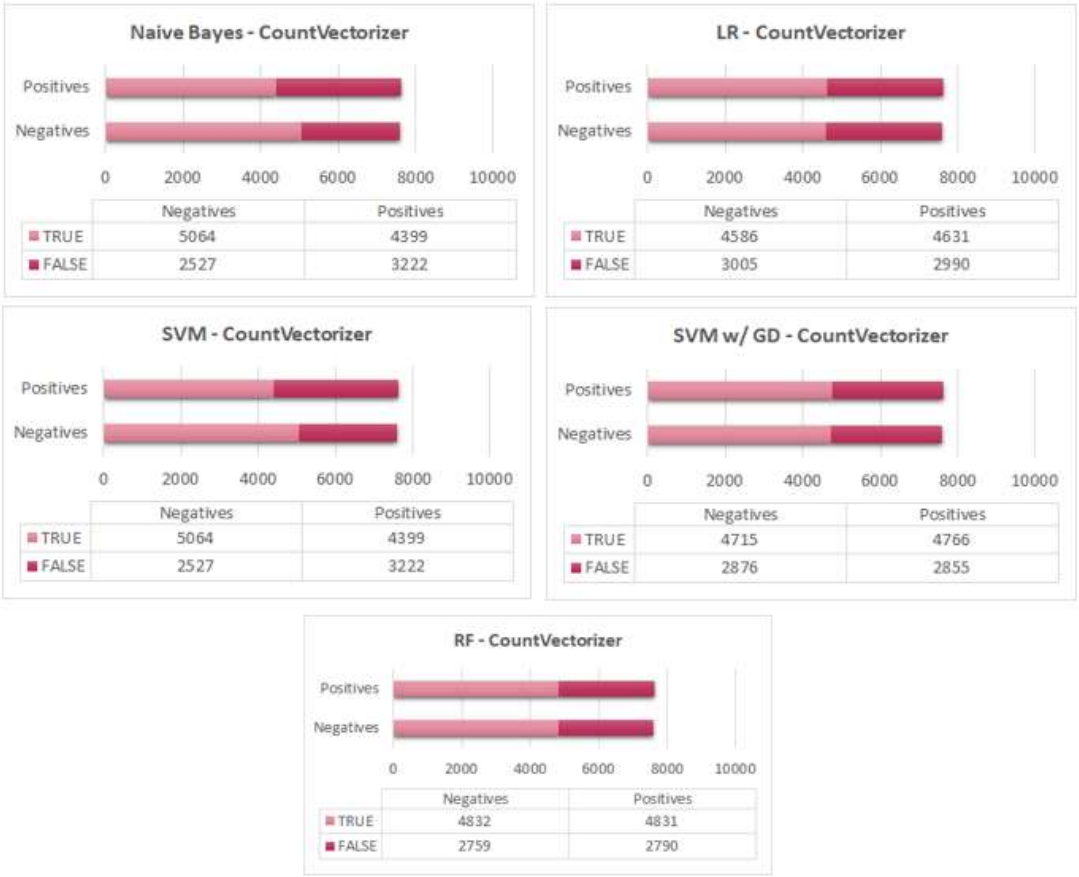
SVM με Stohastic Gradient Descent

	precision	recall	f1-score	support
0.0	0.91	0.50	0.65	636
1.0	0.55	0.92	0.69	418
accuracy			0.67	1054
macro avg	0.73	0.71	0.67	1054
weighted avg	0.77	0.67	0.66	1054

Random Forest

	precision	recall	f1-score	support
0.0	0.90	0.54	0.68	636
1.0	0.57	0.91	0.70	418
accuracy			0.69	1054
macro avg	0.73	0.73	0.69	1054
weighted avg	0.77	0.69	0.69	1054

Εικόνα 16 - Αποτελέσματα εφαρμογής μεθόδων μετά τη διακριτοποίηση κειμένου βάσει της μεθόδου CountVectorizer στο υποσύνολο εκπαίδευσης



Εικόνα 17 - Γραφήματα αξιολόγησης ταξινόμησης (Confusion Graphs) της μεθόδου CountVectorizer, στο υποσύνολο εκπαίδευσης

3.6.2 Αποτελέσματα ταξινόμησης μετά τη διακριτοποίηση μέσω nGram(3) – TF-IDF

Επεκτείνοντας την προηγούμενη μεθοδολογία σε σύνολα 3 λέξεων και διαμορφώνοντας αντίστοιχα τα προς εξέταση χαρακτηριστικά κειμένου σύμφωαν με τη μεθοδολογία TF-IDF, παρατηρείται πως η ακρίβεια πρόβλεψης δε βελτιώνεται σημαντικά, όπως καταδεικνύεται από τα αντίστοιχα γραφήματα. Παρατηρείται πως ανάλογα με τη μέθοδο, υπάρχει μικρή η και καμία αύξηση της ακρίβειας πρόβλεψης/ταξινόμησης. Πιο συγκεκριμένα, ακριβέστερες προβλέψεις φαίνεται να παρουσιάζουν οι μέθοδοι πολυωνυμικής Naïve Bayes, Λογιστικής Παλινδρόμησης αλλά και SVM.

Naïve Bayes

	precision	recall	f1-score	support
0.0	0.87	0.61	0.71	636
1.0	0.59	0.86	0.70	418
accuracy			0.71	1054
macro avg	0.73	0.73	0.71	1054
weighted avg	0.76	0.71	0.71	1054

Logistic Regression

	precision	recall	f1-score	support
0.0	0.89	0.61	0.72	636
1.0	0.60	0.88	0.71	418
accuracy			0.72	1054
macro avg	0.74	0.74	0.72	1054
weighted avg	0.77	0.72	0.72	1054

Support Vector Machine

	precision	recall	f1-score	support
0.0	0.87	0.62	0.72	636
1.0	0.60	0.86	0.70	418
accuracy			0.71	1054
macro avg	0.73	0.74	0.71	1054
weighted avg	0.76	0.71	0.72	1054

SVM με Stohastic Gradient Descent

	precision	recall	f1-score	support
0.0	0.98	0.19	0.32	636
1.0	0.45	0.99	0.62	418
accuracy			0.51	1054
macro avg	0.71	0.59	0.47	1054
weighted avg	0.77	0.51	0.43	1054

Random Forest

	precision	recall	f1-score	support
0.0	0.84	0.54	0.66	636
1.0	0.55	0.84	0.66	418
accuracy			0.66	1054
macro avg	0.69	0.69	0.66	1054
weighted avg	0.72	0.66	0.66	1054

Εικόνα 18 - Αποτελέσματα εφαρμογής μεθόδων μετά τη διακριτοποίηση κειμένου βάσει της μεθόδου nGram , TF-IDF (3)

Εικόνα 19 - Γραφήματα αξιολόγησης ταξινόμησης (Confusion Graphs) της μεθόδου ngram(3) – TF-IDF στο υποσύνολο εκπαίδευσης.

3.6.3 Αποτελέσματα ταξινόμησης μετά την αναζήτηση βέλτιστων παραμέτρων εκτέλεσης για τους αλγορίθμους Λογιστικής Παλινδρόμησης και Random Forest

Επιλέγοντας τους αλγορίθμους της Λογιστικής Παλινδρόμησης και τον Random Forest, πραγματοποιήθηκε επιπλέον ανάλυση αναφορικά με την εύρεση των βέλτιστων παραμέτρων χρήσης τους. Πιο συγκεκριμένα, εξετάστηκε το κατά πόσο επηρεάζει την προβλεπτική τους ικανότητα η μεταβολή του μεγέθους του συνόλου λέξεων που χρησιμοποιείται κατά τη διακριτοποίηση των κειμένων. Έτσι δημιουργήθηκαν σενάρια εφαρμογής επαναϋπολογισμού των χαρακτηριστικών TF-IDF για σύνολα λέξεων από 1-5. Επιπλέον, πραγματοποιήθηκαν σενάρια με διαφορετικές παραμέτρους του κάθε μοντέλου ξεχωριστά, όπως για παράδειγμα το βάθος για κάθε δέντρο απόφασης του Random Forest, η χρήση των TF-IDF για τη λογιστική παλινδρόμηση κ.α.

```
#random forest classifier parameters
parameters = {'rf_tfidf__ngram_range': [(1, 1), (1, 2),(1,3),(1,4),(1,5)],
              'rf_tfidf__use_idf': (True, False),
              'rf_clf__max_depth': (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15)
}

gs_clf = GridSearchCV(random_forest_ngram, parameters, n_jobs=-1)
gs_clf = gs_clf.fit(data1['statement'][:15212],data1['FakeFlag'][:15212])

gs_clf.best_score_
gs_clf.best_params_
gs_clf.cv_results_
```

Εικόνα 20 - GridSearch αναζήτηση των βέλτιστων παραμέτρων για τον αλγόριθμο Random Forest

```
#logistic regression parameters
parameters = {'LogR_tfidf__ngram_range': [(1, 1), (1, 2),(1,3),(1,4),(1,5)],
              'LogR_tfidf__use_idf': (True, False),
              'LogR_tfidf__smooth_idf': (True, False)
}

gs_clf = GridSearchCV(logR_pipeline_ngram, parameters, n_jobs=-1)
gs_clf = gs_clf.fit(data1['statement'][:15212],data1['FakeFlag'][:15212])

gs_clf.best_score_
gs_clf.best_params_
gs_clf.cv_results_
```

Εικόνα 21 - GridSearch αναζήτηση βέλτιστων παραμέτρων για τον αλγόριθμο Logistic Regression

Επιλέχθηκαν τελικά με ελάχιστες βελτιώσεις ο Random Forest 10 επιπέδων και υποσύνολο 3 λέξεων αλλά και η Λογιστική Παλινδρόμηση υποσυνόλου 5 λέξεων

```
random_forest_final = Pipeline([
        ('rf_tfidf',TfidfVectorizer(stop_words='english',ngram_range=(1,3),use_idf=True,smooth_idf=True)),
        ('rf_clf',RandomForestClassifier(n_estimators=300,n_jobs=3,max_depth=10))
        ])

random_forest_final.fit(data1['statement'],data1['FakeFlag'])
predicted_rf_final = random_forest_final.predict(data2['statement'])
np.mean(predicted_rf_final == data2['FakeFlag'])
print(classification_report(data2['FakeFlag'], predicted_rf_final))
```

```
logR_pipeline_final = Pipeline([
        #('LogRCV',countV_ngram),
        ('LogR_tfidf',TfidfVectorizer(stop_words='english',ngram_range=(1,5),use_idf=True,smooth_idf=False)),
        ('LogR_clf',LogisticRegression(penalty="l2",C=1))
        ])

logR_pipeline_final.fit(data1['statement'],data1['FakeFlag'])
predicted_LogR_final = logR_pipeline_final.predict(data2['statement'])
np.mean(predicted_LogR_final == data2['FakeFlag'])
#accuracy = 0.62
print(classification_report(data2['FakeFlag'], predicted_LogR_final))
```

Τα τελικά αποτελέσματα, παρά το γεγονός ότι χρησιμοποιήθηκαν οι βέλτιστες παράμετροι δε βελτιώθηκαν πάνω από 5% καταλήγοντας σε ποσοστά περίπου της τάξης του 72% ακρίβειας ταξινόμησης.

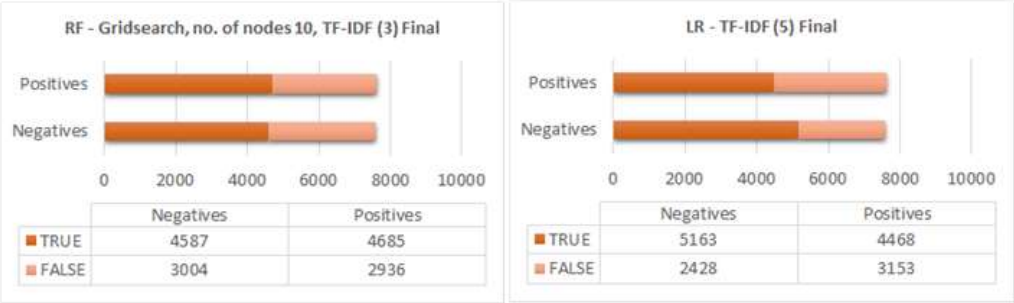
Random Forest

	precision	recall	f1-score	support
0.0	0.86	0.49	0.63	636
1.0	0.53	0.88	0.66	418
accuracy			0.65	1054
macro avg	0.69	0.68	0.64	1054
weighted avg	0.73	0.65	0.64	1054

Logistic Regression

	precision	recall	f1-score	support
0.0	0.88	0.61	0.72	636
1.0	0.60	0.88	0.71	418
accuracy			0.71	1054
macro avg	0.74	0.74	0.71	1054
weighted avg	0.77	0.71	0.72	1054

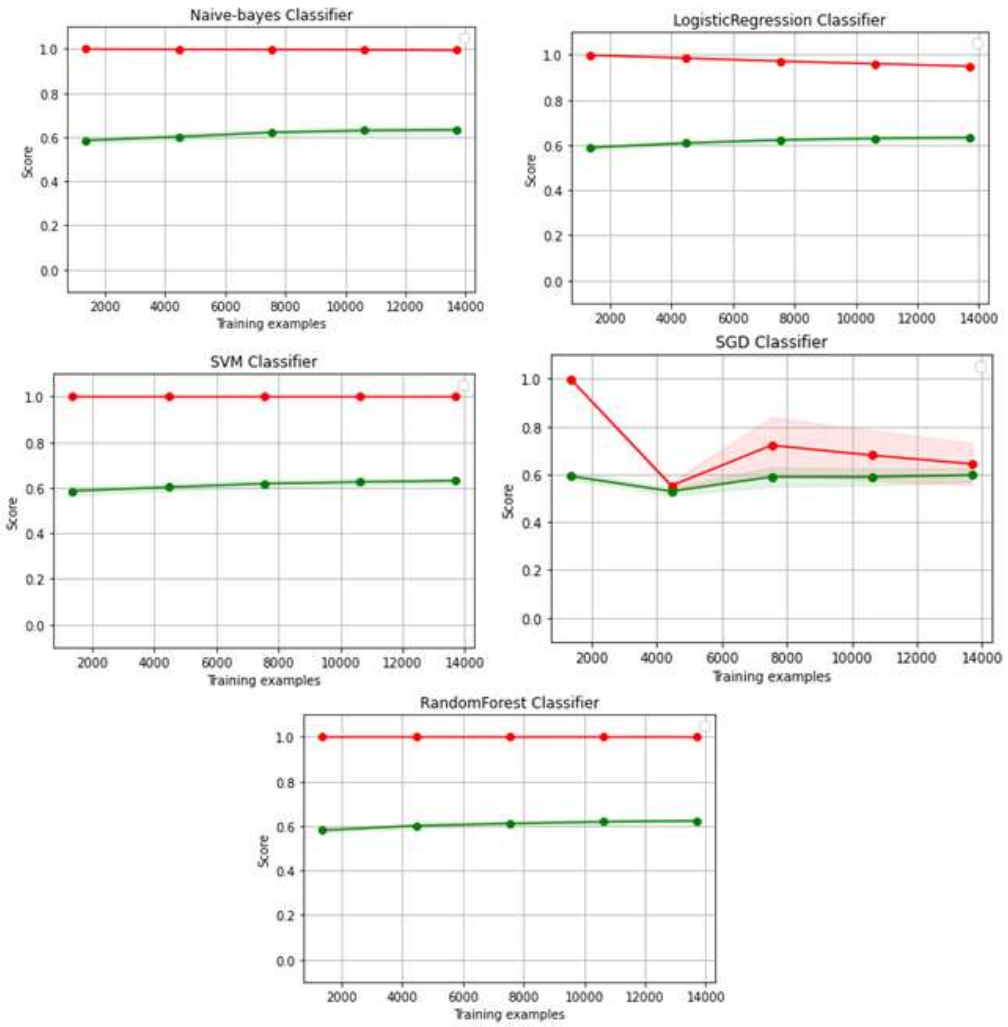
Εικόνα 22 - Απόδοση τελικών μοντέλων RF + LR μετά την εφαρμογή Gridsearch για εύρεση βέλτιστων παραμέτρων



Εικόνα 23 - Γραφήματα αξιολόγησης ταξινόμησης (Confusion Graphs) των αλγορίθμων RF και LR,μετά την επιλογή των βέλτιστων παραμέτρων μέσω GridSearch, στο υποσύνολο εκπαίδευσης

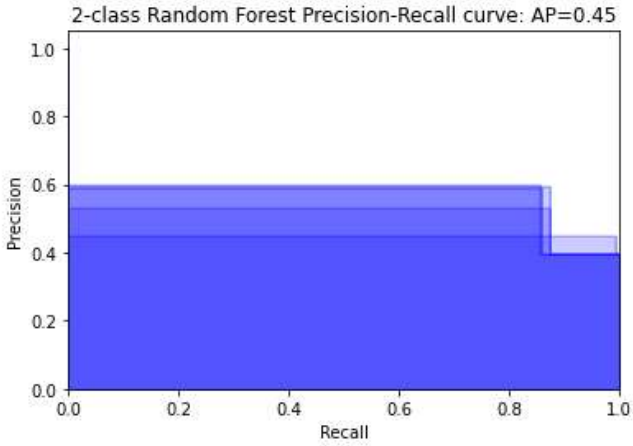
3.6.4 Εξέλιξη μάθησης αλγορίθμων

Μελετήθηκε επίσης η εξέλιξη του ποσοστού ακρίβειας πρόβλεψης ανάλογα με το μέγεθος του υποσυνόλου εκπαίδευσης. Είναι εμφανής η αδυναμία περαιτέρω ενίσχυσης του ποσοστού πρόβλεψης με τις συγκεκριμένες μεθόδους μέσω αύξησης του μεγέθους του δείγματος.



Εικόνα 24 - Γραφήματα εξέλιξης μάθησης των αλγορίθμους υπό εξέταση – Με κόκκινο η εξέλιξη του υποσυνόλου μάθησης και με πράσινο του υποσυνόλου validation.

Τέλος παρουσιάζεται η καμπύλη Precision – Recall για τους 2 αλγορίθμους σε κοινό γράφημα



Εικόνα 25 - Precision - Recall GRaph για τα βέλτιστα RF και LR

3.6.5 Αποτελέσματα ταξινόμησης με χρήση RNN LSTM. Διακριτοποίηση μέσω GloVe / Word Embedding

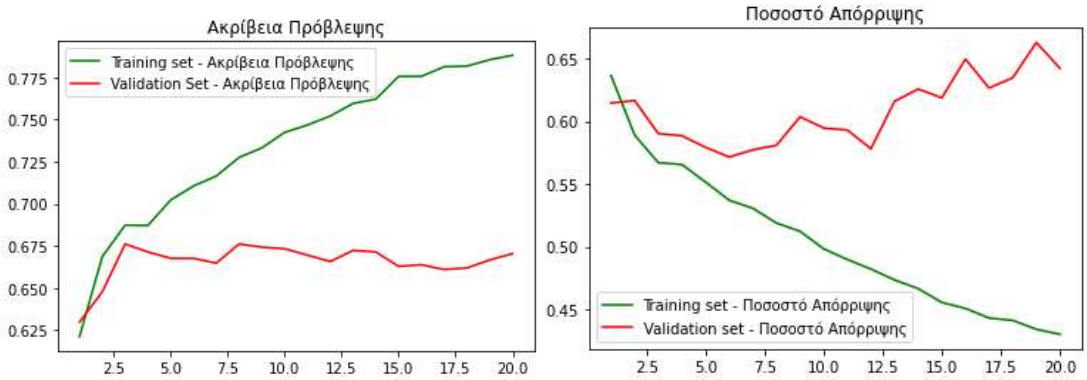
Η διακριτοποίηση μέσω GloVe, μας εξασφαλίζει τη διανυσματική απεικόνιση κάθε λέξης αλλά και συνολικά πρότασης ή κειμένου εξ ολοκλήρου στο διανυσματικό χώρο της επιλογής μας. Για την παρούσα εργασία επιλέχθηκε η απεικόνιση σε χώρο 200 διαστάσεων των λέξεων της αγγλικής γλώσσας (~ 2 εκατομμύρια λέξεις), όπως αυτή προέκυψε μετά και της αντιστοίχιση εφαρμόζοντας τις μεθόδους που αναλύθηκαν παραπάνω για λήμματα εντός Twitter για την περίοδο 2007-2018.

Το αντίστοιχο υποσύνολο λέξεων που αφορούσε την παρούσα εφαρμογή (~ 16 χιλιάδες λέξεις) χρησιμοποιήθηκε για την δημιουργία συνολικών διανυσμάτων για τις 30 πρώτες λέξεις της στήλης “Statement” που στο μεγαλύτερο ποσοστό του δείγματος εκπαίδευσης ήταν επαρκείς για να καλύψουν όλη την έκταση του νοήματος της είδησης αλλά πραγματοποιήθηκε επίσης και δοκιμή για το πλήρες κείμενο 600 λέξεων. Κατά αυτό τον τρόπο δημιουργήθηκαν τελικά αντιπροσωπευτικά διανύσματα για κάθε είδηση που τροφοδότησαν ένα Επαναλαμβανόμενο Νευρωνικό δίκτυο (RNN) με σκοπό την εκπαίδευσή του και την τελική χρήση του ως εργαλείο ταξινόμησης των ειδήσεων σε ψευδείς και αληθείς. Το δίκτυο αποτελείται από περίπου 68.000 παραμέτρους προς εκπαίδευση μέσα από 64 στρώματα κόμβων.

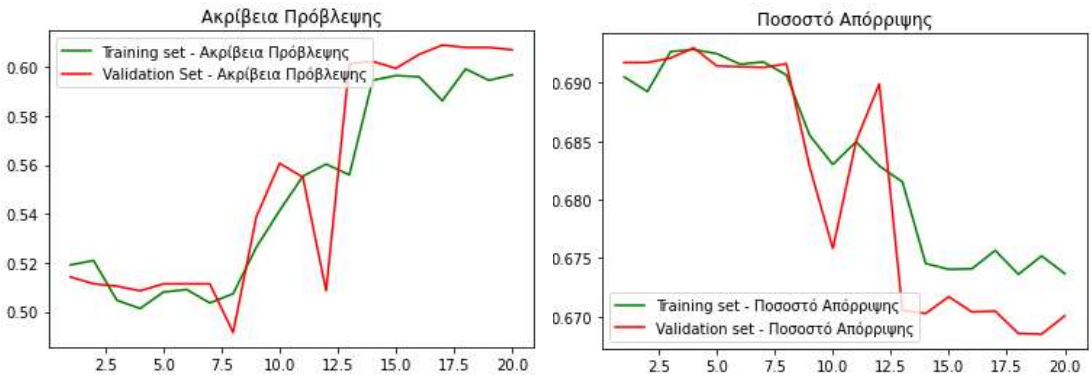
```
Summary of the built model...
Model: "sequential_5"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_5 (Embedding)      (None, 35, 200)           3187400
_________________________________________________________________
lstm_5 (LSTM)                (None, 64)                67840
_________________________________________________________________
dense_5 (Dense)              (None, 1)                 65
=================================================================
Total params: 3,255,305
Trainable params: 67,905
Non-trainable params: 3,187,400
_________________________________________________________________
None
```

Εικόνα 26 - Στοιχεία του LSTM RNN μοντέλου ταξινόμησης

Εφαρμόζοντας τη διαδικασία εκπαίδευσης και διατηρώντας ένα δείγμα αντίστοιχο με αυτό των προηγούμενων δοκιμών για διασταύρωση (validation sample – 1056 ειδήσεων) της απόδοσης, παρατηρήθηκε αντίστοιχη προβλεπτική ικανότητα της μεθόδου με τις προηγούμενες. Ειδικότερα, παρατηρώντας και τις καμπύλες μάθησης, η προβλεπτική ικανότητα δεν ξεπερνά το 70%.



Εικόνα 27 - RNN LSTM - Κείμενο 600 λέξεων



Εικόνα 28 - RNN LSTM - Περίληψη 30 λέξεων

Κεφάλαιο 4 - Συμπεράσματα & μελλοντική έρευνα

4.1 Συμπεράσματα

Τα συμπεράσματα που παρατίθενται παρακάτω, αφορούν στα αποτελέσματα των ανωτέρω εφαρμογών των αλγορίθμων μηχανικής μάθησης για την ταξινόμηση ψευδών ειδήσεων του συγκεκριμένου ανοιχτής πρόσβασης συνόλου ειδήσεων (Fake News Inference Dataset).

Τα χαμηλά ποσοστά προβλεπτικής ικανότητας των μεθόδων που χρησιμοποιήθηκαν ανωτέρω, καταδεικνύουν πως η δυνατότητα ταξινόμησης των ειδήσεων βασιζόμενη μόνο στα λεκτικά χαρακτηριστικά είναι σχετικά μικρή. Η εναπομείνασα αβεβαιότητα καθιστά την αποκλειστική χρήση μεθόδων ανάλυσης της φυσικής γλώσσας ως μέθοδο ταξινόμησης, ανεπαρκή, γεγονός που ενισχύει την ευρέως διαδεδομένη αντίληψη πως απαιτούνται υβριδικές μορφές ελέγχου αξιοπιστίας των ειδήσεων.

Κανείς από τους χρησιμοποιούμενους αλγορίθμους μηχανικής μάθησης δεν περιορίστηκε αναφορικά με το πλήθος των δεδομένων προς εκπαίδευση, γεγονός επίσης που καταδεικνύει πως τόσο το ίδιο το σύνολο ήταν επαρκές αλλά και ενδεχόμενος εμπλουτισμός του με νέες ειδήσεις (πάντα με αντίστοιχα λεκτικά χαρακτηριστικά), φαίνεται να μην μπορεί να οδηγήσει σε μεγάλη βελτίωση της προβλεπτικής ικανότητας.

Η εναλλάξ διερεύνηση των λεκτικών χαρακτηριστικών τόσο της δοθείσας περίληψης (“Statement”, ~ 30-40 λέξεις) των ειδήσεων και της πλήρους έκτασης του κειμένου (~600-700 λέξεις) δεν οδήγησε σε ασφαλή συμπεράσματα αναφορικά με την βέλτιστη έκταση του κειμένου της είδησης που πρέπει να χρησιμοποιείται σε τέτοιες μεθόδους και φαίνεται να εξαρτάται σε μεγάλο βαθμό από το εννοιολογικό περιεχόμενο του συνόλου που εξετάζεται κάθε φορά. Έτσι, για παράδειγμα η εφαρμογή αντίστοιχων μεθόδων σε σύνολα κειμένων ειδήσεων διαφορετικής φύσης (π.χ. οικονομικές ή υγειονομικές ειδήσεις) ενδεχομένως να παραγάγει διαφορετικά αποτελέσματα.

4.2 Προτάσεις για μελλοντική έρευνα

Οι βασικοί άξονες μελλοντικής έρευνας σύμφωνα με ένα μεγάλο μέρος της βιβλιογραφίας εστιάζονται στους εξής άξονες:

1) Μηχανισμοί γνωστικής λειτουργίας και αναγνώρισης ψευδών ειδήσεων από χρήστες (Technocognition) - Προτάσεις για συνδυαστικές λύσεις που ενεργοποιούν τόσο τεχνολογικής προέλευσης αλγορίθμους όσο και ψυχολογικές αντιδράσεις των χρηστών. (Lewandowsky, Ecker, & Cook, 2017). Επιπλέον, ήδη υφίστανται ερευνητικές πρωτοβουλίες που εστιάζουν σε μοντέλα μέτρησης αλλαγής στάσης απέναντι σε ειδήσεις
2) Μεγαλύτερης ευκρίνειας μέθοδοι εντοπισμού ψευδών ειδήσεων. Σύμφωνα με τις πιο πρόσφατες βιβλιογραφικές έρευνες, οι πιο αποτελεσματικές μέθοδοι εντοπισμού ψευδών ειδήσεων εφαρμόζουν τεχνικές βαθιάς μηχανικής μάθησης, οι οποίες όμως χαρακτηρίζονται από πολύ χαμηλή υπολογιστική ευκρίνεια, με αποτέλεσμα να μην υπάρχει η δυνατότητα εύκολης υπολογιστικής ερμηνείας των αποτελεσμάτων τους από τους ανθρώπους. (Human-machine intelligence) (Shu, Cui, Wang, Lee, & Liu, 2019)
3) Αποδοτικότερα μέτρα πρόληψης διάχυσης ψευδών ειδήσεων μέσω πλατφορμών ενημέρωσης και κοινωνικών δικτύων.
    - Μηχανισμοί φιλτραρίσματος βασισμένοι σε ψηφιακούς «φύλακες», οι οποίοι μπορούν να ανακόψουν τη διάχυση των ψευδών ειδήσεων καθιστώντας ανενεργούς τους μηχανισμούς διάδοσης τους.
    - Υβριδικής προσέγγισης έλεγχος δεδομένων με συνεργασία ανθρώπου-μηχανής. Μέσω αρχικών αυτοματοποιημένων ελέγχων, «ύποπτα» αναφορικά με την αξιοπιστία τους άρθρα ειδήσεων, θα μπορούσαν ν προωθούνται σε δευτερογενείς ελέγχους από ανθρώπους ειδικούς.
4) Έγκαιρη αναγνώριση ψευδών ειδήσεων (early detection). Καθώς οι μηχανισμοί που έχουν προταθεί ως ικανοποιητικοί για τη μοντελοποίηση της διασποράς των ψευδών ειδήσεων ακολουθούν πρότυπα των γεωμετρικής προόδου, η όσο το δυνατόν ταχύτερη αναγνώριση των

ψευδών ειδήσεων συνεισφέρει και στη δυναμική αντιμετώπισης της παραπληροφόρησης συνολικά. Οι αλγόριθμοι βαθιάς μηχανικής μάθησης έχουν τη δυνατότητα να προσφέρουν αυξημένες δυνατότητες στην ανάλυση του συνολικού χρόνου ζωής των ψευδών ειδήσεων

5) Μέθοδοι γρήγορης αναίρεσης και αποδόμησης των ψευδών ειδήσεων. Οργανισμοί ελέγχου γεγονότων, διάφορες στρατηγικές επικοινωνίας αλλά και αλγόριθμοι αναγνώρισης αξιοπιστίας των πηγών είναι οι βασικότεροι άξονες που ερευνώνται.
6) Η αύξηση της κοινωνικής επιρροής των δικτύων προστασίας από τις ψευδείς ειδήσεις. Ειδικότερα σε κοινοτικό επίπεδο αλλά και ευρύτερα διεθνές επίπεδο μελετώνται νέοι ψηφιακοί και υβριδικοί θεσμοί που ενισχύουν την ανθρώπινη αναγνώριση των ψευδών ειδήσεων.

Η Συλλογική Νοημοσύνη (Crowd Intelligence) αναγνωρίζεται επίσης διεθνώς βιβλιογραφικά ως ένας πλούσιος για έρευνα χώρος. Ο όρος Συλλογική Νοημοσύνη (Crowd Intelligence), περιλαμβάνει συνολικά στοιχεία ή κοινωνικά δείγματα από το σύνολο των χρηστών της Ψηφιακής ενημέρωσης καθ' όλο τον κύκλο ζωής της πληροφορίας, από τη δημιουργία της μέχρι και τον τρόπο διάχυσής της.

Μέσω των συμπερασμάτων της παρούσας εργασίας, αναγνωρίστηκε η ανάγκη διεύρυνσης των διαθέσιμων συνόλων δεδομένων σε περισσότερες θεματικές ενότητες, όπως επίσης και η εξέταση υβριδικών μεθόδων αξιολόγησης τόσο και άλλων μορφών περιεχομένου (ψηφιακά μέσα, πίνακες δεδομένων κλπ) όσο και της συμπεριφοράς των εκάστοτε χρηστών της πλατφόρμας συνδυαστικά.

4.2.1 Ευρωπαϊκές Ερευνητικές Πρωτοβουλίες

Όπως αναφέρθηκε και στην αρχή της αναφοράς, η δέσμευση των λειτουργιών της ευρωπαϊκής ένωσης στην ελευθερία του λόγου και τα λοιπά κατοχυρωμένα συνταγματικά δικαιώματα είναι δεδομένη και οι ενέργειες αναφορικά με την αναγνώριση ψευδών ειδήσεων θα πρέπει να πραγματοποιούνται χωρίς την παρεμπόδιση της δημόσιας πρόσβασης στην πληροφορία εν γένει. Το ανωτέρω αποτελεί δομικό λίθο στις προσπάθειες και γενικούς στόχους της ευρωπαϊκής ένωσης για το σχεδιασμό αντίστοιχων στρατηγικών.

Με τον ανωτέρω γενικό στόχο και τη βεβαιότητα πως πολύ σπάνια μια είδηση μπορεί να ταξινομηθεί ως αληθής ή ψευδής, αξιοποιώντας μονάχα κεντρικοποιημένες πηγές αξιολόγησης και πληροφόρησης, ευρύτερες δράσεις της ΕΕ περιλαμβάνουν μεταξύ άλλων:

- Τη δημιουργία πρόσφορου εδάφους από τις δημόσιες αρχές για την προώθηση του πλουραλισμού.
- Την προώθηση και ενίσχυση πρωτοβουλιών συνεργατικών πλατφορμών αναγνώρισης, παρακολούθησης και καταγραφής ειδήσεων που προέρχονται από κράτη εκτός ΕΕ
- Την προώθηση της συνεργασίας μεταξύ ΜΜΕ, εταιρειών τεχνολογίας και ερευνητών για τη διαφανή ανάπτυξη και εφαρμογή κριτηρίων και εργαλείων που βοηθούν στην αναγνώριση αξιόπιστου περιεχομένου, υποβοηθώντας και όχι αντικαθιστώντας ανθρώπους και οργανισμούς στην αναγνώριση ψευδών ειδήσεων

4.2.2 Συνδυαστικές μορφές αξιολόγησης εγκυρότητας – Σύνδεση με ευρωπαϊκές στρατηγικές

Η συνεργασία μεταξύ ανεξάρτητων οίκων αξιολόγησης της αυθεντικότητας των ειδήσεων θα ενίσχυε την διακρατική και διακλαδική προσπάθεια αναγνώρισης ψευδών ειδήσεων, βελτιώνοντας παράλληλα τις μεθόδους αναγνώρισης μέσω εφαρμογής τεχνολογιών αιχμής. Οι υφιστάμενες συνεργασίες μέσω των διαφόρων πλατφορμών είναι χρήσιμο να εξαπλωθούν μεταξύ ολόκληρης της ευρωπαϊκής ένωσης συνοδευόμενες από έναν χάρτη διαμοιρασμού δεδομένων μεταξύ ακαδημαϊκών και άλλων εμπλεκόμενων φορέων. Είναι επίσης επιθυμητή η εφαρμογή ενός κοινά αποδεκτού κώδικα δεοντολογίας για όλους τους φορείς ελέγχου αυθεντικότητας ειδήσεων, ο οποίος θα διασφαλίζει συνεχή αύξηση της διαφάνειας των εφαρμοζόμενων μεθόδων. Η επέκταση τέτοιων συνεργασιών θα μπορούσε να εξελίξει τις ήδη υπάρχουσες συνέργειες μεταξύ των φορέων μεταξύ των διαδικτυακών πλατφορμών ενώ θα μπορούσε να αποτελέσει πρόσφορο έδαφος για επέκταση και εξέλιξη καινοτόμων πρωτοβουλιών και ανοιχτού κώδικα τεχνολογιών αιχμής για την αναγνώριση ψευδών ειδήσεων και την

αξιολόγηση αξιοπιστίας των πηγών πληροφόρησης. Ο τελικός στόχος θα μπορούσε να είναι μια ανοικτή αγορά ελέγχου αξιοπιστίας (fact-checking) που θα αποφεύγει το «μονοπώλιο της αλήθειας», αλλά θα προωθεί τον πλουραλισμό. Τέτοιες συνεργασίες θα μπορούσαν να υποβοηθηθούν μέσω κοινοπραξιών δημόσιου και ιδιωτικού τομέα, και το δίκτυο το οποίο θα δημιουργούταν από τέτοιες συνεργασίες θα μπορούσε να ελέγχεται από ανεξάρτητες επιτροπές ειδικών κοινής αποδοχής.

4.2.3 Πλαίσιο τεχνορρύθμισης

Η επιτροπή εμπειρογνωμόνων της ΕΕ, προτείνει τη δημιουργία ευρωπαϊκών κέντρων για διεπιστημονική και ανεξάρτητη έρευνα στα προβλήματα της παραπληροφόρησης. Μέσω των κέντρων αυτών θα μπορεί να διασφαλιστεί ένας ικανός συντονισμός των συνεργατικών προσπαθειών μεταξύ δημόσιων και ιδιωτικών φορέων για τους ανωτέρω σκοπούς.

Λειτουργώντας εντός του νομικού πλαισίου της ΕΕ, οι διάφορες δικτυακές πλατφόρμες επικοινωνίας, προτείνεται, από την επιτροπή, να διευρύνουν τη δυνατότητα διαμοιρασμού των δεδομένων τους με τα ανεξάρτητα ευρωπαϊκά κέντρα έρευνας ώστε να διευκολύνεται η μελέτη φαινομένων παραπληροφόρησης, οι προσπάθειες αναγνώρισης ψευδών ειδήσεων αλλά και η μελέτη της δυναμικής συμπεριφοράς των χρηστών που βοηθά στη διασπορά των τελευταίων.

Αναφορικά με τη δυναμική συμπεριφορά των χρηστών και τη διάχυση των ειδήσεων μέσω δικτύων επικοινωνίας, οι (Kirchner & Reuter, 2020), πραγματοποίησαν μελέτες διάδραση ανθρώπου υπολογιστή με σκοπό να αποτιμήσουν την αποτελεσματικότητα μεθόδων μετάδοσης πληροφορίας σχετικά με την αυθεντικότητα των ειδήσεων. Δείγματα τέτοιων μεθόδων που περιλαμβάνουν μηνύματα προειδοποίησης, αρχική απόκρυψη συνολικά του αμφισβητούμενης αξιοπιστίας περιεχομένου, άρθρα που εκφράζουν την αντίθετη άποψη από τη δημοσιοποιούμενη αλλά και στατιστικά αξιοπιστίας της πηγής δημοσίευσης της κάθε είδησης, είναι μερικά από τα μέσα που έχουν εφαρμοστεί και μελετηθεί βιβλιογραφικά. Συνολικά, λαμβάνοντας υπόψη τις στατιστικές

μελέτες μέσω των οποίων ποσοτικοποιήθηκαν οι επιθυμίες των χρηστών των δικτυακών πλατφορμών συνοψίζονται στη αρχικά θετική στάση τους απέναντι σε προειδοποιήσεις αναφορικά με την αξιοπιστία των διαθέσιμων ειδήσεων, με την αυστηρή προϋπόθεση όμως της πλήρους διαφάνειας όλων των διαθέσιμων μεθόδων και δεδομένων αξιολόγησής της.

Τέλος, η παρακολούθηση και η ανταλλαγή βέλτιστων πρακτικών για τη δίκαιη και ανεξάρτητη αξιολόγηση της εγκυρότητας των ειδήσεων, είναι επίσης βασική επιθυμητή παράμετρος της λειτουργίας των ευρωπαϊκών αυτών κέντρων (European Commission, 2018). Οι διεθνείς προτάσεις για περαιτέρω, μελλοντική έρευνα που αναφέρθηκαν και στην προηγούμενη ενότητα, παραπάνω, θα μπορούσαν να αποτελέσουν υποσύνολο αυτών των βέλτιστων πρακτικών, πάντα μετά από διαβούλευση τόσο με τις ανεξάρτητες αρχής εμπειρογνωμόνων όσο και με τους χρήστες αλλά και διαχειριστές των δικτυακών πλατφορμών επικοινωνίας.

Βιβλιογραφία

Ahmed, H., Traore, I., & Saad, S. (2017). Detecting opinion spams and fake news using text classification. *Wiley*, 1-15.

Ajao, O., Bhowmik, D., & Zargari, S. (2018). Fake News Identification on Twitter with Hybrid CNN and RNN Models. *International Conference on Social Media & Society* (σ. 5). Copenhagen,Denmark: ACM.

Alassad, H. M., & Agarwal, N. (2021). Comprehensive decomposition optimization method for locating key sets of commenters spreading conspiracy theory in complex social networks. *Central European Journal of Operations Research*, 1-28.

Aldwairi, M., & Alwahedi, A. (2018). Detecting Fake News in Social Media Networks. *Procedia Computer Science 141*, 215–222.

Allcott, H., & Gentzkow, M. (2017). Social Media and Fake News in the 2016 Election. *Journal of Economic Perspectives, Volume 31, No. 2*, 211 - 237.

Alrubaian, M., Al-Qurishi, M., Alamri, A., Al-Rakhami, M., Mehedi, H. M., & Fortino, G. (2018). Credibility in Online Social Networks: A Survey. *IEEE, SECTION ON APPLICATIONS OF BIG DATA IN SOCIAL SCIENCES*, 1-28.

AlRubaian, M., Al-Qurishi, M., Al-Rakhami, M., Mehedi Hassan, M., & Alamri, A. (2016). CredFinder: a real-time tweets credibility assessing system. *International Conference on Advances in Social Networks Analysis and Mining* (σ. 5). San Fransisco, CA, USA: IEEE.

AlRubaian, M., Al-Qurishi, M., Al-Rakhami, M., Rahman, S. M., & Alamri, A. (2015). A Multistage Credibility Analysis Model for Microblogs. *International Conference on Advances in Social Networks Analysis and Mining* (σσ. 1-7). Paris, France: IEEE.

Alrubaian, M., Al-Qurishi, M., Hassan, M. M., & Alamri, A. (2018). A Credibility Analysis System for Assessing Information on Twitter. *TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, VOL. 15,* (σσ. 1-14). IEEE.

Aphiwongsophon, S., & Prabhas, C. (2018). Detecting Fake News with Machine Learning Method. *15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology* (σ. 4). IEEE.

Bajaj, S. (2017). *"The Pope Has a New Baby!", Fake News Decetion using Deep Learning.* Stanford University.

Baptista, J. P., & Gradim, A. (2020). Understanding Fake News Consumption: A Review. *Social Sciences 9*, 185-207.

Bonet-Jover, A., Piad-Morffis, A., Saquette, E., & Martinez-Barco, P. (2021). Exploiting discourse structure of traditional digital media to enhance automatic fake news detection. *Expert Systems With Applications*, 1-19.

Brashier, N. M., Pennycook, G., Berinsky, A. J., & Rand, D. G. (2021). Timing matters when correcting fake news. *PNAS 8*, 1-3.

Buntain, C., & Golbeck, J. (2017). Automatically Identifying Fake News in Popular Twitter Trends. *International Conference on Smart Cloud* (σ. 8). IEEE.

Canini, K. R., Suh, B., & Pirolli, P. L. (2011). Finding Credible Information Sources in Social Networks Based on Content and Social Structure. *International Conference on Privacy, Security, Risk, and Trust* (σσ. 1-8). IEEE.

Canini, K. R., Suh, B., & Pirolli, P. L. (2015). Finding Credible Information Sources in Social Networks Based on Content and Social Structure. *IEEE*, 1-8.

Caramancion, K. M. (2020). Understanding the Impact of Contextual Clues in Misinformation Detection. (σσ. 1-6). IEEE.

Castillo, C., Mendoza, M., & Poblete, B. (2011). Information Credibility on Twitter. *International World Wide Web Conference* (σ. 10). Hyderabad, India: International World Wide Web Conference Committee.

Chen, X., Zhang, K., Trajevski, G., Zhong, T., & Zhang, F. (2019). Information Diffusion Prediction via Recurrent Cascades Convolution. *35th International Conference on Data Engineering (ICDE)* (σσ. 1-12). IEEE.

Choras, M., Demestichas, K., Gielczyk, A., Herrero, A., Ksieniewicz, P., REmoundou, K., . . . Wozniak, D. (2021). Advanced Machine Learning techniques for fake news (online disinformation) detection: A systematic mapping study. *Applied Soft Computing Journal 101*, 1-15.

Clayton, K., Blair, S., Busam, J. A., Forstner, S., Glance, J., Green, G., . . . Nyhan, B. (2020). Real Solutions for Fake News? Measuring the Effectiveness of General Warnings and Fact-Check Tags in Reducing Belief in False Stories on Social Media. *Political Behavior 42*, 1073–1095.

Cloudhary, A., & Arora, A. (2021). Linguistic feature based learning model for fake news detection and classification. *Expert Systems With Applications 169*, 1-15.

Conroy, N. J., Rubin, V. L., & Chen, Y. (2015). Automatic Deception Detection: Methods for Finding Fake News. *ASIST* (σσ. 1-5). IEEE.

Dong, M., Yao, L., Wang, X., Benatallah, B., Huang, C., & Ning, X. (2020). Opinion fraud detection via neural autoencoder decision forest. *Pattern Recognition Letters 132*, 1-9.

Dordevic, M., Pourghomi, P., & Safieddine, F. (χ.χ.). Identifying Fake News from the Variables that Governs the Spread of Fake News. *IEEE*, 1-6.

D'Ulizia, A., Cascherra, M. C., Ferri, F., & Grifoni, P. (2021). Fake news detection: a survey of evaluation datasets. *COmputer Sciences*, 1-34.

Ebadi, N., Jozani, M., Choo, K.-K. R., & Rad, P. (2021). A Memory Network Information Retrieval Model for Identification of News Misinformation. *Transactions on Big Data*, 1-13.

Elmurngi, E., & Gherbi, A. (2017). An empirical study on detecting fake reviews using machine learning techniques. *The Seventh International Conference on Innovative Computing Technology* (σ. 8). IEEE.

Esmaelizadeh, S., Peh, G. X., & Xu, A. (2020). Neural Abstractive Text Summarization and Fake News Detection. *IEEE*, 1-10.

European Commission. (2018). *A multi-dimensional approach to disinformation.* European Commission.

Fagan, F. (2020). Optimal social media content moderation and platform immunities. *European Journal of Law and Economics 50*, 437–449.

Feng, S., Banerjee, R., & Choi, Y. (2012). Syntactic Stylometry for Deception Detection. *Research Gate*, 1-6.

Fontanarava, J., Pasi, G., & Viviani, M. (2017). An Ensemble Method for the Credibility Assessment of User-Generated Content. *WI '17* (σ. 6). Leipzig,Germany: ACM.

Frank, M. G., Paolantonio, N., Feeley, T. H., & Servoss, T. J. (2004). Individual and small group accuracy in judging truthful and deceptive communication. *Groupo Decision and Negotiation 13*, 45-49.

Fuller, C. M., Biros, D. P., & Wilson, R. I. (2009). Decision support for determining veracity via linguistic-based cues. *Decision Support Systems 46*, 695-703.

Gabriel, M., & Buning, M. (2018). *Report of the independent High Level GRoup on fake news and on;line disinformation.* Directorate-General for Communication Networks, Content and Technology.

Gereme, F., Zhu, W., Ayall, T., & Alemu, D. (2021). Combating Fake News in "Low-Resource" Languages: Amharic Fake News Detection Accompanied by Resource Crafting. *Information 12*, 1-9.

Ghaisani, A. P., Munajat, Q., & Handayani, P. W. (2020). Information Credibility Factors on Information Sharing Activites in Social Media. IEEE.

Giachanou, A., Zhang, G., & Rosso, P. (2020). Multimodal Multi-image Fake News Detection. *7th International Conference on Data Science and Advanced Analytics (DSAA)* (σσ. 1-8). IEEE.

Gilda, S. (2017). Evaluating Machine Learning Algorithms for Fake News Detection. *15th Student Conference on Research and Development* (σσ. 110-117). IEEE.

Girgis, S., Amer, E., & Gadallah, M. (2020). Deep Learning Algorithms for Detecting Fake News in Online Text. (σ. 5). IEEE.

Goldani, M. H., Momtazi, S., & Safabaksh, R. (2021). Detecting fake news with capsule neural networks. *Applied Soft Computing Journal 101*, 1-8.

Goldani, M. H., Safabakhsh, R., & Momtazi, S. (2021). Convolutional neural network with margin loss for fake news detection. *Information Processing and Management 58*, 1-12.

Gomes, J., de Souza, F. F., de Oliveira, A. M., & de Souza, J. F. (2020). A systematic mapping on automatic classification of fake news in social media. *Social Network Analysis and Mining (2020) 10:48*, 1-21.

Granik, M., & Mesyura, V. (2017). Fake News Detection Using Naive Bayes Classifier. *First Ukraine COnference on Electrical and COmputer Engineering.* IEEE.

Gravanis, G., Vakali, A., Diamantaras, K., & Karadais, P. (2019). Behind the cues: A benchmarking study for fake news detection. *Expert Systems With Applications 128*, 201-213.

Guo, B., Ding, Y., MA, S., MA, S., Li, K., & Yu, Z. (2021). The mass, fake news, and cognition security. *Front. Comput. Sci., 15*, 1-13.

Guo, B., Ding, Y., Yao, L., Liang, Y., & Yu, Z. (2020). The Future of False Information Detection on Social Media: New Perspectives and Trends. *ACM Computing Surveys, Vol. 53, No. 4, Article 68*, 1-36.

Gupta, A., Kumaraguru, P., Castillo, C., & Meier, P. (2014). TweetCred: Real-Time Credibility Assessment of Content on Twitter. *Social Informatics*, 228-243.

Gurbuz, B., Mawengkang, H., Husein, I., & Weber, G.-W. (2021). Rumour propagation: an operational research approach by computational and information theory. *Central European Journal of Operations Research*, 1-21.

Hakak, S., Alazab, M., Khan, S., Gadekallu, T. R., Maddikunta, P. K., & Khan, W. Z. (2021). An ensemble machine learning approach through effective feature

extraction to classify fake news. *Future Generation Computer Systems 117*, 47-58.

Han, H., Nakawatase, H., & Oyama, K. (2014). Evaluating credibility of interest reflection on Twitter. *International Journal of Web Information Systems*, 1-21.

Han, Q., & Fan, W. (2017). Rumor Spreading and Monitoring Deployment in Online Social Networks. *International Conference on Communication Technology* (σσ. 1-5). IEEE.

Hancock, J. T., Curry, L. E., Goorha, S., & Woodworth, M. (2008). On Lying and Being Lied To: A Linguistic Analysis of Deception in Computer- Mediated Communication. *Discourse Processes*, 1-25.

Horne, B. D., & Adali, S. (2017). This just in: Fake News Packs a Lot in Title, uses Simpler, Repetitive Content in Text Body, More Similar to Satire than Real News. *Association fo the Advancement of Artificial Intelligence*, 1-9.

Huang, Y.-F., & Chen, P.-H. (2020). Fake news detection using an ensemble learning model based on Self-Adaptive Harmony Search algorithms. *Expert Systems with Applications 159*, 1-17.

Ishida, Y., & Kuaraya, S. (2018). Fake News and its Credibility Evaluation by Dynamic Relational Networks : A Bottom Up Approach. *Procedia Computer Science 126*, 2228-2237.

Islam, M. R., Liu, S., Wang, X., & Xu, G. (2020). Deep learning for misinformation detection on online social networks: a survey and new perspectives. *Social Network Analysis and Mining (2020) 10:82*, 1-20.

Jiang, T., Haq, A. U., Saboor, A., & Ali, A. (2021). A Novel Stacking Approach for Accurate Detection of Fake News. *IEEE Access*, 1-14.

Jin, Z., Cao, J., Jiang, Y.-G., & Zhang, Y. (2014). News Credibility Evaluation on Microblog with a Hierarchical Propagation Model. *2014 IEEE International Conference on Data Mining*, (σσ. 230-239).

Kaliyar, R. K., Goswami, A., & Narang, P. (2021). DeepFakE: improving fake news detection using tensor decomposition-based deep neural network. *The Journal of Supercomputing 7*, 1015–1037.

Kaliyar, R. K., Goswami, A., & Narang, P. (2021). EchoFakeD: improving fake news detection in social media with an efficient deep neural network. *Neural Computing and Applications*, 1-17.

Kaliyar, R. K., Kumar, P., Kumar, M., Narkhede, M., Namboodiri, S., & Mishra, S. (2020). DeepNet: An Efficient Neural Network for Fake News Detection using News-User Engagements. *IEEE*, 1-6.

Kang, B., O'Donovan, J., & Hollerer, T. (2012). Modeling Topic Specific Credibility in Twitter. *Social Computing & Media* (σσ. 1-11). Lisbon, Portugal: ACM.

Kapusta, J., Hajek, P., Munk, M., & Benko, L. (2020). Comparison of fake and real news based on morphological analysis. *Procedia Computer Science 171*, 2285-2293.

Katarya, R., & Massoudi, M. (2020). Recognizing Fake News in Social Media with Deep Learning: A Systematic Review. *4th International Conference on Computer, Communication and Signal Processing (ICCCSP)* (σσ. 1-4). IEEE.

Kennedy, C., & Griffith, J. (2020). Using markup language to differentiate between reliable and unreliable news. *7th International Conference on Data Science and Advanced Analytics (DSAA)* (σσ. 1-7). IEEE.

Kirchner, J., & Reuter, C. (2020). Countering Fake News: A Comparison of Possible Solutions Regarding User Acceptance and Effectiveness. *ACM Hum.-Comput. Interact. 4,*, 1-27.

Kwon, S., Cha, M., Jung, K., & Chen, W. (2013). Prominent Features of Rumor Propagation in Online Social Media. *13th International Conference on Data Mining* (σσ. 1-6). ACM.

Le, N. C., Dao, M.-T., Nguyen, H.-L., Nguyen, T.-N., & Vu, H. (2020). An Application of Random Walk on Fake Account Detection Problem: A Hybrid Approach. *IEEE*, 1-6.

Lewandowsky, S., Ecker, U., & Cook, J. (2017). Beyond Misinformation Understanding and Coping with the Post-Truth era. *Journal of Applied Research in Memory and Cognition 6*, 353-369.

Liao, Q., Chai, H., Han, H., Zhang, X., Wang, X., Xia, W., & Ding, Y. (2021). An Integrated Multi-Task Model for Fake News Detection. *Transactions on Knowledge and Data Engineering* , 1-12.

Lindermann, D. J. (2020). What's Your Reality? Ealuating Sources and Addressing "Fake News". *Sociological Forum*, 1-8.

Lorek, K., Suehiro-Wicinski, J., Jankiwski-Lorek, M., & Gupta, A. (2015). Automated Credibility Assesment on Twitter. *Computer Science 16*, 2-13.

Ma, J., Gao, W., Wei, Z., Lu, Y., & Wong, K.-F. (2015). Detect Rumors Using Time Series of Social Context Information on Microblogging Websites. *CIKM'15, October 19–23* (σσ. 1-5). Melbourne, Australia: ACM.

Malhotra, B., & Vishwakarma, D. K. (2020). Classification of Propagation Path and Tweets for RUmor Detection using GRaphical Convolutional Networks and Transformer based Encodings. *Sixth International Conference on Multimedia Big Data* (σ. 8). IEEE.

Manning, C., Socher, R., Mundra, R., Peng, E., Sohmshetty, A., & Kamath, A. (2019). Lecture Notes: Part II Word Vectors II: GloVe, Evaluation and Training. *CS224n: Natural Language Processing with Deep learning, Lecture Notes*. Stanford University.

McGee, R. G., & Dawson, A. C. (2020). Fake news and fake research: Why meta-research matters more than ever. *Journal of Paediatrics and Child Health 56*, 1868–1871.

Medeiros, F. D., & Braga, R. B. (2020). Fake News Detection in Social Media: A Systematic Review. *ACM proceedings* (σσ. 1-8). ACM.

Meel, P., & Vishwakarma, D. K. (2020). Fake news, rumor, information pollution in social media and web: A contemporary survey of state-of-the-arts, challenges and opportunities. *Expert Systems With Applications 153*, 1-26.

Mesnil, G., Mikolov, T., Ranzato, A., & Bengio, Y. (2015). ENSEMBLE OF GENERATIVE AND DISCRIMINATIVE TECHNIQUES FOR SENTIMENT ANALYSIS OF MOVIE REVIEWS. *ICLR* (σσ. 1-5). IEEE.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word ρepresentations in vector space. 1301.

Molina, M. D., Sundar, S. S., Le, T., & Lee, D. (2021). "Fake News" Is Not Simply False Information: A Concept Explication and Taxonomy of Online Content. *American Behavioural Scientist 65*, 180-212.

Nikam, S. S., & Dalvi, R. (2019). Machine Learning Algorithm based model for classification of fake news on Twitter. 1-4.

Potthast, M., Kiesel, J., Reinartz, K., Janek, B., & Stein, B. (2017). A Stylometric Inquiry into Hyperpartisan and Fake News. *aclprints*, 1-10.

Qazvinian, V., Rosengren, E., Radey, D. R., & Mei, Q. (2011). *Empirical Methods in Natural Language Processing*, (σσ. 1589-1599).

R. De Oiliveira, N., Pisa, P. S., Lopez, M. A., & de Medeiros, D. S. (2021). Identifying Fake News on Social Networks Based on Natural Language Processing: Trends and Challenges. *Information 12*, 1-32.

Rivadeneira, L., Yang, J.-B., & Lopez-Ibanez, M. (2021). Predicting tweet impact using a novel evidential reasoning prediction method. *Expert Systems With Applications 169*, 1-13.

Rokach, L. (2010). Ensemble-based classifiers. *Artificial Intelligence Review 33*, 1-39.

Rubin, V. L. (2010). On Deception and Deception Detection: Content Analysis of Computer-Mediated Stated Beliefs. *ASIST 2010*, (σσ. 1-10). Pittsburg, PA, USA.

Rubin, V. L., Conroy, N. J., Chen, Y., & Sarah, C. (2016). FAke News or Truth? Using satirical cues to detect potentially misleading news. *Proceedings of NAACL-HLT* (σ. 11). San Diego, CA, USA: Association for Computational Linguistics.

Rubin, V., Conroy, N., & Chen, Y. (2015). Towards news verification: Deception detection methods for news discourse. *Hawaii International Conference on System Sciences*, (σσ. 1-12). Kauai.

Sadeghi, F., Bidgoly, A. J., & Amirkhani, H. (2020, August 2018). FNID: Fake News Inference Dataset. IEEE Dataport.

Sahoo, S. R., & Gupta, B. (2021). Multiple features based approach for automatic fake news detection on social networks using deep learning. *Applied Soft Computing Journal 100*, 1-16.

Sharma, O., Ahuja, S., & Kamal, P. (2021). Demystifying automatic rumour detection in social networks: A Review. *IOP Conf. Ser.: Mater. Sci. Eng. 1022*, 1-13.

Shrivastava, G., Jumar, P., Ojha, R. P., Srivastava, P. K., & Mohan, S. (2020). Defensive Modeling of Fake News Through Online Social Networks. *TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS*, 1-9.

Shu, K., Bhattacharjee, A., Alatawi, F., Nazer, T. H., Ding, K., Karami, M., & Liu, H. (2020). Combating disinformation in a social media age. *Wiley*, 1-23.

Shu, K., Cui, L., Wang, S., Lee, D., & Liu, J. (2019). dEFEND: Explainable Fake News Detection. *KDD '19: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (σσ. 395–405). ACM.

Sikdar, S., Adali, S., Amin, M., Abdelzaher, T., Chan, K., Cho, J.-H., . . . O'Donovan, J. (2016). Finding True and Credible Information on Twitter. *ResearchGate*, 1-9.

Sikdar, S., Byungkyu Kang, J., & Adali, S. (2013). Cutting Through the Noise:: Defining Ground Truth in Information Credibility on Twitter. *ASE 2012* (σσ. 1-18). ACM.

Singh, L. (2020). Fake News Detection: a comparison between available Deep Learning techniques in vector space. (σσ. 1-4). IEEE.

Singh, V. K., Ghosh, I., & Sonagara, D. (2020). Detecting fake news stories via multimodal analysis. *Wiley*, 1-15.

Song, C., Ning, N., Zhang, Y., & Wu, B. (2021). A multimodal fake news detection model based on crossmodal attention residual and multichannel convolutional neural networks. *Information Processing and Management 58*, 1-14.

Stewart, E. (2020). Detecting Fake News: Two Problems for Content Moderation. *Philosophy & Technology*, 1-18.

Sulis, E., & Tambuscio, M. (2020). Simulation of misinformation spreading processes in social networks: an application with NetLogo. *7th International Conference on Data Science and Advanced Analytics (DSAA)* (σσ. 1-5). IEEE.

Tacchini, E., Ballarin, G., Della Vedova, M. L., Moret, S., & de Alfaro, L. (2017). *Automated Fake News Detection in Social Networks.* Santa Cruz, CA: School of Engineering, University of California.

Thandar, M., & Usanavasin, S. (2015). Measuring Opinion Credibility in Twitter. *Advances in Intelligent Systems and Computing 361*, 205-215.

Varshney, D., & Vishwakarma, D. K. (2021). A review on rumour prediction and veracity assessment in online social network. *Expert Systems With Applications 168*, 1-21.

Vijayaraghavan, S., Guo, Z., Wang, Y., Voong, J., Xu, W., Li, L., . . . Cai, J. (2020). Fake News Detection with Different Models. *IEEE* (σσ. 1-15). IEEE.

Vishwakarma, D. K., Varshney, D., & Yadav, A. (2019). Detection and veracity analysis of fake news via scrapping and authenticating the web search. *Cognitive Systems Research 58*, 217-229.

Visser, J., LAwrence, J., & Read, C. (2020). Reason-Checking Fake News. *Communications of ACM 63*, 38-40.

Vogel, I., & Meghana, M. (2020). Detecting Fake News Spreaders on Twitter from a Multilingual Perspective. *7th International Conference on Data Science and Advanced Analytics (DSAA)* (σσ. 1-8). IEEE.

Vziatysheva, V. (2020). How Fake News Spreads Online? *International Journal of Media and Information Literacy 5*, 1-10.

Wang, C.-C. (2020). Fake News and Related Concepts: Definitions and Recent Research Development. *Contemporary Management Research 16*, 145-174.

Wang, Y., Wang, L., Yang, Y., & Lian, T. (2021). SemSeq4FD: Integrating global semantic relationship and local sequential order to enhance text representation for fake news detection. *Expert Systems With Applications 166*, 1-12.

Yoshikawa, K., Awa, T., Kusano, R., Sato, H., Ichino, M., & Yoshiura, H. (2020). A Fake News Dissemination Model Based on Updating Reliability and Doubt among Individuals. *11th International Conference on Awareness Science and Technology (iCAST)* (σσ. 1-8). IEEE.

Zhang, H., Fan, Z., Zeng, J., & Liu, Q. (2012). An improving deception detection method in Computer-Mediated Communication. *Journal of Networks 7*, 11.

Zhou, L., Burgoon, J. K., Nunamaker, J. F., & Twitchell, D. (2004). Automating Linguistics-Based Cues for Detecting Deception in Text-based Asynchronous Computer-Mediated Communication. *Group Decision and Negotiation 13*, 81-106.

Zotova, E., Agerri, R., & Rigau, G. (2021). Semi-automatic generation of multilingual datasets for stance detection in Twitter. *Expert Systems With Applications 170*, 1-13.

Παράρτημα

Εφαρμογές αθροιστών λέξεων και TF-IDF

```
# -*- coding: utf-8 -*-
"""
Created on Mon Apr 5 08:48:25 2021

@author: tkoralis
"""

import numpy as np
import pandas as pd
import os
import re
import matplotlib.pyplot as plt

#Data Preparation
import seaborn as sb

#Feature Selection
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.pipeline import Pipeline
import nltk
import nltk.corpus
from nltk.tokenize import word_tokenize

#Classifiers
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.pipeline import Pipeline
from sklearn.naive_bayes import MultinomialNB
from sklearn.linear_model import  LogisticRegression
from sklearn.linear_model import SGDClassifier
from sklearn import svm
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import KFold
from sklearn.metrics import confusion_matrix, f1_score,
classification_report
from sklearn.model_selection import GridSearchCV
from sklearn.model_selection import learning_curve
import matplotlib.pyplot as plt
from sklearn.metrics import precision_recall_curve
from sklearn.metrics import average_precision_score

from mlxtend.plotting import plot_confusion_matrix

path = "C:/Users/tkoralis/Fake News detection/Drafts/Final
Scripts/data"
os.chdir( path)
data1 = pd.read_csv('../data/fnn_train.csv')
data1.loc[data1['label_fnn'] =="fake", 'FakeFlag'] = int(1)
```

```
data1.loc[data1['label_fnn'] !="fake", 'FakeFlag'] = int(0)

data2 = pd.read_csv('../data/fnn_test.csv')
data2.loc[data2['label_fnn'] =="fake", 'FakeFlag'] = int(1)
data2.loc[data2['label_fnn'] !="fake", 'FakeFlag'] = int(0)

data3 = pd.read_csv('../data/fnn_dev.csv')
data3.loc[data3['label_fnn'] =="fake", 'FakeFlag'] = int(1)
data3.loc[data3['label_fnn'] !="fake", 'FakeFlag'] = int(0)

data = pd.concat([data1,data2,data3])

data.loc[data['label_fnn'] =="fake", 'FakeFlag'] = int(1)
data.loc[data['label_fnn'] !="fake", 'FakeFlag'] = int(0)
data.head()

#Στατιστικά Συνόλου προς επεξεργασία - DataPrep & Feat Selection

def create_distribution(dataFile):

    return sb.countplot(x='FakeFlag', data=dataFile, palette='hls')


#Κατανομές train,test και validation υποσυνόλων
countV = CountVectorizer()
train_count = countV.fit_transform(data['statement'].values)

print(countV)
print(train_count)


create_distribution(data)

################        Αθροιστές λέξεων Bag of Words - Επιλογή
Χαρακτηριστικών
def get_countVectorizer_stats():

    #Μέγεθος Λεξιλογίου
    train_count.shape

    #check vocabulary using below command
    print(countV.vocabulary_)

    #get feature names
    print(countV.get_feature_names()[:25])


#Δημιουργία TF-IDF χαρακτηριστικών
tfidfV = TfidfTransformer()
train_tfidf = tfidfV.fit_transform(train_count)

def get_tfidf_stats():
    train_tfidf.shape
    #Χαρακτηριστικά (features) των δεδομένων εκπαίδευσης
    print(train_tfidf.A[:10])

#Αθροιστές λέξεων - bag of words - with n-grams
```

```
tfidf_ngram =
TfidfVectorizer(stop_words='english',ngram_range=(1,5),use_idf=True,s
mooth_idf=True)

#POS Tagging
tagged_sentences = nltk.corpus.treebank.tagged_sents()

cutoff = int(.75 * len(tagged_sentences))
training_sentences = data['statement']

print(training_sentences)

#Εκπαίδευση POS tagger βασισμένο σε λέξεις
def features(sentence, index):
    """ πρόταση: [w1, w2, ...], δείκτης: ο δείκτης της λέξης """
    return {
        'word': sentence[index],
        'is_first': index == 0,
        'is_last': index == len(sentence) - 1,
        'is_capitalized': sentence[index][0].upper() ==
sentence[index][0],
        'is_all_caps': sentence[index].upper() == sentence[index],
        'is_all_lower': sentence[index].lower() == sentence[index],
        'prefix-1': sentence[index][0],
        'prefix-2': sentence[index][:2],
        'prefix-3': sentence[index][:3],
        'suffix-1': sentence[index][-1],
        'suffix-2': sentence[index][-2:],
        'suffix-3': sentence[index][-3:],
        'prev_word': '' if index == 0 else sentence[index - 1],
        'next_word': '' if index == len(sentence) - 1 else
sentence[index + 1],
        'has_hyphen': '-' in sentence[index],
        'is_numeric': sentence[index].isdigit(),
        'capitals_inside': sentence[index][1:].lower() !=
sentence[index][1:]
    }

#Ταξινομητές

#K-Fold cross validation - Διαδικασία
def build_confusion_matrix(classifier):

    k_fold = KFold(n_splits=5)
    scores = []
    confusion = np.array([[0,0],[0,0]])

    for train_ind, test_ind in k_fold.split(data1):
        train_text = data.iloc[train_ind]['statement']
        train_y = data.iloc[train_ind]['FakeFlag']

        test_text = data2.iloc[test_ind]['statement']
        test_y = data2.iloc[test_ind]['FakeFlag']

        classifier.fit(train_text,train_y)
        predictions = classifier.predict(test_text)
```

```
        confusion += confusion_matrix(test_y,predictions)
        score = f1_score(test_y,predictions)
        scores.append(score)

    return (print('Total statements classified:', len(data1)),
    print('Score:', sum(scores)/len(scores)),
    print('score length', len(scores)),
    print('Confusion matrix:'),
    print(confusion))
    #sb.heatmap(confusion, annot=True, annot_kws={"size": 16}) # font
size
    #plt.imshow(confusion, cmap='binary')

#K-Fold cross validatoin - Διαδικασία

#K-fold cross validation for all classifiers

#Ταξινομητής Naive Bayes
nb_pipeline = Pipeline([
        ('NBCV',countV),
        ('nb_clf',MultinomialNB())])

nb_pipeline.fit(data1['statement'],data1['FakeFlag'])
predicted_nb = nb_pipeline.predict(data2['statement'])
np.mean(predicted_nb == data2['FakeFlag'])


#Ταξινομητής Λογιστικής Παλινδρόμησης
logR_pipeline = Pipeline([
        ('LogRCV',countV),
        ('LogR_clf',LogisticRegression(max_iter=1000))
        ])

logR_pipeline.fit(data1['statement'],data1['FakeFlag'])
predicted_LogR = logR_pipeline.predict(data2['statement'])
np.mean(predicted_LogR == data2['FakeFlag'])


#Ταξινομητής SVM
svm_pipeline = Pipeline([
        ('svmCV',countV),
        ('svm_clf',svm.LinearSVC(max_iter=10000))
        ])

svm_pipeline.fit(data1['statement'],data1['FakeFlag'])
predicted_svm = svm_pipeline.predict(data2['statement'])
np.mean(predicted_svm == data2['FakeFlag'])


#Ταξινομητής SVM με Stochastic Gradient Descent
sgd_pipeline = Pipeline([
        ('svm2CV',countV),
        ('svm2_clf',SGDClassifier(loss='hinge', penalty='l2',
alpha=1e-3, max_iter=1000))
        ])

sgd_pipeline.fit(data1['statement'],data1['FakeFlag'])
```

```
predicted_sgd = sgd_pipeline.predict(data2['statement'])
np.mean(predicted_sgd == data2['FakeFlag'])


#Ταξινομητής random forest
random_forest = Pipeline([
        ('rfCV',countV),
        ('rf_clf',RandomForestClassifier(n_estimators=200,n_jobs=3))
        ])

random_forest.fit(data1['statement'],data1['FakeFlag'])
predicted_rf = random_forest.predict(data2['statement'])
np.mean(predicted_rf == data2['FakeFlag'])


build_confusion_matrix(nb_pipeline)
build_confusion_matrix(logR_pipeline)
build_confusion_matrix(svm_pipeline)
build_confusion_matrix(sgd_pipeline)
build_confusion_matrix(random_forest)

#=====================================================================
===============

print(classification_report(data2['FakeFlag'], predicted_nb))
print(classification_report(data2['FakeFlag'], predicted_LogR))
print(classification_report(data2['FakeFlag'], predicted_svm))
print(classification_report(data2['FakeFlag'], predicted_sgd))
print(classification_report(data2['FakeFlag'], predicted_rf))

## Χρήση με αθροιστές λέξεων και TF-IDF
#Ταξινομητής Naive Bayes
nb_pipeline_ngram = Pipeline([
        ('nb_tfidf',tfidf_ngram),
        ('nb_clf',MultinomialNB())])

nb_pipeline_ngram.fit(data1['statement'],data1['FakeFlag'])
predicted_nb_ngram = nb_pipeline_ngram.predict(data2['statement'])
np.mean(predicted_nb_ngram == data2['FakeFlag'])


#Ταξινομητής Λογιστικής Παλινδρόμησης
logR_pipeline_ngram = Pipeline([
        ('LogR_tfidf',tfidf_ngram),
        ('LogR_clf',LogisticRegression(penalty="l2",C=1))
        ])

logR_pipeline_ngram.fit(data1['statement'],data1['FakeFlag'])
predicted_LogR_ngram =
logR_pipeline_ngram.predict(data2['statement'])
np.mean(predicted_LogR_ngram == data2['FakeFlag'])


#Ταξινομητής SVM
svm_pipeline_ngram = Pipeline([
        ('svm_tfidf',tfidf_ngram),
        ('svm_clf',svm.LinearSVC())
```

```
        ])

svm_pipeline_ngram.fit(data1['statement'],data1['FakeFlag'])
predicted_svm_ngram = svm_pipeline_ngram.predict(data2['statement'])
np.mean(predicted_svm_ngram == data2['FakeFlag'])


#Ταξινομητής SVM με Stochastic Gradient Descent
sgd_pipeline_ngram = Pipeline([
         ('sgd_tfidf',tfidf_ngram),
         ('sgd_clf',SGDClassifier(loss='hinge', penalty='l2',
alpha=1e-3, max_iter=1000))
         ])

sgd_pipeline_ngram.fit(data1['statement'],data1['FakeFlag'])
predicted_sgd_ngram = sgd_pipeline_ngram.predict(data2['statement'])
np.mean(predicted_sgd_ngram == data2['FakeFlag'])


#Ταξινομητής random forest
random_forest_ngram = Pipeline([
        ('rf_tfidf',tfidf_ngram),
        ('rf_clf',RandomForestClassifier(n_estimators=300,n_jobs=3))
        ])

random_forest_ngram.fit(data1['statement'],data1['FakeFlag'])
predicted_rf_ngram = random_forest_ngram.predict(data2['statement'])
np.mean(predicted_rf_ngram == data2['FakeFlag'])


#Αξιολόγηση όλων των ταξινομητών
build_confusion_matrix(nb_pipeline_ngram)
build_confusion_matrix(logR_pipeline_ngram)
build_confusion_matrix(svm_pipeline_ngram)
build_confusion_matrix(sgd_pipeline_ngram)
build_confusion_matrix(random_forest_ngram)

print(classification_report(data2['FakeFlag'], predicted_nb_ngram))
print(classification_report(data2['FakeFlag'], predicted_LogR_ngram))
print(classification_report(data2['FakeFlag'], predicted_svm_ngram))
print(classification_report(data2['FakeFlag'], predicted_sgd_ngram))
print(classification_report(data2['FakeFlag'], predicted_rf_ngram))

data2['FakeFlag'].shape

"""
Από τα μοντέλα που δοκιμάστηκαν, επιλέχθηκαν προς δοκιμή με τη
συνάρτηση GridSearch
τρία εξ αυτών, για οικονομία χρόνου υπολογισμών.
Επιλέχθηκαν τα μοντέλα της Λογιστικής Παλινδρόμησης, του SVM
και του Random Forest.
"""

#Εύρεση βέλτιστων παραμέτρων μέσω grid-search
#Παράμετροι του ταξινομητή Random Forest
parameters = {'rf_tfidf__ngram_range': [(1, 1), (1,
2),(1,3),(1,4),(1,5)],
```

```
                'rf_tfidf__use_idf': (True, False),
                'rf_clf__max_depth':
(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15)
}

gs_clf = GridSearchCV(random_forest_ngram, parameters, n_jobs=-1)
gs_clf =
gs_clf.fit(data1['statement'][:15212],data1['FakeFlag'][:15212])

gs_clf.best_score_
gs_clf.best_params_
gs_clf.cv_results_


#Παράμετροι του ταξινομητή Λογιστικής Παλινδρόμησης
parameters = {'LogR_tfidf__ngram_range': [(1, 1), (1,
2),(1,3),(1,4),(1,5)],
                'LogR_tfidf__use_idf': (True, False),
                'LogR_tfidf__smooth_idf': (True, False)
}

gs_clf = GridSearchCV(logR_pipeline_ngram, parameters, n_jobs=-1)
gs_clf =
gs_clf.fit(data1['statement'][:15212],data1['FakeFlag'][:15212])

gs_clf.best_score_
gs_clf.best_params_
gs_clf.cv_results_

#Παράμετροι του Γραμμικού SVM
parameters = {'svm_tfidf__ngram_range': [(1, 1), (1,
2),(1,3),(1,4),(1,5)],
                'svm_tfidf__use_idf': (True, False),
                'svm_tfidf__smooth_idf': (True, False),
                'svm_clf__penalty': ('l1','l2'),
}

gs_clf = GridSearchCV(svm_pipeline_ngram, parameters, n_jobs=-1)
gs_clf =
gs_clf.fit(data1['statement'][:15212],data1['FakeFlag'][:15212])

gs_clf.best_score_
gs_clf.best_params_
gs_clf.cv_results_

#Οι παραπάνω εντολές ταξινομούν τα διάφορα instances των μοντέλων με
βάση τη βέλτιστη απόδοση
#γεγονός που εν τέλει μας βοηθά να βρούμε το μοντέλο με τις βέλτιστες
παραμέτρους.

#Εφαρμογή των μοντέλων με τις βέλτιστες παραμέτρους
random_forest_final = Pipeline([

('rf_tfidf',TfidfVectorizer(stop_words='english',ngram_range=(1,3),us
e_idf=True,smooth_idf=True)),
```

```
('rf_clf',RandomForestClassifier(n_estimators=300,n_jobs=3,max_depth=
10))
        ])

random_forest_final.fit(data1['statement'],data1['FakeFlag'])
predicted_rf_final = random_forest_final.predict(data2['statement'])
np.mean(predicted_rf_final == data2['FakeFlag'])
print(classification_report(data2['FakeFlag'], predicted_rf_final))

logR_pipeline_final = Pipeline([
        #('LogRCV',countV_ngram),

('LogR_tfidf',TfidfVectorizer(stop_words='english',ngram_range=(1,5),
use_idf=True,smooth_idf=False)),
        ('LogR_clf',LogisticRegression(penalty="l2",C=1))
        ])

logR_pipeline_final.fit(data1['statement'],data1['FakeFlag'])
predicted_LogR_final =
logR_pipeline_final.predict(data2['statement'])
np.mean(predicted_LogR_final == data2['FakeFlag'])
#accuracy = 0.62
print(classification_report(data2['FakeFlag'], predicted_LogR_final))


"""
Εφαρμόζοντας τα μοντέλα με τις βέλτιστες παραμέτρους παρατηρούμε πολύ
μικρή βελτίωση μεταξύ των μεθόδων
"""

#Γραφήματα εξέλιξης καμπυλών μάθησης
def plot_learing_curve(pipeline,title):
    size = 10 #Δοκιμάστηκε παράμετρος 10000 αλλά ο υπολογιστικός
χρόνος ήταν >1 ημέρας
    cv = KFold(size, shuffle=True)

    X = data1["statement"]
    y = data1["FakeFlag"]

    pl = pipeline
    pl.fit(X,y)

    train_sizes, train_scores, test_scores = learning_curve(pl, X, y,
n_jobs=-1, cv=cv, train_sizes=np.linspace(.1, 1.0, 5), verbose=0)

    train_scores_mean = np.mean(train_scores, axis=1)
    train_scores_std = np.std(train_scores, axis=1)
    test_scores_mean = np.mean(test_scores, axis=1)
    test_scores_std = np.std(test_scores, axis=1)

    plt.figure()
    plt.title(title)
    plt.legend(loc="best")
    plt.xlabel("Training examples")
    plt.ylabel("Score")
    plt.gca().invert_yaxis()
```

```
    # Βάση του γραφήματος
    plt.grid()

    # Εισαγωγή της τυπικής απόκλισης
    plt.fill_between(train_sizes, train_scores_mean -
train_scores_std, train_scores_mean + train_scores_std, alpha=0.1,
color="r")
    plt.fill_between(train_sizes, test_scores_mean - test_scores_std,
test_scores_mean + test_scores_std, alpha=0.1, color="g")

    # Εισαγωγή των γραμμών εκπαίδευσης και validation
    plt.plot(train_sizes, train_scores_mean, 'o-', color="r",
label="Training score")
    plt.plot(train_sizes, test_scores_mean, 'o-', color="g",
label="Cross-validation score")

    # Scaling και προβολή λαθών
    plt.ylim(-.1,1.1)
    plt.show()


#Προβολή γραφημάτων καμπύλης μάθησης
plot_learing_curve(nb_pipeline_ngram,"Naive-bayes Classifier")
plot_learing_curve(logR_pipeline_ngram,"LogisticRegression
Classifier")
plot_learing_curve(svm_pipeline_ngram,"SVM Classifier")
plot_learing_curve(sgd_pipeline_ngram,"SGD Classifier")
plot_learing_curve(random_forest_ngram,"RandomForest Classifier")



"""
Μέσω των γραφημάτων, παρατηρούμε πως με την αύξηση του πλήθους των
ειδήσεων στο
σύνολο εκπαίδευσης (train set), δεν επιτυγχάνεται καλύτερη
προβλεπτική απόδοση για σχεδόν όλους
τους εφαρμοζόμενους αλγορίθμους.
"""

def plot_PR_curve(classifier):

    precision, recall, thresholds =
precision_recall_curve(data2['FakeFlag'], classifier)
    average_precision = average_precision_score(data2['FakeFlag'],
classifier)

    plt.step(recall, precision, color='b', alpha=0.2,
             where='post')
    plt.fill_between(recall, precision, step='post', alpha=0.2,
                     color='b')

    plt.xlabel('Recall')
    plt.ylabel('Precision')
    plt.ylim([0.0, 1.05])
    plt.xlim([0.0, 1.0])
```

```
    plt.title('2-class Random Forest Precision-Recall curve:
AP={0:0.2f}'.format(
              average_precision))

plot_PR_curve(predicted_LogR_ngram)
plot_PR_curve(predicted_rf_ngram)

"""
Εκτίμηση των 50 βασικότερων χαρακτηρηστικών ανά μοντέλο
"""

def show_most_informative_features(model, vect, clf, text=None,
n=50):
    # Εξαγωγή μεθόδου διακριτοποίησης και ταξινόμησης
    vectorizer = model.named_steps[vect]
    classifier = model.named_steps[clf]

     # Έλεγχος δυνατότητας υπολογισμού
    if not hasattr(classifier, 'coef_'):
        raise TypeError(
            "Cannot compute most informative features on {}.".format(
                classifier.__class__.__name__
            )
        )

    if text is not None:
        # Υπολογισμός συντελεστών
        tvec = model.transform([text]).toarray()
    else:
        tvec = classifier.coef_

    # Αντιστοίχιση χαρακτηριστικών με συντελεστές
    coefs = sorted(
        zip(tvec[0], vectorizer.get_feature_names()),
        reverse=True
    )

    topn  = zip(coefs[:n], coefs[:-(n+1):-1])

    # Δημιουργία κειμένου προς εκτύπωση
    output = []

    if text is not None:
        output.append("\"{}\"".format(text))
        output.append(
            "Classified as: {}".format(model.predict([text]))
        )
        output.append("")

    # Δημιουργία 2 στηλών με τα περισσότερο και τα λιγότερο σημαντικά
γνωρίσματα/χαρακτηριστικά
    for (cp, fnp), (cn, fnn) in topn:
        output.append(
            "{:0.4f}{: >15}    {:0.4f}{: >15}".format(
                cp, fnp, cn, fnn
            )
        )
```

```
    print(output)

show_most_informative_features(logR_pipeline_ngram,vect='LogR_tfidf',
clf='LogR_clf')
show_most_informative_features(nb_pipeline_ngram,vect='nb_tfidf',clf=
'nb_clf')
show_most_informative_features(svm_pipeline_ngram,vect='svm_tfidf',cl
f='svm_clf')
show_most_informative_features(sgd_pipeline_ngram,vect='sgd_tfidf',cl
f='sgd_clf')
```

Εφαρμογή RNN LSTM

```
# -*- coding: utf-8 -*-
"""
Created on Mon Απρ 12 10:33:48 2021

@author: tkoralis
"""

import numpy as np
import pandas as pd
import os
import re
import matplotlib.pyplot as plt
from tensorflow.python.keras.preprocessing.text import Tokenizer
from tensorflow.python.keras.preprocessing.sequence import
pad_sequences
from keras.models import Sequential
from keras.layers import Dense, Embedding, GRU, LSTM, Bidirectional
from keras.layers.embeddings import Embedding
from keras.initializers import Constant
from keras.callbacks import ModelCheckpoint
from keras.models import load_model

embeddings_index = {}
embedding_dim = 200
GLOVE_DIR = r"C:\Users\tkoralis\Fake News detection\Drafts\Final
Scripts\data\GloVe"
f = open(os.path.join(GLOVE_DIR, 'glove.twitter.27B.200d.txt'),
encoding = "utf-8")
for line in f:
    values = line.split()
    word = values[0]
    coefs = np.asarray(values[1:], dtype='float32')
    embeddings_index[word] = coefs
f.close()

print('Found %s word vectors.' % len(embeddings_index))
#print(embeddings_index.get('fake'))
path = "C:/Users/tkoralis/Fake News detection/Drafts/Final
Scripts/data"
os.chdir( path)
data1 = pd.read_csv('../data/fnn_train.csv')
data2 = pd.read_csv('../data/fnn_test.csv')
```

```
data3 = pd.read_csv('../data/fnn_dev.csv')
data = pd.concat([data1,data2,data3])

data.loc[data['label_fnn'] =="fake", 'FakeFlag'] = int(1)
data.loc[data['label_fnn'] !="fake", 'FakeFlag'] = int(0)
data.head()

def clean_text(text):
    text = text.lower()

    pattern       =       re.compile('http[s]?://(?:[a-zA-Z]|[0-9]|[$-
_@.&+]|[!*\(\),]|(?:%[0-9a-fA-F][0-9a-fA-F]))+')
    text = pattern.sub('', text)
    text = " ".join(filter(lambda x:x[0]!='@', text.split()))
    emoji = re.compile("["
                           u"\U0001F600-\U0001FFFF"  # emoticons
                           u"\U0001F300-\U0001F5FF"  # σύμβολα
                           u"\U0001F680-\U0001F6FF"  # σύμβολα (συν.)
                           u"\U0001F1E0-\U0001F1FF"      #   στοιχεία
λογισμικού
                           u"\U00002702-\U000027B0"
                           u"\U000024C2-\U0001F251"
                           "]+", flags=re.UNICODE)

    text = emoji.sub(r'', text)
    text = text.lower()
    text = re.sub(r"i'm", "i am", text)
    text = re.sub(r"he's", "he is", text)
    text = re.sub(r"she's", "she is", text)
    text = re.sub(r"that's", "that is", text)
    text = re.sub(r"what's", "what is", text)
    text = re.sub(r"where's", "where is", text)
    text = re.sub(r"\'ll", " will", text)
    text = re.sub(r"\'ve", " have", text)
    text = re.sub(r"\'re", " are", text)
    text = re.sub(r"\'d", " would", text)
    text = re.sub(r"\'ve", " have", text)
    text = re.sub(r"won't", "will not", text)
    text = re.sub(r"don't", "do not", text)
    text = re.sub(r"did't", "did not", text)
    text = re.sub(r"can't", "can not", text)
    text = re.sub(r"it's", "it is", text)
    text = re.sub(r"couldn't", "could not", text)
    text = re.sub(r"have't", "have not", text)
    text = re.sub(r"[,.\"\'!@#$%^&*(){}?/;`~:<>+=-]", "", text)
    return text

import string
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords

def CleanTokenize(df):
    head_lines = list()
    lines = df["statement"].values.tolist()

    for line in lines:
        line = clean_text(line)
```

```
        # Διακριτοποίηση κειμένου
        tokens = word_tokenize(line)
        # Αφαίρεση σημείων στίξης
        table = str.maketrans('', '', string.punctuation)
        stripped = [w.translate(table) for w in tokens]
        # Αφαίρεση όχι αλφαριθμητικών χαρακτήρων
        words = [word for word in stripped if word.isalpha()]
        stop_words = set(stopwords.words("english"))
        # Αφαίρεση "stopwords" - Κοινών συνδετικών λέξεων
        words = [w for w in words if not w in stop_words]
        head_lines.append(words)
    return head_lines

head_lines = CleanTokenize(data)
head_lines[0:10]

from collections import Counter
from wordcloud import WordCloud, ImageColorGenerator
pos_data = data.loc[data['FakeFlag'] == 1]
pos_head_lines = CleanTokenize(pos_data)
pos_lines = [j for sub in pos_head_lines for j in sub]
word_could_dict=Counter(pos_lines)

wordcloud = WordCloud(width = 1000, height =
500).generate_from_frequencies(word_could_dict)
plt.figure(figsize=(15,8))
plt.imshow(wordcloud)
plt.axis("off")

validation_split = 0.061
max_length = 35

tokenizer_obj = Tokenizer()
tokenizer_obj.fit_on_texts(head_lines)
sequences = tokenizer_obj.texts_to_sequences(head_lines)

word_index = tokenizer_obj.word_index
print("unique tokens - ",len(word_index))
vocab_size = len(tokenizer_obj.word_index) + 1
print('vocab size -', vocab_size)

lines_pad = pad_sequences(sequences, maxlen=max_length,
padding='post')
sentiment = data['FakeFlag'].values

indices = np.arange(lines_pad.shape[0])
np.random.shuffle(indices)
lines_pad = lines_pad[indices]
sentiment = sentiment[indices]

num_validation_samples = int(validation_split * lines_pad.shape[0])

X_train_pad = lines_pad[:-num_validation_samples]
y_train = sentiment[:-num_validation_samples]
X_test_pad = lines_pad[-num_validation_samples:]
y_test = sentiment[-num_validation_samples:]
```

```
print('Μορφή του X_train_pad:', X_train_pad.shape)
print('Μορφή του y_train:', y_train.shape)

print('Μορφή του X_test_pad:', X_test_pad.shape)
print('Μορφή του y_test:', y_test.shape)

embedding_matrix = np.zeros((len(word_index) + 1, embedding_dim))
c = 0
for word, i in word_index.items():
    embedding_vector = embeddings_index.get(word)
    if embedding_vector is not None:
        c+=1
        embedding_matrix[i] = embedding_vector
print(c)

embedding_layer = Embedding(len(word_index) + 1,
                            embedding_dim,
                            weights=[embedding_matrix],
                            input_length=max_length,
                            trainable=False)

model = Sequential()
model.add(embedding_layer)
model.add(LSTM(64, dropout=0.2, recurrent_dropout=0.25))
model.add(Dense(1, activation='sigmoid'))

model.compile(loss='binary_crossentropy',           optimizer='adam',
metrics=['acc'])

print('Σύνοψη του μοντέλου...')
print(model.summary())


history = model.fit(X_train_pad, y_train, batch_size=32, epochs=20,
validation_data=(X_test_pad, y_test), verbose=2)

# Γραφήματα αποτελεσμάτων
acc = history.history['acc']
val_acc = history.history['val_acc']
loss = history.history['loss']
val_loss = history.history['val_loss']

epochs = range(1, len(acc)+1)

plt.plot(epochs, acc, 'g', label='Training accuracy')
plt.plot(epochs, val_acc, 'r', label='Validation accuracy')
plt.title('Training and validation accuracy')
plt.legend()

plt.figure()

plt.plot(epochs, loss, 'g', label='Training loss')
plt.plot(epochs, val_loss, 'r', label='Validation loss')
plt.title('Training and validation loss')
plt.legend()

plt.show()
```