ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΗΛΕΚΤΡΙΚΩΝ ΒΙΟΜΗΧΑΝΙΚΩΝ ΔΙΑΤΑΞΕΩΝ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΑΠΟΦΑΣΕΩΝ

Ανίχνευση Bots στο Twitter με Χρήση Συνελικτικών Νευρωνικών Δικτύων και Μοντέλων Transformer

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Ιωάννη Μιχαήλ Α. Καζελίδη

Επιβλέπων: Δημήτριος Ασκούνης
Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2023

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΗΛΕΚΤΡΙΚΩΝ ΒΙΟΜΗΧΑΝΙΚΩΝ ΔΙΑΤΑΞΕΩΝ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΑΠΟΦΑΣΕΩΝ

Ανίχνευση Bots στο Twitter με Χρήση Συνελικτικών Νευρωνικών Δικτύων και Μοντέλων Transformer

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Ιωάννη Μιχαήλ Α. Καζελίδη

Επιβλέπων: Δημήτριος Ασκούνης
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 19η Οκτωβρίου 2023.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
……	……	……
Δημήτριος Ασκούνης	Ιωάννης Ψαρράς	Χρυσόστομος Δούκας
Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2023

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΗΛΕΚΤΡΙΚΩΝ ΒΙΟΜΗΧΑΝΙΚΩΝ ΔΙΑΤΑΞΕΩΝ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΑΠΟΦΑΣΕΩΝ

(Υπογραφή)

……
Ιωάννης Μιχαήλ Α. Καζελίδης
Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Περίληψη

Στην σύγχρονη εποχή, τα μέσα κοινωνικής δικτύωσης έχουν ενσωματωθεί πλήρως στις ζωές των ανθρώπων, αποτελώντας μάλιστα πλέον αναπόσπαστο τμήμα αυτών. Η απλότητα στην χρήση, η διαδραστικότητα, καθώς και η δυνατότητα της άμεσης διάδοσης πληροφορίας που παρέχουν, καθιστούν ολοένα και συχνότερη την αξιοποίησή τους ως μέσο διάδοσης των ειδήσεων, στις μέρες μας. Ωστόσο, η διαρκώς αυξανόμενη χρήση τους στον τομέα της ενημέρωσης έχει προσελκύσει κακόβουλους χρήστες, οι οποίοι αποσκοπούν στην εκμετάλλευση των δυνατοτήτων που προσφέρουν τα μέσα κοινωνικής δικτύωσης, προς όφελός τους. Την τελευταία δεκαετία, έχει σημειωθεί μια ραγδαία αύξηση στην δραστηριότητα των κακόβουλων αυτοματοποιημένων λογαριασμών, γνωστών ως bots, στις πλατφόρμες κοινωνικής δικτύωσης, και ιδιαίτερα στην πλατφόρμα του Twitter, εγείροντας σοβαρές οικονομικές, πολιτικές, καθώς και κοινωνικές ανησυχίες. Απώτερος στόχος των bots είναι η παραπληροφόρηση των χρηστών, μέσω της διάδοσης ψευδών ειδήσεων και την χειραγώγηση του δημοσίου λόγου, ενώ χρησιμοποιούνται και για την διασπορά συνομωσιών και την προώθηση συγκεκριμένων προϊόντων. Δεδομένων των παραπάνω, κρίνεται επιτακτική η ανάγκη κατανόησης της φύσης των bots και των χαρακτηριστικών τους, για την έγκαιρη ανίχνευση και αντιμετώπισή τους.

Αντικείμενο της παρούσας διπλωματικής εργασίας αποτελεί η αντιμετώπιση του προβλήματος ανίχνευσης αυτοματοποιημένων λογαριασμών στο μέσο κοινωνικής δικτύωσης του Twitter, με χρήση Βαθιάς Μηχανικής Μάθησης και Προεκπαιδευμένων Μοντέλων Transformer. Πιο συγκεκριμένα, προτείνονται δύο μέθοδοι κατηγοριοποίησης των χρηστών του Twitter σε πραγματικούς και αυτοματοποιημένους. Κατά την πρώτη μέθοδο, αναπτύσσεται ένα μονοτροπικό μοντέλο, το οποίο, αφού πρώτα κατασκευάσει αλληλουχίες Digital DNA για κάθε χρήστη, που προκύπτουν από την δραστηριότητα του λογαριασμού του, τις μετατρέπει σε τρισδιάστατες εικόνες, τις οποίες ύστερα τροφοδοτεί σε προεκπαιδευμένα Συνελικτικά Νευρωνικά Δίκτυα, προς την εύρεση αυτού που τις κατηγοριοποιεί βέλτιστα. Κατά την δεύτερη μέθοδο, αναπτύσσεται ένα πολυτροπικό μοντέλο, το οποίο αξιοποιεί τόσο τις τρισδιάστατες εικόνες που αντιπροσωπεύουν την δραστηριότητα του λογαριασμού του χρήστη, όσο και την περιγραφή του λογαριασμού του. Για τις αναπαραστάσεις εισόδου των εικόνων και του κειμένου χρησιμοποιήθηκαν, αντίστοιχα, το βέλτιστο μοντέλο της προηγούμενης μεθόδου, VGG16, και το προεκπαιδευμένο μοντέλο Transformer TwHIN-BERT, ενώ για την συγχώνευσή τους, επιστρατεύθηκε ένα σύνολο από μεθόδους συγχώνευσης (Concatenation, Gated Multimodal Unit και Crossmodal Attention). Εκτενή και πολυπληθή πειράματα πάνω στο σύνολο δεδομένων Cresci-2017 επιβεβαίωσαν την αποτελεσματικότητα όλων των προτεινόμενων υλοποιήσεων, με αποκορύφωμα το μοντέλο TwHIN-BERT + VGG16 (Cross-Modal Attention), που αξιοποιεί εικόνες που βασίζονται στο περιεχόμενο των tweets του χρήστη, το οποίο επέτυχε επίδοση 99.98% στην μετρική Accuracy, υπερτερώντας ακόμα και μερικών εκ των state-of-the-art προσεγγίσεων που έχουν δημοσιευθεί έως τώρα από την επιστημονική κοινότητα προς την αντιμετώπιση του ίδιου προβλήματος.

Λέξεις – Κλειδιά: Μέσα Κοινωνικής Δικτύωσης, Twitter, Ανίχνευση Bots, Μηχανική Μάθηση, Βαθιά Μάθηση, Τεχνητή Νοημοσύνη, Επεξεργασία Φυσικής Γλώσσας, Συνελικτικά Νευρωνικά Δίκτυα, Μοντέλα Transformer, Μέθοδοι Συγχώνευσης, Ψηφιακό DNA

Abstract

In the modern era, social media have been fully integrated into people's lives, even becoming an indispensable part of them. The ease of use, the interactivity, and the ability to quickly disseminate information, that they provide, make their use as a means of news dissemination increasingly common in our days. However, the constantly increasing use of social media for spreading news has attracted malicious users, who aim to exploit the possibilities offered by social media platforms for their benefit. Over the past decade, there has been a rapid increase in the activity of malicious automated accounts, known as bots, on social media platforms, especially on Twitter, raising serious economic, political, and social concerns. The ultimate goal of these bots is to misinform users by spreading fake news and manipulating public discourse, while also being used to disperse conspiracies and promote specific products. Given the above, there is an urgent need to understand the nature of bots and their characteristics for their timely detection and mitigation.

The main object of this diploma thesis is the detection of automated accounts on the Twitter social media platform, using Deep Learning and Pretrained Transformer Models. More specifically, two methods for classifying Twitter users into legitimate and automated accounts are proposed. In the first method, a unimodal model is developed, which first constructs Digital DNA sequences for each user, based on their account activity, and then transforms them into three-dimensional images. These images are then fed into pretrained Convolutional Neural Networks (CNNs), to find the one that best classifies the users. In the second method, a multimodal model is developed, which utilizes both the three-dimensional images, representing the account activity of the user, and the account description. For the input representations of the images and the text, the optimal model from the previous method, VGG16, and the pretrained Transformer model TwHIN-BERT, respectively, were used. To merge them, a set of fusion methods (Concatenation, Gated Multimodal Unit, and Crossmodal Attention) was employed. Numerous and extensive experiments on the Cresci-2017 dataset confirmed the effectiveness of all proposed approaches, with the TwHIN-BERT + VGG16 (Cross-Modal Attention) model, with images based on the content of tweets, achieving an Accuracy of 99.98%, surpassing even some of the state-of-the-art approaches published by the scientific community to address the same problem.

Keywords: Social Media, Twitter, Bots Detection, Machine Learning, Deep Learning, Artificial Intelligence, Natural Language Processing, Convolutional Neural Networks, CNNs, Transformer Models, Fusion Methods, Digital DNA

Ευχαριστίες

Με την ολοκλήρωση της διπλωματικής μου εργασίας σηματοδοτείται το πέρας των προπτυχιακών σπουδών μου στο Εθνικό Μετσόβιο Πολυτεχνείο, ενός σπουδαίου κεφαλαίου στην ζωή μου. Θα ήθελα, με την δυνατότητα που μου δίνεται στο πλαίσιο αυτό, να εκφράσω τις θερμές μου ευχαριστίες στους ανθρώπους που συνετέλεσαν, ο καθένας με τον δικό του τρόπο, στην επίτευξη του έργου αυτού.

Αρχικά, θα ήθελα να ευχαριστήσω τον κύριο Δημήτριο Ασκούνη, καθηγητή της Σχολής Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Εθνικού Μετσόβιου Πολυτεχνείου, ο οποίος υπήρξε επιβλέπων της διπλωματικής αυτής και μου έδωσε την ευκαιρία να ασχοληθώ με το συγκεκριμένο επιστημονικό πεδίο, καθώς και με ένα τόσο ενδιαφέρον και επίκαιρο θέμα. Ύστερα, οφείλω ένα μεγάλο ευχαριστώ στον Λουκά Ηλία, υποψήφιο διδάκτορα του Εργαστηρίου Συστημάτων Αποφάσεων και Διοίκησης, για την διαρκή καθοδήγηση και τις συμβουλές του, οι οποίες υπήρξαν καταλυτικές για την αποτελεσματική ολοκλήρωση της διπλωματικής αυτής.

Στην συνέχεια, θα ήθελα να ευχαριστήσω όλους μου τους φίλους, που ήταν πάντοτε εκεί για εμένα, δείχνοντας το αμείωτο ενδιαφέρον τους και την στήριξή τους, σε κάθε μου δυσκολία. Ιδιαίτερα, όμως, θα ήθελα να ευχαριστήσω τους φίλους, με τους οποίους διασταυρώθηκαν οι δρόμοι μας κατά την διάρκεια των φοιτητικών μας χρόνων. Η κοινή ακαδημαϊκή μας πορεία, οι αμέτρητες ώρες εκπόνησης εργασιών και μελέτης, και οι δυσκολίες και οι αγωνίες που μοιραστήκαμε όλα αυτά τα χρόνια, αποτελούν μόνο την αφορμή για τις καλές φιλίες που ανακαλώ στα πρόσωπά τους.

Τέλος, θα ήθελα να ευχαριστήσω από τα βάθη της καρδιάς μου την οικογένειά μου, καθώς και όσους, ακόμα και χωρίς δεσμούς αίματος, θεωρώ οικογένειά μου. Η ανιδιοτελής αγάπη τους, η υποστήριξή τους σε όλες μου τις αποφάσεις και η συμπαράστασή τους σε όλες μου τις δύσκολες στιγμές αποτελούν τα στοιχεία που διαμόρφωσαν το άτομο που είμαι σήμερα, γράφοντας αυτό το κείμενο.

Ιωάννης Μιχαήλ Καζελίδης
Αθήνα, Οκτώβριος 2023

Πίνακας περιεχομένων

Ευρετήριο Πινάκων

Ευρετήριο Σχημάτων

Κεφάλαιο 1

1. Εισαγωγή

1.1 Μέσα Κοινωνικής Δικτύωσης

Τα τελευταία χρόνια, με την ραγδαία ανάπτυξη της επιστήμης και της τεχνολογίας, το διαδίκτυο ενσωματώνεται ολοένα και περισσότερο στις ζωές των ανθρώπων, αποτελώντας αρκετές φορές πλέον αναπόσπαστο τμήμα αυτών, εφόσον χρησιμοποιείται τόσο για ενημέρωση και ψυχαγωγία, όσο και για επικοινωνία και αλληλεπίδραση. Με την ανάπτυξη αυτή του διαδικτύου, επήλθε και η ανάπτυξη και η διάδοση των μέσων κοινωνικής δικτύωσης (social media), τα οποία αποτελούν διαδικτυακές πλατφόρμες που επιτρέπουν στους χρήστες να αναπτύσσουν επικοινωνιακές ομάδες και κοινωνικούς δεσμούς με άλλους χρήστες, με τους οποίους μοιράζονται κοινά ενδιαφέροντα ή ιδεολογίες. Οι ιστοσελίδες κοινωνικής δικτύωσης δίνουν την δυνατότητα δημιουργίας και κοινοποίησης περιεχομένου, όπως μηνύματα, εικόνες και βίντεο, σε πραγματικό χρόνο, καθώς και αλληλεπίδρασης μεταξύ των χρηστών και ανταλλαγής πληροφοριών και ιδεών. Μέσω του περιεχομένου που κοινοποιούν οι χρήστες, όχι μόνο έχουν την δυνατότητα να εκφράζουν τις απόψεις τους πάνω σε θέματα της επικαιρότητας, αλλά και να αποτυπώνουν την συναισθηματική τους κατάσταση, καθιστώντας έτσι τα μέσα κοινωνικής δικτύωσης σημαντική πηγή πληροφοριών για την πρόωρη ανίχνευση διαταραχών ψυχικής υγείας, όπως είναι η κατάθλιψη και το άγχος (stress) [1], [2]. Οι πιο δημοφιλείς ιστοσελίδες κοινωνικής δικτύωσης είναι το Facebook, το Instagram, το TikTok, το YouTube, το LinkedIn και το Twitter.

Η πρόταση αξίας των παραπάνω μέσων κοινωνικής δικτύωσης παρουσιάζει σημαντικές διαφορές, ελκύοντας κάθε φορά χρήστες με διαφορετικές ανάγκες ή απαιτήσεις. Η πληρέστερη εμπειρία κοινωνικής δικτύωσης συναντάται στο Facebook, το οποίο προσφέρει στους χρήστες την δυνατότητα όχι μόνο να αναρτήσουν και να κοινοποιήσουν περιεχόμενο και να αλληλεπιδράσουν με άλλους χρήστες μέσω ανταλλαγής μηνυμάτων, αλλά και να δημιουργήσουν και να ενημερωθούν για ιστοσελίδες εκδηλώσεων, να ενταχθούν σε κοινότητες αγοραπωλησιών, καθώς και να ψυχαγωγηθούν με εφαρμογές παιχνιδιών. Το TikTok και το Instagram τοποθετούν την ψυχαγωγία στο επίκεντρο της πρότασης αξίας τους, επιτρέποντας τον διαμοιρασμό σύντομου, δημιουργικού και ορισμένες φορές ενημερωτικού περιεχομένου. Το YouTube επιτρέπει την δημιουργία, κοινοποίηση και αναζήτηση οπτικοακουστικού περιεχομένου στους χρήστες του, και πλέον θεωρείται από τα κατ' εξοχήν μέσα κοινωνικής δικτύωσης για ενημέρωση, ψυχαγωγία και εκπαίδευση. Το LinkedIn επικεντρώνεται στην κοινωνική δικτύωση μεταξύ επαγγελματιών, επιχειρήσεων και χρηστών που είναι ενεργοί στην αγορά εργασίας, προωθώντας την προβολή της ακαδημαϊκής και επαγγελματικής τους δραστηριότητας και διευκολύνοντας με αυτόν τον τρόπο την αναζήτηση και κάλυψη των διαθέσιμων θέσεων εργασίας. Τέλος, το Twitter, με το οποίο και θα ασχοληθούμε εκτεταμένα στα πλαίσια της διπλωματικής αυτής, προωθεί τον δημόσιο διάλογο και την ελευθερία έκφρασης λόγου, συμβάλλοντας στην άμεση και έγκαιρη ενημέρωση για θέματα που απασχολούν την επικαιρότητα.

1.2 Twitter

Το Twitter ιδρύθηκε στις 21 Μαρτίου του 2006 από τους Jack Dorsey, Noah Glass, Biz Stone και Evan Williams, και ξεκίνησε την λειτουργία του τον Ιούλιο του ίδιου έτους. Ο αρχικός στόχος ήταν η δημιουργία μιας πλατφόρμας όπου οι χρήστες θα μπορούν να στέλνουν σύντομες, γρήγορες ενημερώσεις σχετικά με τις δραστηριότητές τους, και αυτός ήταν και ο λόγος πίσω από τον περιορισμό του μεγέθους του "tweet", όπως και αποκαλούνται οι δημοσιεύσεις των χρηστών στον συγκεκριμένο ιστότοπο. Αυτό που έκανε το Twitter ιδιαίτερα δημοφιλές ήταν αυτό το επιτρεπτό άνω όριο χαρακτήρων που έχει για κάθε tweet. Αρχικά, το μήκος του περιεχομένου των tweets ήταν 140 χαρακτήρες, αλλά τον Νοέμβρη του 2017 ο περιορισμός αυτός αυξήθηκε σε 280 χαρακτήρες, επιτρέποντας με αυτόν τον τρόπο περισσότερη πληροφορία σε κάθε ανάρτηση.

Στον χρήστη, πέρα από την δυνατότητα ανάρτησης ενός tweet, το οποίο μπορεί να περιλαμβάνει κείμενο, συνδέσμους, εικόνες και βίντεο, δίνεται και η δυνατότητα της αναδημοσίευσης (retweet) των tweets άλλων χρηστών, για να τα μοιραστεί με τους δικούς του ακόλουθους. Ο κάθε χρήστης ακολουθεί (Following/Friends) τα άτομα για τα οποία θέλει να ενημερώνεται για την δραστηριότητά τους, καθώς και ακολουθείται αντιστοίχως εκείνος από άλλα άτομα, τα οποία μπορούν να ενημερώνονται για την δική του δραστηριότητα (Followers). Στην αλληλεπίδραση με άλλους χρήστες έγκειται και η δυνατότητα απάντησης (reply) στα tweets άλλων χρηστών, καθώς και αναφοράς σε αυτούς (mention), πληκτρολογώντας τον χαρακτήρα "@" στο κείμενο του tweet τους, ακολουθούμενο από το όνομα του επιθυμητού χρήστη, ενώ δίνεται και η δυνατότητα στους χρήστες να μπορούν να επικοινωνήσουν και με ιδιωτικά μηνύματα, μέσω των DMs (Direct Messages).

Επιπρόσθετα, ο χρήστης μπορεί να χρησιμοποιήσει ετικέτες (hashtags), δηλαδή λέξεις ή φράσεις που έπονται του συμβόλου "#" και χρησιμοποιούνται για την κατηγοριοποίηση των tweets σε συγκεκριμένα θέματα, ενώ ταυτόχρονα μπορεί να αναζητήσει και με βάση ένα hashtag ή μια συγκεκριμένη λέξη ή φράση όλα τα tweets τα οποία το/την περιέχουν, σε αντίστροφη χρονολογική σειρά, ακόμα και αν ο ίδιος δεν ακολουθεί τους χρήστες που δημοσίευσαν τα συγκεκριμένα tweets. Μάλιστα, δίνεται στον χρήστη και η δυνατότητα να παρακολουθήσει τις τάσεις (trends), δηλαδή τις τάσεις θεμάτων και ετικετών που είναι αυτήν την στιγμή δημοφιλή, και να παρακολουθήσει την συζήτηση πάνω σε αυτά. Τα trends καθορίζονται από την συνδυασμένη δραστηριότητα των χρηστών στην πλατφόρμα. Ο αλγόριθμος του Twitter παρακολουθεί συνεχώς την δραστηριότητα και τα δεδομένα των χρηστών, αναλύοντας την συχνότητα των αναρτήσεων, των αναφορών, των σχολίων, των retweets και άλλων δεικτών, για να μπορέσει να προσδιορίσει τα θέματα τα οποία έχουν αυξημένη δημοτικότητα και τα οποία έχουν την δυνατότητα να γίνουν trends. Τέλος, παρέχεται στον χρήστη η δυνατότητα να δημιουργήσει λίστες (lists), για να οργανώσει τους λογαριασμούς που ακολουθεί σε διάφορες κατηγορίες ή ομάδες. Ουσιαστικά, οι λίστες είναι ένας τρόπος διαχείρισης και κατηγοριοποίησης του περιεχομένου που βλέπει ένας χρήστης στο χρονολόγιό του.

Το Twitter εξελίχθηκε από μια απλή πλατφόρμα κοινωνικής δικτύωσης σε ένα ισχυρό εργαλείο για την διάδοση των νέων, την ενημέρωση, την ψυχαγωγία, την δικτύωση και την επικοινωνία. Τον Οκτώβριο του 2022, και ύστερα από την μεγάλη απήχηση που είχε γνωρίσει σαν πλατφόρμα, εξαγοράστηκε από τον Elon Musk, έναν από τους πλουσιότερους ανθρώπους

στον πλανήτη. Έκτοτε, το Twitter έχει υποστεί αρκετές διαφοροποιήσεις, τόσο στο προσωπικό του, όσο και στην γενικότερη πολιτική λειτουργίας του, προκαλώντας αρκετές φορές πλέον αντιδράσεις από το κοινό και τους χρήστες που το χρησιμοποιούν ενεργά.

1.3 Το Πρόβλημα των Bots στο Twitter

Αναλογιζόμενοι τα παραπάνω, λόγω της διαδραστικότητας, της απλότητας στην χρήση, καθώς και της ολοένα και συχνότερης επιλογής τους ως μέσο διάδοσης ειδήσεων στην σύγχρονη εποχή, τα μέσα κοινωνικής δικτύωσης όχι μόνο έχουν ενσωματωθεί πλήρως στην καθημερινότητα των ανθρώπων, αλλά αποτελούν πλέον και το κύριο μέσο για την ενημέρωσή τους. Μέσω των κοινωνικών δικτύων, οι πληροφορίες και τα νέα διαδίδονται στους χρήστες ταχύτατα, άμεσα και με ελάχιστο κόστος. Ωστόσο, η αύξηση της χρήσης των μέσων κοινωνικής δικτύωσης στον τομέα της ενημέρωσης και η δυνατότητα άμεσης διάδοσης της πληροφορίας, έχουν προσελκύσει κακόβουλους χρήστες, οι οποίοι στοχεύουν στην εκμετάλλευση των δυνατοτήτων που προσφέρουν τα μέσα κοινωνικής δικτύωσης προς όφελός τους.

Την τελευταία δεκαετία έχει σημειωθεί μια ραγδαία αύξηση της έντασης της δραστηριότητας των “bots” στις πλατφόρμες κοινωνικής δικτύωσης, και ιδιαίτερα στο Twitter, εγείροντας σοβαρές οικονομικές, πολιτικές, καθώς και κοινωνικές ανησυχίες. Ως bots λογίζονται σύνθετες προγραμματισμένες οντότητες (αυτοματοποιημένοι χρήστες) που ενεργούν αυτόνομα και αυτοματοποιημένα στο διαδίκτυο, κατά τρόπο που μιμείται τον άνθρωπο, δημιουργώντας και δημοσιεύοντας περιεχόμενο, εξυπηρετώντας σκοπούς χειραγώγησης της πληροφορίας, διατάραξης της κοινωνικής συνοχής και διαμόρφωσης της κοινής γνώμης.

Η φύση και ο σκοπός των bots μεταβάλλονται διαρκώς, αξιοποιώντας τις νέες δυνατότητες που παρέχει η εξέλιξη της τεχνολογίας και ενσωματώνοντας ανθρώπινα συμπεριφορικά χαρακτηριστικά, έχοντας ως απώτερο στόχο την επέκταση του πεδίου της δράσης τους, και την θωράκισή τους προς αποφυγή της ανίχνευσής τους. Το πεδίο δράσης των bots δεν περιορίζεται στα στενά πλαίσια της παραπληροφόρησης και της προπαγάνδας, καθώς χρησιμοποιούνται και για την διάδοση ψευδών ειδήσεων, την χειραγώγηση του δημοσίου λόγου, του χρηματιστηρίου και της αγοράς (μέσω της καλλιέργειας ευνοϊκού ή δυσμενούς επενδυτικού κλίματος, αποσκοπώντας σε κέρδη μέσα από ανάλογες επενδυτικές κινήσεις), την διαμόρφωση της κοινής γνώμης, καθώς και την διαφήμιση και προώθηση συγκεκριμένων προϊόντων. Μάλιστα, χρησιμοποιούνται και για την διασπορά συνομωσιών, όπως συνέβη την περίοδο της πανδημίας του Covid-19, οι οποίες αναφέρονταν στην πηγή προέλευσης του ιού και στις επιπτώσεις του εμβολιασμού στην ανθρώπινη υγεία. Με αυτόν τον τρόπο, τα bots διαβρώνουν την εμπιστοσύνη των ανθρώπων και καλλιεργούν ένα κλίμα αμφισβήτησης, με αποτέλεσμα το περιεχόμενο της πλατφόρμας του Twitter (και γενικότερα των μέσων κοινωνικής δικτύωσης στην σύγχρονη εποχή) να αντιμετωπίζεται με επιφυλακτικότητα.

1.4 Συνεισφορά Διπλωματικής

Δεδομένων των παραπάνω, κρίνεται επιτακτική η ανάγκη κατανόησης της φύσης των bots, των χαρακτηριστικών τους και του σκοπού που κάθε φορά επιτελούν, με στόχο την επιστράτευση και ανάπτυξη κατάλληλων τεχνικών και εργαλείων, ικανών να ανιχνεύουν και να απομονώνουν αυτούς τους κακόβουλους αυτοματοποιημένους λογαριασμούς, διασφαλίζοντας με αυτόν τον τρόπο την ενημέρωση και την αλληλεπίδραση των χρηστών, παρέχοντάς τους μια αξιόπιστη εμπειρία χρήσης.

Προς την κατεύθυνση αυτή, έχει διεξαχθεί ένα πλήθος ερευνών από την επιστημονική κοινότητα, οι οποίες ωστόσο εμφανίζουν είτε περιορισμούς, είτε πολυπλοκότητα στην διαδικασία ανάπτυξης της υλοποίησής τους. Πιο συγκεκριμένα, έχουν δημοσιευθεί αρκετές έρευνες που εκπαιδεύουν ρηχά (shallow) μοντέλα μηχανικής μάθησης και προτείνουν μεθόδους εξαγωγής χαρακτηριστικών (feature extraction) [3], [4]. Ωστόσο, η τεχνική της εξαγωγής χαρακτηριστικών συνιστά μια χρονοβόρα διαδικασία, η οποία συνήθως απαιτεί υψηλό βαθμό τεχνογνωσίας πάνω στο συγκεκριμένο αντικείμενο, ενώ υπάρχει και το ενδεχόμενο σε μερικές περιπτώσεις προβλημάτων να μην είναι δυνατή η εύρεση του βέλτιστου συνόλου χαρακτηριστικών, που οδηγεί στο αποδοτικότερο μοντέλο. Παράλληλα, έχουν προταθεί και προσεγγίσεις Βαθιάς Μάθησης για την ανίχνευση των αυτοματοποιημένων λογαριασμών στο Twitter. Όμως, οι προσεγγίσεις αυτές είτε χρησιμοποιούν ως είσοδο ένα μεγάλο αριθμό χαρακτηριστικών, όπως metadata που σχετίζονται με τους χρήστες και τα tweets τους, μεταξύ άλλων, είτε επιστρατεύουν εμφυτεύματα λέξεων, όπως GloVe Embeddings, και εκπαιδεύουν Βαθιά Νευρωνικά Δίκτυα, όπως δίκτυα LSTM (Long Short-Term Memory), όπως στα [5] και [6], αντί να επιστρατεύουν Γλωσσικά Μοντέλα που βασίζονται σε Transformers, τα οποία απαιτούν λιγότερο χρόνο εκπαίδευσης, και εμφανίζουν state-of-the-art επιδόσεις σε ένα πλήθος διαφορετικών πεδίων. Βέβαια, ακόμα και μερικές υλοποιήσεις που επιστρατεύουν τέτοια Γλωσσικά Μοντέλα, όπως BERT, RoBERTa και άλλα, δεν μπορούν να εφαρμοστούν σε εισόδους που χρησιμοποιούν διαφορετική γλώσσα από την αγγλική, περιορίζοντας με αυτόν τον τρόπο αρκετά το πλήθος των πεδίων εφαρμογής τους. Επιπρόσθετα, πρόσφατα έχουν εισαχθεί και προσεγγίσεις που βασίζονται σε γράφους για την ανίχνευση των bots. Παρ' όλα αυτά, και σε αυτήν την περίπτωση χρησιμοποιούνται τεχνικές εξαγωγής χαρακτηριστικών για την δημιουργία των διανυσματικών αναπαραστάσεων των χρηστών με βάση τα χαρακτηριστικά τους, οι οποίες αποτελούν τους κόμβους των γράφων [7], ενώ συχνά δημιουργούνται γράφοι μεγάλης κλίμακας, με αποτέλεσμα το μοντέλο να απαιτεί όχι μόνο αρκετό χρόνο για να εκπαιδευθεί, αλλά και πρόσβαση σε υπολογιστικούς πόρους.

Για την αντιμετώπιση των παραπάνω περιορισμών και την ελαχιστοποίηση της πολυπλοκότητας των υλοποιήσεων, στην παρούσα διπλωματική εργασία αναλύεται η ανίχνευση και η κατηγοριοποίηση χρηστών σε humans (πραγματικούς χρήστες) και bots (αυτοματοποιημένους χρήστες), με την χρήση Βαθιάς Μηχανικής Μάθησης και Προεκπαιδευμένων Μοντέλων Transformer. Το μέσο κοινωνικής δικτύωσης από το οποίο αντλούνται τα δεδομένα της εκπαίδευσης, με τα οποία εκπαιδεύονται τα νευρωνικά δίκτυα, και στο οποίο επιτελείται τελικά η αξιολόγησή τους, είναι το Twitter. Συγκεκριμένα, το σύνολο δεδομένων που χρησιμοποιήθηκε για την εκπαίδευση των μοντέλων και την κατηγοριοποίηση των χρηστών ονομάζεται Cresci-2017, και προτάθηκε από τους Cresci et al. κατά την δημοσίευση ενός paper τους [8], το οποίο αποτελεί μια από τις σημαντικότερες συνεισφορές στον τομέα της ανίχνευσης και αντιμετώπισης των bots στα μέσα κοινωνικής δικτύωσης και

έχει αναγνωριστεί από την επιστημονική κοινότητα ως σημαντική πηγή πληροφοριών και ανάλυσης.

Αναλυτικότερα, στα πλαίσια της διπλωματικής αυτής, αφού κάναμε μια ολοκληρωμένη μελέτη της υπάρχουσας βιβλιογραφίας, προτείνουμε δύο μεθόδους κατηγοριοποίησης των χρηστών του Twitter. Στην πρώτη μέθοδο, αναπτύξαμε ένα μονοτροπικό μοντέλο, το οποίο είναι ικανό να κατηγοριοποιήσει τους χρήστες σε πραγματικούς και αυτοματοποιημένους, χρησιμοποιώντας αποκλειστικά εισόδους που έχουν την μορφή εικόνας, η οποία προκύπτει από την δραστηριότητα του λογαριασμού τους. Πιο συγκεκριμένα, χρησιμοποιούμε τα tweets των χρηστών, και με βάση τόσο τον *τύπο* των tweets, όσο και το *περιεχόμενό* τους, δημιουργούμε δύο αλληλουχίες Digital DNA (Ψηφιακού DNA), για τον εκάστοτε χρήστη. Ύστερα, μετατρέπουμε τις αλληλουχίες Digital DNA του χρήστη σε εικόνες που αποτελούνται από 3 κανάλια, οι οποίες αντιπροσωπεύουν την δραστηριότητα του λογαριασμού του. Οι εικόνες αυτές τροφοδοτούνται σε ένα σύνολο από Προεκπαιδευμένα Συνελικτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks – CNNs), των οποίων συγκρίνουμε την επίδοση με βάση διάφορες γνωστές μετρικές αξιολόγησης, για την εύρεση αυτού που τις κατηγοριοποιεί βέλτιστα. Στην δεύτερη μέθοδο, αναπτύξαμε πολυτροπικά μοντέλα, τα οποία είναι ικανά να κατηγοριοποιήσουν τους χρήστες του Twitter, χρησιμοποιώντας τόσο εισόδους που έχουν την μορφή εικόνας, η οποία προκύπτει από την δραστηριότητα του λογαριασμού του χρήστη (τόσο από τον τύπο των tweets όσο και από το περιεχόμενό τους), όσο και εισόδους που έχουν την μορφή κειμένου, που αποτελεί την περιγραφή που έχει θέσει στον λογαριασμό του ο χρήστης. Για τις εισόδους που έχουν μορφή εικόνας αξιοποιήσαμε το μοντέλο που εμφάνισε τις καλύτερες επιδόσεις στο πρόβλημα κατηγοριοποίησης της εικόνας του χρήστη, δηλαδή το μοντέλο VGG16, ενώ για την περιγραφή του λογαριασμού του χρήστη αξιοποιήσαμε το Προεκπαιδευμένο Μοντέλο Transformer TwHIN-BERT (Twitter Heterogeneous Information Network - Bidirectional Encoder Representations from Transformers). Για την συγχώνευση των δύο αναπαραστάσεων, επιστρατεύσαμε ένα σύνολο από Fusion Methods (Concatenation, Gated Multimodal Unit και Crossmodal Attention), αναπτύσσοντας έτσι υλοποιήσεις που επέτυχαν εξαιρετικές επιδόσεις, οι οποίες ξεπέρασαν σε μερικές περιπτώσεις επιδόσεις state-of-the-art προσεγγίσεων.

Συνοπτικά, οι κύριες συνεισφορές της παρούσας διπλωματικής εργασίας παρουσιάζονται παρακάτω:

- Κατασκευάζουμε δύο αλληλουχίες Ψηφιακού DNA (Digital DNA) για κάθε χρήστη, βασιζόμενοι στην δραστηριότητα του λογαριασμού του, δηλαδή στον τύπο των tweets που δημοσιεύει, και στο περιεχόμενό τους, και τις μετατρέπουμε σε τρισδιάστατες εικόνες.
- Επιστρατεύουμε και αξιολογούμε ένα πλήθος Προεκπαιδευμένων Συνελικτικών Νευρωνικών Δικτύων (Convolutional Neural Networks – CNNs) για την κατηγοριοποίηση των τρισδιάστατων εικόνων που προκύπτουν από τις αλληλουχίες Digital DNA.
- Ύστερα από εκτεταμένη αναζήτηση στην επιστημονική βιβλιογραφία, θεωρούμε πως η διπλωματική αυτή αποτελεί την πρώτη έρευνα που εφαρμόζει fine-tuning στο προεκπαιδευμένο μοντέλο TwHIN-BERT, για την δημιουργία μονοτροπικού μοντέλου που αξιοποιεί μόνο την περιγραφή του λογαριασμού του χρήστη για την κατηγοριοποίηση των χρηστών του Twitter σε πραγματικούς και αυτοματοποιημένους, τα αποτελέσματα του οποίου παρουσιάζονται στο Κεφάλαιο 6 της παρούσας εργασίας.

- Αντίστοιχα, ύστερα από εκτεταμένη αναζήτηση στην επιστημονική βιβλιογραφία, θεωρούμε πως η διπλωματική αυτή αποτελεί την πρώτη έρευνα που εισάγει και αξιοποιεί multimodal μοντέλα για την ανίχνευση κοινωνικών spambots στο Twitter, χρησιμοποιώντας αποκλειστικά την περιγραφή του λογαριασμού του χρήστη και τρισδιάστατες εικόνες, οι οποίες αντιπροσωπεύουν την δραστηριότητα του λογαριασμού του.
- Επιστρατεύουμε ένα σύνολο από μεθόδους συγχώνευσης, και αφότου εξετάσουμε την συμπεριφορά τους και τον τρόπο με τον οποίο αξιοποιούν τις αναπαραστάσεις εικόνας και κειμένου για την τελική κατηγοριοποίηση του εκάστοτε χρήστη, αξιολογούμε και συγκρίνουμε τις επιδόσεις τους.

1.5 Δομή Διπλωματικής

Η Διπλωματική Εργασία διαρθρώνεται σε επτά διακριτά κεφάλαια. Στο **Κεφάλαιο 1** δόθηκε μια σύντομη εισαγωγή και περιγραφή του προβλήματος, το οποίο εξετάζει σε βάθος η εν λόγω διπλωματική, ενώ παρουσιάστηκε και η συνεισφορά της, σε μια προσπάθεια αντιμετώπισης του προβλήματος και ελάττωσης των επιπτώσεων και των κινδύνων που το ίδιο μπορεί να επιφέρει. Στο **Κεφάλαιο 2** παρουσιάζεται μια πλήρης βιβλιογραφική ανασκόπηση των μεθόδων, των εργαλείων, των τεχνικών, των χαρακτηριστικών και των συνόλων δεδομένων που έχουν αναπτυχθεί και επιστρατευθεί τα τελευταία χρόνια από την επιστημονική κοινότητα για την ανίχνευση των αυτοματοποιημένων λογαριασμών στο Twitter και την αποτελεσματική αντιμετώπισή τους. Στο **Κεφάλαιο 3** παρέχεται ένα πλήρες θεωρητικό υπόβαθρο, τόσο γενικών αρχών και εννοιών Μηχανικής Μάθησης, Βαθιάς Μάθησης και Νευρωνικών Δικτύων, για την κατανόηση του γενικότερου και ολοκληρωμένου πλαισίου τους, όσο και συγκεκριμένων τεχνολογιών, μεθόδων και μοντέλων, τα οποία χρησιμοποιούνται για την εκπόνηση της παρούσας διπλωματικής εργασίας. Στο **Κεφάλαιο 4** παρουσιάζεται αναλυτικά το σύνολο δεδομένων (dataset) το οποίο χρησιμοποιήθηκε, καθώς και η προεπεξεργασία στην οποία υποβλήθηκαν τα υποσύνολα δεδομένων που το ίδιο εμπεριέχει, για να προκύψει εν τέλει το σύνολο δεδομένων το οποίο και χρησιμοποιήθηκε για την εκπαίδευση των νευρωνικών δικτύων και την αξιολόγησή τους. Στο **Κεφάλαιο 5** παρουσιάζονται οι μέθοδοι, τα μοντέλα, η πειραματική διάταξη και τα αποτελέσματα του πρώτου σκέλους της υλοποίησής μας, που αφορά την μονοτροπική ανίχνευση bots στο Twitter με χρήση μόνο εικόνας. Στο **Κεφάλαιο 6** παρουσιάζονται οι μέθοδοι, οι τεχνικές, τα μοντέλα, η πειραματική διάταξη και τα αποτελέσματα του δεύτερου σκέλους της υλοποίησής μας, που αφορά την πολυτροπική (multimodal) ανίχνευση bots στο Twitter, με χρήση τόσο εικόνας (visual modality), όσο και κειμένου (textual modality). Τέλος, το **Κεφάλαιο 7** παρέχει μια τελική σύνοψη της εργασίας, ενώ καταγράφονται τα βασικά συμπεράσματα που προέκυψαν, καθώς και προτάσεις για μελλοντικές επεκτάσεις της έρευνας που διεξήχθη στα πλαίσια της εργασίας αυτής.

Κεφάλαιο 2

2. Συναφής Βιβλιογραφία

Στο κεφάλαιο αυτό παρουσιάζεται μια πλήρης βιβλιογραφική ανασκόπηση των μεθόδων, των εργαλείων, των τεχνικών, των χαρακτηριστικών και των συνόλων δεδομένων που έχουν αναπτυχθεί και επιστρατευθεί τα τελευταία χρόνια για την ανίχνευση των αυτοματοποιημένων λογαριασμών στο Twitter και την αποτελεσματική αντιμετώπισή τους, καθώς και των μετρικών βάσει των οποίων διεξήχθη η αξιολόγηση των επιδόσεών τους. Η βιβλιογραφία παρουσιάζεται κατηγοριοποιημένη με βάση το είδος της προσέγγισης που ακολουθήθηκε. Συγκεκριμένα, στις επόμενες ενότητες, θα εξετάσουμε βιβλιογραφία η οποία είναι βασισμένη σε παραδοσιακούς αλγορίθμους Μηχανικής Μάθησης (Traditional Machine Learning Algorithms), σε Βαθιά Μάθηση και Μοντέλα Transformer (Deep Learning και Transformer-based Approaches), σε Γράφους (Graph-Based Approaches) και σε Μη-Επιβλεπόμενη Μάθηση (Unsupervised Learning).

2.1 Παραδοσιακοί Αλγόριθμοι Μηχανικής Μάθησης

Οι Dukić et al. [9], χρησιμοποιώντας μόνο τα χαρακτηριστικά που προκύπτουν από τα tweets των χρηστών, ανέπτυξαν τόσο ένα μοντέλο επιβλεπόμενης μηχανικής μάθησης, το μοντέλο Λογιστικής Παλινδρόμησης (Logistic Regression Model), όσο και ένα Βαθύ Νευρωνικό Δίκτυο Εμπρόσθιας Τροφοδότησης (Feedforward Deep Neural Network), ακολουθώντας δύο διαφορετικές προσεγγίσεις εξαγωγής embeddings. Στην πρώτη προσέγγιση, που είναι και αυτή που μας απασχολεί στην συγκεκριμένη ενότητα, χρησιμοποίησαν word2vec embeddings. Σαν μετρική αξιολόγησης των παραπάνω χρησιμοποιήθηκε το Weighted F1-Score, ενώ σαν dataset τα tweets αγγλικής γλώσσας από το σύνολο δεδομένων PAN-19 [10].

Ο Narayan [11] χρησιμοποίησε τα χαρακτηριστικά του λογαριασμού του χρήστη για να κατηγοριοποιήσει τους χρήστες σε πραγματικούς ή αυτοματοποιημένους. Ως ταξινομητές χρησιμοποιήθηκαν τα Δέντρα Αποφάσεων (Decision Trees), Random Forest και Multinomial Naive Bayes, ενώ η επίδοσή τους αξιολογήθηκε με βάση την μετρική Accuracy. Σαν σύνολο δεδομένων δεν επιστρατεύθηκε κάποιο υπάρχον, αλλά χρησιμοποιήθηκε το Twitter-API για την εξαγωγή των χαρακτηριστικών και των tweet των χρηστών, ενώ πριν χρησιμοποιηθούν τα δεδομένα αυτά, φιλτραρίστηκαν και καθαρίστηκαν αναλόγως.

Οι Fagni et al. [12], χρησιμοποιώντας μόνο τα tweets, καθώς και τα χαρακτηριστικά που προκύπτουν από το περιεχόμενο που εξάγεται από τα συμφραζόμενά τους, ακολούθησαν τέσσερις προσεγγίσεις, με σκοπό να εξετάσουν την αποτελεσματικότητα ενός συνόλου από τεχνικές Μηχανικής Μάθησης και Βαθιάς Μάθησης. Στην πρώτη προσέγγιση, που μας αφορά στα πλαίσια της ενότητας αυτής, χρησιμοποίησαν τις τεχνικές Bag Of Words (BoW) και Term Frequency - Inverse Document Frequency (TF-IDF), και τους ταξινομητές Logistic

Regression, Random Forest και SVM. Ως μετρικές αξιολόγησης χρησιμοποιήθηκαν τα Precision, Recall, F1 και Accuracy, ενώ σαν σύνολο δεδομένων χρησιμοποιήθηκε το "TweepFake - A Twitter Deep Fake Dataset", το οποίο προτάθηκε από τους ίδιους τους συγγραφείς στο εν λόγω paper. Για την δημιουργία του dataset αυτού, συλλέξαν tweets από ένα σύνολο από 23 bots, που μιμούνταν 17 λογαριασμούς πραγματικών χρηστών, ενώ συλλέξαν και τυχαία tweets από αυτούς τους πραγματικούς λογαριασμούς, τους οποίους τα bot μιμούνταν, για να δημιουργήσουν συνολικά ένα ισορροπημένο dataset από 25.572 tweets (μισά από πραγματικούς χρήστες και μισά από bots), το οποίο και κοινοποίησαν για δημόσια χρήση στην πλατφόρμα Kaggle.

Οι Ramalingaiah et al. [13], χρησιμοποιώντας ως σύνολο δεδομένων το "Detecting Twitter Bot Data" [14], ένα δημοσίως διαθέσιμο σύνολο δεδομένων στην πλατφόρμα Kaggle, το οποίο περιέχει διάφορα χαρακτηριστικά του λογαριασμού του χρήστη, εφάρμοσαν τεχνικές feature engineering και feature extraction (εξαγωγής χαρακτηριστικών) πάνω σε αυτά. Στα πλαίσια της πρώτης τεχνικής, χρησιμοποιείται ένα Bag Of Bots' Words, ενώ χρησιμοποιώντας τα εξαγόμενα χαρακτηριστικά γίνεται εφαρμογή των ταξινομητών Decision Trees, Logistic Regression, K Nearest Neighbors (KNN), Naive Bayes καθώς και ενός καινούργιου ταξινομητή που προτείνεται στο εν λόγω paper. Ως μετρικές αξιολόγησης χρησιμοποιούνται τα Accuracy, ROC (Receiver Operating Characteristics) και AUC (Area Under Curve).

Οι Shukla et al. [15] πρότειναν μια μέθοδο η οποία ασχολείται με την στατική ανάλυση του λογαριασμού του χρήστη. Συγκεκριμένα, μετά την αρχική προεπεξεργασία των δεδομένων, η οποία περιλάμβανε κλιμακοποίηση (scaling) των αριθμητικών τιμών και εφαρμογή κωδικοποίησης Weight of Evidence (WoE) στα κατηγορικά χαρακτηριστικά, χρησιμοποίησαν τεχνικές αυτόματης επιλογής χαρακτηριστικών (PCA, Univariate feature selection και feature selection από το μοντέλο). Τα χαρακτηριστικά αυτά τροφοδοτήθηκαν σε ταξινομητές όπως Random Forest, Adaboost και Τεχνητά Νευρωνικά Δίκτυα, και τα αποτελέσματα αυτά σε συνδυασμό με τα δεδομένα εισόδου συνδυάστηκαν χρησιμοποιώντας τον Random Forest, για την επίτευξη της καλύτερης δυνατής επίδοσης. Ως μετρικές αξιολόγησης χρησιμοποιήθηκαν τα Accuracy, Precision, Recall, F1-Score και AUC, ενώ σαν σύνολο δεδομένων χρησιμοποιήθηκε ένα dataset από την πλατφόρμα Kaggle, το οποίο περιέχει 37.438 διαφορετικά profile χρηστών, με 18 χαρακτηριστικά το καθένα.

Οι Pramitha et al. [16] αφότου πρώτα έκαναν προεπεξεργασία στα δεδομένα τους, εφάρμοσαν Exploratory Data Analysis (EDA) για να επιλέξουν το καταλληλότερο μοντέλο ανάλυσης και ύστερα εξέτασαν τα μοντέλα μηχανικής μάθησης Random Forest και XGBoost. Λόγω της ανισορροπίας που εμφάνισαν τα δεδομένα, χρησιμοποιήθηκε η τεχνική SMOTE (Synthetic Minority Oversampling Technique), ενώ έλαβε μέρος και ρύθμιση υπερπαραμέτρων (hyperparameter tuning), για την εύρεση του καλύτερου αλγορίθμου. Από τα 15 χαρακτηριστικά που περιείχε το dataset, αποδείχθηκε πως μόνο 3 είχαν άμεση επίδραση στο αποτέλεσμα. Σαν μετρικές αξιολόγησης χρησιμοποιήθηκαν τα Accuracy, Recall, Precision και F1-Score, ενώ σαν σύνολο δεδομένων χρησιμοποιήθηκε ένα από το Kaggle, το οποίο ωστόσο περιείχε μόνο τα ids των χρηστών και την κλάση που ανήκε ο κάθε χρήστης, και συνεπώς απαιτήθηκε χρήση του Twitter API για την ανάκτηση και των υπόλοιπων χαρακτηριστικών του λογαριασμού.

Οι Tyagi et al. [17], χρησιμοποιώντας ένα σύνολο δεδομένων που περιείχε αποκλειστικά και μόνο χαρακτηριστικά του λογαριασμού του χρήστη, και όχι πληροφορίες για το

περιεχόμενο που δημοσιεύει, εξέτασαν τις επιδόσεις των μοντέλων μηχανικής μάθησης Naive Bayes, Random Forest και Decision Trees, χρησιμοποιώντας ως μετρική αξιολόγησης την ROC. Τα χαρακτηριστικά που κυρίως συνέβαλαν στην διάκριση των χρηστών σε πραγματικούς ή αυτοματοποιημένους ήταν το πλήθος των followers και των following, το όνομα χρήστη, η περιγραφή του λογαριασμού, καθώς και αν ο λογαριασμός ήταν verified (επικυρωμένος) ή όχι.

Οι Shevtsov et al. [18], χρησιμοποιώντας το Twitter API για να αντλήσουν δεδομένα από τις αμερικάνικες προεδρικές εκλογές του 2020, κατάφεραν να συλλέξουν συνολικά 15,6 εκατομμύρια tweets και 3,2 εκατομμύρια χρήστες. Τα χαρακτηριστικά του dataset αυτού μπορούν να διακριθούν σε τέσσερις κατηγορίες: προφίλ του χρήστη, περιεχόμενο του χρήστη, χρονισμός του χρήστη, και αλληλεπίδραση του χρήστη. Στα χαρακτηριστικά αυτά εφαρμόστηκε feature selection με τις μεθόδους Lasso, Random Forest Feature Selection και Model Feature Importance (σπουδαιότητα χαρακτηριστικού), στην συνέχεια εφαρμόστηκε ρύθμιση υπερπαραμέτρων, ενώ εξετάστηκε και η επίδοση των μοντέλων Random Forest, SVM και Extreme Gradient Boosting με γνώμονα τις μετρικές αξιολόγησης PR-AUC (Area Under the Precision-Recall Curve), ROC-AUC (Receiver Operating Curve) και F1-Score.

Οι Heidari et al. [19] πρότειναν μια νέα μέθοδο ανάλυσης συναισθήματος για να εξάγουν νέα χαρακτηριστικά από το περιεχόμενο ενός tweet, και στην συνέχεια εξέτασαν την επίδοση των ταξινομητών Random Forest, SVM, Logistic Regression και ενός Feedforward Neural Network, τόσο χωρίς τα νέα χαρακτηριστικά που εξήγαγαν, όσο και με την προσθήκη τους, με σκοπό να αναδείξουν την επίδρασή τους στην επίδοση των μοντέλων αυτών. Το επιλεγμένο set αυτών των νέων χαρακτηριστικών αποτελείται από το πλήθος των ουδέτερων, θετικών και αρνητικών tweets ενός χρήστη, το άθροισμα των βαθμών πολικότητας (polarity scores) των θετικών και των αρνητικών tweets, καθώς και τον μέσο όρο θετικής και αρνητικής πολικότητας των θετικών και αρνητικών tweets αντίστοιχα. Οι μετρικές αξιολόγησης που χρησιμοποιήθηκαν είναι τα Accuracy, MCC και F1-Score, ενώ σαν σύνολο δεδομένων χρησιμοποιήθηκε των Cresci et al. [8].

Οι Fonseca Abreu et al. [20], χρησιμοποιώντας ως σύνολο δεδομένων των Cresci et al. [8], και επιλέγοντας τα βασικά χαρακτηριστικά του λογαριασμού του χρήστη μέσα από feature selection, εξέτασαν την επίδοση των ταξινομητών Random Forest, SVM, Naive Bayes και One-Class SVM (ocSVM), με γνώμονα τις μετρικές Accuracy, Recall, AUC και F1-Score. Με χρήση του ίδιου συνόλου δεδομένων, και χρησιμοποιώντας 13 χαρακτηριστικά για την περιγραφή κάθε λογαριασμού, που πήγαζαν από την χρήση και τους τύπους πληροφορίας του λογαριασμού, οι Rodríguez-Ruiz et al. [21], με γνώμονα την μετρική AUC, εξέτασαν αρχικά τις επιδόσεις δυαδικών ταξινομητών, και ύστερα πάνω στα ίδια δεδομένα τις σύγκριναν με τις επιδόσεις ταξινομητών μιας κλάσης (one-class classifiers). Πιο συγκεκριμένα, ως δυαδικοί ταξινομητές χρησιμοποιήθηκαν οι Bayes Network, J48, Random Forest, Adaboost, Bagging, KNN, Logistic Regression, MLP, Naive Bayes και SVM, ενώ σαν one-class classifiers οι Bagging-TPMiner, Bagging-RandomMiner, OCKRA (One-Class K-means with Randomly-projected features Algorithm), ocSVM και Naive Bayes.

Ο Knauth [22], χρησιμοποιώντας το dataset των Cresci et al. [8], εξήγαγε μια πληθώρα χαρακτηριστικών, τα οποία βασίζονται σε δύο βασικές κατηγορίες: στον λογαριασμό του χρήστη και στο περιεχόμενό του. Αφού πρώτα αντιμετώπισε τις ανισορροπίες που εμφάνισαν τα δεδομένα κάνοντας χρήση της μεθόδου SMOTE-ENN, χρησιμοποίησε τα χαρακτηριστικά

που είχε εξάγει για να κατασκευάσει διαφορετικά σύνολα χαρακτηριστικών, τα οποία και τροφοδότησε σε διαφορετικά μοντέλα μηχανικής μάθησης. Συγκεκριμένα, τα μοντέλα που εξετάστηκαν ήταν Logistic Regression, SVM, Random Forest, Multi-Layer Perceptron (MLP), ενώ τα καλύτερα αποτελέσματα προέκυψαν με χρήση του AdaBoost, κάνοντας χρήση των μετρικών Precision, Recall, F1-Score, Accuracy και AUC-ROC για την σύγκριση και αξιολόγηση των αποτελεσμάτων.

Οι Kosmajac & Keselj [23] βασίστηκαν στην ιδέα μοντελοποίησης των χρηστών με την μέθοδο του Digital DNA (Ψηφιακό DNA). Για να το επιτύχουν αυτό, αντιστοίχισαν κωδικούς σε κάθε tweet τους, αναλόγως τον τύπο του. Πιο συγκεκριμένα, αντιστοίχισαν τον κωδικό 8 σε tweets που ήταν retweets, τον κωδικό 16 σε tweets που ήταν reply, ενώ τον κωδικό 0 σε περίπτωση που δεν ήταν τίποτα από τα δύο (απλό tweet). Ύστερα, στους κωδικούς αυτούς πρόσθεσαν τους κωδικούς που αντιστοίχισαν βασιζόμενοι στο περιεχόμενο των tweets, δηλαδή τους αριθμούς 1, 2 και 4, σε περίπτωση που το tweet περιείχε hashtags, mentions και urls, αντίστοιχα. Έτσι, μετέτρεψαν το κάθε tweet σε έναν αριθμό, τον οποίο αριθμό στην συνέχεια μετέτρεψαν σε χαρακτήρα ASCII. Εφαρμόζοντας την διαδικασία αυτή για όλα τα tweets του κάθε χρήστη, με την χρονολογική σειρά που δημοσιεύθηκαν, δημιούργησαν μια αλληλουχία DNA για κάθε χρήστη, το λεγόμενο Digital DNA του, από την οποία εξήγαγαν n-grams, όπου το n ήταν 1, 2 ή 3. Τέλος, όρισαν 5 διαφορετικά στατιστικά μέτρα, προκειμένου να αποτυπώσουν τις διαφορές μεταξύ των δύο κλάσεων, και να εξάγουν τα χαρακτηριστικά που θα αποτελέσουν είσοδο στους αλγορίθμους μηχανικής μάθησης. Συγκεκριμένα, εξέτασαν τους Gaussian Naive Bayes, SVM, Logistic Regression, K Nearest Neighbors, Random Forest και Gradient Boosting, με μετρική αξιολόγησης την F1-Score. Για την ανάπτυξη των παραπάνω, χρησιμοποιήθηκαν τα δημόσια διαθέσιμα datasets των Cresci-2017 [8] και Varol-2017 [24].

Οι Ilias & Roussaki [3] πρότειναν δύο μεθόδους για την κατηγοριοποίηση των χρηστών σε πραγματικούς και αυτοματοποιημένους, οι οποίες βασίζονται κυρίως σε Natural Language Processing (Επεξεργασία Φυσικής Γλώσσας). Στην πρώτη μέθοδο, που είναι και αυτή που μας απασχολεί στα πλαίσια της ενότητας αυτής, αρχικά προτείνεται μια διαδικασία εξαγωγής χαρακτηριστικών (feature extraction), από την οποία προκύπτουν 71 χαρακτηριστικά για κάθε χρήστη. Στην συνέχεια, ύστερα από την εφαρμογή τεχνικών feature selection (Mutual Information, ANOVA F-Value και Chi-squared test) για την απόρριψη περιττών χαρακτηριστικών, και μετέπειτα την αντιμετώπιση ανισορροπιών που εμφανίζονται στο σύνολο δεδομένων, το υποσύνολο των χαρακτηριστικών που τελικά επιλέχθηκε τροφοδοτείται στα μοντέλα μηχανικής μάθησης SVM, Decision Trees, Random Forests, AdaBoost, Logistic Regression και K Nearest Neighbors. Ως μετρικές αξιολόγησης χρησιμοποιήθηκαν τα Precision, Recall, F-measure/F1-Score, Area Under the ROC curve (AUROC) και Accuracy, ενώ σαν σύνολα δεδομένων χρησιμοποιήθηκαν τα Cresci-2017 [8] και Social Honeypot Dataset [25].

Οι Davoudi et al. [26] χρησιμοποίησαν το Botometer (πρότερα BotOrNot), το οποίο είναι ένα δημόσια διαθέσιμο σύστημα ανίχνευσης bot που προτάθηκε από τους Davis et al. [27], σαν benchmark για την έρευνά τους. Συγκεκριμένα, σκοπός τους ήταν να αξιολογήσουν το Botometer σε σύνολα δεδομένων που είχαν συλλεχθεί με βάση το περιεχόμενό τους, το οποίο ήταν σχετικό με την υγεία, καθώς και να το επεκτείνουν για καλύτερη επίδοση και εφαρμογή στον τομέα της υγείας. Χρησιμοποίησαν το score του Botometer για κάθε χρήστη (score που δείχνει πόσο πιθανό είναι ένας χρήστης να είναι πραγματικός ή αυτοματοποιημένος) ως χαρακτηριστικό για την εκπαίδευση ενός ταξινομητή Gradient Boosting, ενώ αντιμετώπισαν

και τις ανισορροπίες του συνόλου δεδομένων χρησιμοποιώντας την τεχνική SMOTE. Για την εκπαίδευση του ταξινομητή, χρησιμοποιήθηκε μια πληθώρα χαρακτηριστικών, τα οποία δεν χρησιμοποιούνται από το Botometer, ενώ σαν μετρικές αξιολόγησης της επίδοσης χρησιμοποιήθηκαν οι Precision, Recall και F1-Score.

Οι Daouadi et al. [28] χρησιμοποίησαν το Twitter API σε συνδυασμό με δύο έτοιμα σύνολα δεδομένων, που δημοσιεύθηκαν από τους Subrahmanian et al. [29] και Lee et al. [25], με σκοπό να κρατήσουν μόνο τους χρήστες που ήταν ενεργοί, και από αυτούς να κρατήσουν μόνο τα 200 πιο πρόσφατα tweets τους. Προτείνουν μια στατιστική προσέγγιση, η οποία ανιχνεύει τα bots χρησιμοποιώντας τα metadata των profile των χρηστών και των δημοσιεύσεών τους. Για κάθε ένα από τα datasets, κατασκεύασαν τρία μοντέλα: ένα βασιζόμενο μόνο στα metadata του profile του χρήστη, ένα βασιζόμενο μόνο στα metadata των δημοσιεύσεών του, και ένα βασιζόμενο και στα δύο από κοινού. Εξέτασαν περισσότερους από 30 παραδοσιακούς αλγορίθμους επιβλεπόμενης μηχανικής μάθησης, μεταξύ των οποίων οι Bagging, Multi-Layer Perceptron, AdaBoost, Random Forest, Simple Logistic και άλλοι, ενώ όλοι χρησιμοποιήθηκαν με τις βασικές παραμέτρους τους, για να είναι δυνατή η σύγκριση μεταξύ τους. Επίσης, στην προσπάθειά τους να εμπλουτίσουν τα δεδομένα τους με περισσότερα δείγματα πραγματικών και αυτοματοποιημένων λογαριασμών, χρησιμοποίησαν την τεχνική SMOTE. Σαν μετρικές αξιολόγησης χρησιμοποιήθηκαν οι Precision, Recall, F-measure, Accuracy και AUC.

Τέλος, οι Rodrigues et al. [30] πρότειναν ένα σύστημα το οποίο διακρίνει αν ένα tweet θεωρείται “spam” ή “ham” και αξιολογεί το συναίσθημα που εμπεριέχεται σε αυτό. Για την ανίχνευση των spam tweets χρησιμοποιήθηκε ένα SMS dataset, που περιέχει 4.825 ham tweets και 747 spam tweets, ενώ για την ανάλυση του συναισθήματος χρησιμοποιήθηκε ένα μεγάλο dataset από την πλατφόρμα Kaggle που περιέχει 31.015 tweets, εκ των οποίων 12.548 θεωρούνται neutral (ουδέτερα), 9.685 positive (θετικά) και 8.782 negative (αρνητικά). Αρχικά, τα tweets υποβλήθηκαν στο στάδιο της προεπεξεργασίας, το οποίο περιλάμβανε filtering (φιλτράρισμα), tokenization (λεκτική ανάλυση), stop word removal (αφαίρεση λέξεων όπως “i”, “is”, “a” κλπ), stemming (αποκατάληξη) και lemmatization (λημματοποίηση). Στην συνέχεια, χρησιμοποιήθηκαν τεχνικές όπως TF-IDF και Bag Of Words για την εξαγωγή των απαραίτητων χαρακτηριστικών από το κείμενο, τα οποία χρησιμοποιήθηκαν από τους ταξινομητές Decision Tree, Logistic Regression, Multinomial Naive Bayes, Support Vector Machine, Random Forest, και Bernoulli Naive Bayes για την ανίχνευση των spam tweets, ενώ για την ανάλυση του συναισθήματος χρησιμοποιήθηκαν οι Stochastic Gradient Descent, Support Vector Machine, Logistic Regression, Random Forest και Naive Bayes, καθώς και μέθοδοι Βαθιάς Μάθησης, οι οποίες δεν μας αφορούν στα πλαίσια της ενότητας αυτής. Ως μετρικές αξιολόγησης των παραπάνω χρησιμοποιήθηκαν οι Accuracy, Recall, Specificity (ή Negative Recall), Precision και F1-Score.

2.2 Βαθιά Μάθηση και Μοντέλα Transformer

Οι Wei & Nguyen [31], χρησιμοποιώντας μόνο τα tweets του χρήστη, και με χρήση word embeddings (συγκεκριμένα ενός προεκπαιδευμένου 200-dimensional GloVe), τριών επιπέδων Bidirectional Long Short-Term Memory (BiLSTM) και ενός fully connected softmax layer στην έξοδο, πέτυχαν αρκετά υψηλές επιδόσεις (recall ίσο με 0.976). Ως μετρικές αξιολόγησης

της επίδοσης χρησιμοποίησαν τα Precision, Recall, Specificity, Accuracy, F-measure και Matthews Correlation Coefficient (MCC), ενώ σαν σύνολο δεδομένων (dataset) χρησιμοποίησαν αυτό το οποίο προτάθηκε από τους Cresci et al. στο paper τους [8].

Οι Dukić et al. [9], χρησιμοποιώντας μόνο τα χαρακτηριστικά που προκύπτουν από τα tweets των χρηστών, ανέπτυξαν τόσο ένα μοντέλο επιβλεπόμενης μηχανικής μάθησης, το μοντέλο Λογιστικής Παλινδρόμησης (Logistic Regression Model), όσο και ένα Βαθύ Νευρωνικό Δίκτυο Εμπρόσθιας Τροφοδότησης (Feedforward Deep Neural Network), ακολουθώντας δύο διαφορετικές προσεγγίσεις εξαγωγής embeddings. Στην δεύτερη προσέγγιση, η οποία είναι και αυτή που μας απασχολεί στην ενότητα αυτή, και η οποία επέφερε καλύτερα αποτελέσματα, χρησιμοποίησαν ένα προεκπαιδευμένο μοντέλο Transformer, το BERT, για να εξάγουν embeddings τα οποία προέκυπταν από τα tweets και τα συμφραζόμενά τους (contextualized embeddings). Σαν χαρακτηριστικά εν τέλει χρησιμοποιήθηκαν τα contextualized embeddings που είχαν εξαχθεί με χρήση του BERT, τα emojis (μικρά εικονίδια και «φατσούλες» που εκφράζουν συναίσθημα ή κατάσταση), τα οποία είχαν αναπαρασταθεί με 300-dimensional emoji embeddings, ονόματι emoji2vec, καθώς και τα κατηγορικά χαρακτηριστικά της ύπαρξης των hashtags, mentions και URLs μέσα στο tweet και το αν το tweet αποτελούσε retweet (αναδημοσίευση). Σαν μετρική αξιολόγησης των παραπάνω χρησιμοποιήθηκε το Weighted F1-Score, ενώ σαν dataset τα tweets αγγλικής γλώσσας από το σύνολο δεδομένων PAN-19 [10].

Οι Heidari & Jones [32] χρησιμοποίησαν GloVe embeddings και το BERT για να κατηγοριοποιήσουν τα tweets με βάση το συναίσθημα το οποίο μετέδιδαν (sentiment classification), σε θετικά ή αρνητικά, με σκοπό να ανιχνεύσουν και να εξάγουν χαρακτηριστικά από τα tweets, τα οποία ήταν ανεξάρτητα από το θέμα το οποίο τα ίδια πραγματεύονταν. Σαν χαρακτηριστικά χρησιμοποιήθηκαν το πλήθος των ουδέτερων, θετικών και αρνητικών tweets ενός χρήστη, το άθροισμα των βαθμών πολικότητας (polarity scores) των θετικών και των αρνητικών tweets, καθώς και ο μέσος όρος θετικής και αρνητικής πολικότητας των θετικών και αρνητικών tweets αντίστοιχα. Εξετάστηκε η επίδοση των μοντέλων Random Forest, Support Vector Machine (SVM), Logistic Regression και ενός Feedforward Neural Network, το οποίο εν τέλει είχε και την καλύτερη, και για αυτό επιλέχθηκε για την συμπλήρωση του τελευταίου σταδίου του μοντέλου, δηλαδή την κατηγοριοποίηση μεταξύ human και bot. Οι μετρικές αξιολόγησης που χρησιμοποιήθηκαν είναι τα Accuracy, MCC και F1-Score, ενώ σαν σύνολο δεδομένων χρησιμοποιήθηκε των Cresci et al. [8].

Οι Saravani et al. [33] επιστράτευσαν ένα σύνολο από αρχιτεκτονικές νευρωνικών δικτύων και εξέτασαν πληθώρα διαφορετικών συνδυασμών (configurations) μεταξύ τους, με σκοπό να επιλέξουν εκείνο το οποίο πετυχαίνει την καλύτερη επίδοση στην διαδικασία ανίχνευσης των tweets που έχουν δημιουργηθεί από αυτοματοποιημένους χρήστες. Ο συνδυασμός ο οποίος τελικά χρησιμοποιήθηκε ήταν εκείνος που περιείχε το BERT στο πρώτο επίπεδο της αρχιτεκτονικής, και συγκεκριμένα το μοντέλο και τον tokenizer του CTBERT-v2, ένα Bidirectional LSTM στο δεύτερο επίπεδο, το νευρωνικό VLAD (Vector of Locally Aggregated Descriptors) στο τρίτο επίπεδο, και συγκεκριμένα το NeXtVLAD, ενώ το τελευταίο επίπεδο αποτελείται από δύο fully-connected dense layers, για να είναι δυνατή η ταξινόμηση στις δύο κλάσεις. Τα χαρακτηριστικά που χρησιμοποιήθηκαν επήλθαν αποκλειστικά από τα tweets και το περιεχόμενο που εξάγεται από τα συμφραζόμενά τους. Ως μετρικές αξιολόγησης χρησιμοποιήθηκαν τα Precision, Recall, F1 και Accuracy, ενώ σαν

σύνολο δεδομένων χρησιμοποιήθηκε το “TweepFake - A Twitter Deep Fake Dataset”, το οποίο προτάθηκε από τους Fagni et al. κατά την δημοσίευση του paper τους [12].

Οι Fagni et al. [12], χρησιμοποιώντας και αυτοί μόνο τα tweets, καθώς και τα χαρακτηριστικά που προκύπτουν από το περιεχόμενο που εξάγεται από τα συμφραζόμενά τους, ακολούθησαν τέσσερις προσεγγίσεις, με σκοπό να εξετάσουν την αποτελεσματικότητα ενός συνόλου από τεχνικές Μηχανικής Μάθησης και Βαθιάς Μάθησης. Η πρώτη προσέγγιση περιγράφηκε στην ενότητα 2.1, εφόσον αναφερόταν σε παραδοσιακούς αλγορίθμους μηχανικής μάθησης. Στην δεύτερη προσέγγιση, χρησιμοποιήθηκαν οι ίδιοι ταξινομητές με την πρώτη, δηλαδή οι Logistic Regression, Random Forest και SVM, αλλά αυτήν την φορά για την κωδικοποίηση των tweets χρησιμοποιήθηκε το μοντέλο BERT. Στην τρίτη προσέγγιση χρησιμοποιήθηκαν ένα δίκτυο CNN, ένα δίκτυο GRU (Gated Recurrent Units), καθώς και ο συνδυασμός των δύο δικτύων, για την αναπαράσταση των εσωτερικών embeddings των tweets, ενώ στην τελευταία προσέγγιση πραγματοποιήθηκε fine-tuning (δηλαδή προσαρμογή και εκπαίδευση των προεκπαιδευμένων μοντέλων στα δεδομένα που πραγματεύεται η συγκεκριμένη μελέτη περίπτωσης) των Transformer μοντέλων BERT, DistilBERT, RoBERTa (A Robustly Optimized BERT Pretraining Approach), καθώς και του XLNet. Ως μετρικές αξιολόγησης χρησιμοποιήθηκαν τα Precision, Recall, F1 και Accuracy, ενώ σαν σύνολο δεδομένων χρησιμοποιήθηκε το “TweepFake - A Twitter Deep Fake Dataset”, το οποίο προτάθηκε από τους ίδιους τους συγγραφείς στο εν λόγω paper. Για την δημιουργία του dataset αυτού, συλλέξαν tweets από ένα σύνολο από 23 bots, που μιμούνταν 17 λογαριασμούς πραγματικών χρηστών, ενώ συλλέξαν και τυχαία tweets από αυτούς τους πραγματικούς λογαριασμούς, τους οποίους τα bot μιμούνταν, για να δημιουργήσουν συνολικά ένα ισορροπημένο dataset από 25.572 tweets (μισά από πραγματικούς χρήστες και μισά από bots), το οποίο και κοινοποίησαν για δημόσια χρήση στην πλατφόρμα Kaggle.

Οι Martín-Gutiérrez et al. [34] πρότειναν αρχικά ένα μοντέλο που κωδικοποιεί όλα τα χαρακτηριστικά που είναι βασισμένα στο κείμενο του λογαριασμού του χρήστη μέσω πολυγλωσσικών Language Models (LM), μεταξύ των οποίων μοντέλα Transformer όπως το BERT, RoBERTa ή contextual embeddings που βασίζονται στο κείμενο, ενώ στην συνέχεια πρότειναν και ένα Dense-Based μοντέλο βαθιάς μηχανικής μάθησης. Ως χαρακτηριστικά από κάθε λογαριασμό χρησιμοποιήθηκαν χαρακτηριστικά σχετικά με την δραστηριότητα και την δημοσιότητά του, καθώς και πληροφορίες σχετικές με το προφίλ του χρήστη, ενώ σαν μετρική αξιολόγησης εφαρμόστηκε η F1-Score. Σαν σύνολο δεδομένων επιστρατεύθηκε και χρησιμοποιήθηκε μια πληθώρα γνωστών στην βιβλιογραφία συνόλων δεδομένων: Verified-2019 [35], Botwiki-2019 [35], Midterm-2018 [35], Cresci-stock-2018 [36], Cresci-rtbust-2019 [37], Political-bots-2019 [38], Botometer-feedback-2019 [38], Vendor-purchased-2019 [38], Celebrity-2019 [38], Pronbots-2019 [38], Gilani-2017 [39], Varol-2017 [24] και Cresci-2017 [8].

Οι Heidari et al. [40], χρησιμοποιώντας το dataset των Cresci et al. [8], πρότειναν ένα μοντέλο το οποίο αντλεί προσωπικά δεδομένα των χρηστών (ηλικία, φύλο, προσωπικότητα, εκπαίδευση) από τις δημοσιεύσεις τους και δημιουργεί το προφίλ τους. Για τον σκοπό αυτό, επιστρατεύονται τόσο GloVe embeddings, όσο και ELMO (Embeddings from Language MOdels). Συνολικά, εκπαιδεύονται οκτώ διαφορετικά νευρωνικά δίκτυα πάνω στα χαρακτηριστικά που αναφέρθηκαν, ενώ στο τελευταίο layer τους προστέθηκε ένα ακόμα νευρωνικό δίκτυο, με τις μετρικές Accuracy, F-Measure και MCC να χρησιμοποιούνται για την αξιολόγησή τους.

Οι Arin & Kutlu [41] πρότειναν μια αρχιτεκτονική η οποία αποτελείται από τρία LSTM και ένα fully-connected layer, για να αποτυπώσουν όλα τα δεδομένα που είναι διαθέσιμα σε έναν λογαριασμό, ενώ σαν word embeddings χρησιμοποίησαν το Glove-Twitter-50, το οποίο έχει προεκπαιδευτεί πάνω σε tweets. Τα χαρακτηριστικά που χρησιμοποιήθηκαν διακρίνονται στις κατηγορίες του περιεχομένου των tweets, των metadata των tweets, των metadata του λογαριασμού, καθώς και της περιγραφής που έχει θέσει στον λογαριασμό του ο χρήστης, και τα ενσωματώνουν σε διαφορετικά επίπεδα της αρχιτεκτονικής τους. Σαν μετρικές αξιολόγησης χρησιμοποιήθηκαν οι Precision, Recall, F1 και Accuracy, ενώ σαν σύνολα δεδομένων επιστρατεύθηκαν τα Varol-2017 [24], Cresci-2017 [8], Botometer-feedback-2019 [38] και το Caverlee-2011 [25].

Οι Luo et al. [42], χρησιμοποιώντας ως σύνολο δεδομένων το PAN-19 [10], ανέπτυξαν μια αρχιτεκτονική βαθιάς μάθησης η οποία ανιχνεύει τα bots βασιζόμενη στα tweets και τα χαρακτηριστικά που εξάγονται από αυτά. Πιο συγκεκριμένα, η αρχιτεκτονική αυτή περιλαμβάνει ένα Glove Embedding layer, δυο Bidirectional LSTM layers, ένα Attention layer (μηχανισμός προσοχής) μεταξύ τους, και στο τέλος περιλαμβάνει δυο fully-connected layers με συνάρτηση ενεργοποίησης (activation function) την ReLU, ενώ περιλαμβάνει και δυο Dropout layers, τοποθετημένα ανάμεσα στα fully-connected layers και στο τελικό layer της εξόδου, με σκοπό την αποφυγή του προβλήματος της υπερπροσαρμογής (overfitting). Για την αξιολόγηση της επίδοσης της αρχιτεκτονικής αυτής εφαρμόσθηκαν οι μετρικές Accuracy και ROC.

Οι Tourille et al. [43], κάνοντας χρήση του dataset “TweepFake - A Twitter Deep Fake Dataset” [12], καθώς και ενός dataset που κατασκεύασαν οι ίδιοι, χρησιμοποίησαν αποκλειστικά το tweet και το περιεχόμενό του για να διακρίνουν τους αυτοματοποιημένους λογαριασμούς, προτείνοντας δύο προσεγγίσεις. Στην πρώτη χρησιμοποίησαν δύο μοντέλα RoBERTa και ένα Feedforward νευρωνικό δίκτυο δύο επιπέδων για την κατηγοριοποίηση των χρηστών, ενώ στην δεύτερη χρησιμοποίησαν πάλι δύο μοντέλα RoBERTa, τα οποία έγιναν fine-tuned στα δύο διαφορετικά dataset. Ως μετρικές αξιολόγησης της επίδοσης χρησιμοποιήθηκαν οι Precision, Recall, F1-Score και Accuracy.

Οι Lei et al. [44] δοκίμασαν να ανιχνεύσουν bots με πολυτροπικά δεδομένα, χρησιμοποιώντας τόσο τα χαρακτηριστικά που προκύπτουν από κείμενο και από γράφο, όσο και από semantic consistency vectors. Για το modality του text, που μας ενδιαφέρει στην ενότητα αυτήν, επιστρατεύθηκε ένα προεκπαιδευμένο RoBERTa για την κωδικοποίηση των tweets και της περιγραφής του λογαριασμού. Ως μετρικές αξιολόγησης της επίδοσης χρησιμοποιήθηκαν οι Accuracy και F1-Score, ενώ σαν σύνολα δεδομένων τα TwiBot-20 [45] και Cresci-2015 [46].

Οι Garcia-Silva et al. [47] εξετάζουν την χρήση προεκπαιδευμένων γλωσσικών μοντέλων (LM) για την αντιμετώπιση του προβλήματος της ανίχνευσης των tweets που έχουν δημοσιευθεί από αυτοματοποιημένους λογαριασμούς. Συγκεκριμένα, χρησιμοποίησαν τα μοντέλα BERT (base), GPT και GPT-2, τα οποία είναι μοντέλα συγκρίσιμα σε μέγεθος (σε πλήθος παραμέτρων). Σαν μετρικές αξιολόγησης χρησιμοποιήθηκαν οι Precision, Recall και F-Score, ενώ σαν σύνολο δεδομένων χρησιμοποιήθηκε αυτό που είχε προταθεί από τους Gilani et al. [48].

Οι Feng et al. [49] προτείνουν ένα ακόμα καινοτόμο framework, ονόματι SATAR (Self-supervised Approach to Twitter Account Representation learning), το οποίο υιοθετεί την αυτο-

επιβλεπόμενη μάθηση (self-supervised learning) για να δημιουργήσει την αναπαράσταση του χρήστη και να ανιχνεύσει με αυτόν τον τρόπο τα bots. Συγκεκριμένα, το SATAR χρησιμοποιεί και κωδικοποιεί από κοινού τα tweets του χρήστη (χρησιμοποιώντας ιεραρχικά RNNs διαφορετικού βάθους, συνοδευόμενα από τον μηχανισμό προσοχής), τα χαρακτηριστικά του profile του, καθώς και πληροφορίες για την «γειτονιά» του (σχέσεις following-follower), για να κατασκευάσει ένα διάνυσμα που αναπαριστά πλήρως την κοινωνική κατάστασή του, ενώ αποφεύγει να χρησιμοποιήσει feature engineering, για να μην υπάρχει ανεπιθύμητο bias (προτίμηση/τάση) στα αποτελέσματα. Ως μετρικές αξιολόγησης χρησιμοποιήθηκαν οι Accuracy, F1-Score και MCC, ενώ σαν σύνολα δεδομένων επιστρατεύθηκαν τα TwiBot-20 [45], Cresci-2017 [8] και PAN-19 [10].

Οι Ilias & Roussaki [3] πρότειναν δύο μεθόδους για την κατηγοριοποίηση των χρηστών σε πραγματικούς και αυτοματοποιημένους, οι οποίες βασίζονται κυρίως σε Natural Language Processing (Επεξεργασία Φυσικής Γλώσσας). Στην δεύτερη μέθοδο, που είναι και αυτή που μας απασχολεί στα πλαίσια της ενότητας αυτής, προτείνεται μια αρχιτεκτονική βαθιάς μάθησης με σκοπό την διάκριση μεταξύ των tweets που δημοσιεύθηκαν από πραγματικούς και αυτοματοποιημένους χρήστες. Η αρχιτεκτονική αυτή περιλαμβάνει ένα embedding layer (που είναι ένα 50-dimensional προεκπαιδευμένο GloVe), δύο BiLSTM layers, έναν μηχανισμό προσοχής πάνω από το layer του δεύτερου BiLSTM, και τρία dense layers. Ως μετρικές αξιολόγησης χρησιμοποιήθηκαν τα Precision, Recall, F-measure/F1-Score, Area Under the ROC curve (AUROC) και Accuracy, ενώ σαν σύνολα δεδομένων χρησιμοποιήθηκαν τα Cresci-2017 [8] και Social Honeypot Dataset [25].

Οι Rodrigues et al. [30] πρότειναν ένα σύστημα το οποίο διακρίνει αν ένα tweet θεωρείται “spam” ή “ham” και αξιολογεί το συναίσθημα που εμπεριέχεται σε αυτό. Για την ανίχνευση των spam tweets χρησιμοποιήθηκε ένα SMS dataset, που περιέχει 4.825 ham tweets και 747 spam tweets, ενώ για την ανάλυση του συναισθήματος χρησιμοποιήθηκε ένα μεγάλο dataset από την πλατφόρμα Kaggle που περιέχει 31.015 tweets, εκ των οποίων 12.548 θεωρούνται neutral (ουδέτερα), 9.685 positive (θετικά) και 8.782 negative (αρνητικά). Αρχικά, τα tweets υποβλήθηκαν στο στάδιο της προεπεξεργασίας, το οποίο περιλάμβανε filtering (φιλτράρισμα), tokenization (λεκτική ανάλυση), stop word removal (αφαίρεση λέξεων όπως “i”, “is”, “a” κλπ), stemming (αποκατάληξη) και lemmatization (λημματοποίηση). Στην συνέχεια, χρησιμοποιήθηκαν τεχνικές όπως TF-IDF και Bag Of Words για την εξαγωγή των απαραίτητων χαρακτηριστικών από το κείμενο. Για το μέρος της υλοποίησης που αφορά την ανάλυση του συναισθήματος, και το οποίο μας αφορά στην ενότητα αυτήν, δεδομένων των τεχνικών που επιστρατεύθηκαν, χρησιμοποιήθηκαν μέθοδοι Βαθιάς Μάθησης, όπως απλό Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), Bidirectional Long Short-Term Memory (BiLSTM), και 1D (μονοδιάστατο) Convolutional Neural Network (CNN), καθώς και παραδοσιακοί αλγόριθμοι Μηχανικής Μάθησης, οι οποίοι αναφέρθηκαν στην αντίστοιχη παρουσίαση του paper στην ενότητα 2.1. Ως μετρικές αξιολόγησης των παραπάνω χρησιμοποιήθηκαν οι Accuracy, Recall, Specificity (ή Negative Recall), Precision και F1-Score.

Τέλος, οι Di Paolo et al. [50], χρησιμοποιώντας ως σύνολα δεδομένων τα Cresci-2017 [8], Cresci-stock-2018 [36] και TwiBot-20 [45], πρότειναν μια νέα μέθοδο ανίχνευσης των bots, βασισμένη σε αναγνώριση εικόνας (image recognition). Συγκεκριμένα, με βάση τον τύπο του tweet (αν ήταν retweet, reply ή απλό tweet), αντιστοίχισαν το κάθε tweet ενός χρήστη με ένα σύμβολο. Εφαρμόζοντας την διαδικασία αυτή για όλα τα tweets του κάθε χρήστη, με την

χρονολογική σειρά που δημοσιεύθηκαν, δημιούργησαν μια αλληλουχία DNA, το λεγόμενο Digital DNA του. Στην συνέχεια, πρότειναν έναν αλγόριθμο για να μετατρέψουν το Digital DNA σε εικόνα που αποτελείται από 3 channels (κανάλια) και τροφοδότησαν τις εικόνες αυτές σε προεκπαιδευμένα CNNs, όπως το VGG16, το WideResNet50 και το ResNet50, το οποίο απέδωσε και τα καλύτερα αποτελέσματα. Σε περιπτώσεις που ο αριθμός των tweets του χρήστη ήταν μικρός, όπως συμβαίνει στο σύνολο δεδομένων TwiBot-20 [45], όπου σε κάθε χρήστη αντιστοιχούν το πολύ 200 tweets, οι εικόνες που προέκυπταν ήταν μικρές και μη ποιοτικές. Για τον λόγο αυτόν, εμπλούτισαν τις εικόνες αυτές χρησιμοποιώντας τον αλγόριθμο SuperTML, για να μετατρέψουν χαρακτηριστικά του profile του χρήστη και να τα ενσωματώσουν στο Digital DNA του. Ως μετρικές αξιολόγησης της επίδοσης των παραπάνω χρησιμοποιήθηκαν τα Accuracy, Precision, Recall, F1 και Matthews Correlation Coefficient (MCC).

2.3 Δίκτυα Γράφων

Οι Chakraborty et al. [51] χρησιμοποίησαν αρχικά feature engineering (μηχανική χαρακτηριστικών), επεξεργασία φυσικής γλώσσας και ανάλυση γράφων για να εξάγουν συνολικά 54 χαρακτηριστικά, ενώ στην συνέχεια χρησιμοποίησαν δύο μεθόδους επιλογής χαρακτηριστικών (SHapley Additive exPlanations - SHAP και Correlation Matrix) για να επιλέξουν τα πιο σχετικά χαρακτηριστικά, τα οποία στην συνέχεια θα αποτελούσαν το σύνολο εκπαίδευσης των μοντέλων (training set), και τα οποία ήταν 16 σε αριθμό. Για την εξαγωγή χαρακτηριστικών μέσω των γράφων, αρχικά χρησιμοποιήθηκαν Graph Analytics για την δημιουργία ενός κοινωνικού γράφου, με τους χρήστες σαν κόμβους και τις σχέσεις follower-following σαν ακμές μεταξύ τους, ενώ στην συνέχεια στον γράφο αυτόν εφαρμόστηκε το framework Node2Vec, με σκοπό την αξιοποίηση της πληροφορίας της γειτονιάς του κάθε χρήστη-κόμβου. Ως ταξινομητές χρησιμοποιήθηκαν τα SVMs (Polynomial και RBF Kernel), Logistic Regression, Random Forest, Gradient Boosting, XGBoost, καθώς και ένα νευρωνικό δίκτυο με 3 layers. Ως μετρικές αξιολόγησης χρησιμοποιήθηκαν τα Accuracy και F1-Score, ενώ σαν σύνολο δεδομένων χρησιμοποιήθηκαν σε συνδυασμό τα Midterm-2018-candidates [52], Gilani-2017 [39], Varol-2017 [24], καθώς και το Twitter-API, για την συμπλήρωση συνολικά 126.503 διαφορετικών λογαριασμών. Μετά την αφαίρεση των ανενεργών λογαριασμών, το σύνολο δεδομένων που τελικά χρησιμοποιήθηκε αποτελούταν από 46.354 λογαριασμούς.

Οι Bui et al. [53] κατασκεύασαν γράφους, ξεκινώντας από ένα tweet ενός λογαριασμού και αναδρομικά συλλέγοντας retweets και follower relationships για να αναπαραστήσουν τις ακμές μεταξύ των κόμβων, που θεωρούνταν οι χρήστες, με σκοπό να παρατηρήσουν την διαφορά στην μετάδοση της πληροφορίας μεταξύ ενός κοινωνικού δικτύου ενός πραγματικού χρήστη, και ενός αυτοματοποιημένου. Χρησιμοποίησαν ως χαρακτηριστικά τα χαρακτηριστικά των γράφων (κόμβοι, απομονωμένοι κόμβοι, ακμές, αριθμός συνεκτικών συνιστωσών κλπ), ενώ σαν σύνολο δεδομένων το TwiBot-20 [45], το οποίο εμπεριέχει ένα πλήρες δείγμα της σφαίρας του Twitter (Twittersphere), κατηγοριοποιημένο σε διαφορετικά πεδία, όπως πολιτική, επιχειρήσεις, ψυχαγωγία και αθλητικά.

Οι Feng et al. [7], χρησιμοποιώντας ως σύνολο δεδομένων το TwiBot-20 [45], προτείνουν ένα bot detection framework, με το όνομα BotRGCN (Bot Detection with Relational Graph Convolutional Networks). Το framework αυτό κωδικοποιεί από κοινού

πολυτροπικά δεδομένα του χρήστη (multi-modal information), δηλαδή δεδομένα που προέρχονται από διαφορετικά αισθητήρια (συχνά κείμενο, εικόνα και ήχο), κατασκευάζει έναν ετερογενή γράφο για να αναπαραστήσει τον πραγματικό κόσμο του Twitter, και τέλος εφαρμόζει Relational Graph Convolutional Networks (Συσχετιστικά Συνελικτικά Δίκτυα Γράφου). Σαν χαρακτηριστικά του χρήστη χρησιμοποιούνται η περιγραφή και τα tweets του, καθώς και αριθμητικά και κατηγορικά χαρακτηριστικά του, ενώ σαν μετρικές αξιολόγησης του framework χρησιμοποιούνται οι Accuracy, F1-Score και MCC.

Οι Feng et al. [54], χρησιμοποιώντας και πάλι ως σύνολο δεδομένων το TwiBot-20 [45], κατασκεύασαν ένα ετερογενές δίκτυο πληροφοριών με τους χρήστες σαν κόμβους και τις μεταξύ τους σχέσεις σαν ακμές, και εφάρμοσαν σε αυτό Relational Graph Transformers και Semantic Attention Networks (Σημασιολογικά Δίκτυα Προσοχής) για να συναθροίσουν τις ενέργειες και την αλληλεπίδραση μεταξύ των χρηστών, και να ανιχνεύσουν έτσι τους αυτοματοποιημένους λογαριασμούς. Ως χαρακτηριστικά, πάνω στα οποία βασίστηκε και η κατασκευή του Heterogeneous Information Network (HIN), χρησιμοποιήθηκαν τα χαρακτηριστικά των χρηστών και του λογαριασμού τους, ενώ σαν μετρικές αξιολόγησης της επίδοσης του μοντέλου χρησιμοποιήθηκαν οι Accuracy και F1-Score.

Οι Lei et al. [44] δοκίμασαν να ανιχνεύσουν bots με πολυτροπικά δεδομένα, χρησιμοποιώντας τόσο τα χαρακτηριστικά που προκύπτουν από κείμενο και από γράφο, όσο και από semantic consistency vectors. Για το modality του γράφου, που μας ενδιαφέρει στην ενότητα αυτήν, χρησιμοποιήθηκαν οι ίδιες μέθοδοι που επιστράτευσαν στο paper τους οι Feng et al. [7] για την παρουσίαση του BotRGCN framework. Ως μετρικές αξιολόγησης της επίδοσης χρησιμοποιήθηκαν οι Accuracy και F1-Score, ενώ σαν σύνολα δεδομένων τα TwiBot-20 [45] και Cresci-2015 [46].

Οι Ali Alhosseini et al. [55] πρότειναν μια μέθοδο επαγωγικού Representation Learning για την ανίχνευση των αυτοματοποιημένων λογαριασμών, βασιζόμενοι στα χαρακτηριστικά του προφίλ του χρήστη και στον γράφο κοινωνικού δικτύου που κατασκεύασαν από τα δεδομένα, ενώ για να αξιολογήσουν την επίδοση της μεθόδου αυτής, εφάρμοσαν τις μετρικές Precision, Recall και F1-Score για κάθε μια από τις κλάσεις, και ύστερα παρουσίασαν τον μέσο όρο και για τις δύο κλάσεις για κάθε μια από τις μετρικές (τεχνική γνωστή και ως Macro Score).

Τέλος, οι Beskow & Carley [56] θέλησαν να παρουσιάσουν όχι μόνο το χρονολόγιο (timeline) του χρήστη, αλλά και των φίλων του. Για αυτόν τον λόγο, χρησιμοποίησαν 3 διαφορετικά σύνολα δεδομένων για bot, καθώς και το Twitter API για την εξαγωγή πραγματικών λογαριασμών. Εξετάστηκε ο αλγόριθμος Random Forest, ενώ το tuning του διεξήχθη με τυχαία επιλογή τιμών παραμέτρων, χρησιμοποιώντας ένα 3-fold cross-validation. Τα χαρακτηριστικά που χρησιμοποιήθηκαν κατηγοριοποιήθηκαν σε 3 βαθμίδες, με την πρώτη να περιέχει τα χαρακτηριστικά του χρήστη και του δικτύου, το περιεχόμενο του χρήστη και στοιχεία σχετικά με τον χρόνο (ηλικία λογαριασμού κλπ). Η δεύτερη βαθμίδα περιέχει τα χαρακτηριστικά της πρώτης, με περισσότερες προσθήκες στα χαρακτηριστικά του δικτύου, του περιεχομένου και του χρόνου, ενώ η τρίτη βαθμίδα περιέχει τα χαρακτηριστικά της δεύτερης, με περισσότερες προσθήκες στα χαρακτηριστικά του δικτύου, οι οποίες προέκυψαν από την κατασκευή ενός Snowball Sampling Ego-Centric Network. Ως μετρικές αξιολόγησης της επίδοσης χρησιμοποιήθηκαν τα AUC, ROC, Precision και Recall.

2.4 Μη-Επιβλεπόμενη Μάθηση

Οι Wu et al. [57] χρησιμοποίησαν δυο αλγορίθμους μη-επιβλεπόμενης μηχανικής μάθησης, τους Agglomerative και K-Means. Ανέλυσαν το αποτέλεσμα διαφορετικού πλήθους συστάδων, και κατέληξαν πως τα καλύτερα αποτελέσματα προκύπτουν με 4 συστάδες (clusters). Ως χαρακτηριστικά εν τέλει χρησιμοποιήθηκαν 12, τα οποία είχαν σχέση με τα χαρακτηριστικά του λογαριασμού του χρήστη, καθώς και με το πλήθος των tweets του και την συχνότητα δημοσίευσής τους ανά μέρα, ενώ σαν μετρικές αξιολόγησης των αλγορίθμων χρησιμοποιήθηκαν τα Accuracy Macro, F1 Macro, FPR Macro, Kappa, Overall Accuracy, PPV Macro, TPR Macro και Zero-one Loss.

Οι Anwar & Yaqub [58] χρησιμοποίησαν το Twitter API για να συλλέξουν 546.728 tweets σχετικά με τις εκλογές που έλαβαν μέρος το 2019 στον Καναδά, τα οποία ανήκαν σε 103.791 διαφορετικούς χρήστες. Το κείμενο των tweets τελικά αφέθηκε, εφόσον η ανάλυση δεν βασίστηκε στο συναίσθημα του περιεχομένου του, αλλά στα metadata των tweets και τα χαρακτηριστικά του λογαριασμού. Για την εύρεση της βαρύτητας καθενός εκ των 13 χαρακτηριστικών που χρησιμοποιήθηκαν, επιστρατεύτηκαν οι μέθοδοι του Correlation Matrix και Principal Component Analysis (PCA), ενώ εξετάστηκε και η επίδοση του αλγορίθμου K-means, όπου χρησιμοποιήθηκαν 2 clusters.

Τέλος, οι Gera & Sinha [59], χρησιμοποιώντας το Twitter API για να εξάγουν τα δεδομένα τους, τα οποία αρχικά ήταν unlabeled (δεν είχαν ετικέτα που να προσδιορίζει αν ο χρήστης ήταν πραγματικός ή αυτοματοποιημένος), πρότειναν μια μέθοδο ανίχνευσης αυτοματοποιημένων χρηστών η οποία αποτελούταν από 4 στάδια. Στο πρώτο στάδιο, και εφόσον τα δεδομένα ήταν unlabeled, εφάρμοσαν διαφορετικές τεχνικές συσταδοποίησης (DBScan, Agglomerative και Birch Clustering Algorithms) και αξιολόγησαν το silhouette score τους, με την τελευταία να πετυχαίνει το καλύτερο. Στο δεύτερο στάδιο πραγματοποιήθηκε feature selection χρησιμοποιώντας τις τεχνικές Entropy και Information Gain (για τον καθορισμό της συνεισφοράς καθενός εκ των χαρακτηριστικών στην ανίχνευση των bots), που ακολουθήθηκε στο τρίτο στάδιο από ένα σχηματισμό κανόνων (rule formation), οι οποίοι εφαρμόστηκαν ιεραρχικά στους χρήστες στο τέταρτο στάδιο για την ανίχνευση των bots. Στο τελικό labeled dataset εξετάστηκε η επίδοση των μοντέλων SVM και Random Forest, με βάση τις μετρικές Precision και Recall.

Κεφάλαιο 3

3. Θεωρητικό Υπόβαθρο

Στο κεφάλαιο αυτό παρέχεται στον αναγνώστη το απαραίτητο θεωρητικό υπόβαθρο, τόσο γενικών αρχών και εννοιών Μηχανικής Μάθησης και Νευρωνικών Δικτύων, για την κατανόηση του γενικότερου και ολοκληρωμένου πλαισίου τους, όσο και συγκεκριμένων τεχνολογιών και μεθόδων, οι οποίες χρησιμοποιούνται για την εκπόνηση της παρούσας διπλωματικής εργασίας. Συγκεκριμένα, η **πρώτη ενότητα** (3.1) καλύπτει βασικές έννοιες στο πλαίσιο της Μηχανικής Μάθησης, καθώς και τους κλάδους στους οποίους κατηγοριοποιείται. Στην **δεύτερη ενότητα** (3.2) γίνεται αναφορά στις βασικότερες τεχνολογίες Βαθιάς Μάθησης, όπως είναι τα Τεχνητά Νευρωνικά Δίκτυα, οι Συναρτήσεις Ενεργοποίησης, οι Συναρτήσεις Κόστους, ο Αλγόριθμος Οπίσθιας Διάδοσης και η Μεταφορά Μάθησης. Στην **τρίτη ενότητα** (3.3) παρουσιάζονται τεχνολογίες και αρχιτεκτονικές Βαθιών Νευρωνικών Δικτύων, με έμφαση στα Συνελικτικά Νευρωνικά Δίκτυα (CNNs), στους Μηχανισμούς Προσοχής και στα μοντέλα Transformer, τα οποία χρησιμοποιούμε στα πλαίσια της εργασίας αυτής. Στην **τέταρτη ενότητα** (3.4) περιγράφονται θεμελιώδεις έννοιες, τεχνικές και μοντέλα από τον χώρο της Επεξεργασίας Φυσικής Γλώσσας (Natural Language Processing - NLP). Τέλος, το Κεφάλαιο ολοκληρώνεται με την **πέμπτη ενότητα** (3.5), η οποία πραγματεύεται μεθόδους και μετρικές αξιολόγησης του μοντέλου.

3.1 Μηχανική Μάθηση

Η Μηχανική Μάθηση (Machine Learning) αποτελεί κλάδο της Τεχνητής Νοημοσύνης (Artificial Intelligence), στον οποίο μελετώνται και αναπτύσσονται συστήματα και μέθοδοι που δύνανται να μιμηθούν τον τρόπο με τον οποίο εκπαιδεύονται οι άνθρωποι, με σκοπό την παραγωγή προβλέψεων ή την λήψη αποφάσεων για μια συγκεκριμένη εργασία, δίχως ρητό προγραμματισμό. Οι αλγόριθμοι που αναπτύσσονται εκπαιδεύονται με ένα σύνολο δεδομένων, το οποίο έχει προκύψει μέσω παρατηρήσεων και καταγραφών, και εντοπίζουν μοτίβα σε αυτά (pattern matching), βάσει των οποίων προβαίνουν στις εκάστοτε προβλέψεις τους. Ο κλάδος της Μηχανικής Μάθησης διαφοροποιείται από τις παραδοσιακές υπολογιστικές προσεγγίσεις, εφόσον οι κλασικοί αλγόριθμοι συνιστούν ρητά προγραμματισμένες εντολές. Έτσι, οι αλγόριθμοι Μηχανικής Μάθησης χρησιμοποιούνται σε ένα μεγάλο εύρος εφαρμογών, όπου είναι δύσκολη ή ανέφικτη η ανάπτυξη συμβατικών, ρητά προγραμματισμένων, αλγορίθμων για την εκτέλεση των εργασιών αυτών.

Η Μηχανική Μάθηση αποτελεί ταυτόχρονα ένα σημαντικό, οριακά αναπόσπαστο πλέον, συστατικό του τομέα της Επιστήμης Δεδομένων (Data Science), ο οποίος γνωρίζει ολοένα και μεγαλύτερη απήχηση τα τελευταία χρόνια. Οι αλγόριθμοι Μηχανικής Μάθησης, αφού δημιουργηθούν, τροφοδοτούνται συνεχώς με νέες εισόδους συνόλων δεδομένων (datasets), τα οποία συνιστούν μια διαρκώς αναπτυσσόμενη βιβλιοθήκη που βρίσκει χρησιμότητα σε ποικίλες εργασίες, που σχετίζονται με την αναγνώριση και κατηγοριοποίηση εικόνας,

κειμένου, καθώς ακόμα και ήχου. Οι ίδιοι, λόγω του «βομβαρδισμού» που δέχονται από αυτά τα datasets, έρχονται αντιμέτωποι διαρκώς με νεότερα δεδομένα, μέσω των οποίων βελτιώνονται, έτσι ώστε να είναι σε θέση να παράξουν όσο το δυνατόν καλύτερες προβλέψεις όταν έρθουν αντιμέτωποι με δεδομένα με τα οποία έρχονται για πρώτη φορά σε «επαφή».

Αναλόγως την διαδικασία εκπαίδευσης του υπολογιστικού συστήματος και τον τρόπο με τον οποίο διαχειρίζεται τα υπό εξέταση δεδομένα με σκοπό την παραγωγή αποτελέσματος, ο τομέας της Μηχανικής Μάθησης διακρίνεται σε ευρείς κατηγορίες. Η κάθε κατηγορία συνοδεύεται με τα προτερήματά της και τις αδυναμίες της, ενώ για διαφορετικούς τύπους προβλημάτων επιστρατεύεται και διαφορετικό είδος μάθησης. Οι τρεις πιο διαδεδομένες κατηγορίες μάθησης του κλάδου είναι: Επιβλεπόμενη Μάθηση (Supervised Learning), Μη Επιβλεπόμενη Μάθηση (Unsupervised Learning) και Ενισχυτική Μάθηση (Reinforcement Learning). Πιο συγκεκριμένα:

- **Επιβλεπόμενη Μάθηση (Supervised Learning)**

Σε αυτήν την κατηγορία Μηχανικής Μάθησης, ως είσοδος στο μοντέλο παρέχονται δεδομένα με γνωστή κατηγοριοποίηση. Δηλαδή, τα δεδομένα εκπαίδευσης αποτελούνται από χαρακτηριστικά εισόδου (features), που συνοδεύονται με τις αντίστοιχες τιμές εξόδου - κατηγορίες (labelled data). Με τον τρόπο αυτό, το μοντέλο μαθαίνει και εκπαιδεύεται σε κάθε κατηγορία δεδομένων και έπειτα προβαίνει στη διαδικασία της αξιολόγησης, η οποία πραγματοποιείται σε ένα άλλο σύνολο χαρακτηριστικών, των οποίων την κατηγορία δεν γνωρίζουμε και προσπαθούμε να εξάγουμε (test set). Οι αλγόριθμοι επιβλεπόμενης μάθησης διακρίνονται σε **αλγόριθμους ταξινόμησης (classification)**, και **αλγόριθμους παλινδρόμησης (regression)**. Στην πρώτη περίπτωση, η τιμή της εξόδου (target value) παίρνει διακριτές τιμές (discrete values). Παραδείγματα τέτοιου είδους προβλημάτων (classification) αποτελούν η κατηγοριοποίηση χρηστών μέσων κοινωνικής δικτύωσης σε κακόβουλους ή μη, ανίχνευση του αν ένας άνθρωπος πάσχει από την ασθένεια του Alzheimer ή του Parkinson ή μη, καθώς και κατηγοριοποίηση μιας εικόνας σε κατηγορίες ζώων, όπως γάτα, σκύλος, λιοντάρι και άλλα, που αποτελεί multi-label classification, δηλαδή ταξινόμηση σε πολλές κλάσεις, και όχι δυαδική, όπως αποτελούσαν τα προηγούμενα προβλήματα που παρατέθηκαν. Στην δεύτερη περίπτωση, αυτή των προβλημάτων παλινδρόμησης, η τιμή της εξόδου (target value) παίρνει συνεχείς τιμές (continuous values). Παραδείγματα τέτοιων προβλημάτων αποτελούν η πρόβλεψη της τιμής ενός σπιτιού βάσει των χαρακτηριστικών του, η πιθανότητα αγοράς ενός προϊόντος από έναν πελάτη, καθώς και η πιθανότητα απώλειας ενός πελάτη ύστερα από αύξηση στην τιμή των προϊόντων.

- **Μη Επιβλεπόμενη Μάθηση (Unsupervised Learning)**

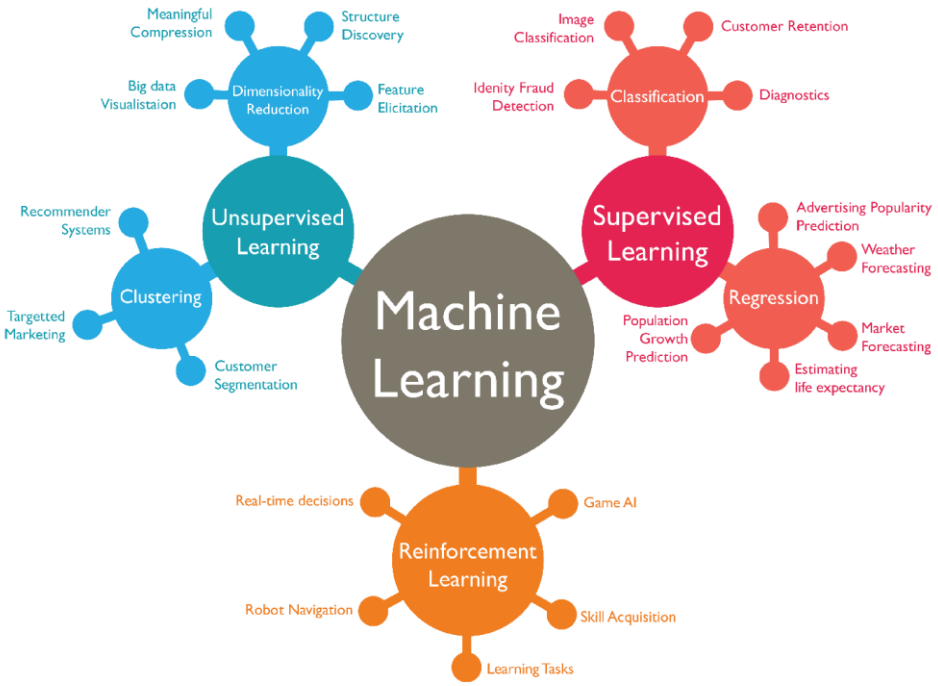
Σε αυτήν την κατηγορία Μηχανικής Μάθησης, τα δεδομένα εκπαίδευσης αποτελούνται από εισόδους (features), τα οποία όμως αυτήν την φορά δεν συνοδεύονται από τις αντίστοιχες τιμές εξόδου (unlabelled data). Συνεπώς, ο αλγόριθμος μάθησης καλείται να βρει ομοιότητες ανάμεσα στα δεδομένα εισόδου και να διακρίνει πρότυπα και μοτίβα, για τα οποία οι τιμές στόχοι είναι μη παρατηρήσιμες ή αδύνατον να συλλεχθούν. Το γεγονός πως τα μη επισημασμένα δεδομένα, δηλαδή τα δεδομένα τα οποία δεν έχουν label, συναντώνται πολύ πιο συχνά από τα επισημασμένα, λόγω της μεγαλύτερης αφθονίας τους, καθιστά τις τεχνικές μη επιβλεπόμενης μάθησης ιδιαίτερα ωφέλιμες και χρήσιμες. Η μη επιβλεπόμενη μάθηση αποσκοπεί τόσο στην αναζήτηση και εύρεση κρυφών προτύπων

μέσα σε ένα σύνολο δεδομένων, όσο και στην μάθηση χαρακτηριστικών (τεχνική γνωστή ως feature learning), η οποία καθιστά την υπολογιστική μηχανή ικανή να διακρίνει με αυτοματοποιημένο τρόπο τις αναπαραστάσεις που χρειάζονται για την ταξινόμηση ανεπεξέργαστων δεδομένων. Οι τέσσερις υποκατηγορίες προβλημάτων της μη επιβλεπόμενης μάθησης είναι **συσταδοποίηση (clustering)**, **συσχέτιση (association)**, **ανίχνευση ανωμαλιών (anomaly detection)** και **αυτόματοι κωδικοποιητές (autoencoders)**, με την συσταδοποίηση να αποτελεί και την μεγαλύτερη υποκατηγορία της. Στόχος της είναι η ομαδοποίηση παρατηρήσεων με τρόπο τέτοιο, ώστε τα μέλη που ανήκουν στην ίδια ομάδα (συστάδα) να είναι παρόμοια μεταξύ τους και να διαφέρουν από τα μέλη άλλων συστάδων, χωρίς βέβαια συχνά να μπορούμε να γνωρίζουμε το πλήθος των συστάδων που θα χρειαστούν εν τέλει ή την μορφή που θα πρέπει να έχουν. Τέλος, χρήση της μη επιβλεπόμενης μάθησης αποτελεί επίσης και η μείωση του αριθμού των χαρακτηριστικών σε ένα μοντέλο, μέσω της διαδικασίας μείωσης διαστάσεων.

- **Ενισχυτική Μάθηση (Reinforcement Learning)**

Σε αυτήν την κατηγορία Μηχανικής Μάθησης, το μοντέλο, που συχνά καλείται agent (πράκτορας), καλείται να εκπαιδευθεί σε ένα διαδραστικό περιβάλλον, μέσω της διαδικασίας «δοκιμής και σφάλματος» (trial and error), λαμβάνοντας ανατροφοδότηση και αξιολόγηση (feedback) από τις ίδιες του τις αποφάσεις. Η Ενισχυτική Μάθηση είναι η διαδικασία μάθησης κατά την οποία γίνεται εκπαίδευση μοντέλων Μηχανικής Μάθησης για την λήψη μιας σειράς αποφάσεων. Ένας τρόπος να την κατανοήσουμε είναι μέσω των παιχνιδιών, εφόσον το κάθε μοντέλο αντιμετωπίζει μια κατάσταση που μοιάζει με παιχνίδι. Ουσιαστικά, ο αλγόριθμος χρησιμοποιεί trial and error για να καταλήξει σε λύση για το τρέχον πρόβλημα. Αυτό επιτυγχάνεται με την λήψη ανταμοιβών και ποινών για τις ενέργειες που αποφασίζει να εκτελέσει, με τελικό στόχο την μεγιστοποίηση της συνολικής ανταμοιβής. Την πολιτική της επιβράβευσης, καθώς και της ποινής, την καθορίζει ο σχεδιαστής του προβλήματος, ο οποίος δεν παρέχει στο μοντέλο καμία υπόδειξη για την ενδεχόμενη λύση του προβλήματος. Αυτό έχει σαν τελικό αποτέλεσμα το μοντέλο να ξεκινάει από εντελώς τυχαίες δοκιμές, και μέσω της προσπάθειάς του να μεγιστοποιήσει το κέρδος (την ανταμοιβή), να καταλήγει με εξελιγμένες ικανότητες και τεχνικές.

Τέλος, υπάρχει και ένα ακόμα συχνά εμφανιζόμενο είδος Μηχανικής Μάθησης, που τοποθετείται ανάμεσα στην Επιβλεπόμενη και Μη Επιβλεπόμενη Μάθηση, το οποίο είναι γνωστό ως **Ημι-Επιβλεπόμενη Μάθηση (Semi-Supervised Learning)**, όπου το μοντέλο πρέπει να διαχειριστεί ταυτόχρονα τόσο δεδομένα που διαθέτουν ετικέτα κατηγοριοποίησης, όσο και δεδομένα που δεν διαθέτουν ετικέτα κατηγοριοποίησης.



Σχήμα 1: Κατηγορίες Μηχανικής Μάθησης και Πεδία Εφαρμογής [60]

3.2 Βαθιά Μάθηση

Η Βαθιά Μάθηση αποτελεί μια υποκατηγορία της Τεχνητής Νοημοσύνης που επικεντρώνεται στην ανάπτυξη αλγορίθμων και μοντέλων που μιμούνται τη λειτουργία του ανθρώπινου εγκεφάλου για την επεξεργασία πληροφορίας. Συγκεκριμένα, η βαθιά μάθηση επικεντρώνεται στη χρήση και ανάπτυξη πολύπλοκων αρχιτεκτονικών με πολλαπλά επίπεδα (γνωστές και ως βαθιά νευρωνικά δίκτυα) για την εκμάθηση και ανάλυση πολύπλοκων μοτίβων από δεδομένα. Οι δύο βασικότερες διαφορές που εμφανίζονται ανάμεσα στην Βαθιά Μάθηση και την Μηχανική Μάθηση είναι το μέγεθος των χρησιμοποιούμενων μοντέλων, καθώς και η εξαγωγή των χαρακτηριστικών από τα δεδομένα. Πιο συγκεκριμένα, στην παραδοσιακή Μηχανική Μάθηση, οι αλγόριθμοι τροφοδοτούνται με ένα «έτοιμο» σύνολο χαρακτηριστικών προς ανάλυση, ενώ στην Βαθιά Μάθηση, οι αλγόριθμοι, μέσω της ανάλυσης ενός τεράστιου όγκου δεδομένων, είναι σε θέση να κρίνουν μόνοι τους την σημασία και την συνεισφορά των χαρακτηριστικών, και να χρησιμοποιούν εν τέλει αυτά τα οποία θεωρούν καταλληλότερα. Η Βαθιά Μάθηση, με τις ολοένα και περισσότερες καινοτομίες που παρουσιάζονται στον κλάδο της, έχει επιφέρει αξιοσημείωτες επιδόσεις σε πολλά πεδία, όπως η Όραση Υπολογιστών, η Επεξεργασία Φυσικής Γλώσσας, η Αναγνώριση Εικόνας και Φωνής, η Ανάλυση Συναισθημάτων, η Αυτόνομη Οδήγηση, καθώς και πολλά άλλα.

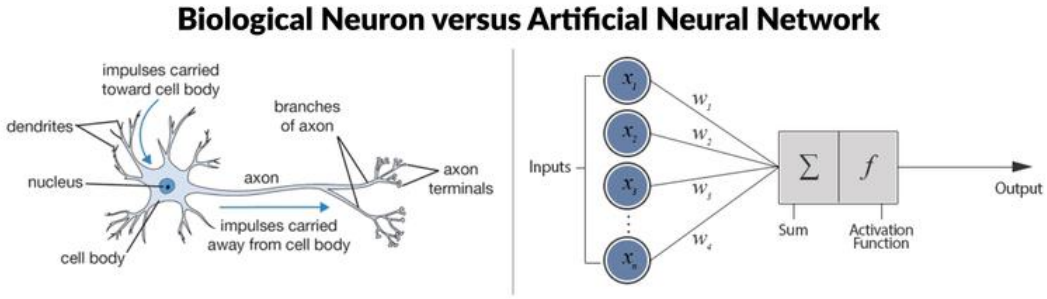
3.2.1 Τεχνητά Νευρωνικά Δίκτυα

Τα Τεχνητά Νευρωνικά Δίκτυα (Artificial Neural Networks - ANNs) είναι υπολογιστικά συστήματα εμπνευσμένα από τα βιολογικά νευρωνικά δίκτυα, που αποτελούν τους

ανθρώπινους εγκεφάλους. Η ανάπτυξη των Τεχνητών Νευρωνικών Δικτύων βασίστηκε στην σημαντική διαφορά που εμφανίζει η αλληλουχία ενεργειών και εργασιών που απαιτούνται για την εκτέλεση υπολογισμών στον ανθρώπινο εγκέφαλο, από την αντίστοιχη διαδικασία σε έναν συμβατικό ψηφιακό υπολογιστή. Ο εγκέφαλος είναι ένα εξαιρετικά πολύπλοκο σύστημα επεξεργασίας πληροφοριών, το οποίο δύναται να επεξεργάζεται παράλληλα και μη γραμμικά τα δεδομένα εισόδου, μέσω της κατάλληλης οργάνωσης των δομικών του στοιχείων, δηλαδή των νευρώνων. Ένα Νευρωνικό Δίκτυο προσομοιώνει την λειτουργία του βιολογικού νευρωνικού δικτύου, αποτελώντας έτσι μια απλοποιημένη μορφή του τρόπου με τον οποίο ο ανθρώπινος εγκέφαλος επεξεργάζεται τις πληροφορίες, ικανή να εκπαιδεύεται, να βελτιώνεται και να προσαρμόζεται στις εκάστοτε συνθήκες.

Ένα Νευρωνικό Δίκτυο αποτελείται συνήθως από τρία δομικά στοιχεία – επίπεδα. Το πρώτο αποτελεί το επίπεδο εισόδου, που αποτελείται από μονάδες που αντιπροσωπεύουν τα πεδία εισόδου x_i. Το δεύτερο δομικό στοιχείο αποτελείται από ένα ή περισσότερα κρυφά επίπεδα h_i, ενώ το τελευταίο αποτελεί το επίπεδο εξόδου, με μια ή περισσότερες μονάδες y_i που αντιπροσωπεύουν το πεδίο στόχο ή τα πεδία στόχους. Τα συνδετικά στοιχεία μεταξύ των μονάδων ονομάζονται βάρη σύνδεσης, w_i, τα οποία μαθαίνονται κατά την διάρκεια της εκπαίδευσης του μοντέλου και καθορίζουν την ισχύ του κάθε νευρώνα, καθώς και την επίδρασή του στους υπόλοιπους.

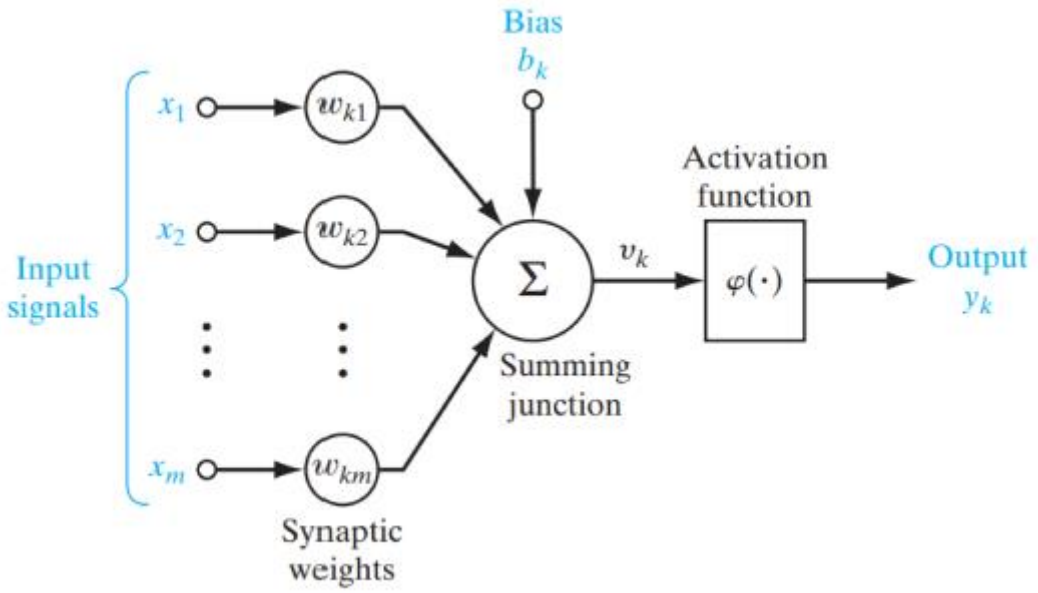
Ουσιαστικά, τα δεδομένα εισόδου εισάγονται στο πρώτο επίπεδο και οι τιμές $w_i x_i$ διαδίδονται από κάθε νευρώνα σε κάθε νευρώνα του επόμενου επιπέδου. Κατά τη διάδοση των τιμών αυτών μέσα στο Νευρωνικό Δίκτυο, αναλόγως την είσοδο σε κάθε επίπεδο ενεργοποιείται κάθε φορά ένας νευρώνας, μέσω μιας Συνάρτησης Ενεργοποίησης f (Activation Function). Αναλυτικότερα για τις συναρτήσεις ενεργοποίησης θα δούμε σε επόμενη ενότητα.



Σχήμα 2: Βιολογικοί Νευρώνες (αριστερά) και Μαθηματική Αναπαράστασή τους (δεξιά) [61]

3.2.1.1 Single-Layer Perceptron (SLP)

Η απλούστερη μορφή τεχνητού νευρώνα είναι το **Μονοεπίπεδο Perceptron (Single Layer Perceptron - SLP)**, το οποίο αποκαλείται και ως «το Perceptron του Rosenblatt» [62].



Σχήμα 3: Single-Layer Perceptron [63]

Αρχικά, στο διάγραμμα φαίνονται τα σήματα εισόδου (input signals) $x_1, x_2, \dots, x_m$, τα οποία σε συνδυασμό με τα συναπτικά βάρη (synaptic weights) αποτελούν ένα σύνολο συνάψεων που καταλήγουν στον κόμβο άθροισης (summing junction). Στον κόμβο αυτό, αθροίζονται τα σήματα εισόδου, σταθμισμένα από τα αντίστοιχα συναπτικά βάρη του νευρώνα, εφόσον μια οποιαδήποτε είσοδος x_j πολλαπλασιάζεται με το αντίστοιχο βάρος w_{kj} της σύναψης j που συνδέεται με τον νευρώνα k. Γενικότερα, το συναπτικό βάρος ενός τεχνητού νευρώνα έχει την δυνατότητα να λάβει τόσο αρνητικές, όσο και θετικές τιμές. Στον κόμβο αυτόν φαίνεται και μια επιπλέον είσοδος, η οποία είναι η εξωτερικά εφαρμοζόμενη πόλωση (bias) b_k. Η πόλωση έχει ως αποτέλεσμα την αύξηση ή μείωση της δικτυακής διέγερσης της συνάρτησης ενεργοποίησης, ανάλογα με το εάν είναι θετική ή αρνητική αντίστοιχα. Τέλος, η έξοδος του αθροιστή αποτελεί είσοδο στην συνάρτηση ενεργοποίησης $\varphi(\cdot)$, η οποία χρησιμοποιείται για τον περιορισμό του πλάτους του σήματος εξόδου ενός νευρώνα, το οποίο τυπικά έχει ως κανονικοποιημένο εύρος τιμών είτε το διάστημα [0, 1], είτε το διάστημα [-1, 1].

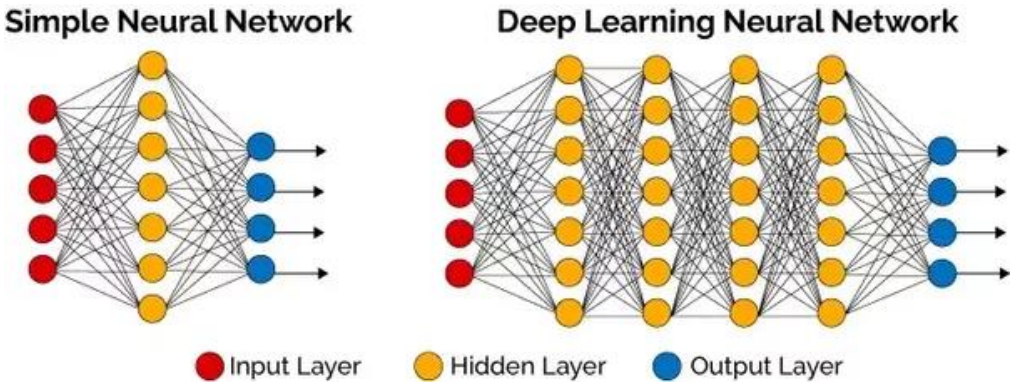
Από τα παραπάνω μπορούν να προκύψουν οι εξής μαθηματικές εξισώσεις για τις εξόδους v_k του κόμβου άθροισης και y_k του νευρώνα, αντίστοιχα:

$$v_k = \sum_{j=1}^{m} w_{kj} x_j + b_k \text{ και } y_k = \varphi(v_k).$$

3.2.1.2 Multi-Layer Perceptron (MLP)

Η απλούστερη περίπτωση αρχιτεκτονικών που συνδυάζουν τεχνητούς νευρώνες είναι το **Πολυεπίπεδο Perceptron (Multi-Layered Perceptron - MLP)**, το οποίο αποτελείται από το επίπεδο εισόδου, τουλάχιστον ένα κρυφό επίπεδο και το επίπεδο εξόδου. Κάθε κόμβος, τόσο των κρυφών επιπέδων, όσο και του επιπέδου εξόδου, έχει την δυνατότητα να διακρίνει μη γραμμικά διαχωρίσιμα δεδομένα, χρησιμοποιώντας μια μη γραμμική συνάρτηση ενεργοποίησης. Ένα Multi-Layered Perceptron, το οποίο αποτελείται από περισσότερα του

ενός κρυφά επίπεδα, αποτελεί ένα **Βαθύ Νευρωνικό Δίκτυο (Deep Neural Network - DNN)**. Λόγω της επεξεργασίας που διεξάγεται σε κάθε ένα από τα κρυφά επίπεδα ενός Deep Neural Network, το αποτέλεσμα της οποίας τροφοδοτείται στο αμέσως επόμενο κρυφό επίπεδο, είναι δυνατή η δημιουργία αναπαραστάσεων υψηλότερου επιπέδου και μεγαλύτερης ανάλυσης. Η τελική αναπαράσταση, με την σειρά της, τροφοδοτείται στο τελευταίο επίπεδο, που είναι το επίπεδο εξόδου, το οποίο είναι υπεύθυνο κάθε φορά για να λαμβάνει την τελική απόφαση πάνω στο εκάστοτε υπό εξέταση πρόβλημα μάθησης, και το οποίο αποτελείται κάθε φορά από τόσους νευρώνες, όσος είναι και ο αριθμός των κλάσεων που επιθυμούμε να προβλέψουμε.



Σχήμα 4: Απλό Νευρωνικό Δίκτυο (αριστερά) και Βαθύ Νευρωνικό Δίκτυο (δεξιά) [64]

3.2.2 Συναρτήσεις Ενεργοποίησης

Στην ενότητα αυτή θα μελετηθούν οι πιο συνήθεις μη γραμμικές συναρτήσεις ενεργοποίησης. Η συμβολή της μη γραμμικότητας που εισάγουν οι συναρτήσεις αυτές είναι καθοριστική στην μοντελοποίηση φαινομένων και συστημάτων που είναι από την φύση τους μη γραμμικά από Τεχνητά Νευρωνικά Δίκτυα. Στην περίπτωση που δεν εφαρμόζονταν, τότε το σήμα εξόδου θα εκφυλλιζόταν σε μια απλή γραμμική συνάρτηση και το Νευρωνικό Δίκτυο θα συμπεριφερόταν ως ένα μοντέλο γραμμικής παλινδρόμησης με περιορισμένες δυνατότητες μάθησης, εφόσον δεν θα μπορούσε να διακρίνει μη γραμμικά διαχωρίσιμα δεδομένα ή να μάθει μη γραμμικές καταστάσεις.

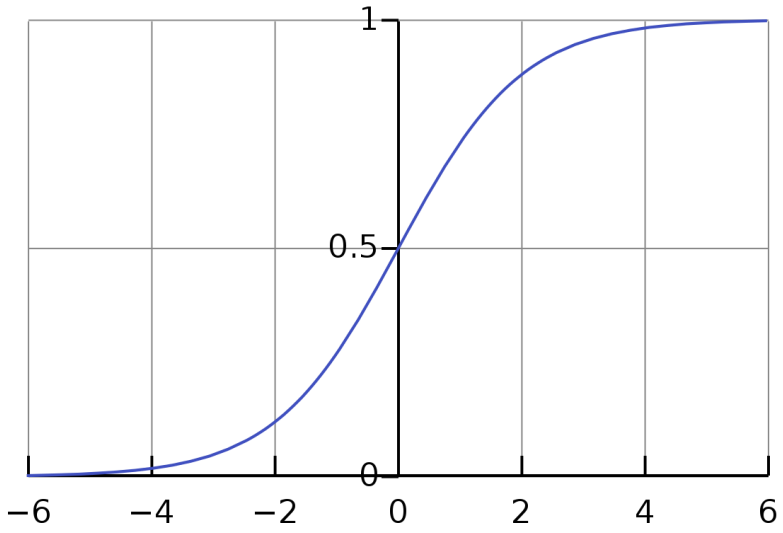
Η Σιγμοειδής Συνάρτηση

Η σιγμοειδής συνάρτηση (sigmoid function) δίνεται από τον τύπο:

$$f(x) = \sigma(x) = \frac{1}{1 + e^{-x}}$$

Η συνάρτηση αυτή, λαμβάνοντας ως είσοδο μία πραγματική τιμή, απεικονίζει στην έξοδο έναν πραγματικό αριθμό στο διάστημα [0, 1]. Είναι μια αρκετά διαδεδομένη συνάρτηση ενεργοποίησης, εφόσον μικρές μεταβολές της εισόδου x οδηγούν σε μεγάλες μεταβολές της εξόδου y, επιτρέποντας έτσι στο δίκτυο να αντιλαμβάνεται ευκολότερα μικρές μεταβολές των χαρακτηριστικών εισόδου. Ως «αδυναμία» της συνάρτησης αυτής θα μπορούσαμε να

παραθέσουμε το γεγονός πως σε κάθε «ουρά» στο 0 ή στο 1, οι τιμές της παραγώγου της είναι πολύ μικρές, συγκλίνοντας στο 0, με αποτέλεσμα τα διανύσματα κλίσης να «εξαφανίζονται» (φαινόμενο γνωστό ως Vanishing Gradient), περιορίζοντας έτσι τις δυνατότητες μάθησης του μοντέλου.



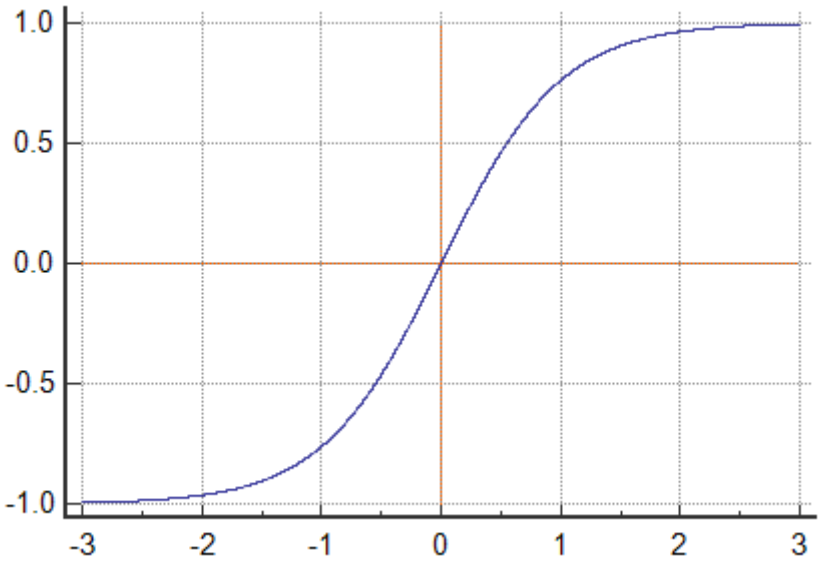
Σχήμα 5: Γραφική Παράσταση της Σιγμοειδούς Συνάρτησης [65]

Η Συνάρτηση Υπερβολικής Εφαπτομένης

Η συνάρτηση υπερβολικής εφαπτομένης (tanh) δίνεται από τον τύπο:

$$f(x) = tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

Οι τιμές εξόδου της συνάρτησης υπερβολικής εφαπτομένης tanh βρίσκονται στο διάστημα [-1, 1]. Ως βασικό προτέρημά της σε βάρος της σιγμοειδούς θα μπορούσαμε να παραθέσουμε το γεγονός ότι η παράγωγός της είναι περισσότερο απότομη, κάτι που δύναται να οδηγήσει σε μεγαλύτερες τιμές εξόδου, προσφέροντας έτσι περισσότερες δυνατότητες για γρήγορη μάθηση και κατάβαση κλίσης. Ωστόσο, παραμένει και στην συνάρτηση αυτή το πρόβλημα σύγκλισης της κλίσης στο 0 κοντά στις δύο «ουρές».



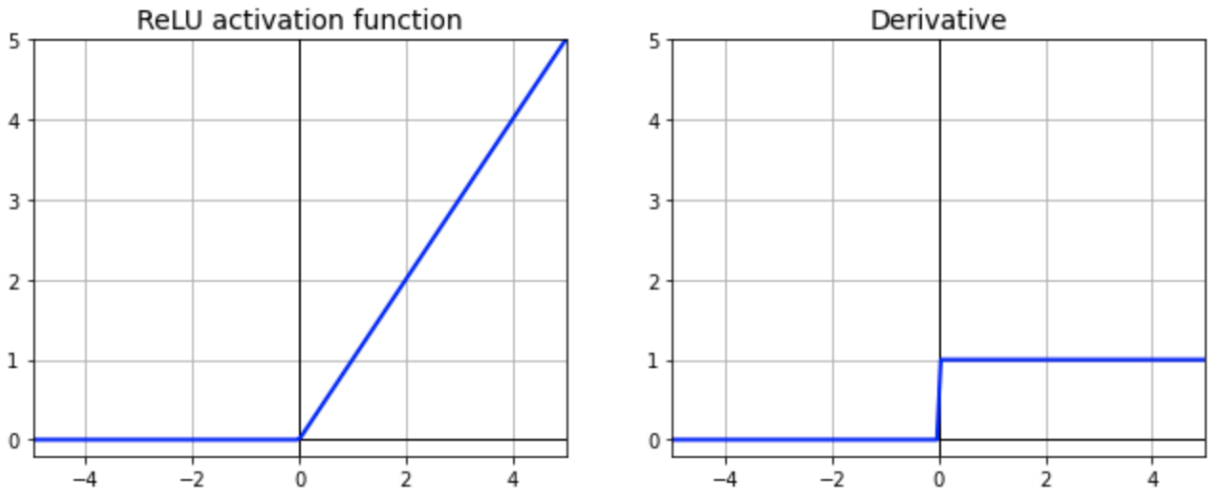
Σχήμα 6: Γραφική Παράσταση της Συνάρτησης Υπερβολικής Εφαπτομένης [66]

Η Συνάρτηση Rectified Linear Unit (ReLU)

Η συνάρτηση ενεργοποίησης Rectified Linear Unit (ReLU) αποτελεί την πιο διαδεδομένη συνάρτηση ενεργοποίησης και χρησιμοποιείται σε αλγορίθμους Βαθιάς Μάθησης (Deep Learning) και σε Συνελικτικά Νευρωνικά Δίκτυα (CNNs). Αντιστοιχεί όλες τις τιμές εισόδου στο διάστημα $[0, \infty)$ και δίνεται από τον τύπο:

$$f(x) = \begin{cases} 0, & \varepsilon\acute{\alpha}\nu\ x \leq 0 \\ x, & \varepsilon\acute{\alpha}\nu\ x > 0 \end{cases}$$

Σε αντίθεση με τις δύο προηγούμενες συναρτήσεις ενεργοποίησης, δεν περιλαμβάνει περίπλοκες υπολογιστικές πράξεις, έχοντας ως αποτέλεσμα να συγκλίνει πιο γρήγορα. Ταυτόχρονα, λόγω της «αραιής» ενεργοποίησης, οι νευρώνες του δικτύου μαθαίνουν πιο σημαντικά χαρακτηριστικά του προβλήματος. Ωστόσο, σε περίπτωση αρνητικής εισόδου, όπου θα έχουμε $f(x) = 0$, η παράγωγος θα είναι επίσης μηδενική, με αποτέλεσμα τα βάρη να μην ανανεώνονται, προκαλώντας με αυτόν τον τρόπο την θανάτωση του νευρώνα (dying ReLU Problem). Τέλος, η συνάρτηση ReLU μπορεί να προκαλέσει φαινόμενα Exploding Gradients, κατά τα οποία οι τιμές των παραγώγων αυξάνονται με ραγδαίο ρυθμό.



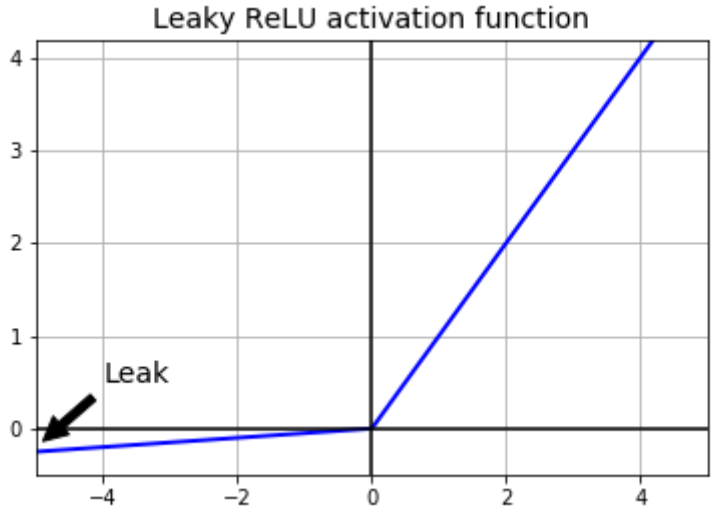
Σχήμα 7: Γραφική Παράσταση της Συνάρτησης ReLU και της Παραγώγου της [67]

Η Συνάρτηση Leaky Rectified Linear Unit (Leaky ReLU)

Η συνάρτηση ενεργοποίησης Leaky ReLU αποτελεί μια προσπάθεια επίλυσης του προβλήματος του dying ReLU, που αναφέρθηκε προηγουμένως. Για τον σκοπό αυτό, αντιστοιχεί όλες τις τιμές εισόδου στο διάστημα $(-\infty, +\infty)$, εφόσον πλέον οι αρνητικές τιμές πολλαπλασιάζονται με μια πολύ μικρή σταθερά c=0.01, αντί να μηδενίζονται, όπως συνέβαινε πριν. Δίνεται από τον τύπο:

$$f(x) = \begin{cases} 0.01x, & \text{εάν } x < 0 \\ x, & \text{εάν } x \geq 0 \end{cases}$$

Η γραφική παράσταση της συνάρτησης δίνεται στο ακόλουθο σχήμα:



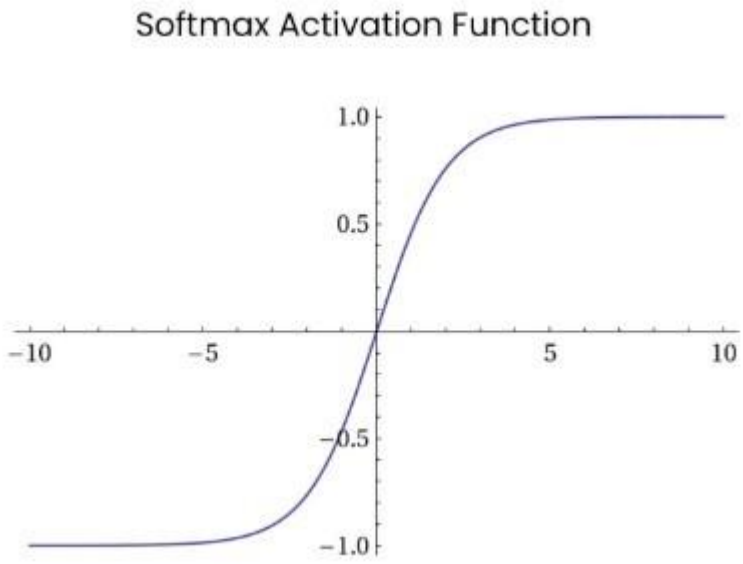
Σχήμα 8: Γραφική Παράσταση της Συνάρτησης Leaky ReLU [68]

Η Συνάρτηση Softmax

Η συνάρτηση ενεργοποίησης Softmax χρησιμοποιείται τόσο σε προβλήματα δυαδικής ταξινόμησης, όσο και σε προβλήματα με περισσότερες κλάσεις. Αντιστοιχεί τους αριθμούς της εισόδου που δέχεται σε πιθανότητες, το άθροισμα των οποίων είναι ίσο με 1. Δίνεται από τον τύπο:

$$Softmax(z_j) = \frac{e^{z_j}}{\sum_{j=0}^{N} e^{z_j}}$$

Η γραφική παράσταση της συνάρτησης δίνεται στο ακόλουθο σχήμα:



Σχήμα 9: Γραφική Παράσταση της Συνάρτησης Softmax [69]

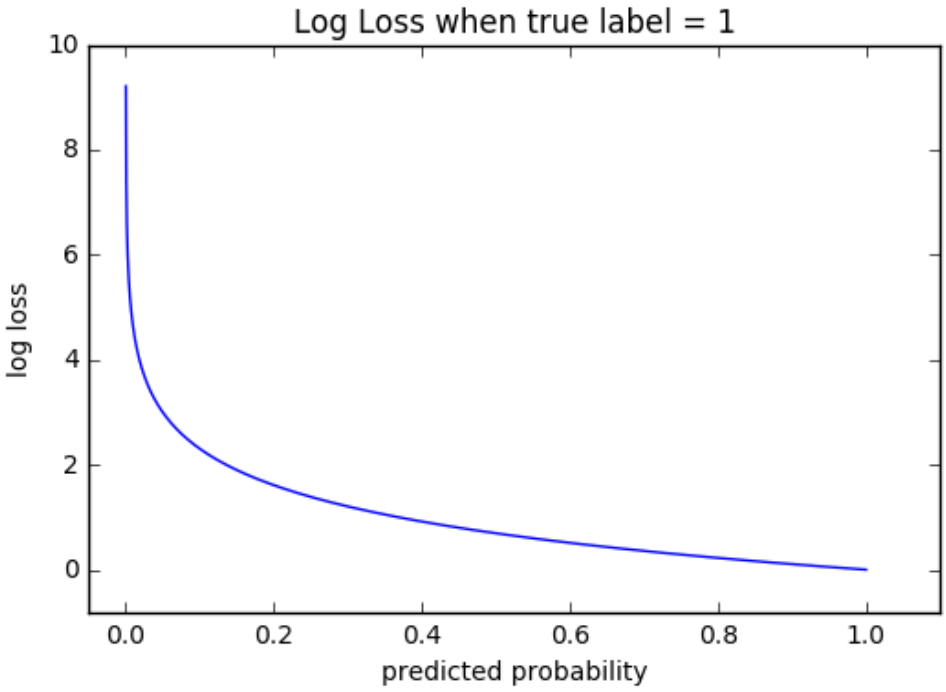
3.2.3 Συναρτήσεις Κόστους

Σε ένα νευρωνικό δίκτυο, σκοπός μας είναι να ελαχιστοποιήσουμε το κόστος, κάτι το οποίο επιτυγχάνουμε βελτιστοποιώντας τα βάρη του. Για την βελτιστοποίηση αυτήν των βαρών και των παραμέτρων του δικτύου, χρησιμοποιούμε την Συνάρτηση Κόστους (Loss Function), η οποία συγκρίνει την πραγματική (actual) με την προβλεπόμενη (predicted) τιμή που έβγαλε το μοντέλο μας. Στην δική μας περίπτωση, όπως θα δούμε σε επόμενο κεφάλαιο, χρησιμοποιήσαμε την μέθοδο Cross Entropy Loss. Ωστόσο, παρακάτω παρουσιάζουμε γενικότερα τις πιο γνωστές συναρτήσεις κόστους:

- **Cross Entropy Loss & Logistic Regression**

Η συγκεκριμένη συνάρτηση κόστους υπολογίζει την επίδοση ενός μοντέλου ταξινομητή, του οποίου η έξοδος είναι μια τιμή πιθανότητας μεταξύ του 0 και το 1. Όσο η πιθανότητα της πρόβλεψης αποκλίνει από την πραγματική τιμή, τόσο αυξάνεται και το Cross Entropy Loss, ενώ είναι 0 σε ένα ιδανικό μοντέλο ταξινόμησης όπου η κατηγοριοποίηση γίνεται

πάντοτε ορθά. Αυτό μπορεί να παρατηρηθεί και στην παρακάτω εικόνα, όπου θεωρούμε ως πραγματική τιμή την 1:



Σχήμα 10: Cross Entropy Loss [70]

Πράγματι, στην περίπτωση που τόσο η πραγματική τιμή, όσο και η προβλεπόμενη από το μοντέλο τιμή είναι 1, το κόστος είναι ίσο με 0, ενώ αυξάνεται με την αύξηση της απόκλισης των δυο τιμών.

Η συνάρτηση κόστους Cross Entropy Loss (για δυαδική ταξινόμηση) δίνεται από τον τύπο:

$$Loss = \frac{1}{m}[\sum_{i=1}^{m} -y^{(i)} \log\left(h_\theta\left(x^{(i)}\right)\right) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)}))]$$

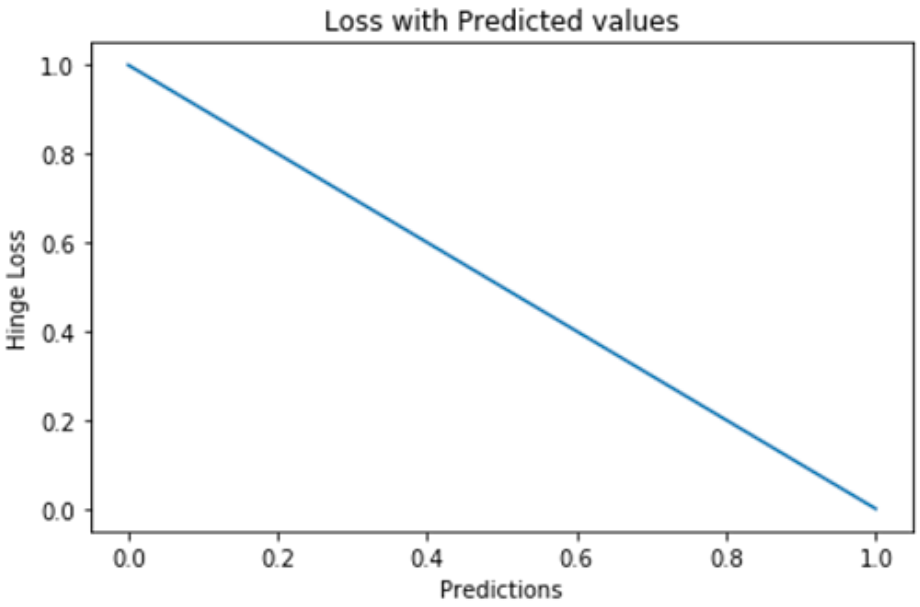
όπου m είναι ο αριθμός των δεδομένων.

- **Hinge Loss / SVM Loss**

Η συνάρτηση κόστους Hinge Loss χρησιμοποιείται επίσης συχνά σε προβλήματα ταξινόμησης. Αναπτύχθηκε κυρίως για την αξιολόγηση του μοντέλου Support Vector Machine (SVM), και για αυτό ονομάζεται και SVM Loss. Δίνεται από τον τύπο:

$$HingeLoss = \sum_{j \neq y_i} \max(0, s_j - s_{yi} + 1)$$

Σαν συνάρτηση «τιμωρεί» τις λάθος προβλέψεις, καθώς και τις σωστές προβλέψεις που δεν είναι «βέβαιες». Στους ταξινομητές SVM, στους οποίους χρησιμοποιείται κυρίως, έχει ως ετικέτες των κλάσεων (class labels) τα -1 και 1.



Σχήμα 11: Hinge Loss / SVM Loss [70]

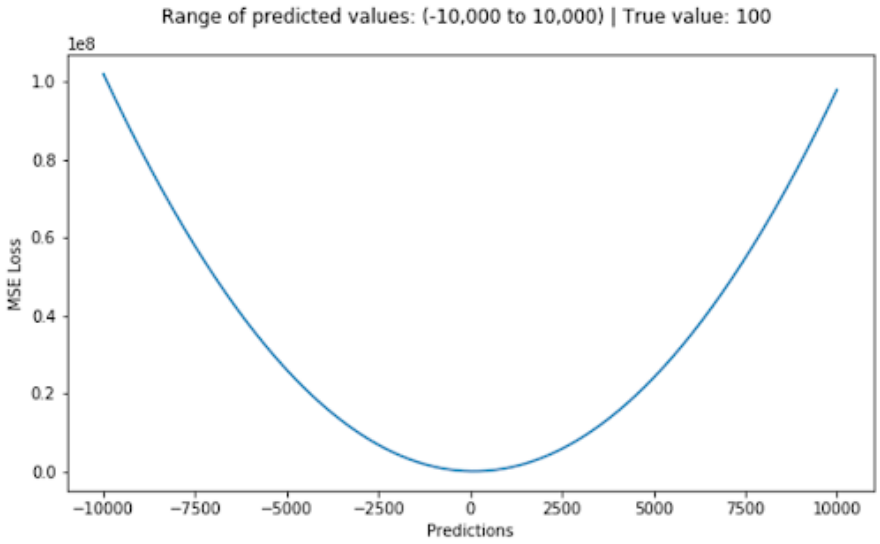
- **Μέσο Τετραγωνικό Σφάλμα (Mean Squared Error)**

Σε προβλήματα παλινδρόμησης (regression), το Μέσο Τετραγωνικό Σφάλμα είναι η συνάρτηση κόστους που χρησιμοποιείται κυρίως. Υπολογίζεται παίρνοντας τον μέσο όρο των τετραγώνων των διαφορών της πραγματικής από την προβλεπόμενη τιμή. Συγκεκριμένα, δίνεται από τον τύπο:

$$J = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Λόγω του τετραγωνισμού, η συνάρτηση αυτή τιμωρεί το μοντέλο όταν κάνει προβλέψεις με μεγάλες αποκλίσεις από τις αντίστοιχες πραγματικές τιμές, γεγονός που την καθιστά λιγότερο εύρωστη σε outliers (ακραίες τιμές).

Στην παρακάτω γραφική παράσταση παρουσιάζεται η συνάρτηση κόστους, σε ένα σενάριο όπου ως πραγματική τιμή έχουμε την 100, και ως προβλεπόμενη τιμή έχουμε ένα εύρος τιμών που κυμαίνεται από -10.000 έως 10.000.



Σχήμα 12: Mean Squared Error [70]

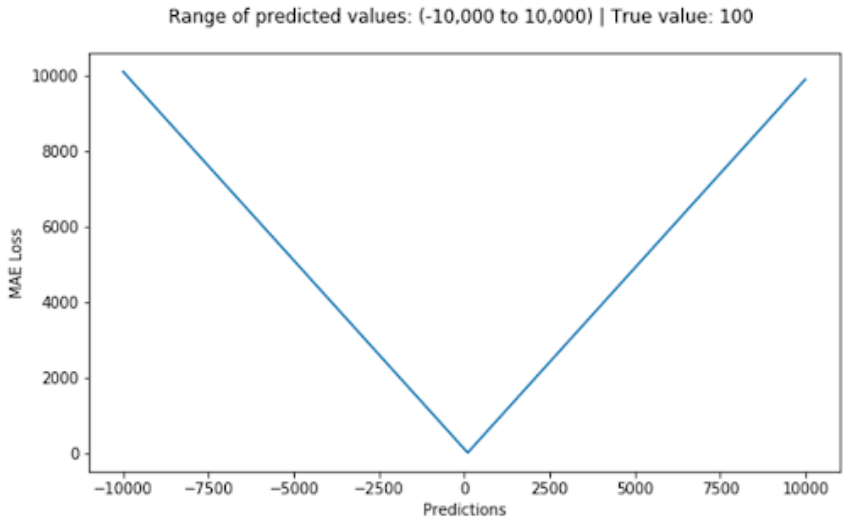
- **Μέσο Απόλυτο Σφάλμα (Mean Absolute Error)**

Το Μέσο Απόλυτο Σφάλμα αποτελεί την δεύτερη πιο συνήθης συνάρτηση κόστους που χρησιμοποιείται σε προβλήματα παλινδρόμησης (regression). Υπολογίζεται παίρνοντας τον μέσο όρο των απόλυτων διαφορών μεταξύ των πραγματικών και των προβλεπόμενων τιμών. Δίνεται από τον τύπο:

$$MAE = \frac{\sum_{i=1}^{n} |y_i - \hat{y_i}|}{n}$$

Συγκριτικά με την Mean Squared Error, η Mean Absolute Error είναι πιο εύρωστη σε outliers, λόγω της απόλυτης τιμής που υπάρχει στον τύπο.

Στην παρακάτω γραφική παράσταση παρουσιάζεται η μορφή της συνάρτησης κόστους, στο ίδιο σενάριο με αυτό το οποίο υποθέσαμε προηγουμένως.



Σχήμα 13: Mean Absolute Error [70]

Ενδεικτικά, άλλες γνωστές συναρτήσεις κόστους που χρησιμοποιούνται είναι οι **Huber Loss**, γνωστή και ως Smooth Mean Absolute Error, **Log-Cosh Loss**, **Quantile Loss**, **KL-Divergence (Kullback-Leibler Divergence)** και **RMSE (Root Mean Squared Error)**.

3.2.4 Αλγόριθμος Οπίσθιας Διάδοσης (Backpropagation)

Ο αλγόριθμος Οπίσθιας Διάδοσης Σφαλμάτων (Backpropagation) αποτελεί μια μέθοδο ελαχιστοποίησης της συνάρτησης κόστους, ανανεώνοντας τα βάρη του νευρωνικού δικτύου. Η κατάλληλη προσαρμογή των βαρών του δικτύου είναι ένα στοιχείο πάνω στο οποίο βασίζεται η διαδικασία εκπαίδευσης των νευρωνικών δικτυών, έτσι ώστε η έξοδός τους να συγκλίνει όσο το δυνατόν περισσότερο στην αναμενόμενη. Η πιο διαδεδομένη και ευρέως χρησιμοποιούμενη μέθοδος ανανέωσης των βαρών του δικτύου, και συνεπώς εύρεσης του ελαχίστου μιας συνάρτησης κόστους, είναι η μέθοδος **Κατάβασης Δυναμικού (Gradient Descent)**. Δίνεται από τον ακόλουθο τύπο:

$$\theta = \theta_{nom} - \alpha \frac{\partial J}{\partial \theta_{nom}}$$

όπου α είναι ο ρυθμός μάθησης (learning rate), θ_{nom} είναι η τρέχουσα τιμή του βάρους, ενώ θ είναι η νέα τιμή του βάρους.

Προκειμένου να ελαχιστοποιηθεί η συνάρτηση κόστους, τα βάρη του δικτύου ανανεώνονται πολλαπλές φορές, καθιστώντας έτσι την μέθοδο Gradient Descent μια επαναληπτική μέθοδο. Διακρίνεται στις ακόλουθες κατηγορίες [71]:

- **Batch Gradient Descent**

Αποτελεί μια κατηγορία Κατάβασης Δυναμικού, κατά την οποία σε κάθε επανάληψη (iteration) γίνεται επεξεργασία όλων των δεδομένων στο σύνολο εκπαίδευσης. Όμως, αν ο αριθμός των δεδομένων στο σύνολο εκπαίδευσης είναι ιδιαίτερα μεγάλος, τότε η μέθοδος αυτή καθίσταται εξαιρετικά ακριβή υπολογιστικά.

- **Stochastic Gradient Descent**

Στην περίπτωση αυτή, σε κάθε επανάληψη γίνεται επεξεργασία κάθε φορά ενός δεδομένου από το σύνολο εκπαίδευσης. Έτσι, ύστερα από κάθε επανάληψη, η ανανέωση των παραμέτρων γίνεται με βάση την επεξεργασία που διεξήχθη σε αυτό το συγκεκριμένο δεδομένο. Ωστόσο, αν ο αριθμός των δεδομένων στο σύνολο εκπαίδευσης είναι μεγάλος, αυτό θα έχει σαν αποτέλεσμα η μέθοδος αυτή να εκτελεί πολλές επαναλήψεις.

- **Mini Batch Gradient Descent**

Η μέθοδος αυτή θεωρείται γρηγορότερη από τις δυο προαναφερθείσες, και προτιμάται η χρήση της σε περιπτώσεις που ο αριθμός των δεδομένων είναι πολύ μεγάλος, ιδιαίτερα σε προβλήματα Βαθιάς Μάθησης. Εν προκειμένω, ο συνολικός αριθμός των δεδομένων του συνόλου εκπαίδευσης, έστω m δείγματα, χωρίζεται σε batches, έστω n σε αριθμό, που το καθένα έχει μέγεθος $\frac{m}{n}$ δείγματα. Έτσι, ο αριθμός των επαναλήψεων είναι ίσος με τον

αριθμό των batches, n, και ο αριθμός των δεδομένων του συνόλου εκπαίδευσης που υφίσταται επεξεργασία σε κάθε επανάληψη είναι ίσος με $\frac{m}{n}$ δείγματα.

Ωστόσο, τα τελευταία χρόνια, εκτός από τις παραπάνω μεθόδους, γίνεται ολοένα και περισσότερο διαδεδομένη η χρήση τεχνικών αυτόματης βελτιστοποίησης, στις οποίες η ρύθμιση του ρυθμού μάθησης γίνεται αυτόματα. Χαρακτηριστικά παραδείγματα τέτοιων μεθόδων αποτελούν οι **Adagrad**, **Adadelta** και **Adam**, η οποία μάλιστα είναι και η πιο διαδεδομένη και ευρέως χρησιμοποιούμενη τεχνική την στιγμή αυτή.

Ο βελτιστοποιητής **Adam (Adaptive Moment Estimation)** [72] χρησιμοποιεί έναν προσαρμοζόμενο ρυθμό μάθησης, γεγονός που σημαίνει πως διατηρεί έναν ρυθμό μάθησης για καθεμία από τις παραμέτρους του μοντέλου, προσαρμόζοντάς τον ξεχωριστά για κάθε βάρος. Ταυτόχρονα, η υπολογιστική του επίδοση σε συνδυασμό με τις χαμηλές απαιτήσεις του σε μνήμη τον καθιστούν ιδανικό για προβλήματα που προϋποθέτουν την εκμάθηση πολλών παραμέτρων, ενώ αποτελεί και τον βελτιστοποιητή που χρησιμοποιήσαμε στα πλαίσια της διπλωματικής αυτής, όπως θα δούμε και στην συνέχεια.

3.2.5 Μεταφορά Μάθησης

Όπως έχει γίνει κατανοητό, σε πολλά προβλήματα Βαθιάς Μάθησης η δημιουργία ενός μοντέλου με ικανοποιητική απόδοση απαιτεί την ύπαρξη και χρήση πληθώρας δεδομένων. Εντούτοις, σε αρκετά προβλήματα επιβλεπόμενης μάθησης, στα οποία είναι αναγκαία η χρήση επισημασμένων δεδομένων, δηλαδή δεδομένων που συνοδεύονται από την ετικέτα της κλάσης στην οποία ανήκουν, η πρόσβαση σε έναν ικανοποιητικό αριθμό από επισημασμένα δεδομένα είναι αρκετές φορές αδύνατη, λόγω χρονικών ή υπολογιστικών περιορισμών. Για τις περιπτώσεις αυτές, έχει αναπτυχθεί μια νέα τεχνική, η οποία ονομάζεται **Μεταφορά Μάθησης (Transfer Learning)**.

Η τεχνική της Μεταφοράς Μάθησης μας επιτρέπει να βελτιώσουμε την επίδοση ενός μοντέλου πάνω σε ένα συγκεκριμένο πρόβλημα μάθησης (στόχος), αξιοποιώντας γνώση που έχει αποκτηθεί κατά την διάρκεια εκπαίδευσης σε ένα άλλο πρόβλημα μάθησης (πηγή). Ουσιαστικά, εξαιτίας της προεκπαίδευσης πάνω στο πρόβλημα - πηγή, το μοντέλο έχει συγκεντρώσει πρότερη γνώση και έχει αποκτήσει την δυνατότητα να αναπτύσσει αναπαραστάσεις υψηλού επιπέδου. Με αυτόν τον τρόπο, εφαρμόζοντας μια διαδικασία fine-tuning πάνω στο πρόβλημα - στόχο, το μοντέλο όχι μόνο είναι σε θέση να προσαρμοστεί άμεσα στο πρόβλημα και να μάθει γρήγορα και αποδοτικά, αλλά επιτυγχάνει και υψηλότερες επιδόσεις αναλογικά με αυτές που θα πετύχαινε, αν δεν είχε προεκπαιδευθεί στο πρόβλημα - πηγή, δηλαδή αν δεν είχε εφαρμοσθεί η τεχνική του Transfer Learning.

3.3 Αρχιτεκτονικές Βαθιών Νευρωνικών Δικτύων

Στην ενότητα αυτή παρουσιάζονται κάποιες από τις βασικότερες αρχιτεκτονικές βαθιών νευρωνικών δικτύων, μέρος των οποίων χρησιμοποιούνται και στα πλαίσια της διπλωματικής αυτής. Για λόγους πληρότητας γίνεται αναφορά στα βαθιά νευρωνικά δίκτυα RNN, LSTM και

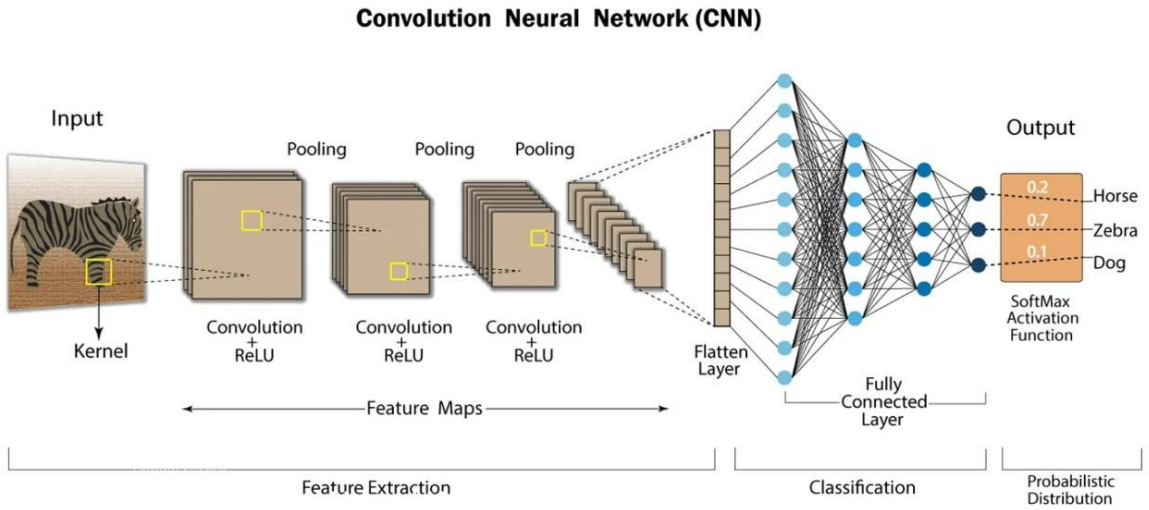
BiLSTM, ενώ καλύπτεται και το θεωρητικό υπόβαθρο γύρω από τα μοντέλα CNN, τους Μηχανισμούς Προσοχής και την αρχιτεκτονική των μοντέλων Transformer Encoder-Decoder, τα οποία χρησιμοποιήθηκαν για την εκπόνηση της εργασίας.

3.3.1 Convolutional Neural Networks (CNN)

Τα Συνελικτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks) είναι ένας τύπος αρχιτεκτονικής βαθιών νευρωνικών δικτύων που έχει σχεδιαστεί ειδικά για την επεξεργασία δεδομένων εικόνας και βίντεο, έχοντας ως στόχο την διαφοροποίηση και ανίχνευση των αντικειμένων που απεικονίζονται, αναθέτοντας βάρη και biases στα αντικείμενα αυτά. Αποτελεί μία από τις κυρίαρχες αρχιτεκτονικές αναφορικά με εφαρμογές που έχουν σχέση με την όραση υπολογιστών, όπως κατηγοριοποίηση και τμηματοποίηση εικόνων, καθώς και ανίχνευση αντικειμένων. Σε αντίθεση με άλλα νευρωνικά δίκτυα, τα οποία δέχονται μια ισοπεδωμένη είσοδο σταθερού μεγέθους, τα CNNs λαμβάνουν ως είσοδο ακατέργαστα pixel εικόνας και μαθαίνουν να εξάγουν από αυτά σχετικά χαρακτηριστικά μέσω συνελικτικών επιπέδων.

Τα Συνελικτικά Νευρωνικά Δίκτυα υπερισχύουν των παραδοσιακών Πλήρως Συνδεδεμένων Νευρωνικών Δικτύων σε δύο βασικά σημεία. Σε πρώτη φάση, σε ένα Πλήρως Συνδεδεμένο Δίκτυο (Fully Connected Network), όλοι οι νευρώνες του κάθε επιπέδου είναι συνδεδεμένοι με όλους τους νευρώνες του επόμενου. Συνεπώς, στην περίπτωση που η είσοδος του δικτύου είναι μία εικόνα, και πόσο μάλλον όταν αποτελείται από περισσότερα του ενός κανάλια, όπως είναι η περίπτωση των RGB εικόνων, ο αριθμός των συνδέσεων μεταξύ των νευρώνων, και συνεπώς ο όγκος των υπερπαραμέτρων, αυξάνονται δραματικά. Αυτό έχει αρνητικές συνέπειες, τόσο στην απαιτούμενη υπολογιστική ισχύ, όσο και στο πλήθος των δεδομένων που χρειάζονται για την διεξαγωγή μιας επιτυχημένης εκπαίδευσης. Αντιθέτως, σε ένα Συνελικτικό Νευρωνικό Δίκτυο, την θέση των πλήρως συνδεδεμένων επιπέδων παίρνουν συνελικτικά επίπεδα, στα οποία ο κάθε νευρώνας ενός επιπέδου συνδέεται με συγκεκριμένους μόνο νευρώνες του επόμενου, μέσα από την διαδικασία συνέλιξης με κατάλληλα φίλτρα. Τα φίλτρα αυτά όχι μόνο έχουν τα ίδια βάρη για όλους τους νευρώνες του ίδιου επιπέδου, αλλά είναι και οργανωμένα σε πλέγμα, έτσι ώστε η εφαρμογή τους να γίνεται σε διαφορετικές περιοχές της εικόνας, έχοντας ως άμεσο αποτέλεσμα την δραματική μείωση του αριθμού των απαιτούμενων παραμέτρων. Σε δεύτερη φάση, στην περίπτωση που τροφοδοτήσουμε ένα Πλήρως Συνδεδεμένο Δίκτυο με μια εικόνα, αυτή θα λειτουργεί σαν μονοδιάστατο διάνυσμα, στερώντας μας από την δυνατότητα να αξιοποιήσουμε τις χωρικές συσχετίσεις που εμφανίζονται μεταξύ των τιμών των γειτονικών pixel. Από την άλλη πλευρά, στα Συνελικτικά Νευρωνικά Δίκτυα είναι δυνατή η διατήρηση αυτής της χωρικής συσχέτισης, μέσω της χρήσης τεχνικών κυλιόμενων παραθύρων στα συνελικτικά επίπεδα.

Στην συνέχεια, παρουσιάζονται συνοπτικά τα διαφορετικά επίπεδα ενός Συνελικτικού Νευρωνικού Δικτύου.



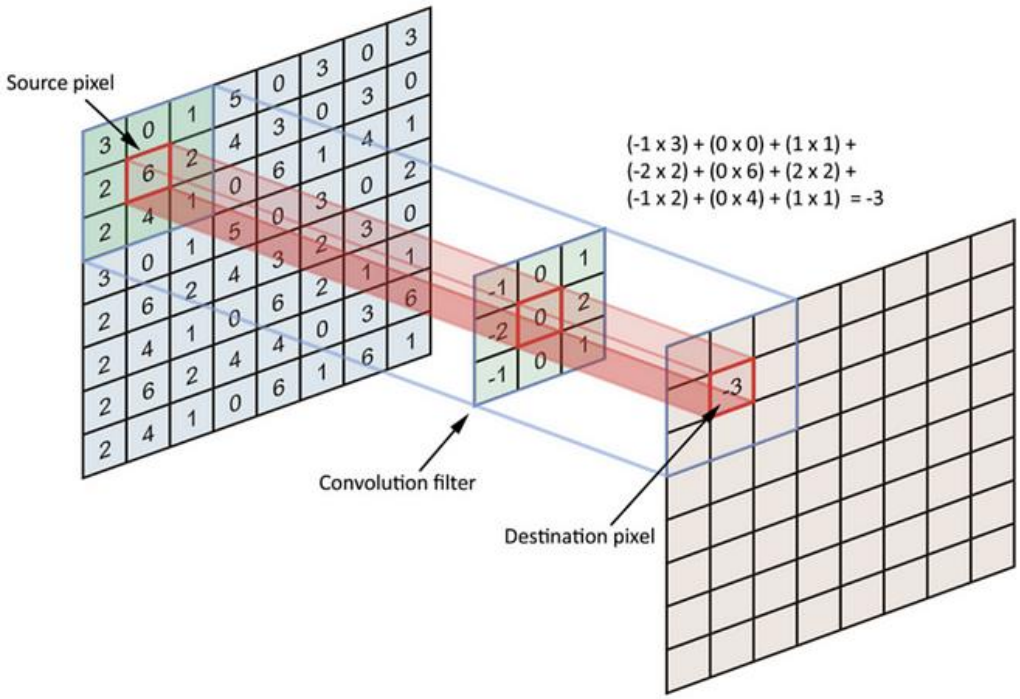
Σχήμα 14: Αρχιτεκτονική Απλού Συνελικτικού Νευρωνικού Δικτύου [73]

Συνελικτικό Επίπεδο (Convolutional Layer)

Για να περιγράψουμε την λειτουργία του Συνελικτικού Επιπέδου, πρέπει πρώτα να ορίσουμε κάποιες βασικές έννοιες. Η έννοια του **πυρήνα (kernel) ή φίλτρου (filter)** αποτελεί ουσιαστικά ένα «παράθυρο» συγκεκριμένης, μικρής διάστασης n x n, το οποίο έχει το ίδιο βάθος με την εικόνα, δηλαδή τον ίδιο αριθμό καναλιών. Ως **συνέλιξη** θεωρούμε την διαδικασία κατά την οποία σε κάθε σημείο της εικόνας, τα στοιχεία του πυρήνα πολλαπλασιάζονται με τα εικονοστοιχεία της αντίστοιχης περιοχής, παράγοντας ένα αποτέλεσμα το οποίο τοποθετείται στην κατάλληλη θέση του πίνακα εξόδου. Η έξοδος αυτή της συνέλιξης ονομάζεται χάρτης ενεργοποίησης (activation map) ή χάρτης χαρακτηριστικών (feature map), εφόσον η κάθε τιμή του χάρτη δίνει την πιθανότητα κατά την οποία το επιθυμητό χαρακτηριστικό βρίσκεται σε αυτήν την περιοχή της αρχικής εικόνας.

Τα βάρη του πυρήνα αποτελούν παραμέτρους, οι οποίες διαμορφώνονται κατά την εκπαίδευση του δικτύου. Αν και τυπικά το βήμα ολίσθησης κατά το οποίο ολισθαίνει ο πυρήνας, το οποίο ονομάζεται stride, είναι μόνο ένα, υπάρχουν περιπτώσεις που χρησιμοποιείται μεγαλύτερο.

Τέλος, συχνά ανά συνελικτικό επίπεδο χρησιμοποιούνται περισσότεροι του ενός πυρήνες, έχοντας ως αποτέλεσμα την δημιουργία πολλαπλών χαρτών ενεργοποίησης που αντιστοιχούν σε διαφορετικά χαρακτηριστικά, ένα για κάθε πυρήνα. Έτσι, η έξοδος του κάθε συνελικτικού επιπέδου αποτελεί ουσιαστικά μια τρισδιάστατη «εικόνα» μεγάλου βάθους, αποτελούμενη από διαφορετικούς χάρτες ενεργοποίησης.



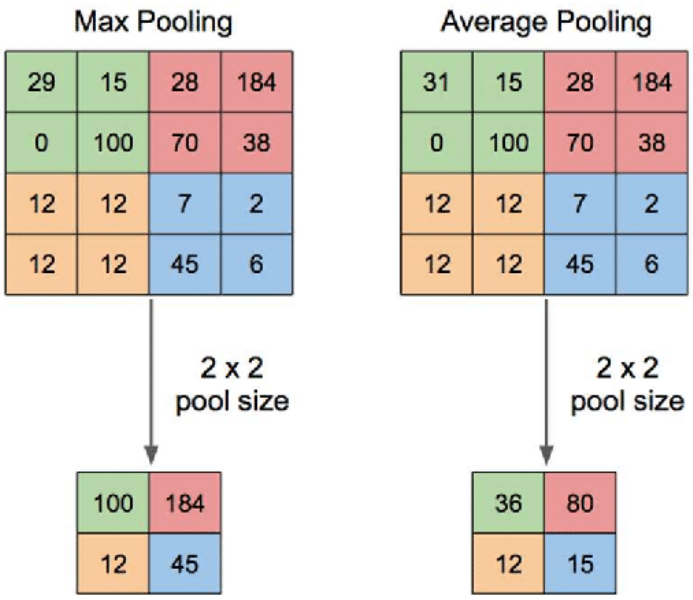
Σχήμα 15: Παράδειγμα Συνέλιξης σε CNN [74]

Επίπεδο Ενεργοποίησης (Activation Layer)

Οι συχνότερες περιπτώσεις χρήσης των Convolutional Neural Networks είναι σε πραγματικά συστήματα και με πραγματικά δεδομένα, γεγονός που συνεπάγεται πως η συμπεριφορά των συστημάτων αυτών δεν είναι απόλυτα γραμμική. Για την εισαγωγή αυτής της απαραίτητης μη-γραμμικότητας στο δίκτυο, μετά από κάθε συνελικτικό επίπεδο τοποθετείται ένα επίπεδο ενεργοποίησης, το οποίο εφαρμόζει στην έξοδο του συνελικτικού επιπέδου μια (μη γραμμική) συνάρτηση ενεργοποίησης. Όπως έχει αναφερθεί και στην ενότητα των Συναρτήσεων Ενεργοποίησης (3.2.2), η πιο ευρέως διαδεδομένη και χρησιμοποιούμενη συνάρτηση ενεργοποίησης είναι η Rectified Linear Unit (ReLU), εφόσον επιτυγχάνει την απλοποίηση του αλγόριθμου backpropagation, επιταχύνοντας έτσι την εκπαίδευση του δικτύου.

Επίπεδο Υποδειγματοληψίας (Pooling Layer)

Η ύπαρξη των Επιπέδων Υποδειγματοληψίας είναι αναγκαία για την μείωση των διαστάσεων των χαρτών ενεργοποίησης που προκύπτουν από τα συνελικτικά επίπεδα, όπως αναλύθηκε παραπάνω. Αφού τεμαχίσουν τον χάρτη ενεργοποίησης σε μη επικαλυπτόμενα τμήματα, κρατάνε για το κάθε ένα από αυτά μια αντιπροσωπευτική τιμή, η οποία στην πιο συνηθισμένη περίπτωση είναι η μέγιστη του εκάστοτε τμήματος (max pooling), αλλά υπάρχουν και περιπτώσεις που χρησιμοποιείται ο μέσος όρος των τιμών του τμήματος (average pooling) ή ακόμα και τυχαία επιλογή από τις τιμές (stochastic pooling). Μέσω της χρήσης των επιπέδων υποδειγματοληψίας μειώνεται και η πιθανότητα υπερ-εκπαίδευσης του δικτύου.



Σχήμα 16: Παραδείγματα Max Pooling & Average Pooling [75]

Επίπεδο Κανονικοποίησης Παρτίδας (Batch Normalization Layer)

Τα Συνελικτικά Νευρωνικά Δίκτυα, όπως είδαμε και παραπάνω, αποτελούνται από διαδοχικά συνελικτικά επίπεδα, με το κάθε ένα να λαμβάνει ως είσοδο την έξοδο του προηγούμενου. Σε περιπτώσεις, ωστόσο, όπου τα Συνελικτικά Νευρωνικά Δίκτυα αποτελούνται από έναν μεγάλο αριθμό επιπέδων, τότε συχνά παρατηρείται ένα φαινόμενο που ονομάζεται «internal covariate shift» [76], με βάση το οποίο κατά την διαδικασία της εκπαίδευσης η προσαρμογή των παραμέτρων του δικτύου προκαλεί αλλαγή στην κατανομή των ενεργοποιήσεων των διαφόρων επιπέδων, ιδιαίτερα των τελευταίων. Για την αντιμετώπιση του προβλήματος αυτού σε τέτοιου είδους περιπτώσεις, είναι συχνή η προσθήκη στο δίκτυο Επιπέδων Κανονικοποίησης Παρτίδας, τα οποία εξασφαλίζουν την κανονικοποίηση των δεδομένων κάθε παρτίδας (batch) σε κάθε επίπεδο. Η κανονικοποίηση αυτή γίνεται με χρήση των στατιστικών χαρακτηριστικών του κάθε υποσυνόλου, έτσι ώστε ο μέσος όρος να ισούται με 0 και η διακύμανση να ισούται με 1.

Πλήρως Συνδεδεμένο Επίπεδο (Fully Connected Layer)

Τα επίπεδα που περιγράψαμε παραπάνω (Συνελικτικά, Ενεργοποίησης, Υποδειγματοληψίας, Κανονικοποίησης) εφαρμόζονται σε διαδοχικές επαναλήψεις στο πρώτο μέρος κάθε Συνελικτικού Νευρωνικού Δικτύου και αποτελούν το Δίκτυο Εξαγωγής Χαρακτηριστικών. Το δίκτυο αυτό, όπως αναφέρθηκε και παραπάνω, έχει ως τελική έξοδο έναν τελικό τρισδιάστατο χάρτη ενεργοποίησης μεγάλου βάθους. Σε περιπτώσεις που έχουμε δίκτυα ταξινόμησης, τότε η έξοδος αυτή του Δικτύου Εξαγωγής Χαρακτηριστικών γίνεται flatten και μετατρέπεται σε μονοδιάστατο πίνακα (διάνυσμα). Το διάνυσμα αυτό, τροφοδοτείται ως είσοδος σε ένα σύνολο από Πλήρως Συνδεδεμένα Επίπεδα, τα οποία αναλαμβάνουν την τελική ταξινόμηση του εκάστοτε αντικειμένου, το οποίο συνήθως είναι εικόνα. Ως συναρτήσεις ενεργοποίησης, τα Πλήρως Συνδεδεμένα Επίπεδα συνήθως

χρησιμοποιούν την ReLU, εκτός από το τελευταίο επίπεδο που χρησιμοποιεί συνήθως την Softmax, για την τελική ταξινόμηση του εκάστοτε αντικειμένου.

Ο τομέας της Μεταφοράς Μάθησης χρησιμοποιείται ενεργά και στα Συνελικτικά Νευρωνικά Δίκτυα, εφόσον στην σύγχρονη εποχή είναι ευρέως διαδεδομένη η χρήση προεκπαιδευμένων (pre-trained) CNNs. Τα pre-trained CNNs είναι μοντέλα τα οποία έχουν πρώτα εκπαιδευθεί σε σύνολα τεράστιων όγκων δεδομένων, όπως είναι το ImageNet, και τα οποία στην συνέχεια μπορούν να προσαρμοστούν στο εκάστοτε υπό-εξέταση πρόβλημα, μέσω μιας διαδικασίας fine-tuning. Χρησιμοποιούνται ευρέως στον τομέα της επεξεργασίας εικόνων για να βελτιώσουν την απόδοση σε διάφορες εφαρμογές, όπως αναγνώριση εικόνων, ενσωμάτωση εικόνων σε συστήματα επαυξημένης πραγματικότητας, ανίχνευση αντικειμένων σε αυτόνομα οχήματα, κλινική διάγνωση με βάση τις εικόνες και άλλα.

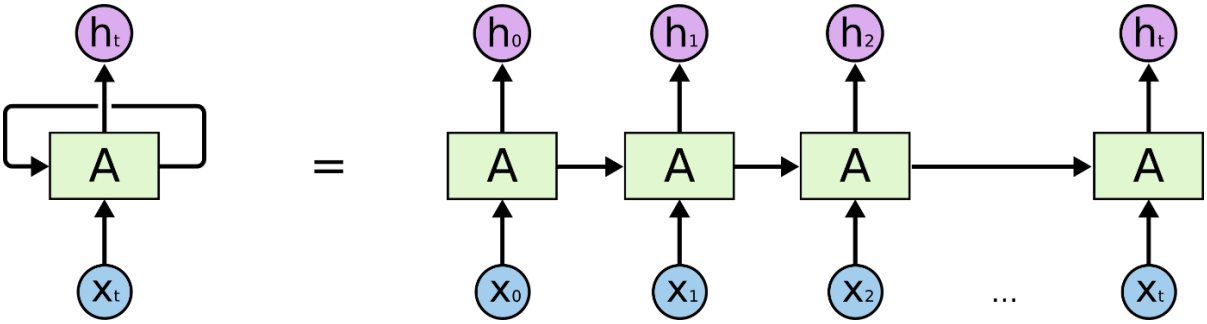
Ένα από τα βασικότερα πλεονεκτήματα της χρήσης προεκπαιδευμένων CNN είναι η δυνατότητά τους να επιταχύνουν και να απλοποιούν την διαδικασία εκπαίδευσης, εφόσον γίνεται χρήση των βαρών και των χαρακτηριστικών που έμαθαν κατά την διαδικασία προεκπαίδευσής τους, χωρίς να χρειάζεται να μαθευτούν εκ νέου, και τα οποία προσαρμόζονται κατάλληλα στο εκάστοτε πρόβλημα. Έτσι, μέσω της αξιοποίησης της γνώσης και της εμπειρίας του προεκπαιδευμένου μοντέλου αποφεύγονται προβλήματα overfitting και underfitting, ενώ είναι και αισθητά μικρότερες οι ανάγκες σε πλήθος δεδομένων εισόδου και υπολογιστικής ισχύς, εξοικονομώντας έτσι χρόνο και χρηματικούς πόρους. Ένα ακόμα πλεονέκτημα που χαρακτηρίζει την χρήση pre-trained μοντέλων είναι η ικανότητά τους να βελτιώσουν και να γενικεύσουν το μοντέλο. Αυτό οφείλεται στην έκθεση των μοντέλων αυτών σε έναν μεγάλο όγκο διαφορετικών συνόλων από εικόνες κατά την διαδικασία της εκπαίδευσής τους, το οποίο τους επιτρέπει να είναι σε θέση να εξάγουν τόσο high-level, όσο και low-level χαρακτηριστικά από τα δεδομένα, κάτι που καθίσταται ιδιαίτερα χρήσιμο σε πλήθος εφαρμογών. Αξιοποιώντας τα χαρακτηριστικά αυτά, είναι δυνατή η βελτίωση της ακρίβειας και της ευρωστίας του μοντέλου, καθώς και της προσαρμοστικότητάς του σε διαφορετικά προβλήματα και πεδία εφαρμογής. Με γνώμονα, λοιπόν, τα παραπάνω, επιστρατεύεται ένα σύνολο από προεκπαιδευμένα CNNs κατά την διαδικασία εκπόνησης της παρούσας εργασίας, όπως θα δούμε αναλυτικότερα και στην συνέχεια.

3.3.2 Recurrent Neural Networks (RNN)

Τα Non-Recurrent ή Feedforward δίκτυα δεν έχουν συνδέσεις επανατροφοδοσίας, δηλαδή συνδέσεις μέσω βαρών που να ξεκινούν από την έξοδο ενός στρώματος και να καταλήγουν ως είσοδος στο ίδιο ή σε επόμενο στρώμα. Αυτό έχει σαν αποτέλεσμα τα δίκτυα αυτά να μην έχουν «μνήμη», με αποτέλεσμα η έξοδός τους να καθορίζεται πάντα από την παρούσα είσοδο και τις τιμές των βαρών.

Αντιθέτως, τα Αναδρομικά Νευρωνικά Δίκτυα (Recurrent Neural Networks), τα οποία αποτελούν μια ειδική κατηγορία νευρωνικών δικτύων, έχουν την δυνατότητα να επεξεργάζονται αποτελεσματικά ακολουθιακά δεδομένα, όπως είναι η φωνή ή η γραφή, τα οποία απαιτούν την ύπαρξη «μνήμης» στο σύστημα. Σε προβλήματα που ζητείται η πρόβλεψη της επόμενης λέξης σε μια πρόταση, για να είναι σε θέση το μοντέλο να κάνει σωστή

πρόβλεψη, θα πρέπει να θυμάται τις προηγούμενες λέξεις της. Για τον λόγο αυτόν επιστρατεύονται τα Αναδρομικά Νευρωνικά Δίκτυα, τα οποία λόγω της ύπαρξης της εσωτερικής τους κρυφής κατάστασης (hidden state), είναι σε θέση να κρατάνε πληροφορία για την εκάστοτε ακολουθία. Στα δίκτυα αυτά, η έξοδος του προηγούμενου βήματος τροφοδοτείται σε συνδυασμό με την είσοδο του επόμενου βήματος ως συνολική είσοδος στο επόμενο βήμα.



Σχήμα 17: Recurrent Neural Network [77]

Στην παραπάνω εικόνα φαίνεται η αναδρομική λειτουργία του RNN, καθώς και το «ξετύλιγμά» της στον χρόνο. Το RNN, δεχόμενο το κάθε ένα από τα στοιχεία της ακολουθίας με την σειρά, ενημερώνει την εσωτερική (κρυφή) του κατάσταση. Οι εξισώσεις που διέπουν την λειτουργία ενός αναδρομικού δικτύου μπορούν να πάρουν την ακόλουθη γενική μορφή:

$$h_t = \varphi(Wx_t + Uh_{t-1} + b)$$

όπου:

- h_t είναι η κρυφή αναπαράσταση την χρονική στιγμή t,
- x_t είναι το διάνυσμα του στοιχείου της ακολουθίας την χρονική στιγμή t,
- W είναι ο πίνακας παραμέτρων που επιδρούν πάνω στην είσοδο x_t,
- U είναι ο πίνακας παραμέτρων που επιδρούν πάνω στην έξοδο του δικτύου h_{t-1} την προηγούμενη χρονική στιγμή,
- b είναι ένα διάνυσμα πόλωσης (bias),
- φ είναι μια μη-γραμμική συνάρτηση ενεργοποίησης.

Γενικότερα, τα Αναδρομικά Νευρωνικά Δίκτυα κατασκευάσθηκαν με σκοπό να έχουν την δυνατότητα να επεξεργάζονται αποτελεσματικά ακολουθίες μεγάλου μήκους. Εντούτοις, λόγω του προβλήματος Vanishing Gradient ή Exploding Gradient, στην πράξη είναι αποτελεσματικά μόνο σε περιορισμένες ακολουθίες μικρού μήκους. Για την αντιμετώπιση αυτού του περιορισμού αναπτύχθηκαν τα Δίκτυα Μακράς Βραχυπρόθεσμης Μνήμης (Long Short-Term Memory ή LSTM), τα οποία θα εξετάσουμε στην επόμενη ενότητα.

3.3.3 Long Short-Term Memory Networks (LSTM)

Οι Hochreiter & Schmidhuber [78] το 1997 πρότειναν μια παραλλαγή του RNN. Συγκεκριμένα, το μοντέλο το οποίο πρότειναν διαφέρει από τα RNN στην αρχιτεκτονική του κρυφού του επιπέδου, το οποίο αποκαλείται κύτταρο LSTM (Long Short-Term Memory). Τα Δίκτυα Μακράς Βραχυπρόθεσμης Μνήμης (LSTM) έχουν την δυνατότητα να αντιμετωπίζουν το πρόβλημα των Vanishing ή Exploding Gradient, καθιστώντας τα ικανά να μαθαίνουν μέσα από πολύ μεγάλες ακολουθίες εισόδου και να τις επεξεργάζονται αποτελεσματικά.

Ένα κύτταρο LSTM αποτελείται από 3 θύρες: την Θύρα Λήθης (Forget Gate), την Θύρα Εισόδου (Input Gate) και την Θύρα Εξόδου (Output Gate). Πιο συγκεκριμένα έχουμε:

- **Θύρα Λήθης (Forget Gate)**

Αρμοδιότητα της Θύρας Λήθης είναι να αποφασίζει για το ποια πληροφορία θα διαγραφεί από την μνήμη. Λαμβάνει ως είσοδο την τρέχουσα είσοδο x_t σε συνδυασμό με την έξοδο του προηγούμενου βήματος h_{t-1}, όπως συνέβαινε και στα RNN, και μέσω μιας σιγμοειδούς συνάρτησης παράγει ως έξοδο έναν αριθμό μεταξύ του 0 και του 1. Τέλος, η έξοδος αυτή πολλαπλασιάζεται με κάθε αριθμό του διανύσματος C_{t-1} της προηγούμενης κατάστασης, καθορίζοντας έτσι την πληροφορία που θα ξεχαστεί. Η μαθηματική αναπαράσταση δίνεται από τον ακόλουθο τύπο:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f)$$

- **Θύρα Εισόδου (Input Gate)**

Αρμοδιότητα της Θύρας Εισόδου είναι να αποφασίζει για το ποια πληροφορία θα αποθηκευτεί στην μνήμη. Αφού πρώτα αποφασιστεί ποιες τιμές θα ενημερωθούν, στην θύρα εισόδου τροφοδοτείται σαν είσοδος η τρέχουσα είσοδος x_t και η έξοδος του προηγούμενου βήματος h_{t-1}. Στην συνέχεια, οι δύο αυτές οντότητες τροφοδοτούνται σε ένα μονοεπίπεδο νευρωνικό δίκτυο, το οποίο με χρήση της υπερβολικής εφαπτομένης σαν συνάρτηση ενεργοποίησης δημιουργεί τις νέες υποψήφιες τιμές $\widetilde{c_t}$ που πρόκειται να αποθηκευτούν στην μνήμη. Οι μαθηματικές αναπαραστάσεις της θύρας εισόδου και της υποψήφιας μνήμης δίνονται αντίστοιχα από τους τύπους:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \text{ και } \widetilde{c_t} = tanh(W_c x_t + U_c h_{t-1} + b_c).$$

- **Μνήμη (Cell State)**

Αρμοδιότητα του βήματος αυτού είναι η ανανέωση της παλιάς μνήμης c_{t-1} στην καινούργια c_t. Για να επιτευχθεί αυτό, η θύρα λήθης πολλαπλασιάζεται με τις τιμές της παλιάς μνήμης c_{t-1} (αποφεύγοντας όρους που αποφασίστηκε να ξεχαστούν), ενώ προστίθεται και ο όρος $i_t \odot \widetilde{c_t}$, ο οποίος ουσιαστικά αποτελεί τις νέες υποψήφιες τιμές, οι οποίες έχουν κλιμακωθεί αναλογικά με το πόσο έχουμε σκοπό να ενημερώσουμε την τρέχουσα κατάσταση. Το σύμβολο $\odot$ υποδεικνύει το γινόμενο Hadamard, δηλαδή πολλαπλασιασμό στοιχείο προς στοιχείο. Η μαθηματική αναπαράσταση των παραπάνω δίνεται από τον ακόλουθο τύπο:

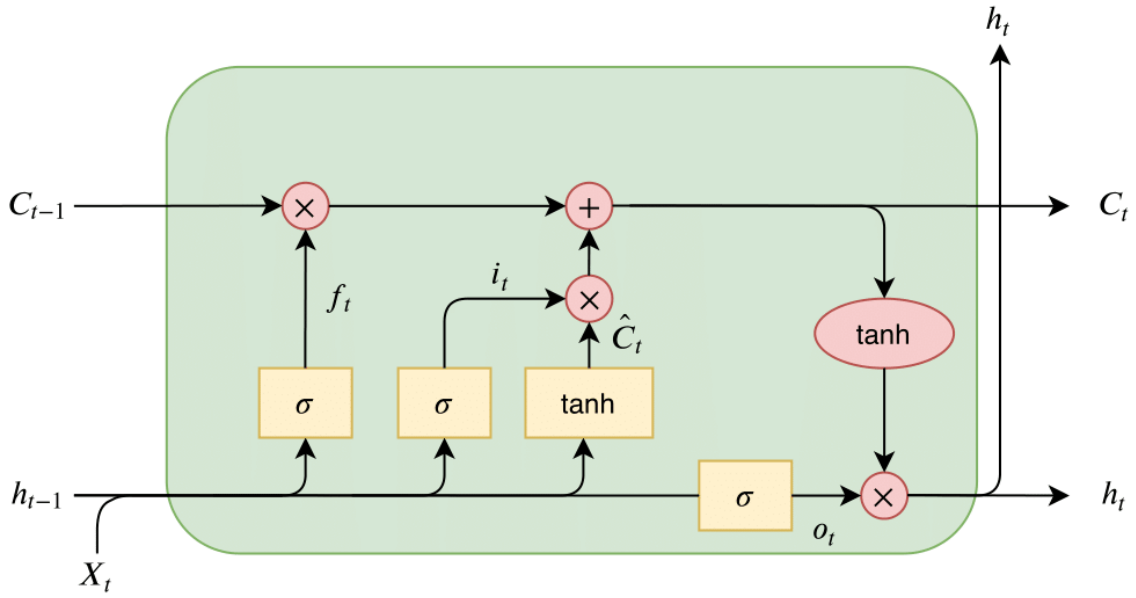
$$c_t = f_t \odot c_{t-1} + i_t \odot \widetilde{c_t}$$

- **Θύρα Εξόδου (Output Gate)**

Η Θύρα Εξόδου αποτελεί την έξοδο του κυττάρου LSTM και βασίζεται σε μια «φιλτραρισμένη» εκδοχή της κατάστασης της μνήμης. Πιο συγκεκριμένα, πάλι οι δυο οντότητες της τρέχουσας εισόδου x_t και της εξόδου του προηγούμενου βήματος h_{t-1} τροφοδοτούνται σε ένα μονοεπίπεδο νευρωνικό δίκτυο, το οποίο με συνάρτηση ενεργοποίησης την σιγμοειδή αποφασίζει ποια από τα μέρη της κατάστασης μνήμης θα χρησιμοποιηθούν τελικά στην έξοδο της Θύρας Εξόδου. Στην συνέχεια, αφού η κατάσταση της μνήμης c_t περάσει μέσα από την συνάρτηση ενεργοποίησης της υπερβολικής εφαπτομένης, πολλαπλασιάζεται στοιχείο προς στοιχείο με την θύρα εξόδου, για να αποφασιστεί τελικά ποια από τα μέρη της κατάστασης αυτής θα χρησιμοποιηθούν στην τελική έξοδο του κυττάρου. Οι μαθηματικές αναπαραστάσεις της θύρας εξόδου και της τελικής εξόδου του κυττάρου δίνονται αντίστοιχα από τους τύπους:

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \text{ και } h_t = o_t \odot \tanh(c_t).$$

Η συνολική μορφή του κυττάρου LSTM φαίνεται στην παρακάτω εικόνα:



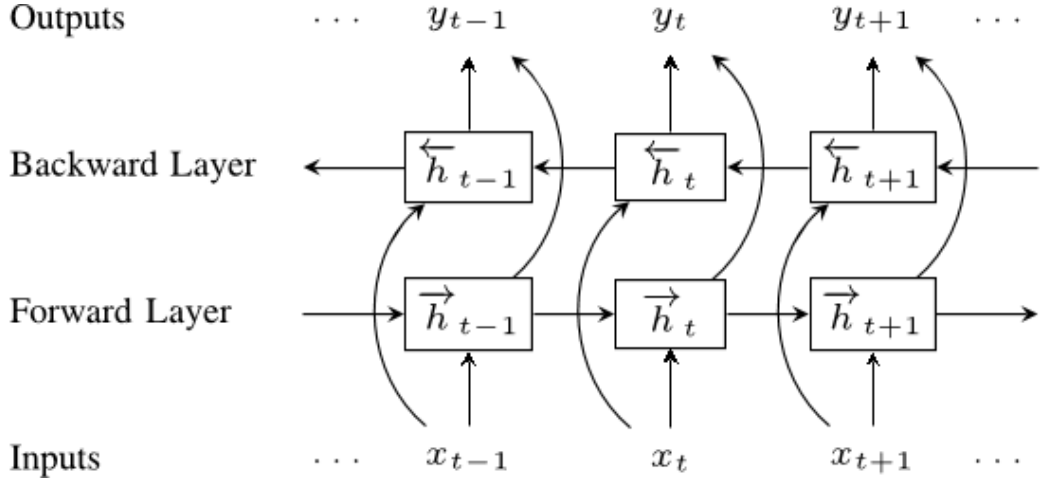
Σχήμα 18: Κύτταρο Long Short-Term Memory (LSTM) [79]

3.3.4 Bidirectional LSTM (BiLSTM)

Τα Αμφίδρομα LSTM (Bidirectional LSTM ή BiLSTM) αποτελούν ουσιαστικά μια επέκταση των κλασικών LSTM που αναλύσαμε στην προηγούμενη ενότητα, τα οποία επιτυγχάνουν αισθητή βελτίωση στην απόδοση του μοντέλου σε ακολουθιακά προβλήματα ταξινόμησης, συγκριτικά με τα παραδοσιακά LSTM.

Ένα Αμφίδρομο LSTM δεν είναι τίποτα άλλο πέρα από ένας συνδυασμός δύο διαφορετικών μοντέλων LSTM, που το κάθε ένα επεξεργάζεται την ίδια ακολουθία εισόδου,

ταυτόχρονα, αλλά με διαφορετική (αντίθετη) φορά. Αυτό έχει σαν αποτέλεσμα, κάθε χρονική στιγμή η έξοδος να εξαρτάται τόσο από τα προηγούμενα, όσο και από τα επόμενα στοιχεία της ακολουθίας. Μια κλασική μελέτη περίπτωσης είναι η αντιμετώπιση προβλημάτων πρόβλεψης μιας λέξης που λείπει από μια πρόταση. Στην περίπτωση αυτή, το BiLSTM λαμβάνει υπόψιν του τόσο τις λέξεις που προηγούνται της λέξης που ψάχνουμε, όσο και αυτές που έπονται, για να κατανοήσει το περιεχόμενο της πρότασης (ή αλλιώς τα συμφραζόμενα) και να είναι σε θέση να παράξει μια αποδοτική πρόβλεψη.



Σχήμα 19: Bidirectional LSTM [80]

Όπως φαίνεται και στην παραπάνω εικόνα, δρουν ταυτόχρονα δύο διαφορετικά LSTM, ένα επεξεργαζόμενο την ακολουθία από τα αριστερά προς το δεξιά (Forward Layer), και ένα από τα δεξιά προς τα αριστερά (Backward Layer). Έτσι, σε κάθε χρονική στιγμή t, το δεξιόστροφο LSTM με κρυφή κατάσταση $\vec{h}$ δέχεται στην είσοδό του τόσο την είσοδο x_t της εκάστοτε τωρινής χρονικής στιγμής t, όσο και την προηγούμενη κρυφή κατάσταση $\overrightarrow{h_{t-1}}$, ενώ ταυτόχρονα το αριστερόστροφο LSTM με κρυφή κατάσταση $\overleftarrow{h}$ δέχεται στην είσοδό του πάλι την είσοδο x_t της εκάστοτε τωρινής χρονικής στιγμής t, αλλά και την μελλοντική κρυφή κατάσταση $\overleftarrow{h_{t+1}}$. Συνεπώς, σαν συνολική κρυφή κατάσταση σε κάθε χρονική στιγμή t, έχουμε το concatenation των εκάστοτε κρυφών καταστάσεων του δεξιόστροφου και του αριστερόστροφου LSTM, όπως φαίνεται και στον ακόλουθο τύπο:

$$h_i = [\overrightarrow{h_i^T}, \overleftarrow{h_i^T}]^T$$

3.3.5 Μηχανισμοί Προσοχής

Οι Μηχανισμοί Προσοχής (Attention Mechanisms) αποτελούν δομές που επιτρέπουν στα μοντέλα να επικεντρώνονται σε μέρος της πληροφορίας εισόδου που θεωρείται σημαντικότερη. Με αυτόν τον τρόπο δύνανται να υπερβούν το πρόβλημα που δημιουργείται σε πολύπλοκες ακολουθίες εισόδου μεγάλου μήκους. Τα Στρώματα Προσοχής (Attention Layers) αποτελούν την βασική δοκιμή μονάδα των μοντέλων Transformers, τα οποία επιστρατεύονται σε προβλήματα Επεξεργασίας Φυσικής Γλώσσας (και όχι μόνο) για τον

εντοπισμό και την σύλληψη πολυπλοκότερων και μακρυνότερων εξαρτήσεων στις ακολουθίες εισόδου, αντικαθιστώντας έτσι σε μεγάλο βαθμό τα RNNs και LSTMs, τα οποία αδυνατούν να το επιτύχουν σε ικανοποιητικό βαθμό. Στον κλάδο της Όρασης Υπολογιστών, ο Μηχανισμός Προσοχής παρομοιάζεται με την ικανότητα του ανθρώπινου οφθαλμού να εστιάζει σε συγκεκριμένες περιοχές μιας εικόνας με μεγαλύτερη ανάλυση, αγνοώντας παράλληλα άλλες περιοχές που έχουν μικρότερο ενδιαφέρον, όπως είναι το φόντο.

Χρησιμοποιώντας τους Μηχανισμούς Προσοχής λαμβάνουμε διανύσματα συμφραζόμενων (context vectors) c_i, τα οποία εμπεριέχουν τρεις πληροφορίες: την κρυφή κατάσταση του κωδικοποιητή (encoder), την κρυφή κατάσταση του αποκωδικοποιητή (decoder), καθώς και την στοίχιση μεταξύ πηγής και στόχου. Θεωρώντας πως το δίκτυο του encoder έχει ως κρυφές καταστάσεις τις $h_1^{enc}, h_2^{enc}, \dots, h_n^{enc}$, και το δίκτυο του decoder έχει ως κρυφή κατάσταση την h_i^{dec}κατά το εκάστοτε χρονικό βήμα i, το διάνυσμα συμφραζόμενων c_i κατά το χρονικό βήμα i υπολογίζεται ως ο σταθμισμένος μέσος όρος των καταστάσεων του encoder με βάρη τις τιμές προσοχής $a_{i,j}$, όπως φαίνεται στον ακόλουθο τύπο:

$$c_i = \sum_{j=1}^{n} a_{i,j} h_j^{(enc)}$$

όπου τα $a_{i,j}$ δίνονται από τον ακόλουθο τύπο:

$$a_{i,j} = softmax(score(h_i^{(dec)}, h_j^{(enc)}))$$

Η ποσότητα $score(h_i^{(dec)}, h_j^{(enc)})$ υπολογίζει μια μη κανονικοποιημένη τιμή στοίχισης, η οποία εκφράζει το κατά πόσο κάθε κρυφή κατάσταση της πηγής πρέπει να ληφθεί υπόψιν για τον υπολογισμό της κάθε εξόδου. Έτσι, τα βάρη $a_{i,j}$ υπολογίζονται με βάση το κατά πόσο ταιριάζει το ζεύγος της εισόδου στη θέση j με την έξοδο στη θέση i.

Στις μέρες μας έχουν αναπτυχθεί διάφορες παραλλαγές Μηχανισμών Προσοχής. Στην συνέχεια παρουσιάζονται οι δημοφιλέστερες από αυτές, συνοδευόμενες από τον τρόπο λειτουργίας τους.

Generalized Attention

Ο Μηχανισμός Προσοχής Generalized Attention λαμβάνει ως είσοδο ακολουθίες λέξεων ή εικόνες και συγκρίνει την ακολουθία εισόδου με την ακολουθία εξόδου. Αναλυτικότερα, μέσω της σύγκρισης της εισόδου του κωδικοποιητή με την έξοδο του αποκωδικοποιητή, που γίνεται σε κάθε επανάληψη, προκύπτουν κάποιες βαθμολογίες οι οποίες χρησιμοποιούνται από το μοντέλο για να αποφασίσει σε ποια τμήματα της εισόδου θα δώσει μεγαλύτερη έμφαση.

Αυτοπροσοχή (Self Attention)

Η Αυτοπροσοχή (Self Attention) είναι ένας Μηχανισμός Προσοχής που συσχετίζει διαφορετικές θέσεις μιας ακολουθίας εισόδου, έτσι ώστε να υπολογίσει μια αναπαράσταση της ίδιας ακολουθίας. Στην περίπτωση που εφαρμοστεί χωρίς την ύπαρξη επιπλέον πληροφορίας, τότε είναι δυνατό να εξαχθούν παράγοντες συγγένειας μιας πρότασης, επιτρέποντας στο

μοντέλο να μάθει συσχετίσεις ανάμεσα σε μια λέξη και τις προηγούμενές της. Θεωρητικά, η αυτοπροσοχή θα μπορούσε να συνδυαστεί με οποιαδήποτε score συνάρτηση, αντικαθιστώντας απλώς την ακολουθία στόχο με την ίδια ακολουθία εισόδου. Χρησιμοποιείται αρκετά σε μοντέλα που βασίζονται σε Transformers, καθώς και σε πλήθος εφαρμογών του τομέα του NLP, όπως Ανάγνωση Μηχανής, Παραγωγή Περιγραφών Εικόνας (Image Caption) και Αφαιρετική Περίληψη.

Απαλή Προσοχή (Soft Attention)

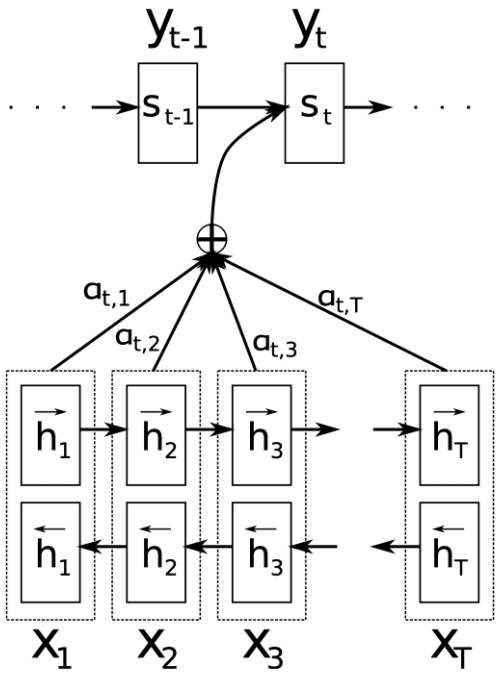
Στην Απαλή Προσοχή, ο Μηχανισμός Προσοχής έχει πρόσβαση σε ολόκληρη την πηγή, τοποθετώντας τα βάρη στοίχισης πάνω σε όλες τις θέσεις της. Στην συγκεκριμένη περίπτωση μηχανισμού, το μοντέλο καθίσταται ομαλό και διαφορίσιμο, με το μειονέκτημα, όμως, πως σε περιπτώσεις μεγάλων ακολουθιών εισόδου, ο ίδιος να είναι αρκετά ακριβός υπολογιστικά.

Σκληρή Προσοχή (Hard Attention)

Στην Σκληρή Προσοχή, ο Μηχανισμός Προσοχής αποφασίζει να εστιάσει σε ένα μόνο μέρος της εισόδου, με αποτέλεσμα να απαιτούνται αισθητά λιγότεροι υπολογισμοί αναλογικά με την προηγούμενη περίπτωση. Ωστόσο, το μοντέλο πλέον παύει να είναι διαφορίσιμο, και για να εκπαιδευθεί απαιτούνται πιο περίπλοκες τεχνικές, όπως μείωση της διασποράς ή ενισχυτική μάθηση.

Additive Attention

Το Additive Attention, γνωστό και ως Bahdanau Attention [81], χρησιμοποιεί alignment scores που υπολογίζονται σε διαφορετικές θέσεις του δικτύου, έτσι ώστε να ευθυγραμμίσει τις ακολουθίες εισόδου με τις ακολουθίες εξόδου, επιτρέποντας έτσι σε απόδοση μεγαλύτερης έμφασης στις σημαντικότερες πληροφορίες. Για την συσχέτιση της εισόδου με την ακολουθία εξόδου λαμβάνονται υπόψη όλες οι κρυφές καταστάσεις του κωδικοποιητή και του αποκωδικοποιητή, για να παραχθούν τα διανύσματα συμφραζόμενων (context vectors). Το μοντέλο χρησιμοποιεί αυτά τα vectors σε συνδυασμό με τις προηγούμενες λέξεις που έχουν παραχθεί για να είναι σε θέση να προβλέψει την εκάστοτε λέξη - στόχο.



Σχήμα 20: Bahdanau Attention Mechanism [81]

Οι σχέσεις από τις οποίες υπολογίζονται τα alignment scores, τα attention weights, αλλά και οι πίνακες, βάσει των οποίων γίνεται η πρόβλεψη, παρουσιάζονται στις ακόλουθες εξισώσεις, αντίστοιχα. Επίσης, δίνεται και η σχέση για την πρόβλεψη της λέξης - στόχου, η οποία χρησιμοποιεί τόσο το διάνυσμα συμφραζόμενων και την έξοδο του αποκωδικοποιητή στο προηγούμενο χρονικό βήμα y_{i-1}, όσο και τις προηγούμενες κρυφές καταστάσεις του αποκωδικοποιητή s_{i-1}.

$$Alignment\_score_{ij} = a(s_{i-1}, h_j)$$

$$Attention\_weight_{ij} = \frac{\exp\left(Alignment\_score_{ij}\right)}{\sum_{k=1}^{T_x} \exp\left(Alignment\_score_{ik}\right)}$$

$$Context_i = \sum_{j=1}^{T_x} Attention\_weight_{ij} h_j$$

$$s_i = f(s_{i-1}, c_i, y_{i-1})$$

Key-Value-Query Attention

Ο Μηχανισμός Προσοχής Key-Value-Query παρουσιάστηκε από τους Vaswani et al. [82] και χρησιμοποιείται στο μοντέλο των Transformers, το οποίο θα παρουσιάσουμε στην επόμενη ενότητα. Ουσιαστικά, η συνάρτηση προσοχής περιγράφεται ως μια απεικόνιση ενός ερωτήματος (query) και ενός συνόλου από ζεύγη από κλειδιά - τιμές (key - value) σε μία έξοδο, όλα εκ των οποίων αποτελούν διανύσματα (ερωτήματα, κλειδιά, τιμές, έξοδοι). Η έξοδος υπολογίζεται κάθε φορά ως ο σταθμισμένος μέσος όρος των τιμών (values), όπου τα βάρη που ανατίθενται σε κάθε τιμή προκύπτουν από μια συνάρτηση συμβατότητας του ερωτήματος (query) με το αντίστοιχο κλειδί (key).

Scaled-Dot-Product Attention

Στον Μηχανισμό Προσοχής Scaled-Dot-Product, η είσοδος αποτελείται από ερωτήματα Q και κλειδιά K διάστασης d_k, καθώς και από τιμές V διάστασης d_v. Ουσιαστικά, στον μηχανισμό αυτόν, αρχικά υπολογίζονται τα γινόμενα πινάκων κάθε ερωτήματος με όλα τα κλειδιά, τα οποία στην συνέχεια διαιρούνται με τον παράγοντα κανονικοποίησης $\sqrt{d_k}$, και στο αποτέλεσμα αυτό εφαρμόζεται η συνάρτηση ενεργοποίησης Softmax για την λήψη των βαρών καθεμίας από τις τιμές. Η έξοδος, εδώ, υπολογίζεται κάθε φορά ως το σταθμισμένο άθροισμα των τιμών με τα αντίστοιχα βάρη. Συνεπώς, η συνάρτηση προσοχής υπολογίζει τα βάρη όλων των τιμών ταυτόχρονα, όπως φαίνεται στον ακόλουθο τύπο:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Αποτελεί έναν αρκετά γρήγορο Μηχανισμό Προσοχής, εφόσον οι πολλαπλασιασμοί πινάκων μπορούν να βελτιστοποιηθούν. Τέλος, ο παράγοντας κανονικοποίησης $\sqrt{d_k}$ χρησιμοποιείται έτσι ώστε να αποτρέπεται το γινόμενο πινάκων από το να λαμβάνει υψηλές τιμές, επιτρέποντας έτσι στην συνάρτηση softmax να λάβει αρκετά χαμηλές τιμές κλίσης.

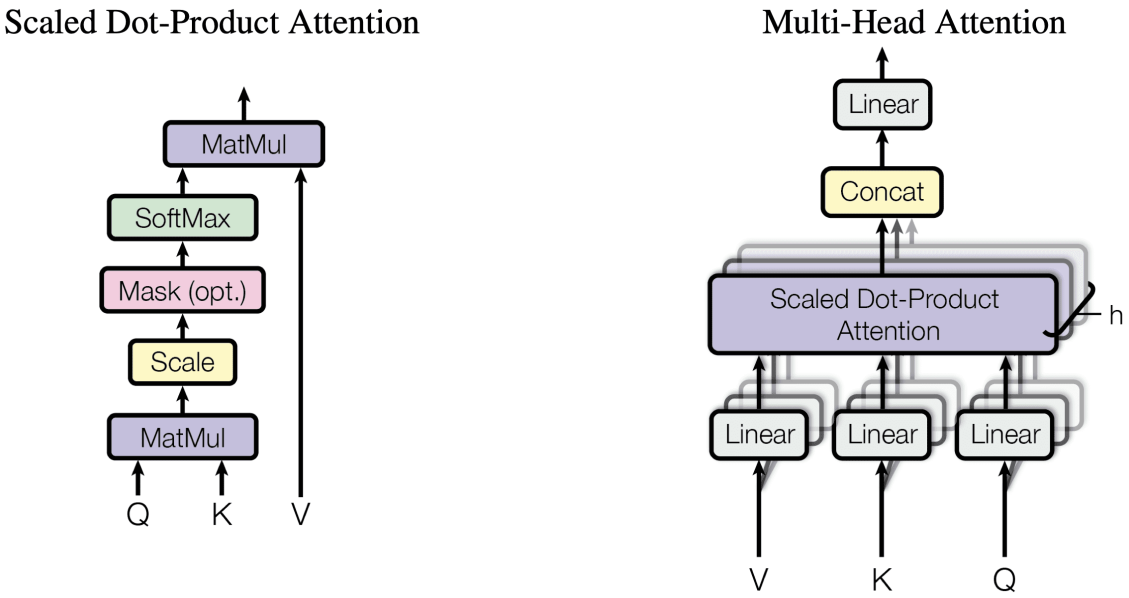
Multi-Head Attention

Τέλος, ο Μηχανισμός Multi-Head Προσοχής αποτελεί μια επέκταση του μηχανισμού Scaled-Dot-Product που αναλύσαμε προηγουμένως. Στην συγκεκριμένη περίπτωση, ωστόσο, αντί να εφαρμόζεται μια και μόνο συνάρτηση προσοχής με κλειδιά, τιμές και ερωτήσεις διαστάσεων d_{model}, τα κλειδιά, οι ερωτήσεις και οι τιμές προβάλλονται γραμμικά h φορές, με διαφορετικές γραμμικές προβολές στις διαστάσεις d_k, d_k και d_v, αντίστοιχα. Με τον τρόπο αυτό, το μοντέλο έχει την δυνατότητα να εστιάσει ταυτόχρονα σε πληροφορία από διαφορετικούς χώρους αναπαράστασης και σε διαφορετικές θέσεις. Τα παραπάνω παριστάνονται από τους ακόλουθους τύπους:

$$MultiHead(Q, K, V) = Concat(head_1, \dots, head_h)W^O$$

$$\text{όπου } head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$$

όπου $W_i^Q \in \mathbb{R}^{d_{model} \times d_k}$, $W_i^K \in \mathbb{R}^{d_{model} \times d_k}$, $W_i^V \in \mathbb{R}^{d_{model} \times d_v}$ και $W^O \in \mathbb{R}^{hd_v \times d_{model}}$. Στο σημείο αυτό αξίζει να αναφερθεί πως οι διαστάσεις επιλέγονται βάσει της σχέσης $d_v = \frac{d_{model}}{h}$, ώστε το υπολογιστικό κόστος με τις μειωμένες διαστάσεις κάθε κεφαλής να παραμένει ίδιο με αυτό του μηχανισμού μιας κεφαλής πλήρους διαστατικότητας.



Σχήμα 21: Μηχανισμοί Προσοχής Scaled-Dot-Product (αριστερά) και Multi-Head (δεξιά) [82]

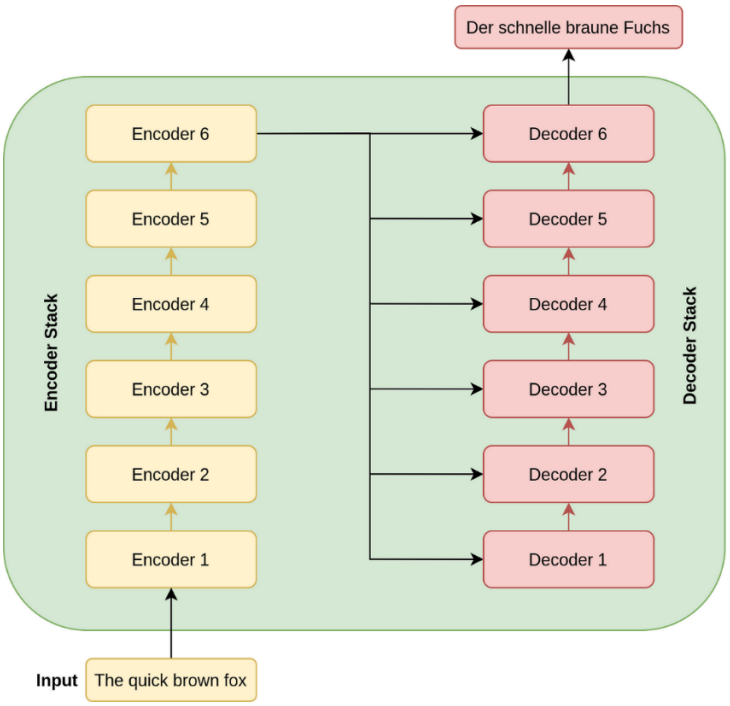
3.3.6 Μοντέλο Transformer Encoder-Decoder

Οι Vaswani et al. [82] πρότειναν για πρώτη φορά το 2017 την αρχιτεκτονική του δικτύου Transformer, το οποίο όχι μόνο επέτρεψε πολύ καλύτερες επιδόσεις αναλογικά με προηγούμενες αρχιτεκτονικές, αλλά και σε αισθητά λιγότερο απαιτούμενο χρόνο για την εκπαίδευσή του. Το μοντέλο Transformer βασίζεται αποκλειστικά σε μηχανισμό αυτοπροσοχής (self-attention) για τον υπολογισμό αναπαραστάσεων των εισόδων και της εξόδου. Ο μηχανισμός αυτοπροσοχής στον οποίο βασίζονται τους επιτρέπει να επικεντρώνονται σε διαφορετικά τμήματα της εισόδου, συλλαμβάνοντας μακρινές εξαρτήσεις και κατανοώντας τις σχέσεις μεταξύ των λέξεων που οδηγούν σε καλύτερες επιδόσεις. Το μοντέλο στηρίζεται στην αρχιτεκτονική Encoder-Decoder (Κωδικοποιητή-Αποκωδικοποιητή), εφόσον αποτελείται αντίστοιχα από δύο στοίβες, μια στοίβα κωδικοποιητή και μια αποκωδικοποιητή. Αρμοδιότητα του κωδικοποιητή είναι να απεικονίζει μια ακολουθία εισόδου, έστω X, σε μια ακολουθία συνεχών αναπαραστάσεων, έστω z. Ύστερα, κάνοντας χρήση της z, αρμοδιότητα του αποκωδικοποιητή είναι να παράξει την ακολουθία εξόδου, έστω Y, μετατρέποντας ένα στοιχείο την φορά. Το μοντέλο, σε κάθε βήμα, χρησιμοποιεί τις αμέσως προηγούμενες αναπαραστάσεις που παρήγαγε σαν επιπρόσθετες εισόδους για την διαδικασία παραγωγής των επόμενων.

Πιο συγκεκριμένα, η στοίβα του κωδικοποιητή (Encoder Stack) αποτελείται συνολικά από 6 πανομοιότυπα στρώματα, κάθε ένα εκ των οποίων αποτελείται από δύο υποστρώματα. Το πρώτο από τα υποστρώματα είναι ένας μηχανισμός Multi-Head αυτοπροσοχής, ενώ το δεύτερο είναι ένα απλό Position-Wise Fully-Connected Feedforward Δίκτυο. Γύρω από κάθε υπόστρωμα χρησιμοποιείται η τεχνική της υπολειπόμενης σύνδεσης (Residual Connection), ενώ ακολουθείται από ένα στρώμα κανονικοποίησης (Normalization Layer).

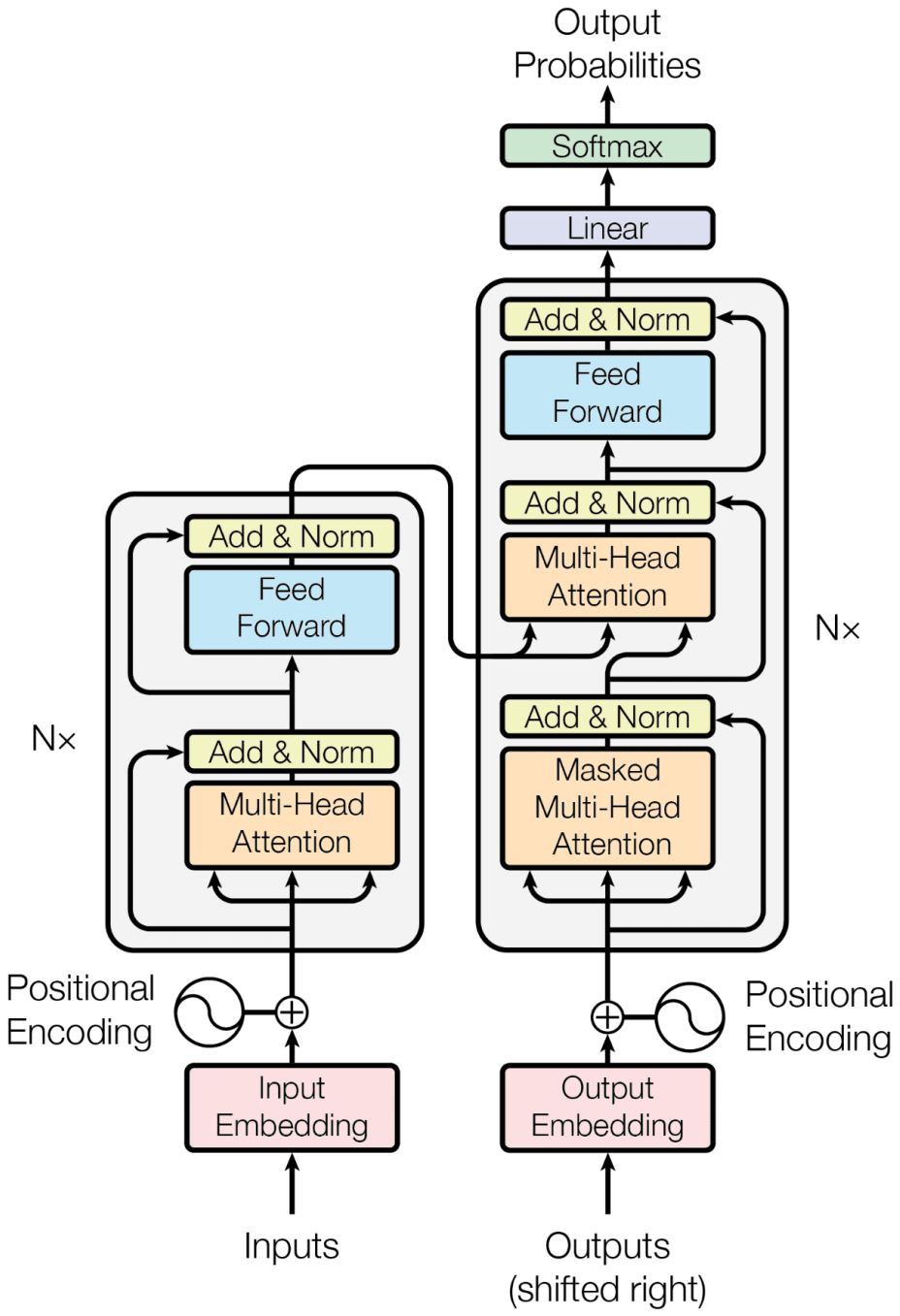
Αντίστοιχα, η στοίβα του αποκωδικοποιητή (Decoder Stack) αποτελείται επίσης από 6 πανομοιότυπα στρώματα. Ωστόσο, στην περίπτωση αυτή, εκτός των δύο υποστρωμάτων που

εμφανίζονται στην στοίβα του κωδικοποιητή, κάθε στρώμα του αποκωδικοποιητή διαθέτει και ένα τρίτο υπόστρωμα, το οποίο εφαρμόζει προσοχή Multi-Head στις εξόδους του Encoder Stack. Οι τεχνικές υπολειπόμενης σύνδεσης και στρωμάτων κανονικοποίησης εφαρμόζονται αντίστοιχα και σε αυτήν την στοίβα, ενώ εφαρμόζεται και ένα είδος μάσκας στον μηχανισμό της αυτοπροσοχής, έτσι ώστε η πρόβλεψη μιας εκάστοτε θέσης να εξαρτάται αποκλειστικά από πρότερες θέσεις.



Σχήμα 22: Στοίβες Κωδικοποιητή και Αποκωδικοποιητή του Μοντέλου Transformer [83]

Στην παραπάνω εικόνα φαίνεται μια οπτικοποίηση υψηλού επιπέδου του μοντέλου Transformer, με τις στοίβες Κωδικοποιητή και Αποκωδικοποιητή και τα 6 στρώματα από τα οποία αποτελείται η κάθε μια. Η συνολική αρχιτεκτονική του μοντέλου φαίνεται στο παρακάτω σχήμα:



Σχήμα 23: Αρχιτεκτονική Μοντέλου Transformer [82]

Εφόσον το μοντέλο δεν διαθέτει αναδρομικότητα ή μηχανισμούς συνέλιξης, δεν αντικατοπτρίζεται με κάποιο τρόπο η έννοια της «σειράς» στην εκάστοτε ακολουθία εισόδου. Για αυτόν τον λόγο, χρησιμοποιείται η τεχνική Positional Encoding, όπως φαίνεται στο παραπάνω σχήμα, αρμοδιότητα της οποίας είναι να αποθηκεύει πληροφορία αναλογικά με την σχετική ή απόλυτη θέση κάθε αντικειμένου της ακολουθίας εισόδου. Στο αρχικό μοντέλο γίνεται χρήση της τεχνικής Positional Encoding που φαίνεται παρακάτω:

$$PE_{(pos,i)} = \begin{cases} \sin\left(\frac{pos}{10000^{\frac{i}{d_{model}}}}\right), \text{εάν } i \text{ άρτιος} \\ cos\left(\frac{pos}{10000^{\frac{i-1}{d_{model}}}}\right), \text{εάν } i \text{ περιττός} \end{cases}$$

όπου pos είναι η θέση, i η διαστατικότητα και d_{model} η διάσταση των διανυσμάτων στα οποία μετατρέπονται η είσοδος και η έξοδος του μοντέλου, χρησιμοποιώντας προεκπαιδευμένα εμφυτεύματα (embeddings), και η οποία ορίζεται ίση με 512. Έτσι, κάθε διάσταση του positional encoding αντιστοιχίζεται σε ένα ημιτονοειδές, καθιστώντας έτσι δυνατή την ενσωμάτωση της έννοιας της «σειράς» μέσα στην ακολουθία.

3.4 Επεξεργασία Φυσικής Γλώσσας (Natural Language Processing)

Η Επεξεργασία Φυσικής Γλώσσας (Natural Language Processing - NLP) αποτελεί υποκλάδο της Επιστήμης Υπολογιστών, της Γλωσσολογίας και της Τεχνητής Νοημοσύνης και πραγματεύεται την αλληλεπίδραση μεταξύ υπολογιστικών συστημάτων και ανθρώπινων (φυσικών) γλωσσών. Ασχολείται με την διαδικασία εκπαίδευσης ενός υπολογιστικού συστήματος, ώστε να είναι σε θέση να εξάγει και να αναλύει χρήσιμη πληροφορία μεγάλου όγκου δεδομένων φυσικής γλώσσας και/ή να παράγει το ίδιο φυσική γλώσσα στην έξοδό του. Το πεδίο του Natural Language Processing ουσιαστικά εμπεριέχει τόσο την αναγνώριση και κατανόηση ομιλίας, όσο και την παραγωγή φυσικής γλώσσας.

Το πεδίο δραστηριότητας του NLP μπορεί να κατηγοριοποιηθεί σε μια ακολουθία από μικρότερα «προβλήματα», καθένα από τα οποία αποτελεί ένα επίπεδο ανάλυσης. Ωστόσο, δεν θεωρείται απαραίτητο να εφαρμοστούν όλα τα επίπεδα για την αντιμετώπιση ενός προβλήματος, ούτε και υπάρχει κάποια αυστηρώς προκαθορισμένη σειρά εφαρμογής τους. Τα διαφορετικά επίπεδα με τα οποία ασχολείται ο κλάδος του NLP είναι τα ακόλουθα:

- **Phonology**

Το επίπεδο αυτό εφαρμόζεται μόνο στις περιπτώσεις που η πηγή ενός κειμένου είναι ο προφορικός λόγος, και ασχολείται με την ερμηνεία των ήχων ομιλίας, τόσο εντός των λέξεων, όσο και μεταξύ τους.

- **Morphology**

Το επίπεδο αυτό πραγματεύεται την κατανόηση διακριτών λέξεων βάσει μορφημάτων, δηλαδή τον σχηματισμό των λέξεων και την τμηματοποίησή τους σε πρόθημα (prefix), κυρίως σώμα και επίθημα (suffix).

- **Lexical**

Το επίπεδο αυτό πραγματεύεται την κατανόηση διακριτών λέξεων βάσει της θέσης τους στον λόγο, της σημασίας τους και της σχέσης τους με άλλες λέξεις. Έτσι, τοποθετείται σε κάθε λέξη μια ετικέτα, η οποία είναι σχετική με το μέρος του λόγου και ονομάζεται part-of-speech tag.

- **Syntactic**

Το επίπεδο αυτό πραγματεύεται την ανάλυση των λέξεων μιας πρότασης, με σκοπό την συντακτική δομή της πρότασης αυτής.

- **Semantic**

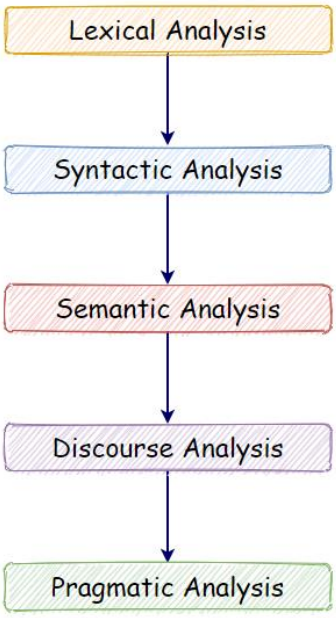
Το επίπεδο αυτό πραγματεύεται την σημασιολογική (semantic) επεξεργασία της εκάστοτε πρότασης, εξάγοντας πιθανά νοήματα που μπορεί να έχει μια λέξη και συσχετίζοντας μεταξύ τους τα συντακτικά χαρακτηριστικά τους.

- **Discourse**

Αυτό που διαχωρίζει το επίπεδο αυτό από τα προηγούμενα, είναι το γεγονός πως χρησιμοποιεί κειμενικά αποσπάσματα, δηλαδή κείμενα, που περιέχουν περισσότερες από μια προτάσεις. Συνεπώς, πραγματεύεται τις ιδιότητες ενός κειμένου σαν σύνολο, και εξάγει σημασιολογικά χαρακτηριστικά και συμπεράσματα βάσει των συνδέσεων και των σχέσεων που εμφανίζονται ανάμεσα στις προτάσεις που το αποτελούν.

- **Pragmatic**

Τέλος, το επίπεδο αυτό ασχολείται με τη χρήση της γλώσσας σε αληθινά σενάρια του πραγματικού κόσμου, ενώ γίνεται χρήση εξωτερικής γνώσης, έτσι ώστε να είναι δυνατή η ερμηνεία του νοήματος που μεταδίδεται από τη φυσική γλώσσα.



Σχήμα 24: Επίπεδα Ανάλυσης ενός Γενικού Προβλήματος NLP [84]

3.4.1 Γλωσσικές Αναπαραστάσεις

Όπως έχουμε δει έως τώρα, η είσοδος σε ένα νευρωνικό δίκτυο γίνεται με την μορφή διανυσμάτων. Συνεπώς, όπως για όλες τις μορφές εισόδου, έτσι και για το κείμενο απαιτείται κάποιου είδους επεξεργασία, ώστε να αποκτήσει μια μορφή που θα είναι κατανοητή από το εκάστοτε νευρωνικό δίκτυο. Έτσι, στην ενότητα αυτή, μελετώνται διάφοροι τρόποι με τους οποίους οι λέξεις ενός κειμένου μετατρέπονται σε μαθηματικές αναπαραστάσεις, ώστε να είναι σε θέση να χρησιμοποιηθούν ως είσοδοι σε μοντέλα που επιλύουν διάφορα NLP προβλήματα.

Τα τελευταία χρόνια, όλο και μεγαλύτερη αναγνώριση λαμβάνουν τεχνικές, οι οποίες αντιμετωπίζουν κάθε λέξη ενός λεξιλογίου ως διανυσματικές αναπαραστάσεις, οι οποίες ονομάζονται Εμφυτεύματα Λέξεων (Word Embeddings). Το βασικότερο κίνητρο αυτής της μεθόδου είναι να δημιουργηθούν διανύσματα λέξεων τα οποία ενθυλακώνουν την έννοια της ομοιότητας ή της διαφοροποίησης ανάμεσα σε παρόμοιες ή διαφορετικές λέξεις αντίστοιχα. Με την μετατροπή αυτήν των λέξεων σε διανυσματικές αναπαραστάσεις, καθίσταται πλέον εφικτό να εφαρμοστούν ως κριτήρια ομοιότητάς τους ένα σύνολο από μετρικές απόστασης, όπως είναι η Ευκλείδεια Απόσταση, η Ομοιότητα Συνημίτονων και άλλες.

Για την κατασκευή λεκτικών αναπαραστάσεων, συναντώνται ουσιαστικά δύο βασικές σημασιολογικές προσεγγίσεις: η Δηλωτική Σημασιολογία (Denotational Semantics) και η Σημασιολογία Κατανομής (Distributional Semantics). Η πρώτη δημιουργεί αραιότερες αναπαραστάσεις, αντιμετωπίζοντας τις λέξεις σαν ξεχωριστά σύμβολα και αδιαφορώντας για την έννοια της ομοιότητας μεταξύ τους. Αντίθετα, η δεύτερη δημιουργεί αναπαραστάσεις με βάση τα συμφραζόμενα (contextual), διατηρώντας την έννοια της ομοιότητας και της διαφοράς μεταξύ των λέξεων. Η προσέγγιση της Σημασιολογίας Κατανομής ουσιαστικά υποστηρίζει την ιδέα της συσχέτισης της κατανομής των λέξεων σε ένα κείμενο με το αντίστοιχο νόημά τους, δηλαδή την τάση που έχουν σημασιολογικά κοντινές λέξεις να εμφανίζονται σε παρόμοιες κατανομές ή συμφραζόμενα, και αντίστροφα.

Στην συνέχεια, θα αναλυθούν οι βασικότερες σύγχρονες μέθοδοι κατασκευής Εμφυτευμάτων Λέξεων (Word Embeddings).

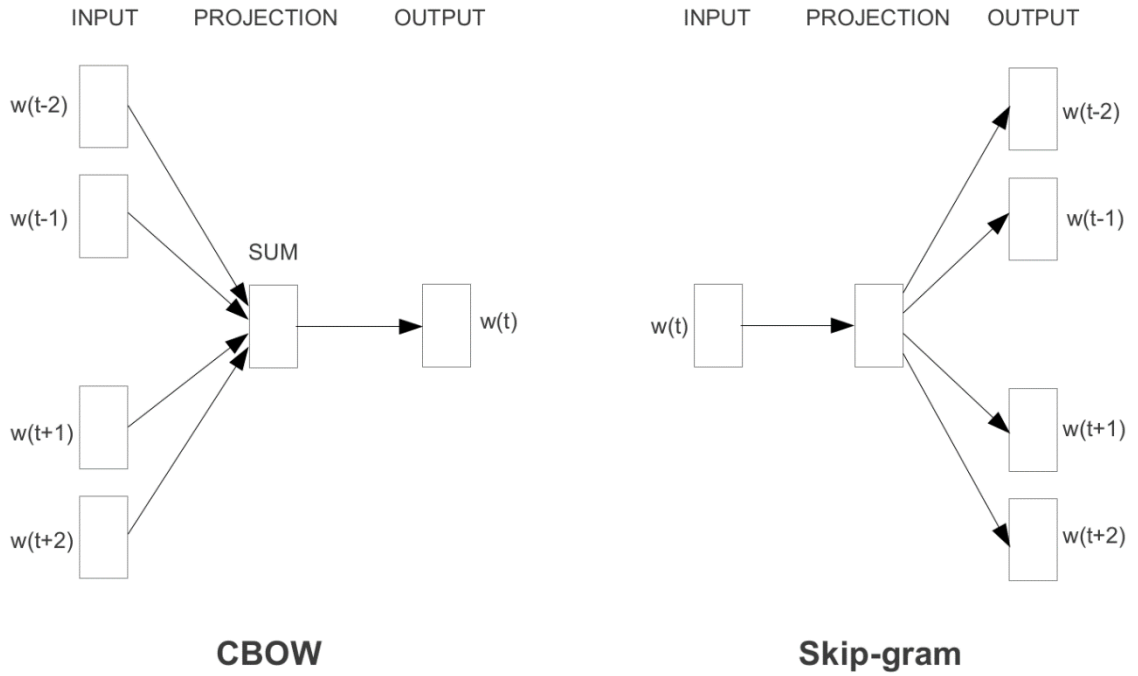
Επαναληπτικές Μέθοδοι Κατασκευής Εμφυτευμάτων - Η μέθοδος Word2Vec

Αρχικά, οι πρώτες απόπειρες που είχαν γίνει για την κατασκευή εμφυτευμάτων στηρίζονταν κυρίως σε συχνοτικές μεθόδους, όπως TF-IDF Vectorization, Count Vectorization, One-Hot Vectorization και άλλες. Ωστόσο, οι τεχνικές αυτές, λόγω των αραιών αναπαραστάσεών τους, ήταν υπολογιστικά ακριβές, δημιουργώντας έτσι την ανάγκη αναζήτησης άλλων, λιγότερο δαπανηρών, τεχνικών. Έτσι, με την πάροδο του χρόνου, οι επαναληπτικές μέθοδοι ήταν αυτές οι οποίες υπήρξαν το επίκεντρο του ενδιαφέροντος στον κλάδο του Natural Language Processing. Η βασικότερη ιδέα που ενστερνίζονταν οι τεχνικές αυτές ήταν η υπόθεση πως σημασιολογικά κοντινές λέξεις θα εμφανίζονταν σε παρόμοια συμφραζόμενα, και άρα θα είχαν διανυσματικές αναπαραστάσεις οι οποίες θα έμοιαζαν μεταξύ τους. Έτσι, τέθηκε ο στόχος της δημιουργίας διανυσματικών αναπαραστάσεων λέξεων δίνοντας έμφαση σε κάθε επανάληψη (εφόσον ήταν επαναληπτικές μέθοδοι) είτε στην πρόβλεψη μιας λέξης από τα συμφραζόμενά της, είτε στην πρόβλεψη των συμφραζόμενων από την λέξη αυτήν. Οι Mikolov et al. [85] πρότειναν την μέθοδο Word2Vec, η οποία αποτελεί την

μέθοδο η οποία γνώρισε την μεγαλύτερη απήχηση στον τομέα του NLP, και την οποία παρουσιάζουμε στην συνέχεια.

Η μέθοδος Word2Vec αποτελεί μια υπολογιστικά αποδοτική προβλεπτική μέθοδο για την εκμάθηση word embeddings από ένα καθαρό κείμενο. Ουσιαστικά, τροφοδοτείται σε ένα νευρωνικό δίκτυο, που αποτελείται από δύο στρώματα, ένα μεγάλο corpus (σώμα) λέξεων, και μέσω της χρήσης των γλωσσολογικών συμφραζόμενων των λέξεων διεξάγεται η εκπαίδευση του δικτύου. Ύστερα, οι διανυσματικές αναπαραστάσεις των λέξεων εισάγονται στον χώρο αναπαραστάσεων με τρόπο τέτοιο, ώστε λέξεις που έχουν παρόμοια συμφραζόμενα στο κείμενο να τοποθετούνται και κοντά μεταξύ τους στον χώρο αυτόν.

Η μέθοδος Word2Vec χρησιμοποιεί δύο προβλεπτικές μεθόδους: την CBOW (Continuous Bag of Words) και την Skip-Gram.

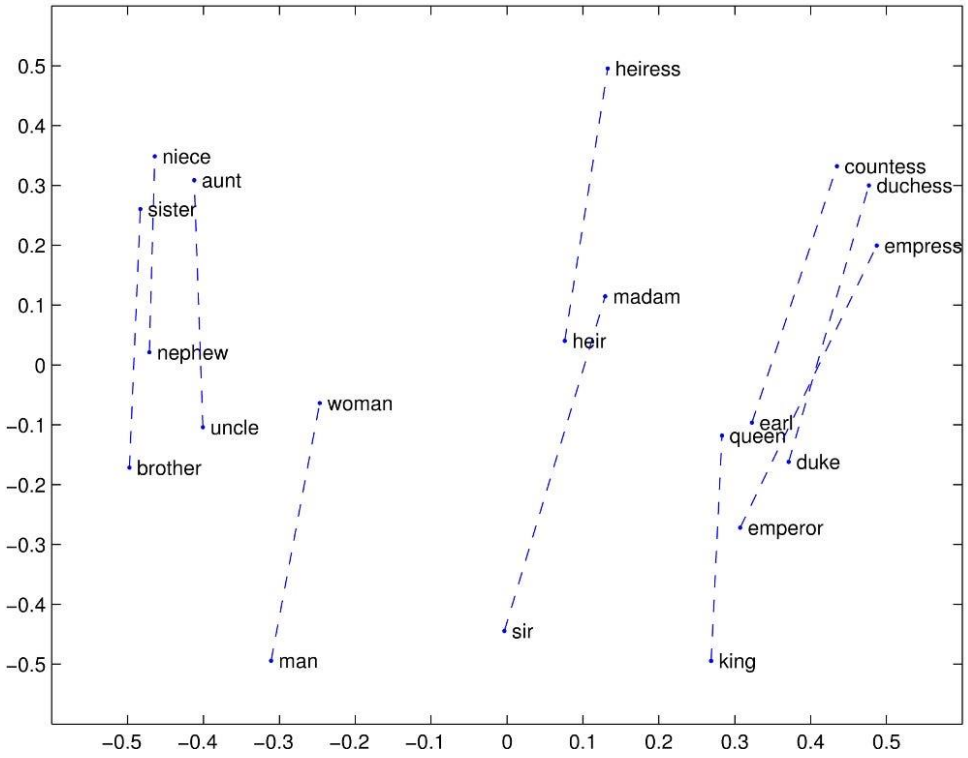


Σχήμα 25: Οι Μηχανισμοί CBOW και Skip-Gram της Μεθόδου Word2Vec [85]

Όπως φαίνεται στο παραπάνω σχήμα, οι δυο αυτοί μηχανισμοί που επιστρατεύονται έχουν «αντίθετη» λειτουργία. Ο μηχανισμός CBOW προσπαθεί να προβλέψει την «κεντρική» λέξη με βάση τα συμφραζόμενα που υπάρχουν γύρω της, ενώ ο μηχανισμός Skip-Gram προσπαθεί να προβλέψει τα γύρω συμφραζόμενα χρησιμοποιώντας την «κεντρική» λέξη. Έτσι, σε περίπτωση που η διανυσματική αναπαράσταση της «κεντρικής» λέξης δεν μπορέσει να προβλέψει αποτελεσματικά τα συμφραζόμενα της λέξης αυτής, γίνεται υπολογισμός ενός σφάλματος, το οποίο μέσω του μηχανισμού backpropagation (οπίσθιας διάδοσης σφαλμάτων) οδηγεί στην ενημέρωση των τιμών των διανυσματικών αναπαραστάσεων.

Η μέθοδος GloVe

Η μέθοδος GloVe (Global Vectors) προτάθηκε από τους Pennington et al. [86] και σε αντίθεση με την Word2Vec, για την δημιουργία των διανυσματικών αναπαραστάσεων δεν βασίζεται αποκλειστικά σε τοπικά στατιστικά, δηλαδή σε τοπικά συμφραζόμενα λέξεων, αλλά εκμεταλλεύεται και ολικά στατιστικά, δηλαδή co-appearances (συν-εμφανίσεις) λέξεων. Αρχικά, προτού ξεκινήσει η διαδικασία εκπαίδευσης του μοντέλου, κατασκευάζεται ένας πίνακας των συνεμφανίσεων των λέξεων, έστω X. Ο πίνακας αυτός, θεωρώντας ένα corpus με V πλήθος λέξεων, είναι διαστάσεων $|V| \times |V|$, όπου το στοιχείο $X_{i,j}$ υποδηλώνει το πλήθος των φορών που η λέξη i έχει εμφανιστεί μαζί με την λέξη j (co-appearance). Ύστερα, αρχικοποιούνται για κάθε λέξη του λεξιλογίου τα διανύσματα λέξεων, αποτελώντας το σημείο που σηματοδοτεί την έναρξη της διαδικασίας εκπαίδευσης, η οποία γίνεται με βασικότερο κριτήριο την ελαχιστοποίηση μιας συνάρτησης κριτηρίου J, όπως είναι το άθροισμα των τετραγώνων των σφαλμάτων. Έτσι, η μέθοδος αυτή είναι σε θέση να εκμεταλλεύεται τόσο τα πλεονεκτήματα των ολικών στατιστικών (με τις συνεμφανίσεις των λέξεων), όσο και τις αποδοτικές γραμμικές υποδομές, που επιστρατεύει η μέθοδος Word2Vec.



Σχήμα 26: Αναπαράσταση Συγγενικών Λέξεων Μοντέλου GloVe [87]

Contextual Embeddings - Εμφυτεύματα με βάση τα συμφραζόμενα

Οι διανυσματικές αναπαραστάσεις των λέξεων που αναλύθηκαν παραπάνω είναι στατικές, δηλαδή παραμένουν σταθερές ανεξάρτητα από την πρόταση στην οποία θα βρεθούν οι λέξεις. Ωστόσο, μια λέξη ενδεχομένως να φέρει τελείως διαφορετικό νόημα, αναλογικά με την πρόταση στην οποία συναντάται και τα συμφραζόμενά της, καθιστώντας έτσι τις έως τώρα μεθόδους ανήμπορες να αναπαραστήσουν επαρκώς τα διαφορετικά πιθανά νοήματα μιας λέξης. Ο περιορισμός αυτός έδωσε έδαφος στην υποστήριξη της ιδέας πως τα συμφραζόμενα

θα πρέπει να λαμβάνονται υπόψιν κατά την δημιουργία των διανυσματικών αναπαραστάσεων των λέξεων, και πως διαφορετικές διανυσματικές αναπαραστάσεις λέξεων που θα βασίζονται στα συμφραζόμενα θα οδηγήσουν σε αποτελεσματικότερες και πληρέστερες αναπαραστάσεις, και επομένως σε καλύτερα αποτελέσματα. Με τον τρόπο αυτό, εμφάνισε σημαντική άνοδο και απέκτησε μεγάλη απήχηση ένα νέο είδος εμφυτευμάτων, αυτά που βασίζονται στα συμφραζόμενα (Contextual Embeddings). Τα εμφυτεύματα αυτά προέρχονται από την εκπαίδευση γλωσσικών μοντέλων, όπως το BERT και το ELMo, το πρώτο εκ των οποίων θα παρουσιάσουμε αναλυτικά στην συνέχεια.

3.4.2 Γλωσσικά Μοντέλα

Σε ένα σύνολο προβλημάτων, που εμφανίζονται στον κλάδο της Επεξεργασίας Φυσικής Γλώσσας, είναι αναγκαίος ο υπολογισμός μιας πιθανότητας εμφάνισης ενός αριθμού λέξεων σε μια πρόταση, και υπεύθυνα για τον υπολογισμό αυτόν είναι τα Γλωσσικά Μοντέλα.

Έστω πως έχουμε μια ακολουθία από λέξεις $\{w_1, w_2, \dots, w_M\}$, M σε αριθμό, και την πιθανότητα εμφάνισής της $P(w_1, w_2, \dots, w_M)$. Η πιθανότητα αυτή υπολογίζεται από τον ακόλουθο τύπο:

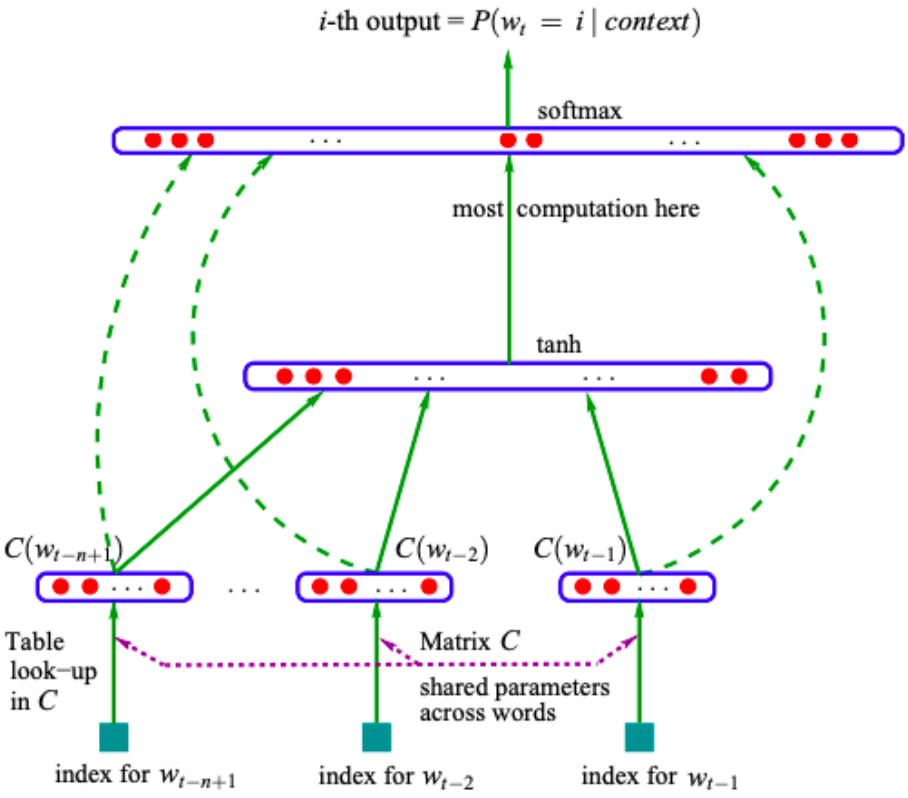
$$P(w_1, w_2, \dots, w_M) = \prod_{i=1}^{M} P(w_i | w_1, \dots, w_{i-1})$$

Γενικότερα, σκοπός είναι η δημιουργία ενός μοντέλου που αποδίδει μικρή πιθανότητα στην εμφάνιση ακολουθιών οι οποίες είναι σπάνιες ή συντακτικά ή γραμματικά λανθασμένες, και αντίθετα μεγάλη πιθανότητα σε συχνές ή συντακτικά και γραμματικά ορθές ακολουθίες.

Παραθυροποιημένα Νευρωνικά Γλωσσικά Μοντέλα

Το 2003, οι Bengio et al. [88] πρότειναν ένα νέο Γλωσσικό Μοντέλο, το Παραθυροποιημένο Νευρωνικό Γλωσσικό Μοντέλο, το οποίο κατάφερε να αντιμετωπίσει αποτελεσματικά ένα από τα κύρια προβλήματα στον χώρο του Natural Language Processing, αυτό της «Κατάρας της Διαστατικότητας» (Curse of Dimensionality).

Τα μοντέλα μη-γραμμικών νευρωνικών δικτύων επιτρέπουν την διαδικασία της μάθησης όταν έχουμε μεγάλα σημασιολογικά μεγέθη, απαιτώντας έτσι υπολογιστικές απαιτήσεις που καθίστανται βιώσιμες, έχοντας ως «κόστος» μια γραμμική αύξηση στον αριθμό των παραμέτρων. Το μοντέλο αποσκοπεί σε δύο ενέργειες: την εκμάθηση ενός διανυσματικού χώρου αναπαραστάσεων λέξεων και την εκμάθηση της κατανομής πιθανότητας για ακολουθίες λέξεων. Ως είσοδος στο μοντέλο τροφοδοτούνται οι αντίστοιχες διανυσματικές αναπαραστάσεις των λέξεων ενός παραθύρου των n προηγούμενων λέξεων. Με τον τρόπο αυτό γίνεται η κωδικοποίηση των λέξεων, των οποίων τα διανύσματα αναπαρίστανται ως $C(w_{t-n+1}), \dots, C(w_{t-2}), C(w_{t-1})$ και ονομάζονται εμφυτεύματα λέξεων ($C(w) \in \mathbb{R}^{d_w}$). Ύστερα, τα εμφυτεύματα αυτά συνενώνονται και δίνονται ως είσοδος σε ένα κρυφό στρώμα του δικτύου, οι έξοδοι του οποίου τροφοδοτούνται ύστερα σε ένα επίπεδο softmax, όπως φαίνεται στην παρακάτω εικόνα:



Σχήμα 27: Το Παραθυροποιημένο Νευρωνικό Γλωσσικό Μοντέλο [88]

Τα παραπάνω μπορούν να αναπαρασταθούν μαθηματικά από τις ακόλουθες εξισώσεις:

$$x = [C(w_{t-n+1}), \dots, C(w_{t-2}), C(w_{t-1})]$$

$$\hat{y} = softmax(\tanh(xW_1 + b_1)\, W_2 + b_2)$$

όπου $w_i \in V$, $W_1 \in \mathbb{R}^{nd_w \times d_{hid}}$, $b_1 \in \mathbb{R}^{d_{hid}}$, $W_2 \in \mathbb{R}^{d_{hid} \times |V|}$, $b_2 \in \mathbb{R}^{|V|}$ και V είναι ένα πεπερασμένο λεξιλόγιο, όπου το μέγεθος του λεξιλογίου $|V|$ κυμαίνεται μεταξύ 1.000 και 1.000.000 λέξεων, με το πιο συχνό μέγεθος να είναι περίπου 70.000 διαφορετικές λέξεις.

Πλέον, για την γλωσσική μοντελοποίηση τα Νευρωνικά Δίκτυα Πρόσθιας Τροφοδότησης έχουν αντικατασταθεί από τα RNNs και LSTMs, ενώ τα μοντέλα που παράγουν τα καλύτερα αποτελέσματα είναι τα μοντέλα Transformers (που παρουσιάσαμε σε προηγούμενη ενότητα) και τα μοντέλα που βασίζονται πάνω σε αυτά, όπως είναι το μοντέλο BERT, που θα παρουσιάσουμε αναλυτικά στην επόμενη υποενότητα.

3.4.3 Μοντέλο BERT

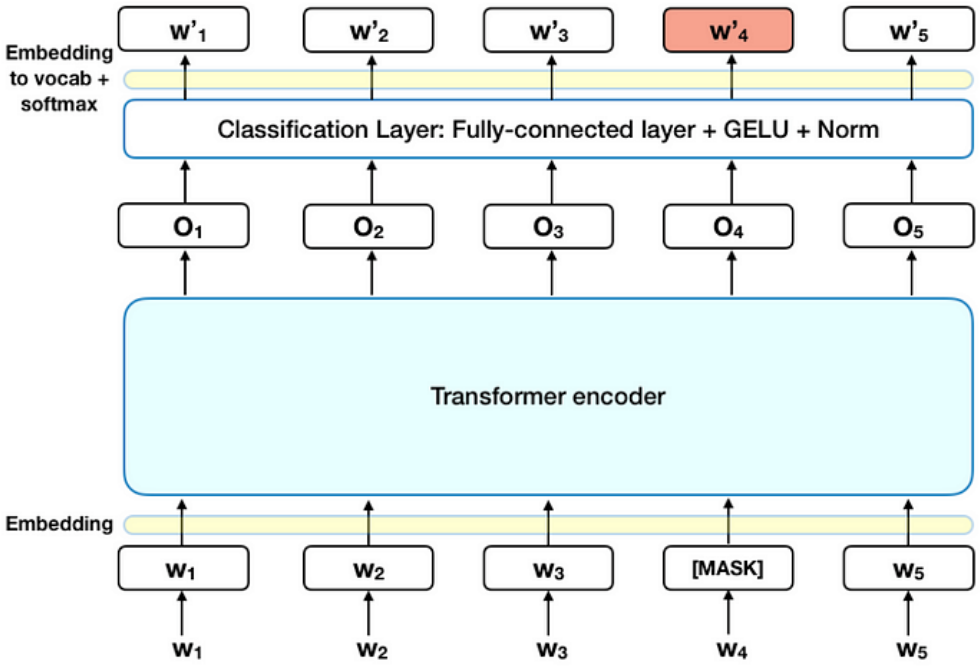
Το μοντέλο BERT (Bidirectional Encoder Representations from Transformers) είναι ένα προεκπαιδευμένο γλωσσικό μοντέλο που αναπτύχθηκε από ερευνητές της Google το 2018 [89] και έφερε επανάσταση στον τομέα της Επεξεργασίας Φυσικής Γλώσσας (Natural Language

Processing - NLP). Χρησιμοποιείται πλέον ευρέως σε μια πληθώρα προβλημάτων επεξεργασίας κειμένου και φυσικής γλώσσας, ενώ με βάση αυτό έχουν αναπτυχθεί και παραλλαγές του, οι οποίες εστιάζουν σε συγκεκριμένες πηγές κειμένου, όπως για παράδειγμα τα μέσα κοινωνικής δικτύωσης. Χαρακτηριστικό παράδειγμα τέτοιας παραλλαγής αποτελεί και το TwHIN-BERT [90], το οποίο θα παρουσιάσουμε στην επόμενη υποενότητα, και το οποίο αποτελεί και το μοντέλο που επιστρατεύσαμε στην παρούσα διπλωματική εργασία για την επεξεργασία και κατηγοριοποίηση της περιγραφής του λογαριασμού του χρήστη (account description), όπως θα δούμε και στην συνέχεια. Άλλες ευρέως διαδεδομένες παραλλαγές αποτελούν το RoBERTa (A Robustly Optimized BERT Pretraining Approach) [91], ALBERT (A Lite BERT) [92], XLNet (eXtreme Learning with Large-scale Networks) [93], DistilBERT [94], BERTweet [95] και άλλες.

Με την ανάπτυξη και διάδοση μοντέλων όπως το BERT, η τεχνική της Μεταφοράς Μάθησης, που αναλύσαμε σε προηγούμενη ενότητα, απέκτησε μεγάλη αναγνώριση, εφόσον όλο και περισσότεροι ερευνητές χρησιμοποιούσαν προεκπαιδευμένα μοντέλα και τα προσάρμοζαν στο εκάστοτε πρόβλημά τους, κάνοντας χρήση της τεχνικής fine-tuning. Η διαδικασία αυτή, όχι μόνο προσέφερε πολύ καλύτερα αποτελέσματα αναλογικά με τις έως τότε χρησιμοποιούμενες τεχνικές, αλλά απαιτούσε και καταβολή αισθητά λιγότερης προσπάθειας, χρόνου και διαθέσιμων πόρων και δεδομένων.

Ένας από τους βασικότερους λόγους, που επιτρέπουν στο BERT να επιτυγχάνει τόσο καλή επίδοση πάνω σε μια πληθώρα διαφορετικών προβλημάτων στον τομέα του NLP, είναι η προεκπαίδευσή του πάνω σε δύο μη επιβλεπόμενα προβλήματα, τα οποία του επιτρέπουν να αντιληφθεί και να κατανοήσει τα μοτίβα που διέπουν μια γλώσσα.

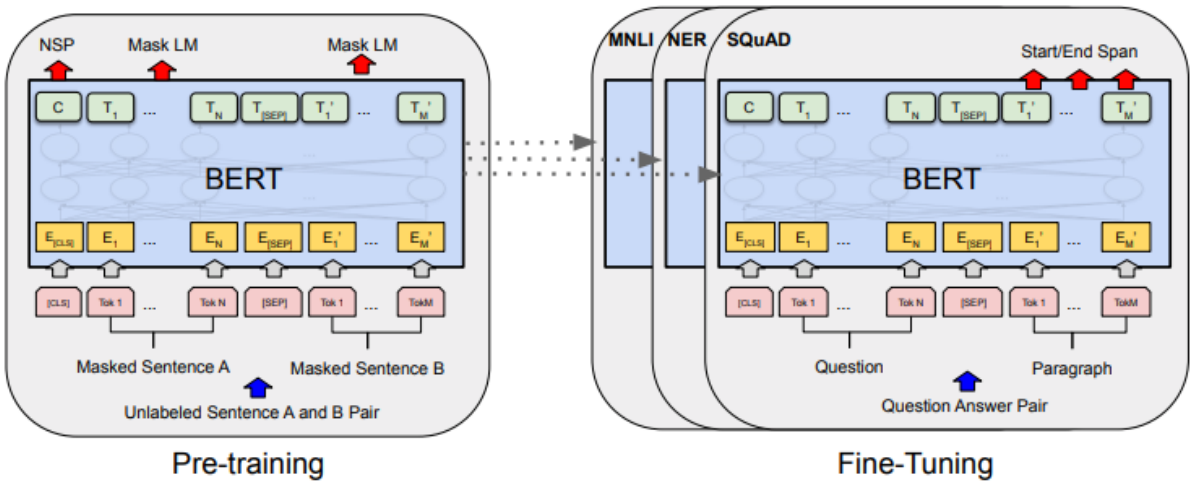
Πιο συγκεκριμένα, το πρώτο πρόβλημα πάνω στο οποίο προεκπαιδεύτηκε το μοντέλο ονομάζεται "Masked Language Modeling" (MLM). Στο συγκεκριμένο πρόβλημα, το 15% όλων των WordPiece ενδείξεων κάθε πρότασης, που ονομάζονται tokens, "κρύβεται" με τυχαίο τρόπο, κάνοντας χρήση του token [MASK]. Ύστερα, το μοντέλο προσπαθεί να προβλέψει την πραγματική τιμή των λέξεων που έχουν αντικατασταθεί με το [MASK] token, βασιζόμενο στο context (περιεχόμενο) άλλων λέξεων της ακολουθίας που δεν έχουν κρυφτεί. Τεχνικά, η πρόβλεψη των λέξεων εξόδου απαιτεί 3 βήματα. Αρχικά, απαιτείται η προσθήκη ενός επιπέδου ταξινόμησης στην κορυφή του αποτελέσματος κωδικοποίησης. Ύστερα, πολλαπλασιάζονται τα διανύσματα εξόδου με τον πίνακα των embeddings, με σκοπό να αποκτήσουν τις διαστάσεις του λεξιλογίου. Τέλος, υπολογίζεται η πιθανότητα κάθε λέξης του λεξιλογίου, κάνοντας χρήση της συνάρτησης ενεργοποίησης Softmax.



Σχήμα 28: Εκπαίδευση Μοντέλου BERT με Masked Language Modeling [96]

Το δεύτερο πρόβλημα ονομάζεται "Next Sentence Prediction" (NSP). Στο συγκεκριμένο πρόβλημα, κατά την διαδικασία της εκπαίδευσης, το μοντέλο λαμβάνει ζευγάρια προτάσεων ως είσοδο και μαθαίνει να προβλέπει εάν η δεύτερη πρόταση του ζεύγους είναι πράγματι η πρόταση που ακολουθεί την πρώτη στο αρχικό κείμενο. Στο 50% των περιπτώσεων των ζευγών εισόδου, η δεύτερη πρόταση όντως ακολουθεί την πρώτη στο αρχικό κείμενο, ενώ στο υπόλοιπο 50% δεν ισχύει αυτό, εφόσον ως δεύτερη πρόταση επιλέγεται μια τυχαία πρόταση από το κείμενο.

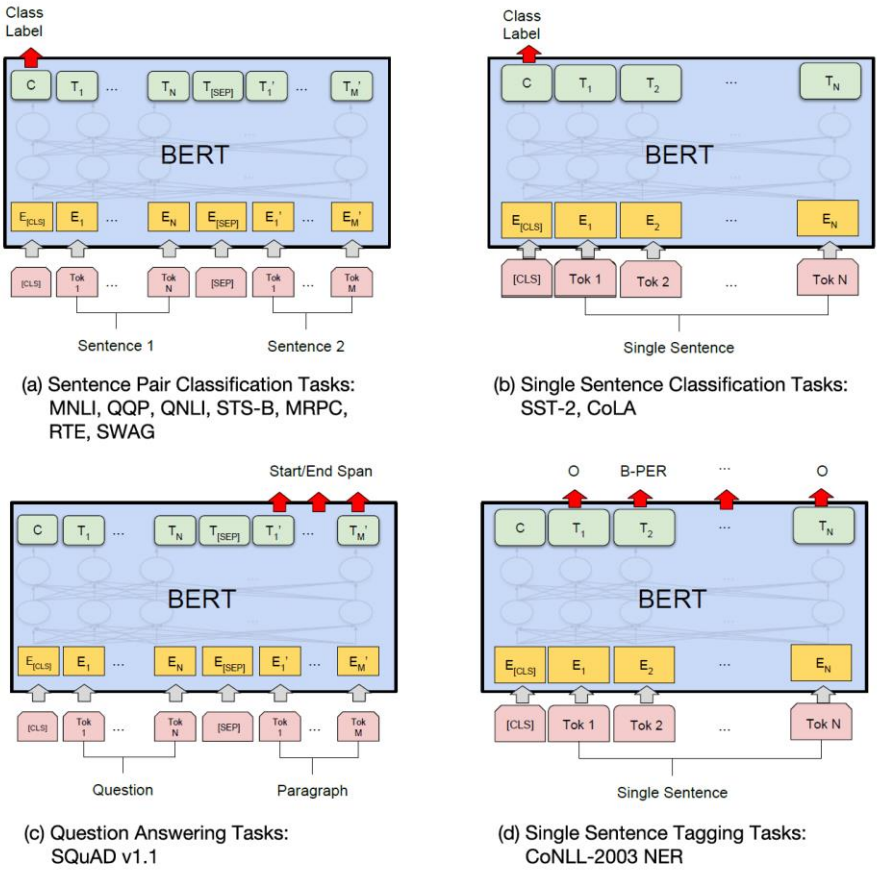
Για την διαδικασία της προεκπαίδευσης, έγινε χρήση τόσο του BooksCorpus (800M λέξεις), όσο και αγγλικών κειμένων της Wikipedia (2,500M λέξεις). Για την διαδικασία του fine-tuning, το μοντέλο BERT αρχικοποιείται με τις προεκπαιδευμένες παραμέτρους, και όλες οι παράμετροι αναπροσαρμόζονται χρησιμοποιώντας επισημασμένα δεδομένα του εκάστοτε προβλήματος - στόχου. Οι διαδικασίες της προεκπαίδευσης και του fine-tuning φαίνονται στο ακόλουθο σχήμα:



Σχήμα 29: Προεκπαίδευση και Fine-Tuning Μοντέλου BERT [89]

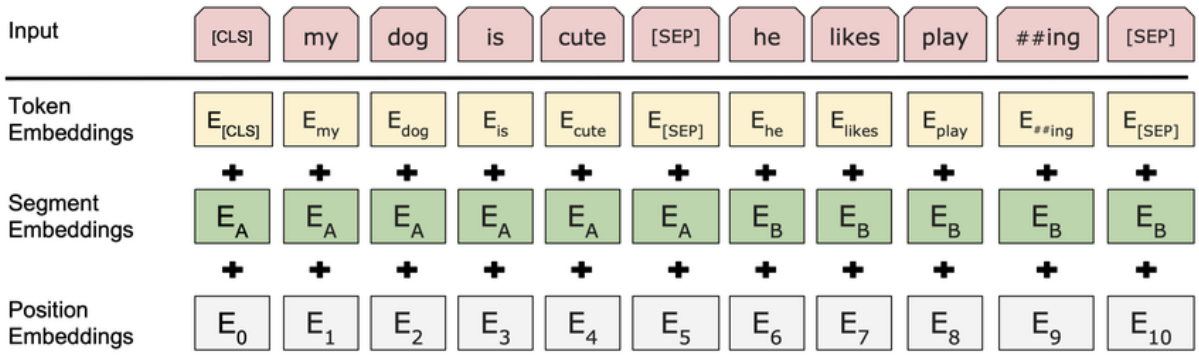
Η αρχιτεκτονική του μοντέλου BERT βασίζεται στην αρχική υλοποίηση του Transformer και ουσιαστικά αποτελεί έναν πολυστρωματικό Bidirectional Transformer κωδικοποιητή, με μηχανισμό Multi-Head αυτοπροσοχής, ο οποίος εστιάζει στην ακολουθία εισόδου και από τις δύο κατευθύνσεις. Το αρχικό μοντέλο παρουσίαζε δύο εκδοχές, του $BERT_{BASE}$ και του $BERT_{LARGE}$. Συγκεκριμένα, το $BERT_{BASE}$ διαθέτει 12 στρώματα στην στοίβα του κωδικοποιητή και το $BERT_{LARGE}$ 24 στρώματα, ενώ και οι δύο εκδοχές περιλαμβάνουν μεγαλύτερα Feedforward δίκτυα, με 768 και 1024 κρυφές μονάδες αντίστοιχα, και περισσότερες κεφαλές προσοχής, 12 και 16 αντίστοιχα, από την πρωτότυπη αρχιτεκτονική των Transformers, η οποία περιείχε 6 στρώματα κωδικοποιητή, 512 κρυφές μονάδες και 8 κεφαλές προσοχής.

Το BERT, πέρα από την εξαγωγή σημασιολογικών γλωσσικών εμφυτευμάτων (embeddings), χρησιμοποιείται και σε άλλα προβλήματα, όπως προβλήματα ταξινόμησης, προβλήματα Ερωτήσεων-Απαντήσεων ή και αναγνώριση οντοτήτων, με την προσθήκη απλώς ενός μικρού δικτύου στην κεφαλή του. Στην παρακάτω εικόνα απεικονίζονται μερικά παραδείγματα χρήσης του μοντέλου:



Σχήμα 30: Παραδείγματα Χρήσης του Μοντέλου BERT [89]

Για να έχει την δυνατότητα το BERT να αντιμετωπίσει αποδοτικά την πληθώρα των προβλημάτων για τα οποία χρησιμοποιείται, η αναπαράσταση εισόδου του είναι ρυθμισμένη έτσι ώστε να μπορεί να απεικονίσει τόσο μια απλή πρόταση, όσο και ένα σύνολο προτάσεων, σε μια ακολουθία από tokens. Για τον λόγο αυτό, αρχικά εφαρμόζουμε λεκτική ανάλυση (tokenization) στην ακολουθία εισόδου, της οποίας πάντοτε το πρώτο token είναι ο ειδικός χαρακτήρας [CLS]. Το token αυτό είναι ιδιαίτερα σημαντικό, εφόσον η τελική κρυφή κατάσταση που αντιστοιχεί σε αυτό χρησιμοποιείται ως η συνολική ακολουθία στα προβλήματα ταξινόμησης. Επίσης, χρησιμοποιείται ο ειδικός χαρακτήρας [SEP] για τον διαχωρισμό των προτάσεων (εισάγεται στο τέλος κάθε πρότασης), ενώ ταυτόχρονα ένα ειδικό εμφύτευμα προστίθεται σε κάθε token, για να υποδεικνύει σε ποια πρόταση ανήκει αυτό. Αν παρατηρήσει κανείς το Σχήμα 29, όπου φαίνεται η προεκπαίδευση και το fine-tuning του μοντέλου, συνειδητοποιεί πως το εμφύτευμα εισόδου συμβολίζεται ως E, το τελικό κρυφό διάνυσμα του ειδικού χαρακτήρα [CLS] ως C, ενώ το τελικό κρυφό διάνυσμα για το i-οστό token εξόδου ως T_i.



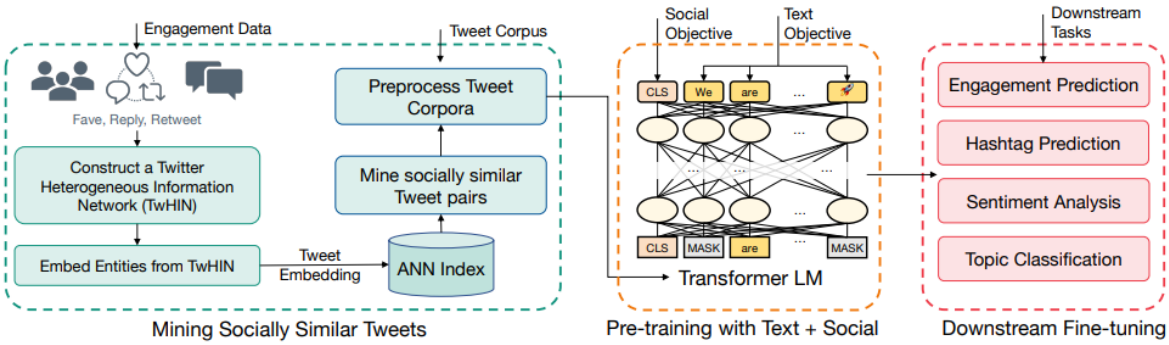
Σχήμα 31: Αναπαράσταση Εισόδου του Μοντέλου BERT [89]

Όπως φαίνεται στο παραπάνω σχήμα, η αναπαράσταση εισόδου ενός token κατασκευάζεται αθροίζοντας τόσο το αντίστοιχο εμφύτευμα του token, όσο και το τμηματικό εμφύτευμα (σε ποια πρόταση ανήκει) και το εμφύτευμα θέσης. Τέλος, αξίζει να σημειωθεί πως, προτού τροφοδοτηθούν στο μοντέλο, όλες οι ακολουθίες εισόδου οφείλουν να είναι συγκεκριμένου (προκαθορισμένου) μήκους, και για αυτό συμπληρώνονται (padding) ή «κόβονται» (truncate) αναλόγως, ενώ υπάρχει και μια μάσκα προσοχής (attention mask) που αρμοδιότητά της είναι να υποδεικνύει ποια tokens υπήρχαν πράγματι στην ακολουθία και ποια όχι, έτσι ώστε να αγνοηθεί η πληροφορία στην δεύτερη περίπτωση.

3.4.4 Μοντέλο TwHIN-BERT

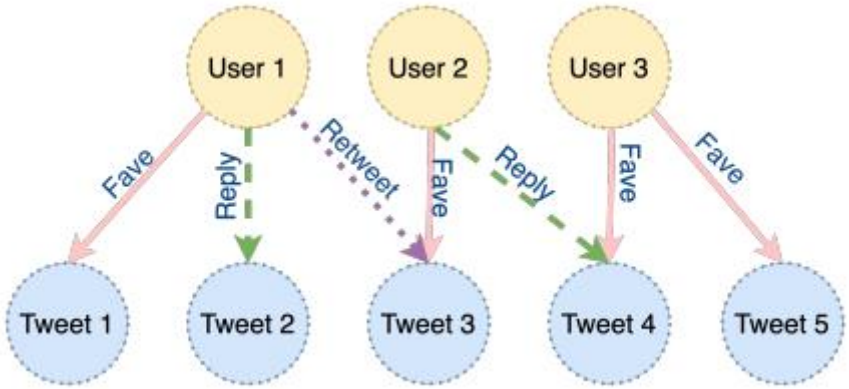
Τα προεκπαιδευμένα Γλωσσικά Μοντέλα (Pre-trained Language Models - PLMs) θεωρούνται θεμελιώδη εργαλεία στις εφαρμογές του χώρου του Natural Language Processing. Ωστόσο, τα περισσότερα από τα υπάρχοντα PLMs δεν είναι συνηθισμένα στο «θορυβώδες» (noisy) κείμενο που γράφουν οι χρήστες στα μέσα κοινωνικής δικτύωσης, ενώ η προεκπαίδευση στην οποία υποβάλλονται δεν λαμβάνει ως παράγοντα τις κοινωνικές αλληλεπιδράσεις που είναι διαθέσιμες σε ένα κοινωνικό δίκτυο. Για αυτόν τον λόγο, οι Zhang et al. [90] παρουσίασαν το TwHIN-BERT, ένα πολυγλωσσικό Γλωσσικό Μοντέλο (Multi-Lingual Language Model) που παράχθηκε ειδικά για το μέσο κοινωνικής δικτύωσης του Twitter.

Το μοντέλο TwHIN-BERT (Twitter Heterogeneous Information Network - BERT) διαφέρει από προηγούμενα PLMs στο γεγονός πως εκπαιδεύθηκε όχι μόνο με self-supervision που ήταν βασισμένο σε κείμενο, αλλά λαμβάνοντας υπόψιν και τις πολύτιμες κοινωνικές δεσμεύσεις που εμφανίζονται σε ένα Ετερογενές Δίκτυο Πληροφοριών του Twitter (TwHIN). Το μοντέλο έχει εκπαιδευθεί πάνω σε 7 δισεκατομμύρια tweets, γραμμένα σε περισσότερες από 100 διαφορετικές γλώσσες, παρέχοντας έτσι πλούσιες αναπαραστάσεις για την μοντελοποίηση μικρών «θορυβωδών» κειμένων που έχουν γραφθεί από πραγματικούς χρήστες του Twitter. Με γνώμονα τα παραπάνω, επιλέχθηκε ως το μοντέλο που επιστρατεύσαμε στην παρούσα διπλωματική εργασία για την επεξεργασία και κατηγοριοποίηση της περιγραφής του λογαριασμού του χρήστη (account description), όπως θα δούμε και στην συνέχεια.



Σχήμα 32: Διαδικασία Προεκπαίδευσης και Fine-Tuning Μοντέλου TwHIN-BERT [90]

Η βασική ιδέα πάνω στην οποία βασίστηκε η κατασκευή του μοντέλου είναι η εκμετάλλευση των «socially similar tweets» για την προεκπαίδευση, δηλαδή των tweets τα οποία ήταν «κοινωνικά πανομοιότυπα». Ως social similar tweets ορίστηκαν άτυπα τα tweets τα οποία συν-δεσμεύονται (co-engaged) από ένα πανομοιότυπο σύνολο από χρήστες. Στο παραπάνω σχήμα φαίνεται η ολοκληρωμένη διαδικασία προεκπαίδευσης και fine-tuning του μοντέλου. Πιο συγκεκριμένα, έχοντας ως είσοδο μια πληθώρα από καταγραφές κοινωνικών δεσμεύσεων, το TwHIN-BERT δημιουργεί ένα Ετερογενές Δίκτυο Πληροφοριών για να ενοποιήσει τις διαφορετικές κατηγορίες των δεσμεύσεων (Favorite, Reply, Retweet και άλλα), το οποίο φαίνεται στο σχήμα που ακολουθεί.



Σχήμα 33: Σύλληψη των Κοινωνικών Δεσμεύσεων από το TwHIN [90]

Ύστερα, χρησιμοποιείται μια κλιμακούμενη μέθοδος εμφυτευμάτων ετερογενούς δικτύου για την σύλληψη των συν-δεσμεύσεων και την αντιστοίχιση των tweets και των χρηστών σε έναν διανυσματικό χώρο. Με αυτόν τον τρόπο, η κοινωνική ομοιότητα μετατρέπεται σε ομοιότητα χώρου εμφυτευμάτων, και κάνοντας χρήση της αναζήτησης του Κατά - Προσέγγιση Πλησιέστερου Γείτονα (Approximate Nearest Neighbor - ANN) πάνω στα εμφυτεύματα των tweets με κριτήριο την απόσταση συνημίτονου (cosine distance), είναι δυνατή η εξαγωγή των ζευγαριών των κοινωνικά πανομοιότυπων tweets. Χρησιμοποιώντας τα tweets αυτά και σε συνδυασμό με το Masked Language Modeling (MLM), που περιγράψαμε στην προηγούμενη υποενότητα, εισάγεται ένα Contrastive Social Objective που επιβάλλει στο μοντέλο να αποφανθεί για το αν ένα ζεύγος από tweets είναι κοινωνικά πανομοιότυπο ή όχι. Ουσιαστικά, αφότου περάσουν τα tweets μέσα από το Language Model, το μοντέλο εκπαιδεύεται ελέγχοντας από κοινού το loss του Contrastive Social Objective και του Masked

Language Modeling. Έτσι, τα Text Objective και Social Objective συνδυάζονται για την αποτελεσματική προεκπαίδευση του μοντέλου, το οποίο στην συνέχεια μπορεί να γίνει fine-tuned για διάφορες εφαρμογές.

Η αρχιτεκτονική του μοντέλου χρησιμοποιεί την ίδια Transformer αρχιτεκτονική με το BERT, ενώ γίνεται χρήση του XLM-R tokenizer [97], που επιτρέπει αρκετά μεγάλη κάλυψη σε όλες τις γλώσσες. Τέλος, το μοντέλο αποτελείται από λεξιλόγιο μεγέθους 250K λέξεων, ενώ το μέγιστο μήκος ακολουθίας (max sequence length) έχει οριστεί σε 128 tokens.

3.5 Μετρικές Αξιολόγησης

Οι Μετρικές Αξιολόγησης, αν και δεν αποτελούν συστατικό του Δικτύου, είναι ιδιαίτερα σημαντικές στον τομέα της Τεχνητής Νοημοσύνης, εφόσον μας επιτρέπουν όχι μόνο να αποκτάμε μια συνολική εικόνα των επιδόσεων ενός μοντέλου, αλλά και να συγκρίνουμε τις επιμέρους επιδόσεις διαφορετικών μοντέλων μεταξύ τους, για την εξαγωγή απαραίτητων συμπερασμάτων.

Μια από τις βασικότερες και ευρέως χρησιμοποιούμενες μετρικές αξιολόγησης είναι ο **Πίνακας Σύγχυσης (Confusion Matrix)**, ο οποίος αποτελεί έναν ειδικό πίνακα που επιτρέπει την οπτικοποίηση της απόδοσης ενός αλγορίθμου Επιβλεπόμενης Μάθησης. Στην Μη-Επιβλεπόμενη Μάθηση, ο πίνακας αυτός ονομάζεται **Πίνακας Ταιριάσματος (Matching Matrix)**. Κάθε γραμμή του Confusion Matrix αναπαριστά τις περιπτώσεις των προβλεπόμενων κλάσεων, ενώ κάθε στήλη του τις περιπτώσεις των πραγματικών, όπως φαίνεται στον ακόλουθο πίνακα:

		Actual Class	
		Positive (**P**)	Negative (**N**)
Predicted Class	Positive (**P**)	True Positive (**TP**)	False Positive (**FP**)
	Negative (**N**)	False Negative (**FN**)	True Negative (**TN**)

Πίνακας 1: Confusion Matrix

Οι μεταβλητές του Confusion Matrix ορίζονται ως εξής:

- **True Positive (TP)**: Αναφέρεται στις περιπτώσεις όπου έχουμε προβλέψει θετικά (Predicted Class), και η πρόβλεψή μας επιβεβαιώνεται (Actual Class).
- **True Negative (TN)**: Αναφέρεται στις περιπτώσεις όπου έχουμε προβλέψει αρνητικά (Predicted Class), και η πρόβλεψή μας επιβεβαιώνεται (Actual Class).
- **False Positive (FP)**: Αναφέρεται στις περιπτώσεις όπου έχουμε προβλέψει θετικά (Predicted Class), αλλά η πρόβλεψή μας δεν επιβεβαιώνεται (Actual Class).

- **False Negative (FN)**: Αναφέρεται στις περιπτώσεις όπου έχουμε προβλέψει αρνητικά (Predicted Class), αλλά η πρόβλεψή μας δεν επιβεβαιώνεται (Actual Class).

Με βάση τις μεταβλητές αυτές, ορίζονται οι ακόλουθες παράμετροι - μετρικές.

Ορθότητα (Accuracy)

Η Ορθότητα ορίζεται ως το ποσοστό των σωστών προβλέψεων του ταξινομητή, επί του συνόλου των προβλέψεων που έγιναν, και δίνεται από τον ακόλουθο τύπο:

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions\ made} = \frac{TP + TN}{TP + TN + FP + FN}$$

Βέβαια, για την αξιολόγηση του μοντέλου μας δεν αρκεί μόνο να εξετάσουμε την ορθότητά του. Αυτό συμβαίνει διότι σε περιπτώσεις που το σύνολο δεδομένων μας παρουσιάζει μεγάλες ανομοιομορφίες στην κατανομή των δειγμάτων των κλάσεων, το μοντέλο θα μάθει να προβλέπει την κλάση που περιέχει τα περισσότερα δείγματα, οδηγώντας έτσι σε ένα μεγάλο ποσοστό της μετρικής Accuracy. Αυτό, ωστόσο, δεν σημαίνει πως το μοντέλο έχει εκπαιδευθεί σωστά και ότι είναι σε θέση να γενικεύει τις προβλέψεις του, εφόσον δεν θα είναι σε θέση να προβλέπει σωστά την κλάση με τα λιγότερα δείγματα. Για την αποφυγή του παραπάνω προβλήματος, ορίζουμε και τις ακόλουθες μετρικές.

Ανάκληση (Recall / True Positive Rate / Sensitivity)

Η Ανάκληση ορίζεται ως το ποσοστό των θετικών δειγμάτων που προβλέφθηκαν σωστά από τον ταξινομητή, και δίνεται από τον ακόλουθο τύπο:

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

Ακρίβεια (Precision)

Η Ακρίβεια ορίζεται ως το ποσοστό των σωστών θετικών προβλέψεων του ταξινομητή, και δίνεται από τον ακόλουθο τύπο:

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

F1-Score

Η μετρική F1-Score συνδυάζει τις μετρικές της Ακρίβειας και της Ανάκλησης, αποτελώντας τον αρμονικό μέσο τους, και δίνεται από τον ακόλουθο τύπο:

$$F1\ score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} = 2 \cdot \frac{TP}{2 \cdot TP + FP + FN}$$

False Positive Rate

Η μετρική False Positive Rate ορίζεται ως το ποσοστό των αρνητικών δειγμάτων που ο ταξινομητής πρόβλεψε λανθασμένα ως θετικά, και δίνεται από τον ακόλουθο τύπο:

$$FPR = \frac{FP}{TN + FP}$$

Specificity (True Negative Rate)

Η μετρική Specificity ορίζεται ως το ποσοστό των αρνητικών δειγμάτων που προβλέφθηκαν σωστά από τον ταξινομητή, και δίνεται από τον ακόλουθο τύπο:

$$Specificity = \frac{TN}{TN + FP}$$

Matthews Correlation Coefficient (MCC)

Η μετρική Matthews Correlation Coefficient ορίζεται ως η εκτίμηση της συσχέτισης μεταξύ της κλάσης που προβλέπεται, και της κλάσης στην οποία ανήκουν πραγματικά τα δείγματα, και δίνεται από τον ακόλουθο τύπο:

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FN) \cdot (TP + FP) \cdot (TN + FP) \cdot (TN + FN)}}$$

ROC Curve and AUC (Area Under Curve)

Τα μοντέλα πρόβλεψης έχουν ως έξοδο την πιθανότητα του κάθε στιγμιότυπου να ανήκει στην κλάση προς ανίχνευση. Για αυτόν τον λόγο απαιτείται ένα κατώφλι (threshold), έτσι ώστε να μετατρέπεται αυτή η πιθανότητα σε 0 ή 1, και να υποδεικνύεται έτσι σε ποια κλάση προβλέπεται να ανήκει το εκάστοτε στιγμιότυπο. Το ROC (Receiver Operating Characteristic) γράφημα ουσιαστικά συνοψίζει όλα τα πιθανά Confusion Matrices που προκύπτουν από κάθε διαφορετικό threshold. Ο άξονας y του γραφήματος είναι το Recall του κάθε πίνακα, ενώ ο άξονας x ορίζεται ως το 1 – Sensitivity. Κάνοντας χρήση του γραφήματος αυτού, μπορούμε να ανιχνεύσουμε την τιμή του threshold που δίνει τα καλύτερα αποτελέσματα για το συγκεκριμένο μοντέλο.

Το AUC (Area Under the receiver operating characteristic Curve) είναι το εμβαδόν μεταξύ ενός ROC γραφήματος και του άξονα των x, αποτελώντας μετρική για τη σύγκριση μεταξύ μοντέλων, εφόσον μεγαλύτερο εμβαδόν υποδεικνύει και καλύτερα αποτελέσματα.

Στα πλαίσια της διπλωματικής αυτής επιστρατεύθηκαν οι μετρικές αξιολόγησης Accuracy, Precision, Recall, Specificity και F1-Score, όπως θα δούμε και στην συνέχεια.

Κεφάλαιο 4

4. Σύνολο Δεδομένων

Στο Κεφάλαιο αυτό παρουσιάζεται αναλυτικά το Σύνολο Δεδομένων (Dataset) το οποίο χρησιμοποιήθηκε για την εκπόνηση της διπλωματικής αυτής, καθώς και η προεπεξεργασία στην οποία υποβλήθηκαν τα υποσύνολα δεδομένων που το ίδιο εμπεριέχει, για να προκύψει εν τέλει το τελικό σύνολο δεδομένων το οποίο και επιστρατεύθηκε για την εκπαίδευση και αξιολόγηση των προτεινόμενων μοντέλων, τα οποία θα παρουσιαστούν σε επόμενα κεφάλαια.

4.1 Περιγραφή Συνόλου Δεδομένων

Το σύνολο δεδομένων που χρησιμοποιήθηκε για την εκπόνηση της παρούσας διπλωματικής, δηλαδή για την εκπαίδευση των προτεινόμενων μοντέλων και την τελική κατηγοριοποίηση των χρηστών του Twitter, ονομάζεται Cresci-2017 και προτάθηκε από τους Cresci et al. κατά την δημοσίευση ενός paper τους [8], το οποίο αποτελεί μια από τις σημαντικότερες συνεισφορές στον τομέα της ανίχνευσης και αντιμετώπισης των bots στα μέσα κοινωνικής δικτύωσης και έχει αναγνωριστεί από την επιστημονική κοινότητα ως σημαντική πηγή πληροφοριών και ανάλυσης.

Στον παρακάτω πίνακα φαίνονται τα βασικότερα χαρακτηριστικά του συνόλου αυτού, τα οποία περιλαμβάνουν τα υποσύνολά του, μια σύντομη περιγραφή τους, το πλήθος των λογαριασμών (accounts) και των δημοσιεύσεων (tweets) που περιέχουν, καθώς και ένα έτος, το οποίο αναπαριστά τον μέσο όρο των ετών κατά τα οποία δημιουργήθηκαν οι λογαριασμοί που ανήκουν στο συγκεκριμένο υποσύνολο - dataset:

		Statistics		
Dataset	**Description**	**Accounts**	**Tweets**	**Year**
genuine accounts	verified accounts that are human-operated	3.474	8.377.522	2011
social spambots #1	retweeters of an Italian political candidate	991	1.610.176	2012
social spambots #2	spammers of paid apps for mobile devices	3.457	428.542	2014
social spambots #3	spammers of products on sale at Amazon.com	464	1.418.626	2011
traditional spambots #1	training set of spammers used by Yang et al. in [98]	1.000	145.094	2009
traditional spambots #2	spammers of scam URLs	100	74.957	2014
traditional spambots #3	automated accounts spamming job offers	433	5.794.931	2013
traditional spambots #4	another group of automated accounts spamming job offers	1.128	133.311	2009
fake followers	simple accounts that inflate the number of followers of another account	3.351	196.027	2012
test set #1	mixed set of 50% genuine accounts + 50% social spambots #1	1.982	4.061.598	-
test set #2	mixed set of 50% genuine accounts + 50% social spambots #3	928	2.628.181	-

Πίνακας 2: Cresci-2017 Dataset

Πιο συγκεκριμένα, το dataset των **genuine accounts** είναι ένα τυχαίο δείγμα από λογαριασμούς που χειρίζονται πραγματικοί χρήστες. Για να το κατασκευάσουν αυτό, οι συγγραφείς του paper επέλεξαν τυχαία χρήστες του Twitter και επικοινώνησαν μαζί τους, ρωτώντας τους μια απλή ερώτηση σε φυσική γλώσσα. Όλες οι απαντήσεις που δόθηκαν από τους χρήστες επαληθεύτηκαν μια προς μια από τους συγγραφείς, και όλα τα 3.474 accounts τα οποία ανταποκρίθηκαν στην ερώτησή τους καταχωρήθηκαν στο dataset των genuine accounts, ως πραγματικοί χρήστες, ενώ τα accounts τα οποία δεν ανταποκρίθηκαν, δεν χρησιμοποιήθηκαν στην επικείμενη μελέτη.

Το dataset των **social spambots #1** δημιουργήθηκε ύστερα από την παρατήρηση της δραστηριότητας ενός καινοτόμου συνόλου από social bots, τα οποία ανακάλυψαν οι συγγραφείς στο Twitter κατά την διάρκεια των δημαρχιακών εκλογών στην Ρώμη, το 2014. Ένας από τους επιλαχόντες δημάρχους προσέλαβε για την προεκλογική του εκστρατεία μια εταιρεία που ασχολούταν με το marketing στα social media, και χρησιμοποίησε περίπου 1.000 αυτοματοποιημένους λογαριασμούς για να κοινοποιεί και να διαφημίζει τις πολιτικές του ιδέες. Προς έκπληξή τους, οι συγγραφείς παρατήρησαν πως οι συγκεκριμένοι αυτοματοποιημένοι λογαριασμοί παρομοίαζαν πραγματικούς λογαριασμούς με κάθε δυνατό τρόπο, εφόσον κάθε profile περιείχε αναλυτικές προσωπικές πληροφορίες του χρήστη, οι οποίες ήταν είτε ψεύτικες (account description, τοποθεσία), είτε κλεμμένες (φωτογραφία profile), ενώ το πλήθος, η

συχνότητα, και το είδος των tweets που δημοσίευαν κατόπτριζε έναν πραγματικό χρήστη. Ωστόσο, κάθε φορά που ο πολιτικός υποψήφιος δημοσίευε ένα νέο tweet από τον επίσημο λογαριασμό του, όλα αυτά τα αυτοματοποιημένα bots το έκαναν retweet μέσα σε λίγα λεπτά, επιτρέποντας έτσι στον υποψήφιο να επεκτείνει την «εμβέλεια» και την απήχηση που γνώριζαν οι πολιτικές του ιδέες σε αρκετά περισσότερα accounts, εκτός αυτών που τον ακολουθούσαν ήδη, καταφέρνοντας να επηρεάσει ακόμα και τις μετρικές του Twitter κατά την διάρκεια της προεκλογικής περιόδου. Τα bots αυτά είχαν χιλιάδες ακόλουθους (followers) και φίλους (friends), η πλειοψηφία των οποίων ήταν πραγματικοί χρήστες, ενώ προσομοίαζαν αληθινούς χρήστες της πλατφόρμας σε τέτοιο βαθμό, που πραγματικοί χρήστες έμπαιναν στην διαδικασία να τα προσεγγίσουν, ξεκινώντας με αυτά διαδικτυακή συζήτηση.

Το dataset των **social spambots #2** αποτελεί ένα σύνολο από social bots, τα οποία προωθούσαν ένα συγκεκριμένο hashtag, το #TALNTS, για ένα χρονικό διάστημα πολλών μηνών. Πιο συγκεκριμένα, το Talnts είναι μια εφαρμογή κινητού τηλεφώνου σχεδιασμένη για την αλληλεπίδραση και πρόσληψη καλλιτεχνών, οι οποίοι εργάζονταν στους τομείς της μουσικής, της ψηφιακής φωτογραφίας και άλλους. Η πλειοψηφία των tweets που δημοσίευαν τα bots αυτά ήταν ακίνδυνα μηνύματα, δημοσιεύοντας περιστασιακά και tweets στα οποία έκαναν mention έναν πραγματικό χρήστη και του πρότειναν να αγοράσει την VIP εκδοχή της εφαρμογής.

Οι συγγραφείς κατάφεραν να ανακαλύψουν και ένα τρίτο group από social bots, το λεγόμενο **social spambots #3**, το οποίο διαφήμιζε προϊόντα σε έκπτωση στο *Amazon.com*, δημοσιεύοντας URLs τα οποία ανακατεύθυναν τον χρήστη στα προϊόντα αυτά. Όμοια με τις δύο προηγούμενες κατηγορίες social spambots, οι αυτοματοποιημένοι αυτοί λογαριασμοί παρεμβάλλαν τα spam tweets τους μεταξύ των ακίνδυνων και φαινομενικά «πραγματικών» δημοσιεύσεών τους.

Εκτός των συνόλων των genuine χρηστών και των 3 ειδών των social spambots, οι συγγραφείς συλλέξαν και 4 είδη από traditional (παραδοσιακά) spambots. Τα **traditional spambots #1** αποτελούν το dataset το οποίο χρησιμοποιήσαν οι Yang et al. [98] ως training set για την εκπαίδευση ενός machine learning ταξινομητή για την ανίχνευση των εξελισσόμενων Twitter spambots. Οι λογαριασμοί που ανήκουν στο dataset **traditional spambots #2** αποτελούν απλοϊκά bots τα οποία επανειλημμένως έκαναν mention άλλους χρήστες σε tweets τα οποία περιείχαν scam URLs. Για να δελεάσουν τους χρήστες να πατήσουν τα κακόβουλα links, το περιεχόμενο των tweets προσκαλούσε τους αναφερόμενους χρήστες σε διεκδίκηση χρηματικού επάθλου. Τα dataset των **traditional spambots #3** και **traditional spambots #4** αναφέρονται σε 2 διαφορετικά σύνολα από bots, τα οποία επανειλημμένως δημοσίευαν tweets για διαθέσιμες θέσεις εργασίας και προσφορές εργασίας.

Οι Fake Followers αποτελούν άλλο ένα είδος κακόβουλων λογαριασμών, οι οποίοι έχουν προξενήσει αρκετό ενδιαφέρον τόσο στους διαχειριστές της πλατφόρμας, όσο και στην επιστημονική κοινότητα. Οι fake followers αποτελούν αρκετά απλοϊκούς λογαριασμούς, τόσο στον σχεδιασμό τους, όσο και στην λειτουργία τους. Τον Απρίλιο του 2013, οι συγγραφείς του paper αγόρασαν 3.351 fake accounts από 3 διαφορετικά διαδικτυακά markets του Twitter, ονόματι *fastfollowerz.com*, *intertwitter.com* και *twittertechnology.com*. Έτσι, συγχωνεύοντας όλα αυτά τα account τα οποία απέκτησαν με αυτόν τον τρόπο, κατάφεραν να δημιουργήσουν το dataset των **fake followers**.

Στον παραπάνω πίνακα εμφανίζονται και δύο dataset, τα **test set #1** και **test set #2**, τα οποία αποτελούν δύο σύνολα τα οποία δημιούργησαν οι συγγραφείς του paper, αναμειγνύοντας μεταξύ τους κάποια από τα άλλα υπάρχοντα dataset.

Με γνώμονα όλα τα παραπάνω και αναλογιζόμενοι και τις ανάγκες της προτεινόμενης υλοποίησης, την οποία θα παρουσιάσουμε στα επόμενα κεφάλαια, αποφασίσαμε να χρησιμοποιήσουμε για την εκπόνηση της παρούσας διπλωματικής τα dataset των **Genuine Accounts** και **Social Spambots #1**. Τα dataset αυτά, όχι μόνο θα μας δώσουν την δυνατότητα να διακρίνουμε τα χαρακτηριστικά και την δραστηριότητα μεταξύ των πραγματικών χρηστών και των spambots της κατηγορίας αυτής, αλλά και να αναπτύξουμε ένα μοντέλο το οποίο θα αξιοποιεί τα παραπάνω για να ανιχνεύει αποτελεσματικά τους κακόβουλους λογαριασμούς. Παράλληλα, ο λόγος πίσω από την επιλογή του dataset των Social Spambots #1 για την εκπροσώπηση των αυτοματοποιημένων χρηστών ήταν διττός. Τα Social Spambots #1, όχι μόνο αποτελούν αυτοματοποιημένους λογαριασμούς οι οποίοι προσομοιάζουν τους πραγματικούς χρήστες με κάθε δυνατό τρόπο, όπως αναφέρθηκε και παραπάνω, αλλά το πλήθος των accounts και των tweets που εμπεριέχει το συγκεκριμένο dataset μας δίνει την δυνατότητα να έχουμε για την διεξαγωγή των πειραμάτων μας τόσο ένα αξιόλογο πλήθος από account descriptions, όσο και ένα αξιόλογο πλήθος αρκετά ποιοτικών εικόνων, που αντιπροσωπεύουν την δραστηριότητα του λογαριασμού του χρήστη, όπως θα δούμε και αναλυτικότερα στα επόμενα κεφάλαια.

4.2 Προεπεξεργασία Συνόλου Δεδομένων

Το κάθε ένα από τα dataset που επιλέξαμε να χρησιμοποιήσουμε για την διεξαγωγή των πειραμάτων μας, **Genuine Accounts** και **Social Spambots #1**, αποτελείται από 2 αρχεία .csv: ένα που αναφέρεται στους χρήστες και τα χαρακτηριστικά τους, και ένα που αναφέρεται στα tweets τους και τα χαρακτηριστικά τους. Τα χαρακτηριστικά και οι πληροφορίες που μας δίνονται για την κάθε οντότητα φαίνονται στο κάθε αρχείο σε στήλες, το όνομα των οποίων παρουσιάζουμε στον παρακάτω πίνακα, ανάλογα με την οντότητα στην οποία αναφερόμαστε:

Dataset	**Στήλες – Χαρακτηριστικά**
Genuine Users Social Spambots #1 Users	"id", "name", "screen_name", "statuses_count", "followers_count", "friends_count", "favourites_count", "listed_count", "url", "lang", "time_zone", "location", "default_profile", "default_profile_image", "geo_enabled", "profile_image_url", "profile_banner_url", "profile_use_background_image", "profile_background_image_url_https", "profile_text_color", "profile_image_url_https", "profile_sidebar_border_color", "profile_background_tile", "profile_sidebar_fill_color", "profile_background_image_url", "profile_background_color", "profile_link_color", "utc_offset", "is_translator", "follow_request_sent", "protected", "verified", "notifications", "description", "contributors_enabled", "following", "created_at", "timestamp", "crawled_at", "updated", "test_set_1", "test_set_2"

Genuine Tweets Social Spambots #1 Tweets	"id", "text", "source", "user_id", "truncated", "in_reply_to_status_id", "in_reply_to_user_id", "in_reply_to_screen_name", "retweeted_status_id", "geo", "place", "contributors", "retweet_count", "reply_count", "favorite_count", "favorited", "retweeted", "possibly_sensitive", "num_hashtags", "num_urls", "num_mentions", "created_at", "timestamp", "crawled_at", "updated"

Πίνακας 3: Στήλες - Χαρακτηριστικά των Dataset Genuine Accounts και Social Spambots #1

Το πλήθος των στηλών (χαρακτηριστικών) των αρχείων που αναφέρονται στους χρήστες είναι 42, ενώ το αντίστοιχο πλήθος για τα αρχεία που αναφέρονται στα tweets είναι 25. Για την επεξεργασία των παραπάνω datasets χρησιμοποιήσαμε την βιβλιοθήκη ***pandas*** [99] της Python, μέσω της οποίας τα μετατρέψαμε σε *dataframes* και εφαρμόσαμε τις απαραίτητες τροποποιήσεις σε αυτά.

Για να αντιμετωπίσουμε το πρόβλημα που προκύπτει από τα imbalanced datasets (μη ισορροπημένα datasets), εφόσον το dataset των genuine accounts περιέχει 3.474 χρήστες, ενώ το dataset των social spambots #1 περιέχει 991, δημιουργούμε ένα dataset το οποίο τελικά (ύστερα από την απαιτούμενη προεπεξεργασία) περιέχει 943 πραγματικούς (genuine) χρήστες και 943 social spambots, για να είναι ακριβώς ισορροπημένος τόσο ο αριθμός των πραγματικών και των αυτοματοποιημένων περιγραφών, όσο και ο αριθμός των πραγματικών και αυτοματοποιημένων εικόνων, οδηγώντας έτσι σε μια ισορροπημένη μάθηση. Αυτή η τεχνική, που αντιμετωπίζει επιτυχώς το πρόβλημα του μη ισορροπημένου dataset με το να εφαρμόζει υποδειγματοληψία στο σύνολο των πραγματικών χρηστών, έχει υιοθετηθεί και από προηγούμενες έρευνες [50] [100].

Πιο συγκεκριμένα, για το dataframe των **genuine users**, αφού πρώτα αφαιρέσαμε από το original dataset τους χρήστες που δεν έχουν περιγραφή (η οποία είναι αναγκαία για την υλοποίηση των πειραμάτων μας), στην συνέχεια κρατάμε όλους τους χρήστες που στην στήλη "test_set_1" έχουν την τιμή 1, δηλαδή ανήκουν στο σύνολο το οποίο είχαν δημιουργήσει και οι ίδιοι οι συγγραφείς του original paper που δημιούργησε το dataset. Ύστερα, από τους χρήστες που έχουν μείνει, αφαιρούμε αυτούς που δεν έχουν δημοσιεύσει κανένα tweet το οποίο να έχει καταχωρηθεί στο dataframe των genuine tweets, το οποίο αποτελεί το dataframe στο οποίο είναι καταχωρημένα τα tweets των χρηστών που υπάρχουν στο dataframe των genuine users. Αυτό εφαρμόστηκε διότι αν ένας χρήστης δεν είχε κανένα καταχωρημένο tweet, τότε δεν θα μπορούσε να προκύψει από αυτόν εικόνα η οποία θα αντιπροσώπευε την δραστηριότητα του λογαριασμού του, η οποία αντίστοιχα είναι απαραίτητη για την διεξαγωγή των πειραμάτων μας. Για αυτόν τον λόγο, έγινε έρευνα στους χρήστες οι οποίοι δεν ανήκαν στο λεγόμενο "test_set_1" (δηλαδή είχαν τιμή 0 σε αυτήν την στήλη, και τους οποίους αρχικά είχαμε αφαιρέσει) για να βρεθούν πραγματικοί χρήστες οι οποίοι διέθεταν τόσο πλήθος από καταχωρημένα tweets, όσο και διαθέσιμες περιγραφές για τον λογαριασμό τους. Έτσι, βρέθηκαν 73 χρήστες που πληρούσαν τα κριτήρια αυτά και προστέθηκαν στο dataset, έτσι ώστε να συμπληρωθεί ο αριθμός των 943 πραγματικών χρηστών που χρειαζόμασταν. Τέλος, αφαιρέθηκαν οι στήλες "test_set_1", "test_set_2", ενώ προστέθηκε μια νέα στήλη, με το όνομα "label", δηλαδή ετικέτα, η οποία θα αποτελούσε την στήλη που υποδείκνυε την κλάση στην οποία ανήκε ο συγκεκριμένος χρήστης, και η οποία αρχικοποιήθηκε με τιμές 0 σε όλα τα δείγματα, εφόσον στα πλαίσια της εργασίας αυτής, θεωρήσαμε την κλάση των bots ως την

θετική κλάση (label = 1), ενώ την κλάση των genuine ως την αρνητική (label = 0). Η σύμβαση αυτή ταιριάζει με αυτήν που χρησιμοποιείται από την επιστημονική κοινότητα, εφόσον είθισται η κλάση η οποία είναι προς ανίχνευση (εδώ τα bots) να θεωρείται η θετική κλάση. Έτσι, το dataframe των genuine users μετά από την επεξεργασία περιείχε 943 χρήστες, ο καθένας από τους οποίους αντιπροσωπευόταν από 41 χαρακτηριστικά - στήλες.

Για το dataframe των **social spambots #1 users**, αφού πρώτα αφαιρέσαμε τους χρήστες που δεν είχαν διαθέσιμη περιγραφή, μετονομάσαμε την στήλη "test_set_1" σε "label", εφόσον η στήλη ήδη περιείχε τιμές 1 για όλα τα δείγματα, εφόσον όλα ανήκαν στο σύνολο "test_set_1". Η στήλη "test_set_2" εδώ δεν υπήρχε, και άρα το τελικό dataframe των bot users που προέκυψε περιείχε και αυτό 943 χρήστες, ο καθένας από τους οποίους αντιπροσωπευόταν από 41 χαρακτηριστικά - στήλες.

Έτσι, το συνολικό **τελικό dataset των χρηστών** προέκυψε από ένα concatenation των δύο παραπάνω επεξεργασμένων dataset, το οποίο περιείχε 1.886 χρήστες (943 πραγματικούς και 943 αυτοματοποιημένους), με 41 χαρακτηριστικά – στήλες για τον κάθε ένα από αυτούς.

Προχωρώντας τώρα στο dataset των **genuine tweets**, αρχικά αφαιρέσαμε τις στήλες "truncated", "retweeted_status_id", "geo", "contributors", "possibly_sensitive", "created_at", "crawled_at", "updated". Στην συνέχεια, κάνοντας χρήση του τελικού dataframe των genuine users, κρατήσαμε μόνο τα tweets τα οποία δημοσιεύθηκαν από τους πραγματικούς χρήστες που εν τέλει χρησιμοποιήσαμε, εφόσον δεν μας απασχολούν tweets τα οποία γράφτηκαν από χρήστες τους οποίους δεν θα κατηγοριοποιούσαμε. Ύστερα, αφαιρέθηκαν όλα τα tweets τα οποία δεν είχαν διαθέσιμο κείμενο, δηλαδή είχαν κενό στην στήλη "text". Τέλος, προστέθηκε μια νέα στήλη "label", η οποία αρχικοποιήθηκε με τιμές 0 για όλα τα δείγματα, για τους λόγους που αναφέρθηκαν προηγουμένως, με αποτέλεσμα να προκύψει ένα dataset που περιείχε 2.428.858 genuine tweets, με 18 στήλες – χαρακτηριστικά για το κάθε ένα από αυτά.

Για το dataset των **social spambots #1 tweets** εφαρμόστηκε η ίδια ακριβώς διαδικασία με πριν, εφόσον έγινε αφαίρεση των ίδιων στηλών, και παρέμειναν στο dataset μόνο τα tweets τα οποία δημοσιεύθηκαν από τους social spambots #1 users οι οποίοι τελικά χρησιμοποιήθηκαν, και τα οποία είχαν διαθέσιμο κείμενο, δηλαδή δεν είχαν κενό στην στήλη "text". Τέλος, προστέθηκε μια στήλη "label", η οποία αρχικοποιήθηκε με τιμές 1 για όλα τα δείγματα, και έτσι προέκυψε ένα dataset το οποίο περιείχε 1.501.592 αυτοματοποιημένα tweets, με 18 στήλες – χαρακτηριστικά για το κάθε ένα από αυτά.

Αντίστοιχα με τους χρήστες, το συνολικό **τελικό dataset των tweets** προέκυψε από ένα concatenation των δύο παραπάνω επεξεργασμένων dataset, το οποίο περιείχε 3.930.450 tweets (2.428.858 genuine και 1.501.592 αυτοματοποιημένα), με 18 στήλες – χαρακτηριστικά για το κάθε ένα από αυτά.

Κάνοντας χρήση των δύο τελικών αυτών dataset, δηλαδή των χρηστών και των tweets, τα οποία περιείχαν αντίστοιχα 1.886 χρήστες με 3.930.450 tweets, προχωρήσαμε στην διεξαγωγή των απαραίτητων βημάτων που απαιτούνταν, για την αποτελεσματική εκπαίδευση και αξιολόγηση των μοντέλων μας, τα οποία θα παρουσιάσουμε αναλυτικά στα επόμενα 2 κεφάλαια.

Κεφάλαιο 5

5. Μονοτροπική Ανίχνευση Bots με Χρήση Εικόνας

Στο Κεφάλαιο αυτό, ύστερα από την ενδελεχή παρουσίαση και ανάλυση του απαραίτητου Θεωρητικού Υπόβαθρου και των Τεχνικών, που χρησιμοποιήθηκαν για τους σκοπούς της παρούσας διπλωματικής εργασίας, καθώς και του Συνόλου Δεδομένων που επιστρατεύθηκε και της προεπεξεργασίας στην οποία υποβλήθηκε το ίδιο, για να είναι σε θέση να αξιοποιηθεί από την προτεινόμενη υλοποίηση, παρουσιάζονται οι μέθοδοι, τα μοντέλα και τα αποτελέσματα του πρώτου σκέλους της υλοποίησής μας, που αφορά την μονοτροπική ανίχνευση bots στο Twitter με χρήση μόνο εικόνας. Πιο συγκεκριμένα, στην ενότητα 5.1 γίνεται η περιγραφή του προτεινόμενου μοντέλου, ενώ παρουσιάζονται αναλυτικά οι τεχνικές και οι μέθοδοι που επιστρατεύθηκαν, με σκοπό την σχεδίαση και ανάπτυξη της προτεινόμενης υλοποίησης, η οποία βασίζεται αποκλειστικά σε εισόδους που έχουν την μορφή εικόνας, η οποία προκύπτει από την δραστηριότητα του λογαριασμού του χρήστη. Ύστερα, στην ενότητα 5.2 γίνεται αναφορά στην πειραματική διάταξη, με βάση την οποία πραγματοποιήθηκαν τα πειράματα, ενώ στην τελευταία ενότητα του Κεφαλαίου, στην ενότητα 5.3, παρουσιάζονται τα αποτελέσματα που προέκυψαν από τα πειράματα, καθώς και μια συγκριτική μελέτη των μετρικών αξιολόγησης, βάσει των οποίων ποσοτικοποιήθηκε η επίδοση των υλοποιημένων μοντέλων.

5.1 Περιγραφή Προτεινόμενου Μοντέλου

Για την ανάπτυξη του μοντέλου που βασίζεται αποκλειστικά στην εικόνα, υιοθετήσαμε την μεθοδολογία που παρουσιάστηκε στα [101] και [102] για την δημιουργία μιας αλληλουχίας DNA, του λεγόμενου Digital DNA (Ψηφιακό DNA). Η βιολογική αλληλουχία DNA, η οποία περιέχει την γενετική πληροφορία ενός έμβιου οργανισμού, αναπαρίσταται από μια αλληλουχία που χρησιμοποιεί 4 χαρακτήρες, οι οποίοι αντιπροσωπεύουν τις 4 νουκλεοτιδικές βάσεις: A (Αδενίνη), C (Κυτοσίνη), G (Γουανίνη) και T (Θυμίνη). Εμπνευσμένοι από την βιολογική αυτήν αλληλουχία, οι συγγραφείς των [101], [102] εισήγαγαν τον όρο του Digital DNA, το οποίο αποτελεί ένα ψηφιακό αντίγραφο του βιολογικού DNA, και το οποίο κωδικοποιεί την συμπεριφορά ενός online λογαριασμού. Πιο συγκεκριμένα, δημιούργησαν μια αλληλουχία DNA βασισμένη είτε στον τύπο των tweets του χρήστη, είτε στο περιεχόμενό τους.

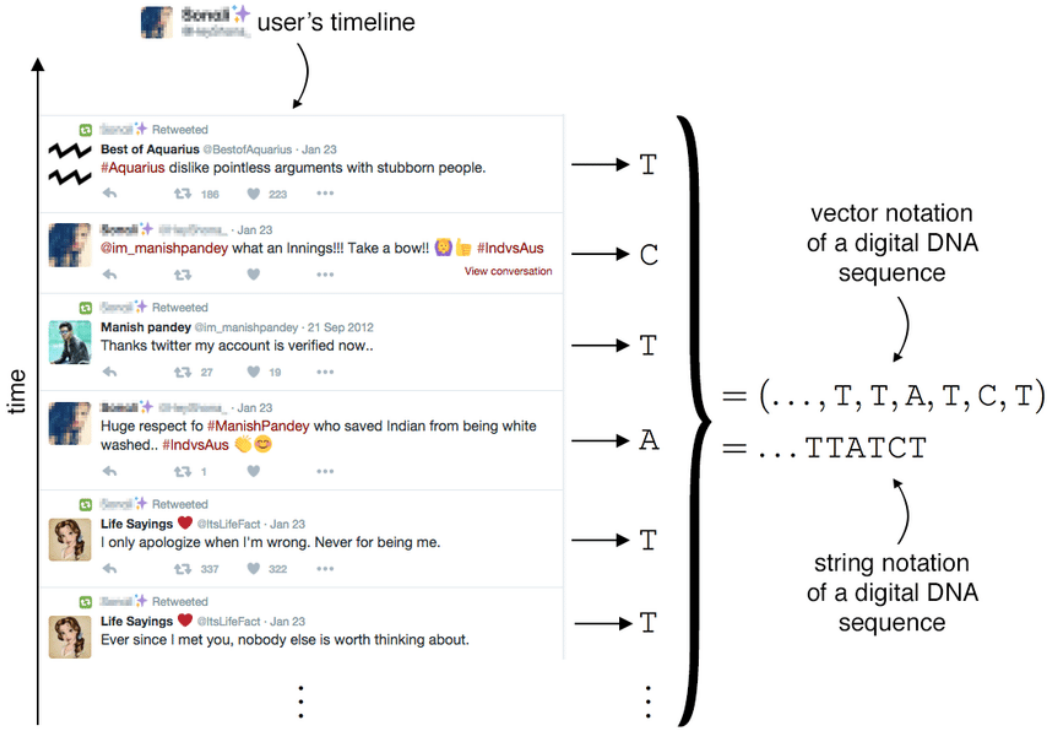
Στην παρούσα διπλωματική εργασία, δημιουργούμε 2 ψηφιακές αλληλουχίες DNA, τόσο βασιζόμενοι στον τύπο των tweets (type of tweets), όσο και στο περιεχόμενό τους (content of tweets). Για να το πραγματοποιήσουμε αυτό, αρχικά ταξινομήσαμε τα tweets κάθε χρήστη κατά χρονολογική σειρά, ξεκινώντας από την πιο παλιά δημοσίευση και καταλήγοντας στην πιο πρόσφατη. Στην πρώτη περίπτωση, στην οποία βασιζόμαστε στον τύπο των tweets, η αλληλουχία DNA που δημιουργούμε αποτελείται από τα σύμβολα – βάσεις A, T, C, και

προκύπτει αντικαθιστώντας κάθε απλό tweet του χρήστη με το σύμβολο A, κάθε retweet με το σύμβολο T, ενώ κάθε reply με το σύμβολο C, όπως φαίνεται και στον παρακάτω πίνακα:

Base	Type of Tweet
A	Tweet (Απλό)
C	Reply (Απάντηση)
T	Retweet (Αναδημοσίευση)

Πίνακας 4: Αντιστοίχιση Νουκλεοτιδικών Βάσεων με τον Τύπο των Tweets

Για να μπορέσουμε να διακρίνουμε τον τύπο ενός tweet σε απλό tweet, retweet ή reply, ελέγξαμε και χρησιμοποιήσαμε τις τιμές που βρίσκονταν στις στήλες “in_reply_to_status_id”, “in_reply_to_user_id” και “text”, που αποτελούν χαρακτηριστικά του tweet. Ένα παράδειγμα δημιουργίας μιας αλληλουχίας DNA με βάση τον τύπο του tweet φαίνεται στην παρακάτω εικόνα [103], όπου κάθε tweet, retweet και reply αντικαθίσταται αντίστοιχα από τους χαρακτήρες A, T και C:



Σχήμα 34: Παράδειγμα Δημιουργίας Αλληλουχίας Digital DNA από τον Τύπο των Tweets [103]

Στην δεύτερη περίπτωση, στην οποία βασιζόμαστε στο περιεχόμενο των tweets, η αλληλουχία DNA που δημιουργούμε αποτελείται από τα σύμβολα – βάσεις N, U, H, M και X. Πιο συγκεκριμένα, το σύμβολο N υποδεικνύει ότι το tweet δεν περιέχει καμία οντότητα, δηλαδή είναι ένα απλό κείμενο. Το σύμβολο U υποδεικνύει πως το tweet περιέχει μόνο την οντότητα URL, σε μια ή περισσότερες εμφανίσεις. Αντίστοιχα, το σύμβολο H υποδεικνύει πως το tweet περιέχει μόνο την οντότητα Hashtag (#), σε μια ή περισσότερες εμφανίσεις, ενώ το σύμβολο M υποδεικνύει πως το tweet περιέχει μόνο την οντότητα Mention (@), σε μια ή

περισσότερες εμφανίσεις. Τέλος, το X υποδεικνύει πως το tweet περιέχει οντότητες από δύο διαφορετικά είδη οντοτήτων και πάνω, όπως φαίνεται χαρακτηριστικά και στον παρακάτω πίνακα:

Base	Content of Tweet
N	Tweet contains no entities (plain text)
U	Tweet contains one or more URLs
H	Tweet contains one or more Hashtags
M	Tweet contains one or more Mentions
X	Tweet contains entities of mixed types

Πίνακας 5: Αντιστοίχιση Νουκλεοτιδικών Βάσεων με το Περιεχόμενο των Tweets

Αντίστοιχα με πριν, για να μπορέσουμε να διακρίνουμε το περιεχόμενο του tweet στις παραπάνω κατηγορίες, ελέγξαμε και χρησιμοποιήσαμε τις τιμές που βρίσκονταν στις στήλες "num_hashtags", "num_urls" και "num_mentions", που αποτελούν χαρακτηριστικά του tweet.

Αφότου δημιουργήσαμε την ψηφιακή αλληλουχία DNA για κάθε έναν από τους χρήστες, τόσο με βάση τον τύπο των tweets που δημοσίευσαν, όσο και με βάση το περιεχόμενό τους, υιοθετούμε την μεθοδολογία που παρουσιάστηκε στο [50] για την μετατροπή του digital DNA σε μια εικόνα που αποτελείται από 3 κανάλια. Σύμφωνα με τους συγγραφείς στο paper αυτό, δεδομένης της ομοιότητας που παρουσιάζεται στις ακολουθίες των ενεργειών των bots, σε σύγκριση με αυτές των πραγματικών χρηστών, ένας ταξινομητής εικόνας θα μπορούσε να επιφέρει αξιόλογα αποτελέσματα στον τομέα της ανίχνευσης αυτοματοποιημένων λογαριασμών.

Αρχικά, εφόσον τα Συνελικτικά Νευρωνικά Δίκτυα δέχονται εικόνες ίδιου μεγέθους, για κάθε μια από τις 2 αλληλουχίες DNA (με βάση τον τύπο και το περιεχόμενο), βρίσκουμε εκείνη που έχει το μεγαλύτερο μήκος αλληλουχίας, και ελέγχουμε αν το μήκος αυτό είναι τέλειο τετράγωνο, δηλαδή αν η ρίζα του αριθμού αυτού είναι ακέραιος αριθμός. Σε περίπτωση που είναι, τότε ορίζουμε το μέγεθος της εικόνας ως την τετραγωνική ρίζα του μήκους αυτού, ενώ σε περίπτωση που δεν είναι, λαμβάνουμε ως μέγιστο μήκος ακολουθίας το τέλειο τετράγωνο που είναι το πιο κοντινό και αυστηρά μεγαλύτερο από το πραγματικό μέγιστο μήκος ακολουθίας, και εφαρμόζουμε εκεί την τετραγωνική ρίζα, για να βρούμε το τελικό μέγεθος της εικόνας. Εφαρμόζοντας την διαδικασία αυτή, είναι δυνατό να μετατρέψουμε όλες τις ακολουθίες DNA σε εικόνες ίδιου μεγέθους. Στην συνέχεια, αντιστοιχούμε κάθε βάση – σύμβολο της αλληλουχίας του DNA, με ένα χρώμα, για την δημιουργία της εικόνας. Έτσι, χρωματίζουμε ένα προς ένα τα pixel της εικόνας, με βάση το χρώμα που έχουμε αναθέσει στο εκάστοτε σύμβολο. Ο χρωματισμός των pixel εφαρμόζεται έως ότου να ολοκληρωθεί η εκάστοτε αλληλουχία DNA. Αυτό, ωστόσο, σημαίνει πως σε περίπτωση που η ακολουθία δεν είναι αυτή με το μέγιστο μήκος, θα υπάρχει μέρος της εικόνας το οποίο δεν θα ξέρουμε πώς να χρωματίσουμε. Στις περιπτώσεις αυτές, τα υπολειπόμενα pixels της εικόνας που μένει να χρωματιστούν, μετά τον χρωματισμό ολόκληρης της αλληλουχίας του DNA, χρωματίζονται με το χρώμα μαύρο. Ο αλγόριθμος που περιγράφηκε φαίνεται σε μορφή ψευδοκώδικα στην παρακάτω εικόνα [50]:

Algorithm 1 From Digital DNA to image: Pseudocode

```
Input: List of DNA sequences
Output: DNA images
n ← Length of the longest DNA sequence
if n is a perfect square then
    L ← √n
else
    L ← get_closest_square_number(n)
end if
P ← dict with symbols and colors
for each DNA sequence do
    I ← create_image(width=L, height=L)
    for row in range(L) do
        for col in range(L) do
            k ← (row * L) + col
            if k < n then
                I[row, col] ← P[DNA[k]]
            end if
        end for
    end for
end for
```

Σχήμα 35: Ψευδοκώδικας Μετατροπής Αλληλουχίας Digital DNA σε Εικόνα [50]

Η αντιστοίχιση καθενός εκ των συμβόλων – βάσεων, από τα οποία αποτελείται κάθε μια από τις 2 αλληλουχίες Digital DNA (ως προς τον τύπο και το περιεχόμενο των tweets), με το χρώμα για την δημιουργία της τελικής εικόνας, έγινε με αυθαίρετο τρόπο, σε μια προσπάθεια να είναι αισθητή η διαφορά του χρώματος των pixels που αντιστοιχούν στις εκάστοτε διαφορετικές κατηγορίες, με βάση τις οποίες κατηγοριοποιήσαμε τα tweets. Η αντιστοίχιση αυτή φαίνεται στους παρακάτω 2 πίνακες, με δεδομένο πως το χρώμα "0" αντιστοιχεί σε μαύρο χρώμα, ενώ το "255" σε άσπρο.

Base	Pixel Color
A	180
C	120
T	70

Πίνακας 6: Αντιστοίχιση Νουκλεοτιδικών Βάσεων του Τύπου των Tweets με το Χρώμα των Pixels της Εικόνας

Base	Pixel Color
N	210
U	168
H	126
M	84
X	42

Πίνακας 7: Αντιστοίχιση Νουκλεοτιδικών Βάσεων του Περιεχομένου των Tweets με το Χρώμα των Pixels της Εικόνας

Ο λόγος που τα χρώματα που απεικονίζονται παραπάνω είναι μονοδιάστατα (210, 168 κλπ) είναι επειδή όλες οι εικόνες που προκύπτουν τελικά, είναι grayscale εικόνες διαστάσεων (1, H, W), όπου το 1 στην αρχή υποδηλώνει το πλήθος των καναλιών, το οποίο εν προκειμένω είναι 1, εφόσον η εικόνα είναι grayscale, ενώ τα H και W υποδηλώνουν αντίστοιχα τα Height (Ύψος) και Width (Πλάτος) της εικόνας. Ακολουθώντας την μεθοδολογία που παρουσιάζεται στο [50], μετατρέπουμε τις εικόνες που δημιουργήσαμε σε εικόνες διαστάσεων (3, H, W), δηλαδή εικόνες 3 καναλιών, χρησιμοποιώντας τον Μετασχηματισμό Grayscale της PyTorch[1]. Αυτή η μετατροπή από 1 σε 3 κανάλια ήταν αναγκαία, διότι όλα τα προεκπαιδευμένα Συνελικτικά Νευρωνικά Δίκτυα (pretrained CNNs) που χρησιμοποιήσαμε απαιτούσαν εικόνες εισόδου οι οποίες θα είχαν προεπεξεργασθεί και κανονικοποιηθεί με τον ίδιο τρόπο. Η κανονικοποίηση αυτή απαιτούσε τόσο την μετατροπή των εικόνων σε RGB εικόνες 3 καναλιών, με τα H και W να έχουν τιμές τουλάχιστον ίσες με 224 (και για αυτό και αλλάξαμε το μέγεθος των εικόνων σε 256 × 256 pixels), όσο και την μετατροπή τους σε Τένσορα (Pytorch Tensor), δηλαδή την κλιμάκωση των εικόνων στο διάστημα [0, 1], και την μετέπειτα κανονικοποίησή τους χρησιμοποιώντας τις τιμές mean = [0.485, 0.456, 0.406] και std = [0.229, 0.224, 0.225].

Έχοντας μετατρέψει όλες τις αλληλουχίες Digital DNA (και των 2 κατηγοριών) όλων των χρηστών σε εικόνες, και έχοντας προεπεξεργασθεί και κανονικοποιήσει κατάλληλα όλες τις εικόνες με τον ίδιο τρόπο, είμαστε σε θέση να τις τροφοδοτήσουμε σε προεκπαιδευμένα Συνελικτικά Νευρωνικά Δίκτυα, τα οποία κάνουμε fine-tune στο πρόβλημά μας. Τα προεκπαιδευμένα Συνελικτικά Νευρωνικά Δίκτυα τα οποία επιστρατεύσαμε στα πλαίσια της παρούσας διπλωματικής εργασίας για την κατηγοριοποίηση των χρηστών σε πραγματικούς ή αυτοματοποιημένους, και τα οποία επέτυχαν πολύ καλές επιδόσεις, όπως θα δούμε και στην συνέχεια, είναι τα ακόλουθα: **GoogLeNet (Inception v1)** [104], **ResNet50** [105], **WideResNet-50-2** [106], **AlexNet** [107], **SqueezeNet1_0** [108], **DenseNet-201** [109], **MobileNetV2** [110], **ResNeXt-50 32×4d** [111], **VGG16** [112] και **EfficientNet-B4** [113]. Μάλιστα, για την επιλογή του μοντέλου EfficientNet-B4, πρώτα πειραματιστήκαμε και αξιολογήσαμε όλη την σειρά των μοντέλων, δηλαδή από το EfficientNet-B0 έως και το EfficientNet-B7, αλλά καταλήξαμε πως το EfficientNet-B4 ήταν το πιο αποδοτικό, επιστρατεύοντας αυτό ως το αντιπροσωπευτικό μοντέλο της σειράς των Efficient-Nets.

5.2 Πειραματική Διάταξη

Όλες οι πτυχές από τις οποίες αποτελείται η προτεινόμενη υλοποίηση, οι οποίες ξεκινούν με την φόρτωση και την προεπεξεργασία στην οποία υποβλήθηκε το σύνολο δεδομένων για να προκύψει η τελική μορφή του που χρησιμοποιήθηκε για την εκπόνηση της εργασίας, συνεχίζουν με την κατασκευή των αλληλουχιών Digital DNA και την μετατροπή τους εικόνα, και καταλήγουν στην διεξαγωγή των πειραμάτων μας, κατά τα οποία επιστρατεύουμε ένα σύνολο προεκπαιδευμένων Συνελικτικών Νευρωνικών Δικτύων και αξιολογούμε και συγκρίνουμε την επίδοσή τους, αναπτύσσονται στην δωρεάν έκδοση του περιβάλλοντος *Google Colab*, σε γλώσσα προγραμματισμού *Python*. Πιο συγκεκριμένα, όλα τα μοντέλα (pretrained CNNs) αναπτύσσονται με χρήση της βιβλιοθήκης *PyTorch* [114], ενώ όλα τα

[1] https://pytorch.org/vision/main/generated/torchvision.transforms.Grayscale.html

πειράματα, τα οποία περιλαμβάνουν την εκπαίδευση (fine-tuning) των pretrained CNNs και την τελική αξιολόγησή τους, διεξάγονται αποκλειστικά πάνω σε μια *Tesla T4 GPU*.

Για την εκπαίδευση και αξιολόγηση των μοντέλων χωρίζουμε το σύνολο δεδομένων, το οποίο προέκυψε μετά από την απαιτούμενη προεπεξεργασία, στα σύνολα train set, validation set και test set. Το train set είναι το σύνολο δεδομένων που χρησιμοποιείται για την εκπαίδευση του μοντέλου, το validation set είναι το σύνολο δεδομένων πάνω στο οποίο εξετάζουμε την απόδοση του μοντέλου μας στο τέλος κάθε εποχής, αφού έχει χρησιμοποιηθεί δηλαδή όλο το train set, ενώ το test set χρησιμοποιείται στο τέλος, μετά το πέρας της εκπαίδευσης του μοντέλου, για την αξιολόγηση των προβλέψεών του πάνω σε άγνωστα δεδομένα. Ο στόχος της επιβλεπόμενης μάθησης, εξάλλου, είναι η δημιουργία ενός μοντέλου που εμφανίζει καλές επιδόσεις σε προβλέψεις δεδομένων με τα οποία δεν είχε πρότερη «επαφή». Για τον λόγο αυτό χρησιμοποιείται το test set, για την προσομοίωση των δεδομένων αυτών. Ο διαχωρισμός του συνόλου δεδομένων, ο οποίος τελικά ακολουθήθηκε, είναι 80% - 10% - 10% αντίστοιχα, και γίνεται κάθε φορά μέσω τυχαίας επιλογής δειγμάτων από το σύνολο δεδομένων, χρησιμοποιώντας διαφορετικά random seeds, για κάθε ένα από τα πέντε «τρεξίματα» (runs) που ολοκληρώθηκαν.

Ως run θεωρούμε μια ολοκληρωμένη διαδικασία κατά την οποία ένα μοντέλο εκπαιδεύεται, και ύστερα αξιολογείται με βάση τις μετρικές αξιολόγησης που επιστρατεύθηκαν. Για λόγους πληρότητας, και προς αποφυγή ακραίων περιπτώσεων που ενδεχομένως να προέκυπταν από την τυχαία επιλογή των δειγμάτων για την δημιουργία των train, validation και test sets, αποφασίστηκε να ολοκληρώνονται 5 runs για κάθε ένα από τα μοντέλα, και η τελική αξιολόγηση των μοντέλων με βάση τις μετρικές να γινόταν χρησιμοποιώντας τον μέσο όρο (average) και την τυπική απόκλιση (standard deviation) των αντίστοιχων μετρικών, σε κάθε ένα από τα 5 runs. Ως μετρικές αξιολόγησης της επίδοσης των μοντέλων επιστρατεύθηκαν οι Accuracy, Precision, Recall, Specificity και F1-Score, οι οποίες υπολογίστηκαν θεωρώντας ως θετική κλάση (label = 1) την κλάση των bots, όπως αναφέρθηκε και στο προηγούμενο κεφάλαιο.

Για την διαδικασία της εκπαίδευσης των μοντέλων, επιλέχθηκε ένας μέγιστος αριθμός 30 εποχών. Οι εποχές αντιπροσωπεύουν τον αριθμό των επαναλήψεων που εκπαιδεύεται το μοντέλο πάνω στο train set. Η επιλογή του πλήθους των εποχών εκπαίδευσης αποτελεί μια σημαντική παράμετρο, εφόσον σε περίπτωση που ο αριθμός των εποχών είναι μεγάλος, ενδέχεται να παρατηρηθούν φαινόμενα υπερπροσαρμογής (overfitting) στο σύνολο εκπαίδευσης, έχοντας ως αποτέλεσμα το εκπαιδευόμενο μοντέλο να χάσει την γενικότητά του (generalization). Για τον λόγο αυτόν, χρησιμοποιούμε την μέθοδο *EarlyStopping*. Κατά την μέθοδο αυτή, γίνεται χρήση του validation set, το οποίο όπως αναφέραμε είναι το σύνολο δεδομένων πάνω στο οποίο εξετάζουμε την απόδοση του μοντέλου μας στο τέλος κάθε εποχής. Με αυτόν τον τρόπο, ελέγχουμε διαρκώς το validation loss, και σε περίπτωση που σταματήσει να μειώνεται για έναν προκαθορισμένο συνεχόμενο αριθμό εποχών, ο οποίος στην περίπτωσή μας ήταν 6 εποχές, σταματάει η διαδικασία της εκπαίδευσης. Ο αριθμός αυτός ήταν 6 εποχές, διότι θέσαμε την μεταβλητή *patience*, η οποία υποδεικνύει την «υπομονή» που έχει το μοντέλο μας σε εποχές στις οποίες δεν μειώνεται το validation loss, δηλαδή δεν γίνεται καλύτερο το ίδιο, σε 5.

Σκοπός μας ήταν η ελαχιστοποίηση της συνάρτησης κόστους, για την οποία επιστρατεύσαμε την συνάρτηση κόστους *Binary Cross-Entropy Loss*. Ως συνάρτηση

βελτιστοποίησης χρησιμοποιήσαμε τον βελτιστοποιητή *Adam* [72], με αρχική τιμή ρυθμού μάθησης (learning rate) την 0.00001, ενώ χρησιμοποιήθηκε και η τεχνική του δρομολογητή (scheduler) *ReduceLROnPlateau*, κατά την οποία ο ρυθμός μάθησης μειώνεται κατά την διάρκεια της εκπαίδευσης κατά τον παράγοντα 0.1, σε περίπτωση που το validation loss έχει σταματήσει να μειώνεται για 4 συνεχόμενες εποχές (patience = 3).

5.3 Αποτελέσματα και Σύγκριση Μοντέλων

Με χρήση των τεχνικών που αναφέρθηκαν και κάτω από τις συνθήκες που παρουσιάστηκαν αναλυτικά στην προηγούμενη ενότητα, έγινε η διεξαγωγή των πειραμάτων μας, μέσω των οποίων προέκυψαν αρκετά αξιόλογα αποτελέσματα, όπως θα δούμε αναλυτικά στην ενότητα αυτήν.

Τα αποτελέσματα που θα παρουσιαστούν για κάθε μοντέλο, δηλαδή οι μετρικές αξιολόγησης που χαρακτηρίζουν το κάθε ένα από αυτά, προέκυψαν ύστερα από την ολοκλήρωση 5 runs, λαμβάνοντας τον μέσο όρο και την τυπική απόκλιση των επιμέρους μετρικών αξιολόγησης, σε κάθε ένα από τα 5 runs, όπως αναφέρθηκε και στην προηγούμενη ενότητα.

5.3.1 Εικόνα Βασισμένη στον Τύπο των Tweets

Στον παρακάτω πίνακα, φαίνονται τα αποτελέσματα και οι επιδόσεις των προτεινόμενων μοντέλων, για την περίπτωση που η εικόνα προκύπτει από αλληλουχία DNA, η οποία κατασκευάστηκε με βάση τον τύπο των tweets (tweet, retweet και reply), ενώ τα καλύτερα αποτελέσματα για κάθε μετρική απεικονίζονται με **bold**.

Images Based on the Type of Tweets					
Architecture	**Evaluation Metrics**				
	Precision	**Recall**	**F1-score**	**Accuracy**	**Specificity**
GoogLeNet (Inception v1)	**100.00** ± 0.00	97.63 ± 1.39	98.79 ± 0.72	98.73 ± 0.79	**100.00** ± 0.00
ResNet50	**100.00** ± 0.00	97.83 ± 1.00	98.90 ± 0.52	98.94 ± 0.47	**100.00** ± 0.00
WideResNet-50-2	99.78 ± 0.44	99.35 ± 1.29	99.57 ± 0.87	99.58 ± 0.85	99.79 ± 0.42
AlexNet	99.59 ± 0.81	97.12 ± 1.23	98.34 ± 0.88	98.31 ± 0.91	99.55 ± 0.91
SqueezeNet1_0	99.17 ± 1.02	97.23 ± 1.41	98.17 ± 0.55	98.20 ± 0.54	99.14 ± 1.05
DenseNet-201	99.37 ± 0.51	98.59 ± 0.98	98.98 ± 0.55	98.94 ± 0.58	99.35 ± 0.53
MobileNetV2	99.58 ± 0.52	97.70 ± 1.28	98.63 ± 0.76	98.62 ± 0.72	99.57 ± 0.53
ResNeXt-50 32 × 4d	99.79 ± 0.43	97.59 ± 1.64	98.67 ± 1.01	98.62 ± 1.04	99.78 ± 0.44
VGG16	99.78 ± 0.44	**99.55** ± 0.54	**99.67** ± 0.44	**99.68** ± 0.42	99.80 ± 0.41
EfficientNet-B4	99.20 ± 1.14	96.31 ± 0.60	97.73 ± 0.47	97.67 ± 0.54	99.08 ± 1.35

Πίνακας 8: Αποτελέσματα Unimodal Μοντέλων με Χρήση Εικόνας που Βασίζεται στον Τύπο των Tweets

Όπως γίνεται αντιληπτό από τον παραπάνω πίνακα, όλα τα μοντέλα εμφανίζουν εξαιρετικές επιδόσεις απέναντι στο πρόβλημα κατηγοριοποίησης των χρηστών του Twitter σε πραγματικούς και αυτοματοποιημένους, χρησιμοποιώντας αποκλειστικά την εικόνα που προκύπτει από την δραστηριότητα του λογαριασμού τους, και συγκεκριμένα από τον τύπο των tweets που δημοσιεύουν.

Ωστόσο, το μοντέλο το οποίο ξεχωρίζει από τα υπόλοιπα είναι το μοντέλο **VGG16**, το οποίο αποτελεί το μοντέλο με τις καλύτερες επιδόσεις, υπερτερώντας των υπολοίπων pretrained μοντέλων στις μετρικές Recall, F1-score και Accuracy. Αξίζει να σημειωθεί εδώ, πως υπάρχουν μοντέλα τα οποία εμφανίζουν ίσες, ή ακόμα και καλύτερες επιδόσεις από το VGG16 στις μετρικές Precision και Specificity. Πιο συγκεκριμένα, το WideResNet-50-2 εμφανίζει ίδιο Precision (99.78%) με το VGG16, ενώ τα GoogLeNet (100%), ResNet50 (100%) και ResNeXt-50 32 × 4d (99.79%) εμφανίζουν υψηλότερα, ενώ παράλληλα τα GoogLeNet και ResNet50 ξεπερνούν το VGG16 και στην μετρική Specificity, με 100% έκαστο, έναντι του 99.80% που εμφανίζει το VGG16. Εντούτοις, το VGG16 υπερτερεί αυτών των μοντέλων σε F1-score, το οποίο αποτελεί τον σταθμισμένο μέσο των Precision και Recall, και το οποίο αποτελεί ταυτόχρονα και σημαντικότερη μετρική από το Specificity, εφόσον υψηλό Specificity συνδυασμένο με χαμηλό F1-score υποδεικνύει πως μερικά bots κατηγοριοποιούνται λανθασμένα σαν πραγματικοί λογαριασμοί. Το VGG16 υπερτερεί των υπόλοιπων μοντέλων στην μετρική Recall κατά 0.2 - 3.24%, στην μετρική F1-score κατά 0.1 - 1.94% και στην μετρική Accuracy κατά 0.1 - 2.01%.

Το δεύτερο καλύτερο μοντέλο, μετά το VGG16, αποτελεί το WideResNet-50-2, το οποίο εμφανίζει ίδιο Precision (99.78%) με το VGG16, συνυπάρχοντας με αυτό στην 3η υψηλότερη επίδοση όσον αφορά στην μετρική αυτή, πίσω από τα GoogLeNet και ResNet50 (100%) και ResNeXt-50 32 × 4d (99.79%). Εμφανίζει το 2ο μεγαλύτερο Recall (99.35%), το 2ο μεγαλύτερο F1-score (99.57%) και το 2ο μεγαλύτερο Accuracy (99.58%), πίσω από το VGG16, ενώ κατέχει και την 3η υψηλότερη επίδοση όσον αφορά στην μετρική Specificity (99.79%), πίσω από τα GoogLeNet και ResNet50 (100%) και VGG16 (99.80%). Πιο συγκεκριμένα, υπερτερεί των υπολοίπων μοντέλων, εκτός του VGG16, στην μετρική Recall κατά 0.76 - 3.04%, στην μετρική F1-score κατά 0.59 - 1.84% και στην μετρική Accuracy κατά 0.64 - 1.91%.

Όλα τα υπόλοιπα μοντέλα, εκτός του EfficientNet-B4, εμφανίζουν Recall που κυμαίνεται από 97.12% (AlexNet) έως 98.59% (DenseNet-201), F1-score που κυμαίνεται από 98.17% (SqueezeNet1_0) έως 98.98% (DenseNet-201) και Accuracy που κυμαίνεται από 98.20% (SqueezeNet1_0) έως 98.94% (DenseNet-201 και ResNet50). Το EfficientNet-B4 αποτελεί το μοντέλο με τις χειρότερες επιδόσεις, εμφανίζοντας τις μικρότερες επιδόσεις στις μετρικές αυτές, δηλαδή Recall 96.31%, F1-score 97.73% και Accuracy 97.67%, οι οποίες ωστόσο αποτελούν εξαιρετικές επιδόσεις από μόνες τους, χωρίς δηλαδή να τις συγκρίνουμε με τις υψηλότερες επιδόσεις των άλλων μοντέλων. Τέλος, όλα τα μοντέλα εμφανίζουν εξαιρετικά μεγάλες επιδόσεις στις μετρικές Precision και Specificity, της τάξεως του 99%, με τα μοντέλα GoogLeNet και ResNet50 να εμφανίζουν 100% και στις δύο.

Αναλογιζόμενοι, λοιπόν, τα παραπάνω, συμπεραίνουμε πως το VGG16 αποτελεί το μοντέλο με τις καλύτερες επιδόσεις απέναντι στο πρόβλημα κατηγοριοποίησης των χρηστών του Twitter σε πραγματικούς και αυτοματοποιημένους, χρησιμοποιώντας αποκλειστικά την εικόνα που βασίζεται στον τύπο των tweets που δημοσιεύουν. Για αυτόν τον λόγο, όπως θα δούμε και στο επόμενο κεφάλαιο, είναι και το μοντέλο που θα χρησιμοποιηθεί για την κατηγοριοποίηση της εικόνας, βασισμένης στον τύπο των tweets, σε πολυτροπικά (multimodal) μοντέλα που θα επιστρατευθούν για την αντιμετώπιση του ίδιου προβλήματος.

5.3.2 Εικόνα Βασισμένη στο Περιεχόμενο των Tweets

Στον παρακάτω πίνακα, φαίνονται τα αποτελέσματα και οι επιδόσεις των προτεινόμενων μοντέλων, για την περίπτωση που η εικόνα προκύπτει από αλληλουχία DNA, η οποία κατασκευάστηκε με βάση το περιεχόμενο των tweets, ενώ τα καλύτερα αποτελέσματα για κάθε μετρική απεικονίζονται με **bold**.

Images Based on the Content of Tweets					
Architecture	**Evaluation Metrics**				
	Precision	**Recall**	**F1-score**	**Accuracy**	**Specificity**
GoogLeNet (Inception v1)	**100.00** ± 0.00	98.09 ± 1.27	99.03 ± 0.65	99.05 ± 0.62	**100.00** ± 0.00
ResNet50	**100.00** ± 0.00	97.81 ± 1.13	98.89 ± 0.58	98.84 ± 0.62	**100.00** ± 0.00
WideResNet-50-2	98.93 ± 0.71	99.34 ± 0.89	99.14 ± 0.69	99.15 ± 0.63	98.95 ± 0.64
AlexNet	**100.00** ± 0.00	98.75 ± 1.20	99.37 ± 0.61	99.37 ± 0.62	**100.00** ± 0.00
SqueezeNet1_0	99.36 ± 0.53	97.64 ± 1.06	98.49 ± 0.64	98.52 ± 0.62	99.36 ± 0.52
DenseNet-201	99.59 ± 0.49	97.90 ± 1.37	98.73 ± 0.58	98.73 ± 0.54	99.56 ± 0.54
MobileNetV2	99.59 ± 0.50	98.98 ± 1.27	99.28 ± 0.74	99.26 ± 0.79	99.56 ± 0.54
ResNeXt-50 32 × 4d	99.35 ± 0.53	98.43 ± 1.31	98.88 ± 0.52	98.94 ± 0.47	99.38 ± 0.51
VGG16	**100.00** ± 0.00	**99.78** ± 0.43	**99.89** ± 0.22	**99.89** ± 0.21	**100.00** ± 0.00
EfficientNet-B4	96.13 ± 2.26	97.98 ± 2.01	97.02 ± 1.38	97.14 ± 1.28	96.36 ± 2.14

Πίνακας 9: Αποτελέσματα Unimodal Μοντέλων με Χρήση Εικόνας που Βασίζεται στο Περιεχόμενο των Tweets

Όπως γίνεται αντιληπτό από τον παραπάνω πίνακα, όλα τα μοντέλα εμφανίζουν αντίστοιχα εξαιρετικές επιδόσεις απέναντι και στο πρόβλημα κατηγοριοποίησης των χρηστών του Twitter σε πραγματικούς και αυτοματοποιημένους, χρησιμοποιώντας αποκλειστικά την εικόνα που προκύπτει από την δραστηριότητα του λογαριασμού τους, και συγκεκριμένα από το περιεχόμενο των tweets που δημοσιεύουν.

Ωστόσο, όπως συνέβη και στην προηγούμενη περίπτωση, το μοντέλο το οποίο ξεχωρίζει από τα υπόλοιπα είναι το μοντέλο **VGG16**, το οποίο αποτελεί το μοντέλο με τις καλύτερες επιδόσεις, υπερτερώντας των υπολοίπων pretrained μοντέλων στις μετρικές Recall, F1-score και Accuracy, και εμφανίζοντας το απόλυτο (100%) στις μετρικές Precision και Specificity. Πιο συγκεκριμένα, υπερτερεί των υπολοίπων μοντέλων στην μετρική Recall κατά 0.44 - 2.14%, στην μετρική F1-score κατά 0.52 - 2.87% και στην μετρική Accuracy κατά 0.52 - 2.75%, ενώ εμφανίζει ίσες επιδόσεις (100%) στις μετρικές Precision και Specificity με τα μοντέλα GoogLeNet, ResNet50 και AlexNet.

Το δεύτερο καλύτερο μοντέλο, μετά το VGG16, αποτελεί το AlexNet, το οποίο εμφανίζει μέγιστες επιδόσεις (100%) στα Precision και Specificity, και τις 2ες καλύτερες επιδόσεις στις μετρικές F1-score (99.37%) και Accuracy (99.37%). Πιο συγκεκριμένα, υπερτερεί των υπολοίπων μοντέλων, εκτός του VGG16, στην μετρική F1-score κατά 0.09 - 2.35% και στην μετρική Accuracy κατά 0.11 - 2.23%.

Όλα τα υπόλοιπα μοντέλα, εκτός του EfficientNet-B4, εμφανίζουν F1-score που κυμαίνεται από 98.49% (SqueezeNet1_0) έως 99.28% (MobileNetV2) και Accuracy που κυμαίνεται από 98.52% (SqueezeNet1_0) έως 99.26% (MobileNetV2). Επίσης, με εξαίρεση το EfficientNet-B4, όλα τα μοντέλα εμφανίζουν εξαιρετικά μεγάλες επιδόσεις στις μετρικές Precision και Specificity, της τάξεως του 99%, με τα μοντέλα GoogLeNet, ResNet50, AlexNet και VGG16 να εμφανίζουν 100% και στις δύο, όπως αναφέρθηκε και προηγουμένως. Ταυτόχρονα, εξαιρώντας το VGG16 που συνιστά το αποδοτικότερο μοντέλο, όλα τα μοντέλα εμφανίζουν Recall που κυμαίνεται από 97.64% (SqueezeNet1_0) έως 99.34% (WideResNet-50-2). Τέλος, όπως και στην προηγούμενη περίπτωση, το EfficientNet-B4 αποτελεί το μοντέλο με τις χειρότερες επιδόσεις, εμφανίζοντας τις χαμηλότερες επιδόσεις σε όλες τις μετρικές, εκτός της μετρικής Recall. Πιο συγκεκριμένα, εμφανίζει επίδοση 96.13% στην μετρική Precision, 97.02% στην μετρική F1-score, 97.14% στην μετρική Accuracy και 96.36% στην μετρική Specificity, οι οποίες ωστόσο αντίστοιχα αποτελούν εξαιρετικές επιδόσεις από μόνες τους, χωρίς δηλαδή να τις συγκρίνουμε με τις υψηλότερες επιδόσεις των άλλων μοντέλων.

Αναλογιζόμενοι τα παραπάνω, όπως και στην προηγούμενη περίπτωση, συμπεραίνουμε πως το VGG16 αποτελεί το μοντέλο με τις καλύτερες επιδόσεις απέναντι στο πρόβλημα κατηγοριοποίησης των χρηστών του Twitter σε πραγματικούς και αυτοματοποιημένους, χρησιμοποιώντας αποκλειστικά την εικόνα που βασίζεται στο περιεχόμενο των tweets που δημοσιεύουν. Για αυτόν τον λόγο, όπως θα δούμε και στο επόμενο κεφάλαιο, είναι και το μοντέλο που θα χρησιμοποιηθεί για την κατηγοριοποίηση της εικόνας, βασισμένης στο περιεχόμενο των tweets, σε πολυτροπικά (multimodal) μοντέλα που θα επιστρατευθούν για την αντιμετώπιση του ίδιου προβλήματος.

Κεφάλαιο 6

6. Πολυτροπική Ανίχνευση Bots με Χρήση Εικόνας και Κειμένου

Στο Κεφάλαιο αυτό, ύστερα από την ενδελεχή παρουσίαση και περιγραφή του προτεινόμενου μονοτροπικού (unimodal) μοντέλου που χρησιμοποιεί αποκλειστικά την εικόνα που προκύπτει από την δραστηριότητα του λογαριασμού του χρήστη για την αντιμετώπιση του προβλήματος ανίχνευσης των bots στο Twitter, που παρουσιάστηκε στο προηγούμενο κεφάλαιο, παρουσιάζονται οι μέθοδοι, οι τεχνικές, τα μοντέλα και τα αποτελέσματα του δεύτερου σκέλους της υλοποίησής μας, που αφορά την πολυτροπική (multimodal) ανίχνευση bots στο Twitter, με χρήση τόσο εικόνας (visual modality), όσο και κειμένου (textual modality). Πιο συγκεκριμένα, στην ενότητα 6.1 παρουσιάζονται τα προτεινόμενα πολυτροπικά μοντέλα που αναπτύχθηκαν, τα οποία χρησιμοποιούν από κοινού την εικόνα που προκύπτει από την δραστηριότητα του λογαριασμού του χρήστη (τόσο από τον τύπο των tweets όσο και από το περιεχόμενό τους), και την περιγραφή του λογαριασμού του χρήστη (account description), για την κατηγοριοποίηση των χρηστών σε πραγματικούς και αυτοματοποιημένους. Ύστερα, στην ενότητα 6.2 γίνεται αναφορά στην πειραματική διαδικασία, η οποία περιλαμβάνει τόσο την παρουσίαση των baselines που χρησιμοποιήθηκαν για την αξιολόγηση των προτεινόμενων υλοποιήσεων και την σύγκριση με άλλες state-of-the-art υλοποιήσεις της επιστημονικής κοινότητας, που εμφανίζουν εξίσου εξαιρετικές επιδόσεις, όσο και της πειραματικής διάταξης, με βάση την οποία πραγματοποιήθηκαν τα πειράματα. Τέλος, στην ενότητα 6.3, που αποτελεί την τελευταία ενότητα του Κεφαλαίου, παρουσιάζονται τόσο τα αποτελέσματα που προέκυψαν από τα πειράματα και τις υλοποιήσεις μας, όσο και οι επιδόσεις των state-of-the-art μοντέλων, τα οποία αξιοποιούμε για να διεξάγουμε μια συγκριτική μελέτη των μετρικών αξιολόγησης και των επιδόσεων, τόσο των δικών μας υλοποιήσεων μεταξύ τους, όσο και με τις state-of-the-art προσεγγίσεις.

6.1 Περιγραφή Προτεινόμενων Πολυτροπικών Μοντέλων – Fusion Methods

Για την ανάπτυξη των πολυτροπικών μοντέλων που αξιοποιούν τόσο την εικόνα (visual modality), όσο και το κείμενο (textual modality), επιστρατεύσαμε ένα σύνολο από Fusion Methods (Μεθόδους Συγχώνευσης), μέσω των οποίων είχαμε την δυνατότητα να συγχωνεύσουμε με διαφορετικούς τρόπους την εικόνα που χαρακτηρίζει την δραστηριότητα του λογαριασμού κάθε χρήστη με την περιγραφή του λογαριασμού του, προς την κατηγοριοποίηση των χρηστών σε πραγματικούς και αυτοματοποιημένους. Πιο συγκεκριμένα, τα Fusion Methods που αξιοποιήσαμε είναι τα Concatenation, Gated Multimodal Unit (GMU) και Crossmodal Attention, τα οποία παρουσιάζονται αναλυτικά στις επόμενες υποενότητες.

Όσον αφορά στην προσθήκη του textual modality, που αποτελεί την ειδοποιό διαφορά σε σχέση με τα unimodal μοντέλα που παρουσιάσαμε στο προηγούμενο κεφάλαιο, τα οποία αξιοποιούσαν μόνο το visual modality, για την κατηγοριοποίηση της περιγραφής του χρήστη αξιοποιήσαμε ένα προεκπαιδευμένο μοντέλο Transformer, το TwHIN-BERT [90]. Πιο συγκεκριμένα, εφόσον το TwHIN-BERT έχει εκπαιδευθεί πάνω σε 7 δισεκατομμύρια tweets, γραμμένα σε περισσότερες από 100 διαφορετικές γλώσσες, όπως έχει αναφερθεί και στην ενότητα παρουσίασης του μοντέλου στο Κεφάλαιο 3, έχει την δυνατότητα να παρέχει πλούσιες αναπαραστάσεις για την μοντελοποίηση μικρών «θορυβωδών» κειμένων που έχουν γραφθεί από πραγματικούς χρήστες του Twitter. Για αυτόν τον λόγο, το κείμενο της περιγραφής δεν υπέστη κάποιου είδους επεξεργασία, εφόσον οποιοσδήποτε «θόρυβος» που μπορεί να υπήρχε στο κείμενο, είτε από URLs, είτε από Hashtags (#), είτε από Mentions (@), είτε από Emojis, είτε από οτιδήποτε που μπορεί να συμπεριληφθεί από έναν χρήστη του Twitter στην περιγραφή του λογαριασμού του, θα μπορούσε να περιέχει πληροφορία καθοριστική για την κατηγοριοποίηση του χρήστη σε πραγματικό και αυτοματοποιημένο. Έτσι, αφού πρώτα το κωδικοποιήσαμε κατάλληλα ώστε να αποκτήσει την μορφή που απαιτεί στην είσοδό του το μοντέλο, το τροφοδοτήσαμε σε αυτό για την εξαγωγή των απαραίτητων κάθε φορά χαρακτηριστικών από τα Fusion Methods, όπως θα δούμε αναλυτικά στην συνέχεια. Για την κωδικοποίηση του κειμένου, χρησιμοποιήσαμε την συνάρτηση *encode_plus*[2], μέσω της οποίας κάναμε tokenize το κείμενο (δηλαδή το μετατρέψαμε σε tokens), προσθέσαμε στο κείμενο τα ειδικά tokens [CLS] και [SEP], και ορίσαμε μέγιστο μήκος ακολουθίας τα 128 tokens, έτσι ώστε όλες οι ακολουθίες εισόδου να έχουν το ίδιο μήκος, εφόσον όπως έχουμε αναφέρει το μοντέλο απαιτεί οι ακολουθίες εισόδου να είναι συγκεκριμένου (προκαθορισμένου) μήκους, και για αυτό συμπληρώνονται (padding) ή «κόβονται» (truncate) αναλόγως. Τελικά, από την κωδικοποίηση του κειμένου προκύπτει μια λίστα με τα *input_ids*, που αποτελεί μια λίστα με τα token IDs που αντιπροσωπεύουν το κείμενο εισόδου, και μια λίστα *attention_mask*, που υποδεικνύει ποια tokens υπήρχαν πράγματι στην ακολουθία και ποια όχι (διότι ενδεχομένως να προέκυψαν από padding), έτσι ώστε να αγνοηθεί η πληροφορία στην δεύτερη περίπτωση. Αυτές οι δύο λίστες είναι και αυτές που αποτελούν τελικά την είσοδο του μοντέλου μας.

6.1.1 Concatenation

Στα πλαίσια της υλοποίησης αυτής, όσον αφορά στο textual modality, δηλαδή την περιγραφή του λογαριασμού του χρήστη, τροφοδοτούμε την περιγραφή (με τον τρόπο που παρουσιάστηκε παραπάνω και ύστερα από την κατάλληλη κωδικοποίησή της) στο μοντέλο TwHIN-BERT, από την έξοδο του οποίου εξάγουμε το [CLS] token. Το token αυτό, όπως έχουμε αναφέρει, είναι ιδιαίτερα σημαντικό, εφόσον η τελική κρυφή κατάσταση που αντιστοιχεί σε αυτό χρησιμοποιείται ως η συνολική ακολουθία στα προβλήματα ταξινόμησης. Θεωρώντας ως f^t την διανυσματική αναπαράσταση του textual modality, θα ισχύει πως $f^t \in \mathbb{R}^{d_t}$, όπου το d_t υποδεικνύει την διαστατικότητα, και είναι ίσο με 768.

Όσον αφορά στο visual modality, ακολουθούμε την διαδικασία που παρουσιάσαμε στην ενότητα 5.1 του προηγούμενου κεφαλαίου, για την κατασκευή δυο τρισδιάστατων εικόνων, μια που προκύπτει από την αλληλουχία Digital DNA του χρήστη, η οποία προκύπτει από τον

[2] https://huggingface.co/docs/transformers/v4.33.2/en/internal/tokenization_utils#transformers.PreTrainedTokenizerBase.encode_plus

τύπο των tweets που ο ίδιος δημοσιεύει, και μια που προκύπτει από την αλληλουχία Digital DNA του χρήστη, η οποία προκύπτει από το περιεχόμενο των tweets του. Για την τελική κατηγοριοποίηση της εικόνας, επιστρατεύσαμε και στις δύο περιπτώσεις (εικόνα βασισμένη σε τύπο και περιεχόμενο των tweets) το μοντέλο VGG16, εφόσον όπως παρουσιάστηκε αναλυτικά στο προηγούμενο κεφάλαιο αποτελεί το μοντέλο με τις καλύτερες επιδόσεις προς την αντιμετώπιση του συγκεκριμένου προβλήματος, υπερτερώντας των υπόλοιπων pretrained μοντέλων. Στην συγκεκριμένη περίπτωση, ωστόσο, για τις ανάγκες αυτής της μεθόδου συγχώνευσης, αφαιρέσαμε το τελευταίο layer του μοντέλου, και αντικαταστήσαμε το προτελευταίο dense layer, το οποίο αποτελείται από 4.096 units, με ένα dense layer που αποτελείται από 768 units. Η διαδικασία αυτή ακολουθήθηκε έτσι ώστε το visual modality να αποκτήσει την ίδια διαστατικότητα (768) με το textual modality. Θεωρώντας ως f^v την διανυσματική αναπαράσταση του visual modality, θα ισχύει πως $f^v \in \mathbb{R}^{d_v}$, όπου το d_v υποδεικνύει την διαστατικότητα, και είναι ίσο με 768.

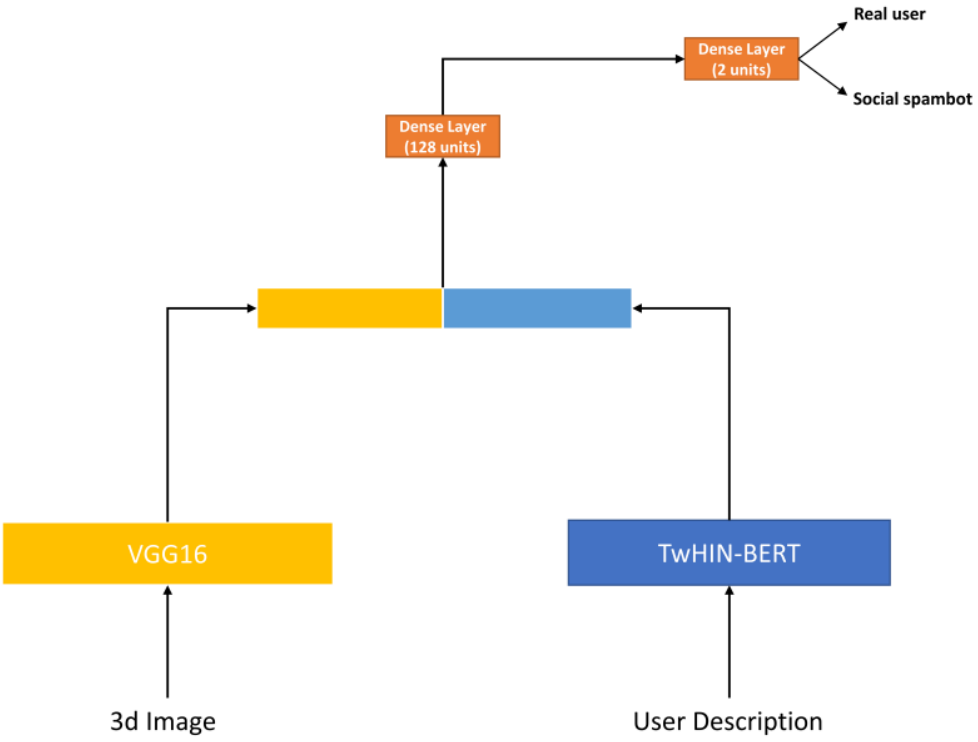
Αφότου υπολογίσαμε τις διανυσματικές αναπαραστάσεις f^t και f^v του textual και του visual modality, αντιστοίχως, εφαρμόζουμε concatenation (συνένωση) των δύο διανυσματικών αναπαραστάσεων σε μια διανυσματική αναπαράσταση, έστω z, όπως φαίνεται από τον ακόλουθο τύπο:

$$z = [f^t; f^v]$$

, όπου το [;] υποδεικνύει την λειτουργία της μεθόδου του concatenation και $z \in \mathbb{R}^d$, όπου $d = d_t + d_v$ είναι η διαστατικότητα της διανυσματικής αναπαράστασης z, και είναι ίσο με το άθροισμα των διαστατικοτήτων των επιμέρους διανυσματικών αναπαραστάσεων, δηλαδή έχει τιμή 1.536.

Ύστερα από τον υπολογισμό της διανυσματικής αναπαράστασης z, εφαρμόζουμε την τεχνική dropout [115] σε αυτήν, με συντελεστή 0,1. Ο λόγος που επιστρατεύουμε την τεχνική dropout είναι για την διασφάλιση της γενικότητας (generalization) των μοντέλων και την αποφυγή φαινομένων υπερπροσαρμογής (overfitting). Αποτελεί μια μέθοδο εξομάλυνσης (regularization), με βάση την οποία σε κάθε βήμα της εκπαίδευσης ένα ποσοστό των νευρώνων απενεργοποιείται τυχαία από το δίκτυο.

Μετά την εφαρμογή της τεχνικής dropout, περνάμε το z μέσα από ένα dense layer που αποτελείται από 128 units, με συνάρτηση ενεργοποίησης την ReLU, ενώ για την εξαγωγή της τελικής πρόβλεψης περνάμε το αποτέλεσμα που προέκυψε από άλλο ένα dense layer, το οποίο αποτελείται από 2 units, τα οποία αντιπροσωπεύουν τις κλάσεις του προβλήματός μας, δηλαδή τις κλάσεις human και bot. Η αρχιτεκτονική της υλοποίησης που παρουσιάστηκε αναλυτικά παραπάνω φαίνεται στο ακόλουθο σχήμα:



Σχήμα 36: Αρχιτεκτονική Multimodal Μοντέλου με Χρήση Concatenation

6.1.2 Gated Multimodal Unit (GMU)

Στα πλαίσια της υλοποίησης αυτής, αντίστοιχα με την προηγούμενη υλοποίηση, όσον αφορά στο textual modality, δηλαδή την περιγραφή του λογαριασμού του χρήστη, τροφοδοτούμε την περιγραφή (με τον τρόπο που παρουσιάστηκε παραπάνω και ύστερα από την κατάλληλη κωδικοποίησή της) στο μοντέλο TwHIN-BERT, από την έξοδο του οποίου εξάγουμε το [CLS] token. Θεωρώντας ως f^t την διανυσματική αναπαράσταση του textual modality, θα ισχύει πως $f^t \in \mathbb{R}^{d_t}$, όπου το d_t υποδεικνύει την διαστατικότητα, και είναι ίσο με 768.

Όσον αφορά στο visual modality, αντίστοιχα με την προηγούμενη υλοποίηση, ακολουθούμε πάλι την διαδικασία που παρουσιάσαμε αναλυτικά στην ενότητα 5.1 του προηγούμενου κεφαλαίου, για την κατασκευή των δυο τρισδιάστατων εικόνων για κάθε χρήστη. Για την τελική κατηγοριοποίηση της εικόνας, επιστρατεύσαμε και στις δύο περιπτώσεις (εικόνα βασισμένη σε τύπο και περιεχόμενο των tweets) το μοντέλο VGG16, εφόσον όπως παρουσιάστηκε αναλυτικά στο προηγούμενο κεφάλαιο αποτελεί το μοντέλο με τις καλύτερες επιδόσεις προς την αντιμετώπιση του συγκεκριμένου προβλήματος, υπερτερώντας των υπόλοιπων pretrained μοντέλων. Αντίστοιχα με την προηγούμενη περίπτωση και για τις ανάγκες και αυτής της μεθόδου συγχώνευσης, αφαιρέσαμε το τελευταίο layer του μοντέλου, και αντικαταστήσαμε το προτελευταίο dense layer, το οποίο αποτελείται από 4.096 units, με ένα dense layer που αποτελείται από 768 units. Θεωρώντας ως f^v την

διανυσματική αναπαράσταση του visual modality, θα ισχύει πως $f^v \in \mathbb{R}^{d_v}$, όπου το d_v υποδεικνύει την διαστατικότητα, και είναι ίσο με 768.

Αφότου υπολογίσαμε τις διανυσματικές αναπαραστάσεις f^t και f^v του textual και του visual modality, αντιστοίχως, επιστρατεύσαμε την μέθοδο συγχώνευσης του Gated Multimodal Unit (GMU), που παρουσιάστηκε στο [116] (και χρησιμοποιείται και από τα [117], [118]), για τον έλεγχο της ροής της πληροφορίας από κάθε ένα από τα δύο modalities και του μεγέθους της συνεισφοράς τους στην τελική κατηγοριοποίηση του χρήστη. Οι εξισώσεις που εκφράζουν την λειτουργία του GMU, και τις οποίες εφαρμόσαμε και εμείς στην δική μας υλοποίηση, φαίνονται ακόλουθα:

$$h^t = \tanh\left(W^t f^t + b^t\right)$$

$$h^v = \tanh(W^v f^v + b^v)$$

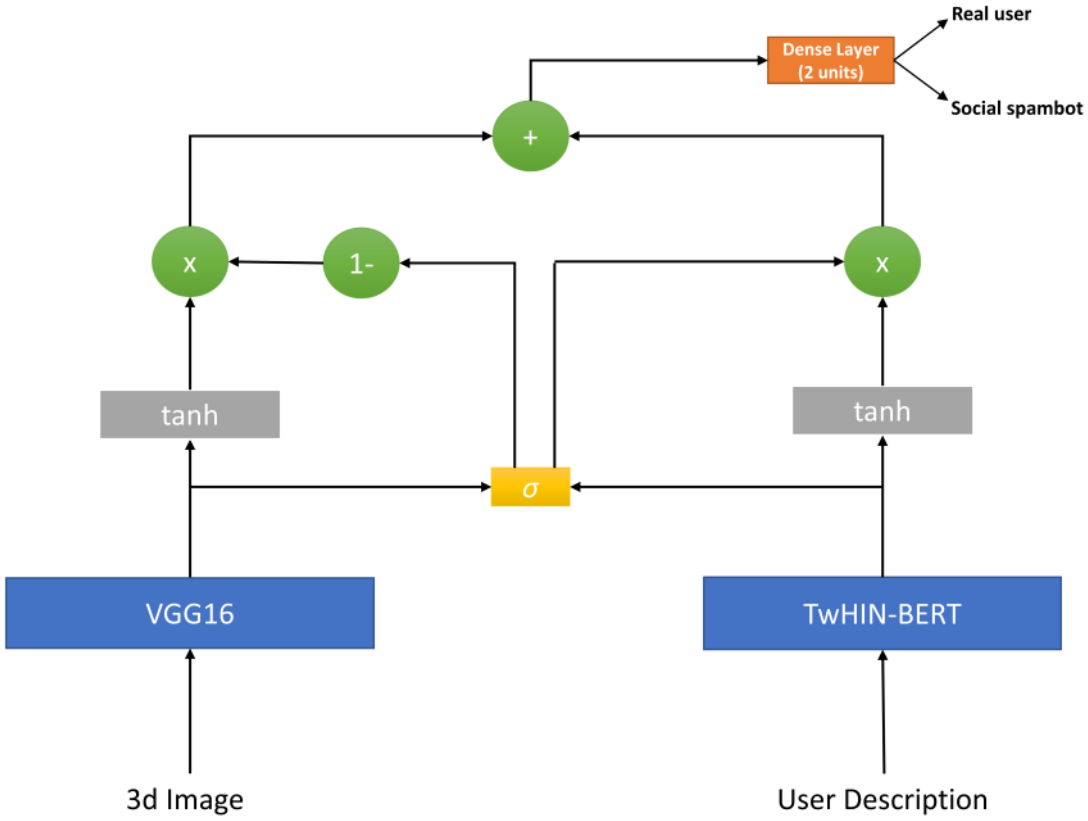
$$z = \sigma(W^z [f^t; f^v] + b^z)$$

$$h = z * h^t + (1 - z) * h^v$$

$$\Theta = \{W^t, W^v, W^z\}$$

, όπου το h^t αντιπροσωπεύει την ροή πληροφορίας του textual modality, το h^v αντιπροσωπεύει την ροή πληροφορίας του visual modality, το z αντιπροσωπεύει το ποσοστό της συνεισφοράς καθενός από τα modalities στο τελικό αποτέλεσμα, και το h αντιπροσωπεύει τον σταθμισμένο συνδυασμό της ροής πληροφοριών των δύο modalities. Το Θ υποδεικνύει τις εκπαιδεύσιμες παραμέτρους, ενώ το $[;]$ υποδεικνύει την λειτουργία της μεθόδου του concatenation, όπως έχουμε αναφέρει.

Πιο συγκεκριμένα, τροφοδοτούμε τόσο την διανυσματική αναπαράσταση f^t, όσο και την διανυσματική αναπαράσταση f^v, σε dense layer που αποτελείται από 128 units, και ύστερα περνάμε τα αποτελέσματα αυτά μέσα από την συνάρτηση ενεργοποίησης της υπερβολικής εφαπτομένης, για να προκύψουν αντίστοιχα τα h^t και h^v. Ύστερα, κάνουμε concatenate τα f^t και f^v, τροφοδοτούμε το αποτέλεσμα σε ένα dense layer που αποτελείται από 128 units, και ύστερα αυτό που προκύπτει το περνάμε μέσα από μια σιγμοειδή συνάρτηση ενεργοποίησης, για να προκύψει το z. Ύστερα, από τον σταθμισμένο συνδυασμό των h^t και h^v, μέσω του z, προκύπτει το h, στο οποίο εφαρμόζουμε την τεχνική dropout με συντελεστή 0,1. Τέλος, για την εξαγωγή της τελικής πρόβλεψης περνάμε το αποτέλεσμα που προέκυψε, δηλαδή το h, από ένα dense layer, το οποίο αποτελείται από 2 units, τα οποία αντιπροσωπεύουν τις κλάσεις του προβλήματός μας, δηλαδή τις κλάσεις human και bot. Η αρχιτεκτονική της υλοποίησης που παρουσιάστηκε αναλυτικά παραπάνω φαίνεται στο ακόλουθο σχήμα:



Σχήμα 37: Αρχιτεκτονική Multimodal Μοντέλου με Χρήση Gated Multimodal Unit

6.1.3 Crossmodal Attention

Στα πλαίσια της υλοποίησης αυτής, όσον αφορά στο textual modality, δηλαδή την περιγραφή του λογαριασμού του χρήστη, τροφοδοτούμε την περιγραφή (με τον τρόπο που παρουσιάστηκε παραπάνω και ύστερα από την κατάλληλη κωδικοποίησή της) στο μοντέλο TwHIN-BERT, και λαμβάνουμε από την έξοδο του μοντέλου την τελευταία κρυφή κατάσταση (last hidden state). Έτσι, θεωρώντας ως f^t την διανυσματική αναπαράσταση του textual modality, στην συγκεκριμένη περίπτωση θα ισχύει πως $f^t \in \mathbb{R}^{N \times d_t}$, όπου το N υποδεικνύει το μήκος της ακολουθίας (sequence length), ενώ το d_t υποδεικνύει την διαστατικότητα και είναι ίσο με 768.

Όσον αφορά στο visual modality, αντίστοιχα με τις προηγούμενες υλοποιήσεις, ακολουθούμε πάλι την διαδικασία που παρουσιάσαμε αναλυτικά στην ενότητα 5.1 του προηγούμενου κεφαλαίου, για την κατασκευή των δυο τρισδιάστατων εικόνων για κάθε χρήστη. Για την τελική κατηγοριοποίηση της εικόνας, επιστρατεύσαμε και στις δύο περιπτώσεις (εικόνα βασισμένη σε τύπο και περιεχόμενο των tweets) το μοντέλο VGG16, εφόσον όπως παρουσιάστηκε αναλυτικά στο προηγούμενο κεφάλαιο αποτελεί το μοντέλο με τις καλύτερες επιδόσεις προς την αντιμετώπιση του συγκεκριμένου προβλήματος, υπερτερώντας των υπόλοιπων pretrained μοντέλων. Σε αντίθεση, όμως, με τις προηγούμενες υλοποιήσεις, στην περίπτωση αυτή θεωρούμε ως έξοδο του μοντέλου, την έξοδο του

τελευταίου CNN layer του VGG16, που βρίσκεται μετά από το max pooling στην αρχιτεκτονική του μοντέλου. Έτσι, θεωρώντας ως f^v την διανυσματική αναπαράσταση του visual modality, θα ισχύει πως $f^v \in \mathbb{R}^{T \times d_v}$, όπου το d_v υποδεικνύει την διαστατικότητα και είναι ίσο με 512, ενώ το T είναι ίσο με 49 (7*7). Ύστερα, για να αποκτήσουν και τα δύο modalities την ίδια διαστατικότητα, εφόσον η διαστατικότητα του visual modality (512) είναι διαφορετική από αυτήν του textual (768), περνάμε το f^v μέσα από ένα dense layer που αποτελείται από 768 units.

Αφότου υπολογίσαμε τις διανυσματικές αναπαραστάσεις f^t και f^v του textual και του visual modality, αντιστοίχως, συμβουλευτήκαμε τα [119], [120] και [117] για να επιστρατεύσουμε την μέθοδο συγχώνευσης του Crossmodal Attention, κατά την οποία κατασκευάσαμε δύο crossmodal attention layers, ένα που απευθύνεται στην συσχέτιση και αλληλεπίδραση των textual χαρακτηριστικών f^t με τα visual χαρακτηριστικά f^v, και ένα που απευθύνεται στην συσχέτιση και αλληλεπίδραση των visual χαρακτηριστικών f^v με τα textual χαρακτηριστικά f^t.

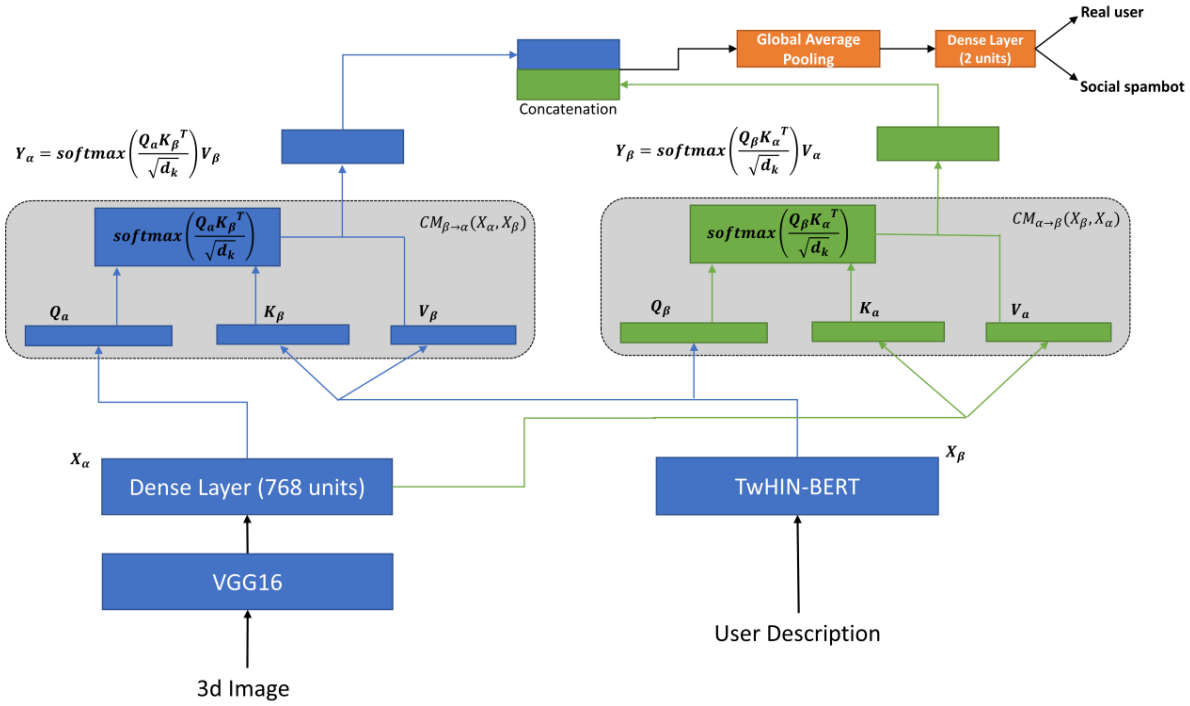
Ουσιαστικά, οι μηχανισμοί που αξιοποιεί το crossmodal attention είναι σχεδιασμένοι για να αντιλαμβάνονται τον τρόπο με τον οποίο πληροφορίες από ένα modality έχουν σχέση (ή ενδεχομένως να συμπληρώνουν ή να εξαρτώνται) με πληροφορίες ενός άλλου modality. Αυτή η μέθοδος συγχώνευσης επιτρέπει στο μοντέλο να αντιλαμβάνεται την σημασία διαφορετικών μερών ή πτυχών ενός modality, βασιζόμενο στο περιεχόμενο ή τα χαρακτηριστικά του άλλου.

Πιο συγκεκριμένα, για την πραγματοποίηση των παραπάνω, υπολογίζουμε τον μηχανισμό προσοχής Scaled-Dot-Product Attention [82], ο οποίος δίνεται από τον ακόλουθο τύπο, όπως έχουμε αναφέρει:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d}}\right)V$$

Στην πρώτη περίπτωση, που κατασκευάζουμε crossmodal attention layer από τα textual στα visual χαρακτηριστικά, το textual modality αντιστοιχεί στο Query (Q), και το visual modality αντιστοιχεί στα Key (K) και Value (V), ενώ αντίστοιχα στην δεύτερη περίπτωση, που κατασκευάζουμε crossmodal attention layer από τα visual στα textual χαρακτηριστικά, το visual modality αντιστοιχεί στο Query (Q), και το textual modality αντιστοιχεί στα Key (K) και Value (V). Για τον υπολογισμό των Q και K, τα εκάστοτε modalities περνούν μέσα από ένα dense layer, που αποτελείται από 200 units, ενώ για τον υπολογισμό των V, τα εκάστοτε modalities περνούν μέσα από ένα dense layer, που αποτελείται από 128 units. Αντικαθιστώντας τα εκάστοτε Q, K, V που προκύπτουν στον παραπάνω τύπο και εκτελώντας τους απαραίτητους υπολογισμούς, προκύπτουν τα αποτελέσματα για τις δύο περιπτώσεις, έστω z και y, αντίστοιχα.

Αφότου υπολογίσαμε τα z και y, εφαρμόζουμε concatenation σε αυτά για την δημιουργία μιας συνολικής fused αναπαράστασης, στην οποία ύστερα εφαρμόζουμε την τεχνική dropout, με συντελεστή 0,1. Το αποτέλεσμα που προκύπτει τροφοδοτείται σε ένα Global Average Pooling Layer, η έξοδος του οποίου περνάει μέσα από ένα dense layer, που αποτελείται από 2 units, που αντιπροσωπεύουν τις κλάσεις του προβλήματός μας, δηλαδή τις κλάσεις human και bot, για την εξαγωγή της τελικής πρόβλεψης. Η αρχιτεκτονική της υλοποίησης που παρουσιάστηκε αναλυτικά παραπάνω φαίνεται στο ακόλουθο σχήμα:



Σχήμα 38: Αρχιτεκτονική Multimodal Μοντέλου με Χρήση Crossmodal Attention

6.2 Πειραματική Διαδικασία

Στα πλαίσια της ενότητας αυτής, θα αναφερθούμε στην πειραματική διαδικασία. Πιο συγκεκριμένα, στην υποενότητα 6.2.1 θα αναφερθούμε στα baselines τα οποία ορίσαμε, για να είμαστε σε θέση να αξιολογήσουμε τις προτεινόμενες υλοποιήσεις μας και να συγκρίνουμε την επίδοσή τους τόσο μεταξύ τους, όσο και με άλλες state-of-the-art υλοποιήσεις που έχουν αναπτυχθεί από την επιστημονική κοινότητα για την αντιμετώπιση του προβλήματος ανίχνευσης των bots στο Twitter. Ύστερα, στην υποενότητα 6.2.2 θα παρουσιαστεί η πειραματική διάταξη, με βάση την οποία πραγματοποιήθηκαν τα πειράματα.

6.2.1 Baselines

Ως baselines, δηλαδή ως επίπεδα αναφοράς, τα οποία χρησιμοποιήσαμε για να είμαστε σε θέση να αξιολογήσουμε τις προτεινόμενες πολυτροπικές υλοποιήσεις μας, που παρουσιάσαμε αναλυτικά στην προηγούμενη ενότητα, χρησιμοποιήσαμε τόσο state-of-the-art υλοποιήσεις που είναι διαθέσιμες στην επιστημονική κοινότητα, όσο και τις δικές μας μονοτροπικές υλοποιήσεις, για να αντιληφθούμε την συνεισφορά της ύπαρξης διαφορετικών modalities και της χρήσης fusion methods, για την συγχώνευσή τους, στις τελικές επιδόσεις των μοντέλων μας.

Πιο συγκεκριμένα, επιστρατεύσαμε τα ακόλουθα baselines:

1. State-of-the-art υλοποιήσεις:

- DeepSBD [121]: Η έρευνα αυτή παρουσίασε ένα Βαθύ Νευρωνικό Δίκτυο που βασιζόταν σε μοντέλα CNN και BiLSTM, συνδυασμένα με έναν Μηχανισμό Προσοχής. Πιο συγκεκριμένα, οι συγγραφείς τροφοδότησαν το profile, καθώς και πληροφορίες σχετικές με τον χρόνο και την δραστηριότητα του λογαριασμού σε ένα BiLSTM δύο επιπέδων, ενώ τροφοδότησαν τις πληροφορίες που ήταν σχετικές με το περιεχόμενο σε ένα βαθύ Συνελικτικό Νευρωνικό Δίκτυο. Τέλος, «πάνω» από την προτεινόμενη υλοποίηση τοποθέτησαν και έναν Μηχανισμό Προσοχής, αναπτύσσοντας έτσι μια υλοποίηση που ξεπέρασε σε επιδόσεις τις άλλες state-of-the-art προσεγγίσεις.
- DNNBD [5]: Στην έρευνα αυτή οι συγγραφείς υιοθέτησαν μεθόδους για την αναγνώριση των bots, τόσο με βάση το account, όσο και με βάση τα tweets. Όσον αφορά στο account, αφότου πρώτα εξήγαγαν ένα σύνολο χαρακτηριστικών, χρησιμοποίησαν τεχνικές όπως SMOTE, SMOTENN και SMOTOMEK και εκπαίδευσαν ένα σύνολο από παραδοσιακούς αλγορίθμους Μηχανικής Μάθησης. Όσον αφορά στα tweets, ανέπτυξαν ένα Βαθύ Νευρωνικό Δίκτυο που περιείχε LSTM και dense layers, ενώ χρησιμοποίησαν σαν είσοδο GloVe embeddings των tweets και τα metadata τους. Για την υλοποίηση αυτή, θα παραθέσουμε τα αποτελέσματα που έχουν καταγραφεί στο [121].
- DBDM [122]: Στην έρευνα αυτή, προτείνεται πάλι ένα Βαθύ Νευρωνικό Δίκτυο που αποτελείται από μοντέλα CNN και LSTM. Πιο συγκεκριμένα, οι συγγραφείς μοντελοποιούν την κοινωνική συμπεριφορά κάθε χρήστη (δηλαδή τις δημοσιεύεις και αναδημοσιεύσεις του) και αξιοποιούν ένα δίκτυο LSTM. Ύστερα, αξιοποιώντας τα παλιά tweets του χρήστη (history tweets) αναπτύσσουν ένα δίκτυο CNN-LSTM, ενώ εξάγουν την τελική τους πρόβλεψη συγχωνεύοντας τις διανυσματικές αναπαραστάσεις που προκύπτουν από τα δύο δίκτυα. Για την υλοποίηση αυτή, θα παραθέσουμε τα αποτελέσματα που έχουν καταγραφεί στο [121].
- DeBD [123]: Σε αυτήν την έρευνα, οι συγγραφείς τροφοδοτούν το περιεχόμενο των tweets και την μεταξύ τους σχέση σε ένα μοντέλο CNN. Επίσης, επιστρατεύεται ένα LSTM για την εξαγωγή των δυνητικών χρονικών χαρακτηριστικών (χαρακτηριστικών που έχουν σχέση με τον χρόνο) από τα metadata των tweets. Στο τέλος, οι συγγραφείς κάνουν concatenate τα χρονικά χαρακτηριστικά με τα χαρακτηριστικά του περιεχομένου για την ανίχνευση των bots. Για την υλοποίηση αυτή, θα παραθέσουμε τα αποτελέσματα που έχουν καταγραφεί στο [121].
- MulBot-Glob_Hier [100]: Στην έρευνα αυτή, οι συγγραφείς παρουσιάζουν το MulBot, που αποτελεί έναν unsupervised ανιχνευτή bot, βασισμένο σε πολυμεταβλητές χρονοσειρές (multivariate time series). Πιο συγκεκριμένα, η προτεινόμενη υλοποίηση αποτελείται από τα ακόλουθα βήματα: εξαγωγή πολυδιάστατων χρονικών χαρακτηριστικών από τα χρονολόγια (timelines) των χρηστών, μείωση διαστατικότητας με χρήση ενός autoencoder, εξαγωγή καθολικών στατιστικών χαρακτηριστικών (που αποτελεί προαιρετικό βήμα), concatenation των χαρακτηριστικών αυτών με τα διανυσματικά χαρακτηριστικά που προέκυψαν από τον encoder (που αποτελεί προαιρετικό βήμα), και η εφαρμογή ενός clustering αλγορίθμου, του Agglomerative Hierarchical Clustering Algorithm. Τα αποτελέσματα που προέκυψαν από την υλοποίηση αυτή είναι ενθαρρυντικά, τόσο στο πρόβλημα της δυαδικής ταξινόμησης, όσο και σε αυτό της ταξινόμησης σε περισσότερες κλάσεις (multiclass classification).

- DNA - sequence (supervised) [102]: Στην έρευνα αυτή, οι συγγραφείς κατασκευάζουν το Digital DNA κάθε χρήστη, χρησιμοποιώντας είτε τον τύπο των tweets, είτε το περιεχόμενό τους. Ύστερα, υπολογίζουν τις ομοιότητες μεταξύ αυτών των ακολουθιών των χρηστών, χρησιμοποιώντας την ομοιότητα του Longest Common Subsequence (LCS), και συσταδοποιούν (ομαδοποιούν) τους χρήστες βασιζόμενοι στα ποσοστά ομοιότητας των ακολουθιών τους.
- Ahmed_DBSCAN [100], [124]: Η μέθοδος αυτή υιοθετεί την υλοποίηση που προτείνεται στο [124] και χρησιμοποιεί το DBSCAN για να βελτιώσει την απόδοση. Για την υλοποίηση αυτή, θα παραθέσουμε τα αποτελέσματα που έχουν καταγραφεί στο [100].
- F. Ahmed and M. Abulaish [124]: Η μέθοδος αυτή αξιοποιεί την Ευκλείδεια Απόσταση μεταξύ των διανυσμάτων των χαρακτηριστικών για να δημιουργήσει έναν γράφο ομοιότητας των accounts. Ύστερα, εφαρμόζονται αλγόριθμοι Graph Clustering και Community Detection για την ανίχνευση ομάδων από πανομοιότυπα accounts στον γράφο. Για την υλοποίηση αυτή, θα παραθέσουμε τα αποτελέσματα που έχουν καταγραφεί στο [102].

2. Το μονοτροπικό (unimodal) μοντέλο TwHIN-BERT, το οποίο χρησιμοποιεί αποκλειστικά το textual modality, δηλαδή την περιγραφή του λογαριασμού του χρήστη, για την κατηγοριοποίηση του χρήστη σε πραγματικό ή αυτοματοποιημένο. Η εκπαίδευση και η αξιολόγηση του μοντέλου, για την παραγωγή των τελικών αποτελεσμάτων, διεξάχθηκε στο ίδιο περιβάλλον και κάτω από ακριβώς τις ίδιες συνθήκες με αυτές που ορίστηκαν για να διεξαχθούν και τα πειράματα για τα unimodal μοντέλα που αξιοποιούσαν μόνο το visual modality, δηλαδή την εικόνα που προκύπτει από την δραστηριότητα του λογαριασμού του χρήστη, και τα οποία παρουσιάσαμε αναλυτικά στο προηγούμενο κεφάλαιο.
3. Τα μονοτροπικά (unimodal) μοντέλα VGG16, που χρησιμοποιούν αποκλειστικά το visual modality, τα οποία αποτελούν τα μοντέλα με τις καλύτερες επιδόσεις, τόσο για την περίπτωση που η εικόνα προκύπτει από τον τύπο των tweets του χρήστη, όσο και για την περίπτωση που προκύπτει από το περιεχόμενό τους.

6.2.2 Πειραματική Διάταξη

Όλες οι πτυχές από τις οποίες αποτελείται η προτεινόμενη υλοποίηση, οι οποίες ξεκινούν με την φόρτωση και την προεπεξεργασία στην οποία υποβλήθηκε το σύνολο δεδομένων για να προκύψει η τελική μορφή του που χρησιμοποιήθηκε για την εκπόνηση της εργασίας, συνεχίζουν με την κατασκευή των αλληλουχιών Digital DNA και την μετατροπή τους εικόνα, περιλαμβάνουν την σχεδίαση και ανάπτυξη των πολυτροπικών υλοποιήσεων με χρήση των fusion methods, και καταλήγουν στην διεξαγωγή των πειραμάτων μας, κατά τα οποία εκπαιδεύουμε τα προτεινόμενα πολυτροπικά μοντέλα και αξιολογούμε και συγκρίνουμε την επίδοσή τους, αναπτύσσονται ακριβώς κάτω από τις ίδιες συνθήκες και χρησιμοποιώντας τις ίδιες τεχνικές που παρουσιάστηκαν αναλυτικά στην ενότητα της Πειραματικής Διάταξης του προηγούμενου κεφαλαίου. Για λόγους πληρότητας, θα τις παρουσιάσουμε συνοπτικά παρακάτω.

Πιο συγκεκριμένα, η προτεινόμενη υλοποίηση αναπτύσσεται στην δωρεάν έκδοση του περιβάλλοντος *Google Colab*, σε γλώσσα προγραμματισμού *Python*. Για την ανάπτυξη των μοντέλων επιστρατεύεται η βιβλιοθήκη *PyTorch* [114], ενώ όλα τα πειράματα, τα οποία περιλαμβάνουν την εκπαίδευση των μοντέλων και την τελική αξιολόγησή τους, διεξάγονται αποκλειστικά πάνω σε μια *Tesla T4 GPU*.

Για την εκπαίδευση και αξιολόγηση των μοντέλων χωρίζουμε το σύνολο δεδομένων, το οποίο προέκυψε ύστερα από την απαιτούμενη προεπεξεργασία, στα σύνολα train set, validation set και test set. Ο διαχωρισμός του συνόλου δεδομένων, ο οποίος τελικά ακολουθήθηκε, είναι 80% - 10% - 10% αντίστοιχα, και γίνεται κάθε φορά μέσω τυχαίας επιλογής δειγμάτων από το σύνολο δεδομένων, χρησιμοποιώντας διαφορετικά random seeds, για κάθε ένα από τα πέντε «τρεξίματα» (runs) που ολοκληρώθηκαν. Για λόγους πληρότητας, και προς αποφυγή ακραίων περιπτώσεων που ενδεχομένως να προέκυπταν από την τυχαία επιλογή των δειγμάτων για την δημιουργία των train, validation και test sets, αποφασίστηκε να ολοκληρώνονται 5 runs για κάθε ένα από τα μοντέλα, και η τελική αξιολόγηση των μοντέλων με βάση τις μετρικές να γινόταν χρησιμοποιώντας τον μέσο όρο (average) και την τυπική απόκλιση (standard deviation) των αντίστοιχων μετρικών, σε κάθε ένα από τα 5 runs. Ως μετρικές αξιολόγησης της επίδοσης των μοντέλων επιστρατεύθηκαν οι Accuracy, Precision, Recall, Specificity και F1-Score, οι οποίες υπολογίστηκαν θεωρώντας ως θετική κλάση (label = 1) την κλάση των bots.

Για την διαδικασία της εκπαίδευσης των μοντέλων, επιλέχθηκε ένας μέγιστος αριθμός 30 εποχών, ενώ γίνεται χρήση της μεθόδου *EarlyStopping*. Με αυτόν τον τρόπο, ελέγχουμε διαρκώς το validation loss, και σε περίπτωση που σταματήσει να μειώνεται για έναν προκαθορισμένο συνεχόμενο αριθμό εποχών, ο οποίος στην περίπτωσή μας ήταν 6 εποχές, σταματάει η διαδικασία της εκπαίδευσης. Ο αριθμός αυτός ήταν 6 εποχές, διότι θέσαμε την μεταβλητή *patience*, η οποία υποδεικνύει την «υπομονή» που έχει το μοντέλο μας σε εποχές στις οποίες δεν μειώνεται το validation loss, δηλαδή δεν γίνεται καλύτερο το ίδιο, σε 5.

Σκοπός μας ήταν η ελαχιστοποίηση της συνάρτησης κόστους, για την οποία επιστρατεύσαμε την συνάρτηση κόστους *Binary Cross-Entropy Loss*. Ως συνάρτηση βελτιστοποίησης χρησιμοποιήσαμε τον βελτιστοποιητή *Adam* [72], με αρχική τιμή ρυθμού μάθησης (learning rate) την 0.00001, ενώ χρησιμοποιήθηκε και η τεχνική του δρομολογητή (scheduler) *ReduceLROnPlateau*, κατά την οποία ο ρυθμός μάθησης μειώνεται κατά την διάρκεια της εκπαίδευσης κατά τον παράγοντα 0.1, σε περίπτωση που το validation loss έχει σταματήσει να μειώνεται για 4 συνεχόμενες εποχές (patience = 3).

Τέλος, για το μοντέλο TwHIN-BERT, που αποτελεί το μοντέλο που αξιοποιήσαμε για την κατηγοριοποίηση του textual modality, δηλαδή της περιγραφής του λογαριασμού του χρήστη, χρησιμοποιήσαμε το *TwHIN-BERT-base version*[3] από την βιβλιοθήκη *transformers* [125] της Python.

[3] https://huggingface.co/Twitter/twhin-bert-base

6.3 Αποτελέσματα και Σύγκριση Μοντέλων

Με χρήση των τεχνικών που αναφέρθηκαν και κάτω από τις συνθήκες που παρουσιάστηκαν αναλυτικά στην προηγούμενη ενότητα, έγινε η διεξαγωγή των πειραμάτων μας, μέσω των οποίων προέκυψαν αρκετά αξιόλογα αποτελέσματα, όπως θα δούμε αναλυτικά στην ενότητα αυτήν.

Σύμφωνα με τα baselines που ορίσαμε, εκτός από τις επιδόσεις των πολυτροπικών μοντέλων που παρουσιάσαμε αναλυτικά στο κεφάλαιο αυτό, θα παρουσιαστούν, επίσης, τόσο τα αποτελέσματα από τις state-of-the-art προσεγγίσεις, τα οποία αντλήθηκαν κάθε φορά από τα papers που αναφέραμε κατά την περιγραφή των ίδιων, όσο και τα αποτελέσματα των unimodal μοντέλων, δηλαδή του TwHIN-BERT, που χρησιμοποιεί αποκλειστικά το textual modality, και των δύο VGG16 μοντέλων, που χρησιμοποιούν αποκλειστικά το visual modality, το οποίο κάθε φορά προκύπτει είτε από τον τύπο των tweets, είτε από το περιεχόμενό τους. Τα αποτελέσματα που θα παρουσιαστούν για κάθε μοντέλο το οποίο αναπτύχθηκε στα πλαίσια της διπλωματικής αυτής, προέκυψαν ύστερα από την ολοκλήρωση 5 runs, λαμβάνοντας τον μέσο όρο και την τυπική απόκλιση των επιμέρους μετρικών αξιολόγησης, σε κάθε ένα από τα 5 runs, όπως αναφέρθηκε και στην προηγούμενη ενότητα.

Στον παρακάτω πίνακα, λοιπόν, φαίνονται τα αποτελέσματα και οι επιδόσεις των μοντέλων και των υλοποιήσεων που αναφέρθηκαν, ενώ τα καλύτερα αποτελέσματα για κάθε μετρική απεικονίζονται με **bold**.

Architecture	Evaluation Metrics				
	Precision	**Recall**	**F1-score**	**Accuracy**	**Specificity**
State-of-the-art Approaches					
DeepSBD [121]	**100.00**	-	99.81	99.83	-
DNNBD [5]	77.66	-	75.63	78.20	-
DBDM [122]	**100.00**	-	98.82	99.32	-
DeBD [123]	97.73	-	97.59	97.74	-
MulBot-Glob_Hier [100]	99.50	99.50	99.00	99.30	-
DNA - sequence (supervised) [102]	98.20	97.70	97.70	97.70	98.10
Ahmed_DBSCAN [100], [124]	93.00	93.00	93.00	92.80	-
F. Ahmed and M. Abulaish [124]	94.50	94.40	94.40	94.30	94.50
Unimodal Approaches (only user description)					
TwHIN-BERT	99.59 ± 0.50	99.18 ± 0.75	99.38 ± 0.39	99.37 ± 0.40	99.56 ± 0.54
Unimodal Approaches (only images)					
VGG16 (type of tweets)	99.78 ± 0.44	99.55 ± 0.54	99.67 ± 0.44	99.68 ± 0.42	99.80 ± 0.41
VGG16 (content of tweets)	**100.00** ± 0.00	99.78 ± 0.43	99.89 ± 0.22	99.89 ± 0.21	**100.00** ± 0.00
Proposed Multimodal Approaches (images based on the type of tweets)					
TwHIN-BERT + VGG16 (Concatenation)	99.78 ± 0.44	99.34 ± 0.88	99.56 ± 0.54	99.58 ± 0.52	99.80 ± 0.41
TwHIN-BERT + VGG16 (GMU)	99.79 ± 0.42	99.79 ± 0.42	99.79 ± 0.42	99.79 ± 0.42	99.78 ± 0.43
TwHIN-BERT + VGG16 (Cross-Modal Attention)	**100.00** ± 0.00	99.79 ± 0.41	99.90 ± 0.21	99.89 ± 0.21	**100.00** ± 0.00
Proposed Multimodal Approaches (images based on the content of tweets)					
TwHIN-BERT + VGG16 (Concatenation)	99.77 ± 0.46	99.77 ± 0.46	99.77 ± 0.46	99.79 ± 0.42	99.80 ± 0.39
TwHIN-BERT + VGG16 (GMU)	**100.00** ± 0.00	99.58 ± 0.52	99.79 ± 0.26	99.79 ± 0.26	**100.00** ± 0.00
TwHIN-BERT + VGG16 (Cross-Modal Attention)	**100.00** ± 0.00	**99.96** ± 0.08	**99.98** ± 0.04	**99.98** ± 0.04	**100.00** ± 0.00

Πίνακας 10: Αποτελέσματα Multimodal Μοντέλων και Σύγκριση με Unimodal Μοντέλα και State-of-the-art Προσεγγίσεις

Όπως γίνεται αντιληπτό από τον παραπάνω πίνακα, όλα τα προτεινόμενα multimodal μοντέλα εμφανίζουν εξαιρετικές επιδόσεις απέναντι στο πρόβλημα κατηγοριοποίησης των χρηστών του Twitter σε πραγματικούς και αυτοματοποιημένους, χρησιμοποιώντας τόσο την εικόνα που προκύπτει από την δραστηριότητα του λογαριασμού τους, όσο και την περιγραφή του λογαριασμού τους. Αξίζει να παρατηρήσουμε πως, εκτός των unimodal προσεγγίσεων που βασίζονται αποκλειστικά στην εικόνα του χρήστη και τις οποίες έχουμε ήδη αναλύσει, ανταγωνιστικές επιδόσεις εμφανίζει και το unimodal μοντέλο που αξιοποιεί μόνο το textual modality, δηλαδή μόνο την περιγραφή του λογαριασμού του χρήστη, για να αποφανθεί εάν είναι human ή bot.

Πιο συγκεκριμένα, όσον αφορά στις προτεινόμενες multimodal προσεγγίσεις, στις οποίες η εικόνα βασίζεται στον τύπο των tweets, γίνεται εύκολα αντιληπτό πως το μοντέλο TwHIN-BERT + VGG16 (Cross-Modal Attention) συνιστά το μοντέλο με τις καλύτερες επιδόσεις, εμφανίζοντας επίδοση 99.89% στην μετρική Accuracy, 99.79% στην μετρική Recall, 99.90% στην μετρική F1-score, και το απόλυτο, δηλαδή 100%, στις μετρικές Precision και Specificity. Υπερτερεί των υπόλοιπων δύο μοντέλων (Concatenation και GMU) της συγκεκριμένης προσέγγισης σε όλες τις μετρικές, εκτός της μετρικής Recall, που εμφανίζει ίδια επίδοση (99.79%) με το TwHIN-BERT + VGG16 (GMU). Πιο συγκεκριμένα, υπερτερεί των άλλων δύο μοντέλων (Concatenation και GMU) στην μετρική Accuracy κατά 0.1 - 0.31%, στην μετρική Precision κατά 0.21 - 0.22%, στην μετρική F1-score κατά 0.11 - 0.34% και στην μετρική Specificity κατά 0.2 - 0.22%. Παράλληλα, υπερτερεί και των μοντέλων TwHIN-BERT και VGG16 (type of tweet), εφόσον υπερτερεί του TwHIN-BERT στην μετρική Precision κατά 0.41%, στην μετρική Recall κατά 0.61%, στην μετρική F1-score κατά 0.52%, στην μετρική Accuracy κατά 0.52% και στην μετρική Specificity κατά 0.44%, ενώ υπερτερεί του μοντέλου VGG16 (type of tweets) στην μετρική Precision κατά 0.22%, στην μετρική Recall κατά 0.24%, στην μετρική F1-score κατά 0.23%, στην μετρική Accuracy κατά 0.21% και στην μετρική Specificity κατά 0.2%. Επιπρόσθετα, αξίζει να παρατηρήσουμε πως το TwHIN-BERT + VGG16 (GMU) υπερτερεί σε όλες τις μετρικές του μοντέλου TwHIN-BERT + VGG16 (Concatenation), εκτός της μετρικής Specificity. Πιο συγκεκριμένα, υπερτερεί στην μετρική Precision κατά 0.01%, στην μετρική Recall κατά 0.45%, στην μετρική F1-score κατά 0.23% και στην μετρική Accuracy κατά 0.21%. Επίσης, ξεπερνάει στην μετρική Accuracy, μεταξύ άλλων, τόσο το TwHIN-BERT, όσο και το VGG16 (type of tweets), κατά 0.42% και 0.11%, αντίστοιχα. Όσον αφορά τώρα στο μοντέλο TwHIN-BERT + VGG16 (Concatenation), παρατηρούμε πως παρά το γεγονός ότι υπερτερεί του μοντέλου TwHIN-BERT σε όλες τις μετρικές, υστερεί σε όλες τις μετρικές του μοντέλου VGG16 (type of tweets), εκτός των μετρικών Precision και Specificity, που εμφανίζουν τα ίδια, δηλαδή 99.78% και 99.80%, αντίστοιχα. Πιο συγκεκριμένα, το VGG16 (type of tweets) υπερτερεί του TwHIN-BERT + VGG16 (Concatenation) στην μετρική Recall κατά 0.21%, στην μετρική F1-score κατά 0.11% και στην μετρική Accuracy κατά 0.1%. Θεωρούμε πως αυτή η μείωση στην επίδοση του μοντέλου οφείλεται στον τρόπο με τον οποίο εφαρμόζεται η μέθοδος συγχώνευσης του Concatenation, κατά την οποία κάθε modality συνεισφέρει κατά το ίδιο ποσοστό στο τελικό αποτέλεσμα, εφόσον ανατίθεται ίδια σημασία σε κάθε ένα από αυτά, αμελώντας έτσι τις έμφυτες συσχετίσεις και εξαρτήσεις που ενδεχομένως να εμφανίζουν μεταξύ τους. Συνολικά, θεωρούμε πως το μοντέλο TwHIN-BERT + VGG16 (Cross-Modal Attention) παρουσιάζει τις καλύτερες επιδόσεις στην συγκεκριμένη περίπτωση, που η εικόνα βασίζεται στον τύπο των tweets, τόσο αναλογικά με τα δύο άλλα multimodal μοντέλα που χρησιμοποιούν fusion methods, όσο και με τα unimodal μοντέλα που χρησιμοποιούν αποκλειστικά είτε το textual, είτε το visual modality. Αυτό μπορεί να αποδοθεί στο γεγονός πως αυτή η μέθοδος συγχώνευσης επιτρέπει στο μοντέλο να αντιλαμβάνεται την σημασία διαφορετικών μερών ή πτυχών ενός modality, βασιζόμενο στο περιεχόμενο ή τα χαρακτηριστικά του άλλου, όπως έχουμε αναφέρει και παραπάνω. Δεύτερο σε επιδόσεις έρχεται το TwHIN-BERT + VGG16 (GMU), το οποίο μέσω της μεθόδου συγχώνευσης του GMU έχει την δυνατότητα να ελέγχει την ροή πληροφοριών από κάθε modality και το μέγεθος της συνεισφοράς του καθενός στο τελικό αποτέλεσμα, ενώ συγκριτικά με τα multimodal μοντέλα, τελευταίο σε επιδόσεις έρχεται το μοντέλο που αξιοποιεί την μέθοδο συγχώνευσης του Concatenation, για τους λόγους που αναφέραμε.

Όσον αφορά, τώρα, στις προτεινόμενες multimodal προσεγγίσεις, στις οποίες η εικόνα βασίζεται στο περιεχόμενο των tweets, γίνεται εύκολα αντιληπτό πως το μοντέλο TwHIN-BERT + VGG16 (Cross-Modal Attention) συνιστά πάλι το μοντέλο με τις καλύτερες επιδόσεις, εμφανίζοντας επίδοση 99.98% στην μετρική Accuracy, 99.96% στην μετρική Recall, 99.98% στην μετρική F1-score, και το απόλυτο, δηλαδή 100%, στις μετρικές Precision και Specificity, υπερτερώντας, με αυτόν τον τρόπο, των άλλων δύο multimodal προσεγγίσεων, δηλαδή των μοντέλων TwHIN-BERT + VGG16 (GMU) και TwHIN-BERT + VGG16 (Concatenation). Πιο συγκεκριμένα, το TwHIN-BERT + VGG16 (Cross-Modal Attention) υπερτερεί του TwHIN-BERT + VGG16 (GMU) στην μετρική Recall κατά 0.38%, στην μετρική F1-score κατά 0.19%, και στην μετρική Accuracy κατά 0.19%. Παρά το γεγονός πως τα 2 μοντέλα εμφανίζουν ίσες επιδόσεις στις μετρικές Precision και Specificity (100%), το μοντέλο TwHIN-BERT + VGG16 (Cross-Modal Attention) εμφανίζει καλύτερο F1-score, το οποίο συνιστά τον σταθμισμένο μέσο των Precision και Recall. Ταυτόχρονα, το TwHIN-BERT + VGG16 (Cross-Modal Attention) υπερτερεί του TwHIN-BERT + VGG16 (Concatenation) στην μετρική Precision κατά 0.23%, στην μετρική Recall κατά 0.19%, στην μετρική F1-score κατά 0.21%, στην μετρική Accuracy κατά 0.19%, και στην μετρική Specificity κατά 0.2%. Συγκριτικά τώρα με τις unimodal προσεγγίσεις, παρατηρούμε πως υπερτερεί των μοντέλων TwHIN-BERT και VGG16 (content of tweets) σε όλες τις μετρικές, εκτός από τις μετρικές Precision και Specificity, στις οποίες εμφανίζει ίσες επιδόσεις, το απόλυτο 100%, με το VGG16 (content of tweets). Πιο συγκεκριμένα, υπερτερεί του μοντέλου TwHIN-BERT στην μετρική Precision κατά 0.41%, στην μετρική Recall κατά 0.78%, στην μετρική F1-score κατά 0.6%, στην μετρική Accuracy κατά 0.61%, και στην μετρική Specificity κατά 0.44%, ενώ υπερτερεί του μοντέλου VGG16 (content of tweets) στις μετρικές Recall κατά 0.18%, F1-score κατά 0.09% και Accuracy κατά 0.09%. Όσον αφορά στο μοντέλο TwHIN-BERT + VGG16 (GMU), παρατηρούμε πως υπερτερεί του μοντέλου TwHIN-BERT + VGG16 (Concatenation) στις μετρικές Precision κατά 0.23%, F1-score κατά 0.02% και Specificity κατά 0.2%, εμφανίζει ίδιο Accuracy με αυτό (99.79%), ενώ υστερεί στην μετρική Recall κατά 0.19%. Αναφορικά με το μοντέλο TwHIN-BERT + VGG16 (Concatenation), αντίστοιχα με την προηγούμενη περίπτωση, παρατηρούμε πως παρά το γεγονός ότι υπερτερεί του μοντέλου TwHIN-BERT σε όλες τις μετρικές, υστερεί σε όλες τις μετρικές του μοντέλου VGG16 (content of tweets), κάτι το οποίο αποδίδουμε στον τρόπο με τον οποίο εφαρμόζεται η μέθοδος συγχώνευσης του Concatenation, όπως αναφέραμε και προηγουμένως. Συνολικά, όπως συνέβη και στην προηγούμενη περίπτωση, θεωρούμε πως το μοντέλο TwHIN-BERT + VGG16 (Cross-Modal Attention) παρουσιάζει τις καλύτερες επιδόσεις στην συγκεκριμένη περίπτωση, που η εικόνα βασίζεται στο περιεχόμενο των tweets, τόσο αναλογικά με τα δύο άλλα multimodal μοντέλα που χρησιμοποιούν fusion methods, όσο και με τα unimodal μοντέλα που χρησιμοποιούν αποκλειστικά είτε το textual, είτε το visual modality.

Άρα, από τα παραπάνω, μπορούμε να συμπεράνουμε πως το TwHIN-BERT + VGG16 (Cross-Modal Attention) αποτελεί το μοντέλο με τις καλύτερες επιδόσεις σε κάθε μια από τις δύο περιπτώσεις τις οποίες εξετάσαμε, κατά τις οποίες το visual modality προκύπτει από τον τύπο και το περιεχόμενο των tweets, αντίστοιχα. Ωστόσο, συγκρίνοντας μεταξύ τους τα μοντέλα που αναπτύχθηκαν στα πλαίσια της κάθε υλοποίησης, καθώς και τις επιδόσεις που τελικά εμφάνισαν, μπορούμε να αποφανθούμε με βεβαιότητα πως το μοντέλο TwHIN-BERT + VGG16 (Cross-Modal Attention) με χρήση εικόνων που βασίζονται στο περιεχόμενο των tweets, υπερτερεί του μοντέλου TwHIN-BERT + VGG16 (Cross-Modal Attention) που χρησιμοποιεί εικόνες που βασίζονται στον τύπο των tweets, και συνεπώς και όλων των

μοντέλων που χρησιμοποιούν εικόνες βασισμένες στον τύπο των tweets, και τα οποία αναπτύξαμε στα πλαίσια της διπλωματικής αυτής. Πιο συγκεκριμένα, υπερτερεί του TwHIN-BERT + VGG16 (Cross-Modal Attention) που χρησιμοποιεί εικόνες που βασίζονται στον τύπο των tweets στις μετρικές Recall κατά 0.17%, F1-score κατά 0.08%, και Accuracy κατά 0.09%, ενώ και οι δύο υλοποιήσεις εμφανίζουν το απόλυτο (100%) στις μετρικές Precision και Specificity. Συνεπώς, **το μοντέλο TwHIN-BERT + VGG16 (Cross-Modal Attention) που χρησιμοποιεί εικόνες που βασίζονται στο περιεχόμενο των tweets συνιστά το καλύτερο μοντέλο μας, εμφανίζοντας τις υψηλότερες επιδόσεις**.

Τέλος, αναφορικά με τις state-of-the-art υλοποιήσεις και τις επιδόσεις που εμφανίζουν, παρατηρούμε πως το καλύτερο μοντέλο μας, δηλαδή το TwHIN-BERT + VGG16 (Cross-Modal Attention) που χρησιμοποιεί εικόνες που βασίζονται στο περιεχόμενο των tweets, υπερτερεί των υλοποιήσεων αυτών στην μετρική Precision κατά 0.5 - 22.34% (εκτός από τα DeepSBD και DBDM, με τα οποία εμφανίζει το απόλυτο 100%), στην μετρική Recall κατά 0.46 - 6.96%, στην μετρική F1-score κατά 0.17 - 24.35%, στην μετρική Accuracy κατά 0.15 - 21.78%, και στην μετρική Specificity κατά 1.9 - 5.5%. Αντίστοιχα, το δεύτερο καλύτερο μοντέλο μας, δηλαδή το TwHIN-BERT + VGG16 (Cross-Modal Attention) που χρησιμοποιεί εικόνες που βασίζονται στον τύπο των tweets, υπερτερεί στην μετρική Precision κατά 0.5 - 22.34% (εκτός από τα DeepSBD και DBDM, με τα οποία εμφανίζει το απόλυτο 100%), στην μετρική Recall κατά 0.29 - 6.79%, στην μετρική F1-score κατά 0.09 - 24.27%, στην μετρική Accuracy κατά 0.06 - 21.69%, και στην μετρική Specificity κατά 1.9 - 5.5%, ενώ και τα υπόλοιπα μοντέλα που αναπτύξαμε στα πλαίσια της διπλωματικής αυτής εμφανίζουν ανταγωνιστικές επιδόσεις συγκριτικά με τις state-of-the-art προσεγγίσεις, υπερτερώντας μάλιστα συχνά σε αρκετές από τις μετρικές, όπως φαίνεται και στον παραπάνω πίνακα.

Συμπερασματικά, οι υλοποιήσεις που αναπτύχθηκαν και παρουσιάστηκαν στα πλαίσια της διπλωματικής αυτής, όχι μόνο εμφανίζουν εξαιρετικές επιδόσεις στην αντιμετώπιση του προβλήματος κατηγοριοποίησης των χρηστών του Twitter σε πραγματικούς και αυτοματοποιημένους, αλλά καταφέρνουν σε αρκετές περιπτώσεις, με αποκορύφωμα τα δύο καλύτερα μοντέλα μας, να ξεπερνούν σε επιδόσεις και να υπερτερούν ακόμα και των state-of-the-art προσεγγίσεων.

Κεφάλαιο 7

7. Επίλογος

Στο Κεφάλαιο αυτό, συνοψίζουμε την έρευνα που πραγματοποιήθηκε στα πλαίσια της διπλωματικής αυτής, τα αποτελέσματα και τα συμπεράσματα που προέκυψαν από την εκπόνησή της, καθώς παρατίθενται και προτάσεις για μελλοντικές επεκτάσεις της, για την περαιτέρω βελτίωσή της απέναντι στο πρόβλημα ανίχνευσης αυτοματοποιημένων λογαριασμών στο Twitter, και ευρύτερα στα μέσα κοινωνικής δικτύωσης.

7.1 Σύνοψη και Συμπεράσματα

Αντικείμενο της παρούσας διπλωματικής εργασίας αποτελεί η αντιμετώπιση του προβλήματος ανίχνευσης αυτοματοποιημένων λογαριασμών, γνωστών ως bots, στο μέσο κοινωνικής δικτύωσης του Twitter. Λόγω της διαδραστικότητας, της απλότητας στην χρήση, καθώς και της ολοένα και συχνότερης επιλογής τους ως μέσο διάδοσης ειδήσεων στην σύγχρονη εποχή, τα μέσα κοινωνικής δικτύωσης έχουν γνωρίσει μεγάλη απήχηση και έχουν γίνει αναπόσπαστο τμήμα της καθημερινότητας των ανθρώπων. Ωστόσο, η διάδοση των μέσων κοινωνικής δικτύωσης, η διαρκώς αυξανόμενη χρήση τους στον τομέα της ενημέρωσης και η δυνατότητα της άμεσης διάδοσης πληροφορίας που παρέχουν, έχουν προσελκύσει κακόβουλους χρήστες, οι οποίοι αποσκοπούν στην εκμετάλλευση των δυνατοτήτων αυτών προς όφελός τους. Την τελευταία δεκαετία έχει σημειωθεί μια ραγδαία αύξηση στην δραστηριότητα των bots στις πλατφόρμες κοινωνικής δικτύωσης, και ιδιαίτερα στην πλατφόρμα του Twitter, την οποία εξετάζουμε στα πλαίσια της διπλωματικής αυτής, ενώ έχουν καταφέρει να ενσωματώσουν στην λειτουργία τους μηχανισμούς και ανθρώπινα συμπεριφορικά χαρακτηριστικά, τα οποία τους επιτρέπουν να μιμούνται την δραστηριότητα ενός πραγματικού χρήστη, προστατεύοντας έτσι την ύπαρξη και την λειτουργία τους από τα υπάρχοντα συστήματα που έχουν αναπτυχθεί έως τώρα για την ανίχνευσή τους.

Για αυτόν τον λόγο, αξιοποιώντας το σύνολο δεδομένων που προτάθηκε από τους Cresci et al. [8], ένα δημόσια διαθέσιμο dataset, το οποίο έχει αναγνωριστεί από την επιστημονική κοινότητα ως σημαντική πηγή πληροφοριών και ανάλυσης στον τομέα της ανίχνευσης και αντιμετώπισης των bots στα μέσα κοινωνικής δικτύωσης, σχεδιάσαμε και αναπτύξαμε δύο μεθόδους ανίχνευσης των bots στο Twitter, με χρήση Βαθιάς Μηχανικής Μάθησης και Προεκπαιδευμένων Μοντέλων Transformer.

Στην πρώτη μέθοδο, αναπτύξαμε ένα μονοτροπικό μοντέλο, το οποίο είναι ικανό να κατηγοριοποιήσει τους χρήστες σε πραγματικούς και αυτοματοποιημένους, χρησιμοποιώντας αποκλειστικά εισόδους που έχουν την μορφή εικόνας, η οποία προκύπτει από την δραστηριότητα του λογαριασμού τους. Πιο συγκεκριμένα, υιοθετήσαμε την μεθοδολογία που παρουσιάστηκε στα [101] και [102] για την δημιουργία μιας αλληλουχίας DNA, του λεγόμενου Digital DNA (Ψηφιακό DNA), το οποίο στην συνέχεια, μέσω της υιοθέτησης της

μεθοδολογίας που παρουσιάστηκε στο [50], μετατράπηκε σε μια εικόνα που αποτελείται από 3 κανάλια. Για κάθε έναν από τους χρήστες, δημιουργήσαμε δύο ψηφιακές αλληλουχίες DNA, οι οποίες βασίζονταν στον τύπο των tweets που δημοσιεύει ο χρήστης, και στο περιεχόμενό τους, αντίστοιχα. Συνεπώς, ο κάθε χρήστης χαρακτηριζόταν από δύο εικόνες, οι οποίες αποτύπωναν με διαφορετικό τρόπο την δραστηριότητα του λογαριασμού του. Στην συνέχεια, τροφοδοτήσαμε τις εικόνες αυτές σε προεκπαιδευμένα Συνελικτικά Νευρωνικά Δίκτυα, τα οποία κάνουμε fine-tune στο πρόβλημά μας για την κατηγοριοποίηση των χρηστών σε πραγματικούς ή αυτοματοποιημένους, και τα οποία επέτυχαν πολύ καλές επιδόσεις, με αποκορύφωμα το μοντέλο VGG16 [112], το οποίο αποτέλεσε το μοντέλο με τις καλύτερες επιδόσεις, τόσο στην περίπτωση που η εικόνα βασίζεται στον τύπο των tweets του χρήστη, όσο και στην περίπτωση που βασίζεται στο περιεχόμενό τους.

Στην δεύτερη μέθοδο, αναπτύξαμε πολυτροπικά μοντέλα, τα οποία είναι ικανά να κατηγοριοποιήσουν τους χρήστες του Twitter, χρησιμοποιώντας τόσο εισόδους που έχουν την μορφή εικόνας (visual modality), η οποία προκύπτει από την δραστηριότητα του λογαριασμού του χρήστη (τόσο από τον τύπο των tweets όσο και από το περιεχόμενό τους), όσο και εισόδους που έχουν την μορφή κειμένου (textual modality), που αποτελεί την περιγραφή που έχει θέσει στον λογαριασμό του ο χρήστης. Πιο συγκεκριμένα, επιστρατεύσαμε ένα σύνολο από Fusion Methods (Concatenation, Gated Multimodal Unit και Crossmodal Attention), μέσω των οποίων είχαμε την δυνατότητα να συγχωνεύσουμε με διαφορετικούς τρόπους το visual και το textual modality προς την κατηγοριοποίηση των χρηστών σε πραγματικούς και αυτοματοποιημένους. Το μοντέλο που αξιοποιήθηκε για το visual modality είναι το VGG16 [112], το οποίο συνιστούσε το καλύτερο μοντέλο στο πρόβλημα κατηγοριοποίησης της εικόνας του χρήστη, όπως φάνηκε από τα αποτελέσματα της προηγούμενης μεθόδου, ενώ για το textual modality αξιοποιήθηκε το προεκπαιδευμένο μοντέλο Transformer TwHIN-BERT [90]. Τα πολυτροπικά μοντέλα που αναπτύχθηκαν επέτυχαν εξαιρετικές επιδόσεις, καλύτερες τις περισσότερες φορές από τα αντίστοιχα μονοτροπικά μοντέλα. Αποκορύφωμα αποτελεί το μοντέλο TwHIN-BERT + VGG16 (Cross-Modal Attention) που χρησιμοποιεί εικόνες που βασίζονται στο περιεχόμενο των tweets, το οποίο όχι μόνο συνιστά το καλύτερο μοντέλο που αναπτύξαμε στα πλαίσια της διπλωματικής αυτής, εμφανίζοντας τις υψηλότερες επιδόσεις, αλλά οι επιδόσεις του ξεπερνούν ακόμα και επιδόσεις state-of-the-art προσεγγίσεων, αποτελώντας έτσι πρόοδο στον τομέα της ανίχνευσης και αντιμετώπισης των bots στα μέσα κοινωνικής δικτύωσης, και ανοίγοντας νέους ορίζοντες στην επιστημονική κοινότητα για την δημιουργία ανταγωνιστικότερων και αποτελεσματικότερων μοντέλων, για την διασφάλιση της ενημέρωσης και της αλληλεπίδρασης των χρηστών, και μιας πιο αξιόπιστης εμπειρίας χρήσης.

Τέλος, αξίζει να αναφερθεί το γεγονός πως η διπλωματική αυτή αποτελεί την πρώτη έρευνα που εισάγει και αξιοποιεί multimodal και crossmodal μοντέλα για την ανίχνευση κοινωνικών spambots στο Twitter, χρησιμοποιώντας αποκλειστικά την περιγραφή του λογαριασμού του χρήστη και τρισδιάστατες εικόνες, οι οποίες αντιπροσωπεύουν την δραστηριότητα του λογαριασμού του.

7.2 Μελλοντικές Επεκτάσεις

Παρά την επιτυχία των προτεινόμενων υλοποιήσεων, οι οποίες όπως αναφέραμε παρουσίασαν εξαιρετικές επιδόσεις, με μέρος των οποίων να ξεπερνούν ακόμα και state-of-

the-art προσεγγίσεις, λόγω των περιορισμών χρόνου και υπολογιστικών πόρων που περιλαμβάνονται στη διεξαγωγή μιας διπλωματικής εργασίας, υπάρχουν μερικές βελτιώσεις που θα μπορούσε κανείς να εφαρμόσει, με σκοπό τόσο την αύξηση της πληρότητας της προτεινόμενης υλοποίησης, όσο και της περαιτέρω έρευνας στην ανίχνευση των bots στα μέσα κοινωνικής δικτύωσης.

Πιο συγκεκριμένα, ένας πρώτος στόχος προς την επέκταση της εργασίας αυτής θα ήταν η χρήση και των άλλων δύο κατηγοριών από social spambots (δηλαδή των social spambots #2 και social spambots #3), τα οποία περιλαμβάνονται στο σύνολο δεδομένων που χρησιμοποιήσαμε. Με αυτόν τον τρόπο, ύστερα από την κατάλληλη επεξεργασία και ισορρόπηση του συνόλου δεδομένων, θα είχαμε τόσο πραγματικούς χρήστες, όσο και αυτοματοποιημένους χρήστες που θα ανήκαν σε διαφορετικές κατηγορίες από bots, δηλαδή η συμπεριφορά τους και η δραστηριότητά τους θα εμφάνιζε διαφορετικά χαρακτηριστικά και ιδιομορφίες. Συνεπώς, θα αποτελούσε μια πρόκληση για το μοντέλο μας, εφόσον θα έπρεπε να είναι σε θέση να αναγνωρίζει τα χαρακτηριστικά των επιμέρους κατηγοριών των social bots, και να τα διακρίνει από τους πραγματικούς χρήστες.

Ένας άλλος στόχος θα ήταν η εφαρμογή των προτεινόμενων υλοποιήσεών μας σε άλλα σύνολα δεδομένων που είναι δημοσίως διαθέσιμα στην επιστημονική κοινότητα και σχετίζονται με το πρόβλημα της ανίχνευσης bots στο Twitter. Με αυτόν τον τρόπο, θα μπορούσαμε να βεβαιωθούμε για την ικανότητα γενίκευσης των μοντέλων που αναπτύξαμε.

Ταυτόχρονα, άλλη μια προσθήκη που θα ενίσχυε την πληρότητα της προτεινόμενης υλοποίησης, και ενδεχομένως και τις επιδόσεις της, θα ήταν η ανάπτυξη και εφαρμογή περισσότερων fusion methods, εκτός των Concatenation, Gated Multimodal Unit και Crossmodal Attention, που ήδη χρησιμοποιήσαμε στα πλαίσια της διπλωματικής αυτής. Πιο συγκεκριμένα, αξιοποιώντας μεθόδους συγχώνευσης που επιστρατεύονται στα [126], [127], [128], και [129], θα ήμασταν σε θέση να αναπτύξουμε περισσότερα πολυτροπικά μοντέλα, να συγκρίνουμε τον τρόπο με τον οποίο συγχωνεύουν τις visual και textual αναπαραστάσεις, και να αξιολογήσουμε τις επιδόσεις τους.

Επιπρόσθετα, ένας ακόμα μελλοντικός στόχος θα ήταν η χρήση μεθόδων ερμηνευσιμότητας (explainability methods). Πιο συγκεκριμένα, αξιοποιώντας την τεχνική LIME [130], που επιστρατεύεται από τα [131], [132], καθώς και άλλες γνωστές τεχνικές ερμηνευσιμότητας, όπως την SHAP (SHapley Additive exPlanations) [133], κατά τις οποίες καθιστούμε ερμηνεύσιμες τις προβλέψεις και τις αποφάσεις που λαμβάνουν τα προτεινόμενα μοντέλα μας, θα ήμασταν σε θέση να αντιληφθούμε καλύτερα την λειτουργία και την συμπεριφορά τους.

Παράλληλα, άλλη μια τεχνική, που θα θέλαμε να εντάξουμε στα πλαίσια της υλοποίησής μας, είναι η τεχνική της ρύθμισης υπερπαραμέτρων (hyperparameter tuning), κατά την οποία προσπαθούμε να ανιχνεύσουμε το καλύτερο σύνολο των τιμών των υπερπαραμέτρων, που οδηγούν τελικά στην καλύτερη επίδοση του μοντέλου. Οι υπερπαράμετροι είναι παράμετροι που δεν μαθαίνονται από το μοντέλο κάνοντας χρήση των χαρακτηριστικών του συνόλου δεδομένων κατά την διάρκεια της εκπαίδευσης, αλλά ορίζονται προτού ξεκινήσει το στάδιο της εκπαίδευσης, όπως ο αριθμός των εποχών, ο ρυθμός μάθησης, ο συντελεστής της τεχνικής dropout, και άλλα.

Μια ακόμα τροποποίηση που θα μπορούσαμε να εφαρμόσουμε θα ήταν στην εκπαίδευση του μοντέλου μας και στον τρόπο με τον οποίο διαχειρίζεται τα δεδομένα του, ώστε να μην χρειάζεται επισημασμένα δεδομένα, δηλαδή δεδομένα που εμπεριέχουν και την ετικέτα της κλάσης στην οποία ανήκουν τα δείγματα. Αυτό θα επέτρεπε την εφαρμογή του μοντέλου μας σε μεγαλύτερο εύρος προβλημάτων, εφόσον σε αρκετά προβλήματα μηχανικής μάθησης, η πρόσβαση σε έναν ικανοποιητικό αριθμό από επισημασμένα δεδομένα είναι αρκετές φορές αδύνατη, λόγω χρονικών ή υπολογιστικών περιορισμών.

Μια άλλη πιθανή κατεύθυνση, που θα μπορούσαμε να ακολουθήσουμε για την ανάπτυξη του μοντέλου μας, θα ήταν η κατηγοριοποίηση των χρηστών με χρήση αποκλειστικά του κειμενικού περιεχομένου των tweets, ανεξάρτητα δηλαδή από την δραστηριότητα του λογαριασμού του χρήστη και από την περιγραφή που έχει παραθέσει στον λογαριασμό του. Έτσι, χωρίς καμία πρότερη γνώση των προφίλ των χρηστών και με χρήση μόνο των tweets τους, το μοντέλο θα ήταν σε θέση να καταλάβει αν τα tweets αυτά ανήκουν σε πραγματικό ή αυτοματοποιημένο λογαριασμό.

Επίσης, στο πλαίσιο της αποτελεσματικής αξιοποίησης των ευρημάτων της παρούσας έρευνας, θα μπορούσαμε να την ενσωματώσουμε σε μια διαδικτυακή εφαρμογή, στην οποία θα παρείχαμε τον λογαριασμό ενός χρήστη του Twitter, και το μοντέλο, χρησιμοποιώντας την περιγραφή του λογαριασμού του χρήστη και δημιουργώντας την εικόνα που προκύπτει από την δραστηριότητά του έως εκείνη την χρονική στιγμή, θα ήταν σε θέση να παρέχει μια εκτίμηση της πιθανότητας ο λογαριασμός αυτός να αποτελεί αυτοματοποιημένο χρήστη, με βάση την εκπαίδευση στην οποία έχει υποβληθεί.

Τέλος, μελλοντικό στόχο αποτελεί και η προσπάθεια επέκτασης του προτεινόμενου μοντέλου, ώστε να είναι σε θέση να ανιχνεύει αυτοματοποιημένους λογαριασμούς και σε άλλα μέσα κοινωνικής δικτύωσης, όπως το Facebook και το Instagram.

Βιβλιογραφία

[1] L. Ilias, S. Mouzakitis and D. Askounis, "Calibration of Transformer-Based Models for Identifying Stress and Depression in Social Media," *IEEE Transactions on Computational Social Systems,* 2023.

[2] L. Ilias and D. Askounis, "Multitask learning for recognizing stress and depression in social media," *arXiv preprint arXiv:2305.18907,* 2023.

[3] L. Ilias and I. Roussaki, "Detecting malicious activity in Twitter using deep learning techniques," *Applied Soft Computing, 107,* p. 107360, 2021.

[4] S. Ouni, F. Fkih and M. N. Omri, "Bots and gender detection on twitter using stylistic features," *International Conference on Computational Collective Intelligence,* pp. 650-660, 2022.

[5] S. Kudugunta and E. Ferrara, "Deep neural networks for bot detection," *Information Sciences,* vol. 467, pp. 312-322, 2018.

[6] K. Hayawi, S. Mathew, N. Venugopal, M. M. Masud and P.-H. Ho, "Deeprobot: a hybrid deep neural network model for social bot detection based on user profile data," *Social Network Analysis and Mining,* vol. 12, no. 1, p. 43, 2022.

[7] S. Feng, H. Wan, N. Wang and M. Luo, "BotRGCN: Twitter bot detection with relational graph convolutional networks," *Proceedings of the 2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining,* pp. 236-239, 2021.

[8] S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi and M. Tesconi, "The paradigm-shift of social spambots: Evidence, theories, and tools for the arms race," *Proceedings of the 26th international conference on world wide web companion,* pp. 963-972, 2017.

[9] D. Dukić, D. Keča and D. Stipić, "Are You Human? Detecting Bots on Twitter Using BERT," *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA),* pp. 631-636, 2020.

[10] "Bots and Gender Profiling 2019," 2019. [Online]. Available: https://pan.webis.de/clef19/pan19-web/author-profiling.html.

[11] N. Narayan, "Twitter Bot Detection using Machine Learning Algorithms," *2021 Fourth International Conference on Electrical, Computer and Communication Technologies (ICECCT),* pp. 1-4, 2021.

[12] T. Fagni, F. Falchi, M. Gambini, A. Martella and M. Tesconi, "TweepFake: About detecting deepfake tweets," *PLoS ONE 16(5), e0251415,* 2021.

[13] A. Ramalingaiah, S. Hussaini and S. Chaudhari, "Twitter bot detection using supervised machine learning," *Journal of Physics: Conference Series,* vol. 1950, no. 1, p. 012006, 2021.

[14] "Kaggle - Detecting Twitter Bot Data," [Online]. Available: https://www.kaggle.com/datasets/charvijain27/detecting-twitter-bot-data.

[15] H. Shukla, N. Jagtap and B. Patil, "Enhanced Twitter bot detection using ensemble machine learning," *2021 6th International Conference on Inventive Computation Technologies (ICICT),* pp. 930-936, 2021.

[16] F. N. Pramitha, R. B. Hadiprakoso, N. Qomariasih and Girinoto, "Twitter Bot Account Detection Using Supervised Machine Learning," *2021 4th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI),* pp. 379-383, 2021.

[17] T. Tyagi, P. Sharma, R. Bansal, K. Jain, P. Bansal and K. Malik, "Twitter Bot Detection using Machine Learning Models," *2023 13th International Conference on Cloud Computing, Data Science & Engineering (Confluence),* pp. 26-30, 2023.

[18] A. Shevtsov, C. Tzagkarakis, D. Antonakaki and S. Ioannidis, "Identification of twitter bots based on an explainable machine learning framework: the US 2020 elections case study," *Proceedings of the International AAAI Conference on Web and Social Media,* vol. 16, pp. 956-967, 2022.

[19] M. Heidari, J. H. J. Jones and O. Uzuner, "An Empirical Study of Machine learning Algorithms for Social Media Bot Detection," *2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS),* pp. 1-5, 2021.

[20] J. V. Fonseca Abreu, C. Ghedini Ralha and J. J. Costa Gondim, "Twitter Bot Detection with Reduced Feature Set," *2020 IEEE International Conference on Intelligence and Security Informatics (ISI),* pp. 1-6, 2020.

[21] J. Rodríguez-Ruiz, J. I. Mata-Sánchez, R. Monroy, O. Loyola-Gonzalez and A. López-Cuevas, "A one-class classification approach for bot detection on Twitter," *Computers & Security, 91,* p. 101715, 2020.

[22] J. Knauth, "Language-agnostic twitter-bot detection," *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019),* pp. 550-558, 2019.

[23] D. Kosmajac and V. Keselj, "Twitter bot detection using diversity measures," *Proceedings of the 3rd International Conference on Natural Language and Speech Processing,* pp. 1-8, 2019.

[24] O. Varol, E. Ferrara, C. A. Davis, F. Menczer and A. Flammini, "Online human-bot interactions: Detection, estimation, and characterization," *Proceedings of the international AAAI conference on web and social media,* vol. 11, no. 1, pp. 280-289, 2017.

[25] K. Lee, B. Eoff and J. Caverlee, "Seven months with the devils: A long-term study of content polluters on twitter," *Proceedings of the international AAAI conference on web and social media,* vol. 5, no. 1, pp. 185-192, 2011.

[26] A. Davoudi, A. Z. Klein, A. Sarker and G. Gonzalez-Hernandez, "Towards automatic bot detection in Twitter for health-related tasks," *AMIA Summits on Translational Science Proceedings 2020,* pp. 136-141, 2020.

[27] C. A. Davis, O. Varol, E. Ferrara, A. Flammini and F. Menczer, "Botornot: A system to evaluate social bots," *Proceedings of the 25th international conference companion on world wide web,* pp. 273-274, 2016.

[28] K. E. Daouadi, R. Z. Rebaï and I. Amous, "Bot detection on online social networks using deep forest," *Artificial Intelligence Methods in Intelligent Algorithms: Proceedings of 8th Computer Science On-line Conference 2019,* vol. 2 8, pp. 307-315, 2019.

[29] V. S. Subrahmanian et al., "The DARPA Twitter Bot Challenge," *Computer,* vol. 49, no. 6, pp. 38-46, 2016.

[30] A. P. Rodrigues, R. Fernandes, A. Shetty, K. Lakshmanna and R. M. Shafi, "Real-time twitter spam detection and sentiment analysis using machine learning and deep learning techniques," *Computational Intelligence and Neuroscience 2022,* 2022.

[31] F. Wei and U. T. Nguyen, "Twitter Bot Detection Using Bidirectional Long Short-Term Memory Neural Networks and Word Embeddings," *2019 First IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA),* pp. 101-109, 2019.

[32] M. Heidari and J. H. Jones, "Using BERT to Extract Topic-Independent Sentiment Features for Social Media Bot Detection," *2020 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON),* pp. 0542-0547, 2020.

[33] S. M. Saravani, I. Ray and I. Ray, "Automated identification of social media bots using deepfake text detection," *International Conference on Information Systems Security,* pp. 111-123, 2021.

[34] D. Martín-Gutiérrez, G. Hernández-Peñaloza, A. B. Hernández, A. Lozano-Diez and F. Álvarez, "A Deep Learning Approach for Robust Detection of Bots in Twitter Using Transformers," *IEEE Access,* vol. 9, pp. 54591-54601, 2021.

[35] K.-C. Yang, O. Varol, P.-M. Hui and F. Menczer, "Scalable and generalizable social bot detection through data selection," *Proceedings of the AAAI Conference on Artificial Intelligence,* vol. 34, no. 1, pp. 1096-1103, 2020.

[36] S. Cresci, F. Lillo, D. Regoli, S. Tardelli and M. Tesconi, "$ FAKE: Evidence of spam and bot activity in stock microblogs on Twitter," *Proceedings of the International AAAI Conference on Web and Social Media,* vol. 12, no. 1, 2018.

[37] M. Mazza, S. Cresci, M. Avvenuti, W. Quattrociocchi and M. Tesconi, "Rtbust: Exploiting temporal patterns for botnet detection on twitter," *Proceedings of the 10th ACM Conference on Web Science,* pp. 183-192, 2019.

[38] K.-C. Yang, O. Varol, C. A. Davis, E. Ferrara, A. Flammini and F. Menczer, "Arming the public with artificial intelligence to counter social bots," *Human Behavior and Emerging Technologies,* vol. 1, no. 1, pp. 48-61, 2019.

[39] Z. Gilani, R. Farahbakhsh, G. Tyson, L. Wang and J. Crowcroft, "Of bots and humans (on twitter)," *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017,* pp. 349-354, 2017.

[40] M. Heidari, J. H. Jones and O. Uzuner, "Deep Contextualized Word Embedding for Text-based Online User Profiling to Detect Social Bots on Twitter," *2020 International Conference on Data Mining Workshops (ICDMW),* pp. 480-487, 2020.

[41] E. Arin and M. Kutlu, "Deep Learning Based Social Bot Detection on Twitter," *IEEE Transactions on Information Forensics and Security,* vol. 18, pp. 1763-1772, 2023.

[42] L. Luo, X. Zhang, X. Yang and W. Yang, "Deepbot: a deep neural network based approach for detecting Twitter bots," *IOP Conference Series: Materials Science and Engineering,* vol. 719, no. 1, p. 012063, 2020.

[43] J. Tourille, B. Sow and A. Popescu, "Automatic Detection of Bot-generated Tweets," *Proceedings of the 1st International Workshop on Multimedia AI against Disinformation,* pp. 44-51, 2022.

[44] Z. Lei, H. Wan, W. Zhang, S. Feng, Z. Chen, J. Li, Q. Zheng and M. Luo, "Bic: Twitter bot detection with text-graph interaction and semantic consistency," *arXiv preprint arXiv:2208.08320,* 2022.

[45] S. Feng, H. Wan, N. Wang, J. Li and M. Luo, "Twibot-20: A comprehensive twitter bot detection benchmark," *Proceedings of the 30th ACM International Conference on Information & Knowledge Management,* pp. 4485-4494, 2021.

[46] S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi and M. Tesconi, "Fame for sale: Efficient detection of fake Twitter followers," *Decision Support Systems, 80,* pp. 56-71, 2015.

[47] A. Garcia-Silva, C. Berrio and J. M. Gomez-Perez, "Understanding transformers for bot detection in Twitter," *arXiv preprint arXiv:2104.06182,* 2021.

[48] Z. Gilani, E. Kochmar and J. Crowcroft, "Classification of twitter accounts into automated agents and human users," *Proceedings of the 2017 IEEE/ACM international conference on advances in social networks analysis and mining 2017,* pp. 489-496, 2017.

[49] S. Feng, H. Wan, N. Wang, J. Li and M. Luo, "Satar: A self-supervised approach to twitter account representation learning and its application in bot detection," *Proceedings*

of the 30th ACM International Conference on Information & Knowledge Management, pp. 3808-3817, 2021.

[50] E. Di Paolo, M. Petrocchi and A. Spognardi, "From Online Behaviours to Images: A Novel Approach to Social Bot Detection," *arXiv preprint arXiv:2304.07535,* 2023.

[51] M. Chakraborty, S. Das and R. Mamidi, "Detection of Fake Users in Twitter Using Network Representation and NLP," *2022 14th International Conference on COMmunication Systems & NETworkS (COMSNETS),* pp. 754-758, 2022.

[52] Y. Hua, M. Naaman and T. Ristenpart, "Characterizing twitter users who engage in adversarial interactions against political candidates," *Proceedings of the 2020 CHI conference on human factors in computing systems,* pp. 1-13, 2020.

[53] T. Bui and K. Potika, "Twitter Bot Detection using Social Network Analysis," *2022 Fourth International Conference on Transdisciplinary AI (TransAI),* pp. 87-88, 2022.

[54] S. Feng, Z. Tan, R. Li and M. Luo, "Heterogeneity-aware twitter bot detection with relational graph transformers," *Proceedings of the AAAI Conference on Artificial Intelligence,* vol. 36, no. 4, pp. 3977-3985, 2022.

[55] S. Ali Alhosseini, R. Bin Tareaf, P. Najafi and C. Meinel, "Detect me if you can: Spam bot detection using inductive representation learning," *Companion proceedings of the 2019 world wide web conference,* pp. 148-153, 2019.

[56] D. M. Beskow and K. M. Carley, "Bot conversations are different: leveraging network metrics for bot detection in twitter," *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM),* pp. 825-832, 2018.

[57] J. Wu, E. Teng and Z. Cao, "Twitter Bot Detection Through Unsupervised Machine Learning," *2022 IEEE International Conference on Big Data (Big Data),* pp. 5833-5839, 2022.

[58] A. Anwar and U. Yaqub, "Bot detection in twitter landscape using unsupervised learning," *The 21st Annual International Conference on Digital Government Research,* pp. 329-330, 2020.

[59] S. Gera and A. Sinha, "A machine learning-based malicious bot detection framework for trend-centric twitter stream," *Journal of Discrete Mathematical Sciences and Cryptography 24,* no. 5, pp. 1337-1348, 2021.

[60] V. Rawatt, "Introduction to Machine Learning," [Online]. Available: https://medium.com/@vaani.rawatt/introduction-to-machine-learning-a5ddc31ce404. [Accessed 28 9 2023].

[61] "Deep Learning: All about this concept," DataScientest, [Online]. Available: https://datascientest.com/en/all-about-deep-learning. [Accessed 28 9 2023].

[62] F. Rosenblatt, "The perceptron: a probabilistic model for information storage and organization in the brain," *Psychological review,* vol. 65, no. 6, pp. 386-408, 1958.

[63] A. Saxena, "Building a Simple Neural Network from Scratch," Towards Data Science, [Online]. Available: https://towardsdatascience.com/building-a-simple-neural-network-from-scratch-a5c6b2eb0c34. [Accessed 28 9 2023].

[64] S. Kampakis, "What Deep Learning Is And Isn't," The Data Scientist, [Online]. Available: https://thedatascientist.com/what-deep-learning-is-and-isnt/. [Accessed 28 9 2023].

[65] "Sigmoid function," Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Sigmoid_function. [Accessed 28 9 2023].

[66] "MedCalc," [Online]. Available: https://www.medcalc.org/manual/tanh-function.php. [Accessed 28 9 2023].

[67] A. Bajpai, "Top Activation Functions at a glance," [Online]. Available: https://medium.com/@anushka.datascoop/activation-functions-at-a-glance-352fb3051abd. [Accessed 28 9 2023].

[68] M. M. ARAT, "Some Basic Activation Functions," [Online]. Available: https://mmuratarat.github.io/2019-02-10/some-basic-activation-functions. [Accessed 28 9 2023].

[69] S. Bag, "Activation Functions - All You Need To Know!," [Online]. Available: https://medium.com/analytics-vidhya/activation-functions-all-you-need-to-know-355a850d025e. [Accessed 28 9 2023].

[70] S. Gupta, "The 7 Most Common Machine Learning Loss Functions," [Online]. Available: https://builtin.com/machine-learning/common-loss-functions. [Accessed 28 9 2023].

[71] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747,* 2016.

[72] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980,* 2014.

[73] N. Shahriar, "What is Convolutional Neural Network - CNN (Deep Learning)," [Online]. Available: https://nafizshahriar.medium.com/what-is-convolutional-neural-network-cnn-deep-learning-b3921bdd82d5. [Accessed 28 9 2023].

[74] M. Stewart, "Simple Introduction to Convolutional Neural Networks," Towards Data Science, [Online]. Available: https://towardsdatascience.com/simple-introduction-to-convolutional-neural-networks-cdf8d3077bac. [Accessed 28 9 2023].

[75] V. Rajput, "Pooling layers in Neural nets and their variants," [Online]. Available: https://medium.com/aiguys/pooling-layers-in-neural-nets-and-their-variants-f6129fc4628b. [Accessed 28 9 2023].

[76] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *International conference on machine learning,* pp. 448-456, 2015.

[77] C. Olah, "Understanding LSTM Networks," [Online]. Available: https://colah.github.io/posts/2015-08-Understanding-LSTMs/. [Accessed 28 9 2023].

[78] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation,* vol. 9, no. 8, pp. 1735-1780, 1997.

[79] T. M. Ingolfsson, "Insights into LSTM architecture," [Online]. Available: https://thorirmar.com/post/insight_into_lstm/. [Accessed 28 9 2023].

[80] J. Zweig, "Elmo Embeddings in Keras with TensorFlow hub," Towards Data Science, [Online]. Available: https://towardsdatascience.com/elmo-embeddings-in-keras-with-tensorflow-hub-7eb6f0145440. [Accessed 28 9 2023].

[81] D. Bahdanau, K. Cho and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473,* 2014.

[82] A. Vaswani et al., "Attention is all you need," *Advances in neural information processing systems 30,* 2017.

[83] R. Agarwal, "Understanding Transformers, the Data Science Way," [Online]. Available: https://www.kdnuggets.com/2020/10/understanding-transformers-data-science-way.html. [Accessed 28 9 2023].

[84] "Natural Language Processing (NLP) with Python - Tutorial," [Online]. Available: https://towardsai.net/p/nlp/natural-language-processing-nlp-with-python-tutorial-for-beginners-1f54e610a1a0. [Accessed 28 9 2023].

[85] T. Mikolov, K. Chen, G. Corrado and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781,* 2013.

[86] J. Pennington, R. Socher and C. D. Manning, "Glove: Global vectors for word representation," *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP),* pp. 1532-1543, 2014.

[87] J. Pennington, R. Socher and C. D. Manning, "GloVe: Global Vectors for Word Representation," [Online]. Available: https://nlp.stanford.edu/projects/glove/. [Accessed 28 9 2023].

[88] Y. Bengio, R. Ducharme and P. Vincent, "A neural probabilistic language model," *Advances in neural information processing systems 13,* 2000.

[89] J. Devlin, M. W. Chang, K. Lee and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805,* 2018.

[90] X. Zhang, Y. Malkov, O. Florez, S. Park, B. McWilliams, J. Han and A. El-Kishky, "Twhin-bert: A socially-enriched pre-trained language model for multilingual tweet representations," *arXiv preprint arXiv:2209.07562,* 2022.

[91] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692,* 2019.

[92] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," *arXiv preprint arXiv:1909.11942,* 2019.

[93] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov and Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," *Advances in neural information processing systems 32,* 2019.

[94] V. Sanh, L. Debut, J. Chaumond and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108,* 2019.

[95] D. Q. Nguyen, T. Vu and A. T. Nguyen, "BERTweet: A pre-trained language model for English Tweets," *arXiv preprint arXiv:2005.10200,* 2020.

[96] R. Horev, "BERT Explained: State of the art language model for NLP," Towards Data Science, [Online]. Available: https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270. [Accessed 28 9 2023].

[97] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," *arXiv preprint arXiv:1911.02116,* 2019.

[98] C. Yang, R. Harkreader and G. Gu, "Empirical evaluation and new design for fighting evolving twitter spammers," *IEEE Transactions on Information Forensics and Security,* vol. 8, no. 8, pp. 1280-1293, 2013.

[99] W. McKinney et al., "Data structures for statistical computing in python," *Proceedings of the 9th Python in Science Conference,* vol. 445, pp. 51-56, 2010.

[100] L. Mannocci, S. Cresci, A. Monreale, A. Vakali and M. Tesconi, "MulBot: Unsupervised Bot Detection Based on Multivariate Time Series," *2022 IEEE International Conference on Big Data (Big Data),* pp. 1485-1494, 2022.

[101] S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi and M. Tesconi, "Dna-inspired online behavioral modeling and its application to spambot detection," *IEEE Intelligent Systems,* vol. 31, no. 5, pp. 58-64, 2016.

[102] S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi and M. Tesconi, "Social fingerprinting: Detection of spambot groups through dna-inspired behavioral modeling," *IEEE Transactions on Dependable and Secure Computing,* vol. 15, no. 4, pp. 561-576, 2018.

[103] S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi and M. Tesconi, "Exploiting digital dna for the analysis of similarities in twitter behaviours," *2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA),* pp. 686-695, 2017.

[104] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich, "Going deeper with convolutions," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* pp. 1-9, 2015.

[105] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* pp. 770-778, 2016.

[106] S. Zagoruyko and N. Komodakis, "Wide residual networks," *arXiv preprint arXiv:1605.07146,* 2016.

[107] A. Krizhevsky, "One weird trick for parallelizing convolutional neural networks," *arXiv preprint arXiv:1404.5997,* 2014.

[108] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally and K. Keutzer, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and< 0.5 mb model size," *arXiv preprint arXiv:1602.07360,* 2016.

[109] G. Huang, Z. Liu, L. Van Der Maaten and K. Q. Weinberger, "Densely connected convolutional networks," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* pp. 2261-2269, 2017.

[110] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition,* pp. 4510-4520, 2018.

[111] S. Xie, R. Girshick, P. Dollár, Z. Tu and K. He, "Aggregated residual transformations for deep neural networks," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* pp. 5987-5995, 2017.

[112] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556,* 2014.

[113] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," *International conference on machine learning,* pp. 6105-6114, 2019.

[114] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai and S. Chintala, "Pytorch: An

imperative style, high-performance deep learning library," *Advances in neural information processing systems, 32,* 2019.

[115] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research,* vol. 15, no. 1, pp. 1929-1958, 2014.

[116] J. Arevalo, T. Solorio, M. Montes-y Gomez and F. A. González, "Gated multimodal networks," *Neural Computing and Applications,* pp. 1-20, 2020.

[117] L. Ilias, D. Askounis and J. Psarras, "Detecting dementia from speech and transcripts using transformers," *Computer Speech & Language,* vol. 79, p. 101485, 2023.

[118] L. Ilias, D. Askounis and J. Psarras, "Multimodal detection of epilepsy with deep neural networks," *Expert Systems with Applications,* vol. 213, p. 119010, 2023.

[119] Y.-H. H. Tsai, S. Bai, P. P. Liang, J. Z. Kolter, L.-P. Morency and R. Salakhutdinov, "Multimodal transformer for unaligned multimodal language sequences," *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics,* vol. 2019, pp. 6558-6569, 2019.

[120] D. Sánchez Villegas and N. Aletras, "Point-of-interest type prediction using text and images," *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing,* pp. 7785-7797, 2021.

[121] M. Fazil, A. K. Sah and M. Abulaish, "Deepsbd: A deep neural network model with attention mechanism for socialbot detection," *IEEE Transactions on Information Forensics and Security,* vol. 16, pp. 4211-4223, 2021.

[122] C. Cai, L. Li and D. Zeng, "Detecting social bots by jointly modeling deep behavior and content information," *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management,* pp. 1995-1998, 2017.

[123] H. Ping and S. Qin, "A social bots detection model based on deep learning algorithm," *2018 IEEE 18th International Conference on Communication Technology (ICCT),* pp. 1435-1439, 2018.

[124] F. Ahmed and M. Abulaish, "A generic statistical approach for spam detection in online social networks," *Computer Communications,* vol. 36, no. 10, pp. 1120-1129, 2013.

[125] T. Wolf et al., "Transformers: State-of-the-Art Natural Language Processing," *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations,* pp. 38-45, 2020.

[126] L. Ilias and D. Askounis, "Multimodal Deep Learning Models for Detecting Dementia From Speech and Transcripts," *Frontiers in Aging Neuroscience,* vol. 14, 2022.

[127] L. Ilias, D. Askounis and J. Psarras, "A Multimodal Approach for Dementia Detection from Spontaneous Speech with Tensor Fusion Layer," *2022 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI),* pp. 1-5, 2022.

[128] L. Ilias and D. Askounis, "Context-aware attention layers coupled with optimal transport domain adaptation and multimodal fusion methods for recognizing dementia from spontaneous speech," *Knowledge-Based Systems,* vol. 277, p. 110834, 2023.

[129] M. Chatzianastasis, L. Ilias, D. Askounis and M. Vazirgiannis, "Neural Architecture Search with Multimodal Fusion Methods for Diagnosing Dementia," *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),* pp. 1-5, 2023.

[130] M. T. Ribeiro, S. Singh and C. Guestrin, ""Why should I trust you?" explaining the predictions of any classifier," *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining,* pp. 1135-1144, 2016.

[131] L. Ilias and D. Askounis, "Explainable Identification of Dementia From Transcripts Using Transformer Networks," *IEEE Journal of Biomedical and Health Informatics,* vol. 26, no. 8, pp. 4153-4164, 2022.

[132] L. Ilias, F. Soldner and B. Kleinberg, "Explainable Verbal Deception Detection using Transformers," *arXiv preprint arXiv:2210.03080,* 2022.

[133] S. M. Lundberg and S. I. Lee, "A unified approach to interpreting model predictions," *Advances in neural information processing systems,* vol. 30, 2017.