Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών
και Μηχανικών Υπολογιστών
Τομέας Σημάτων, Ελέγχου
και Ρομποτικής

Τεχνικές Μεταφοράς Μάθησης σε Βαθιά Νευρωνικά Δίκτυα για Ανάλυση Συναισθήματος και Σημασιολογική Μοντελοποίηση

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΑΛΕΞΑΝΔΡΑ ΧΡΟΝΟΠΟΥΛΟΥ

Επιβλέπων : Αλέξανδρος Ποταμιάνος
Αναπληρωτής Καθηγητής Ε.Μ.Π.

Αθήνα, Μάρτιος 2019

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών
και Μηχανικών Υπολογιστών
Τομέας Σημάτων, Ελέγχου
και Ρομποτικής

Τεχνικές Μεταφοράς Μάθησης σε Βαθιά Νευρωνικά Δίκτυα για Ανάλυση Συναισθήματος και Σημασιολογική Μοντελοποίηση

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΑΛΕΞΑΝΔΡΑ ΧΡΟΝΟΠΟΥΛΟΥ

Επιβλέπων : Αλέξανδρος Ποταμιάνος
Αναπληρωτής Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 20η Μαρτίου 2019.

Αλέξανδρος Ποταμιάνος	**Ίων Ανδρουτσόπουλος**	**Γεώργιος Στάμου**
Αναπληρωτής Καθηγητής Ε.Μ.Π.	Αναπληρωτής Καθηγητής Ο.Π.Α.	Αναπληρωτής Καθηγητής Ε.Μ.Π.

Αθήνα, Μάρτιος 2019

Αλεξάνδρα Χρονοπούλου

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Στη γιαγιά μου, Αλεξάνδρα.

Περίληψη

Στα πλαίσια αυτής της διατριβής [1], εξετάζουμε το ζήτημα της μη ικανοποιητικής απόδοσης σε προβλήματα ταξινόμησης λόγω της έλλειψης δεδομένων με ετικέτες. Για να επιτύχουμε σημαντικές βελτιώσεις στα συγκεκριμένα προβλήματα ταξινόμησης, αξιοποιούμε προεκπαιδευμένες αναπαραστάσεις και εξερευνούμε μεθόδους μεταφοράς μάθησης, τόσο στη μορφή προεκπαιδευμένων ταξινομητών όσο και προεκπαιδευμένων γλωσσικών μοντέλων. Έπειτα, παρουσιάζουμε μια πιο αποτελεσματική και εξειδικευμένη μορφή μεταφοράς μάθησης, η οποία περιέχει μια βοηθητική συνάρτηση κόστους για το γλωσσικό μοντέλο, ταυτόχρονα με την συνάρτηση κόστους του ταξινομητή. Το ζήτημα αυτό είναι καίριο στην βαθιά μάθηση (deep learning) και έχει ως ένα βαθμό αντιμετωπιστεί πρόσφατα στο πεδίο της Επεξεργασίας Φυσικής Γλώσσας, καθώς τα βαθιά νευρωνικά δίκτυα συνήθως απαιτούν έναν εκτεταμένο αριθμό παραδειγμάτων κατά τη διάρκεια της εκπαίδευσης. Ωστόσο, το να αποκτήσει κανείς πληθώρα δεδομένων για να εκπαιδεύσει ένα τέτοιο νευρωνικό δίκτυο είναι συχνά δαπανηρό και δύσκολο να επιτευχθεί.

Αρχικά παρουσιάζουμε μια μέθοδο, κατά την οποία χρησιμοποιούμε ένα προεκπαιδευμένο μοντέλο σε ανάλυση συναισθήματος για να μειώσουμε το σφάλμα πάνω στο σύνολο δεδομένων σε ένα σημασιολογικά παρεμφερές πρόβλημα ταξινόμησης. Η μεταφορά μάθησης από προεκπαιδευμένους ταξινομητές αξιοποιεί την αναπαράσταση που έχει μάθει ένα μοντέλο υπό συνθήκες επιβλεπόμενης μάθησης, σε ένα συγκεκριμένο πρόβλημα με πληθώρα δεδομένων για εκπαίδευση, για να επιτύχει ανταγωνιστικά αποτελέσματα σε ένα παρόμοιο πρόβλημα, όπου μόνο λίγα δεδομένα είναι διαθέσιμα.

Έπειτα χρησιμοποιούμε προεκπαιδευμένες αναπαραστάσεις λέξεων από γλωσσικά μοντέλα, για να αντιμετωπίσουμε ένα πρόβλημα κατηγοριοποίησης κειμένου στα βασικά συναισθήματα. Ένας αλγόριθμος μάθησης μπορεί να χρησιμοποιήσει πληροφορίες που απέκτησε επιλύοντας ένα πρόβλημα μη επιβλεπόμενης μάθησης για να έχει καλύτερη απόδοσης στο στάδιο επιβλεπόμενης μάθησης. Συγκεκριμένα, οι προεκπαιδευμένες αναπαραστάσεις λέξεων που μας προσφέρουν τα γλωσσικά μοντέλα είναι χρήσιμες, διότι κωδικοποιούν πληροφορίες σχετικές με το περιεχόμενο και μοντελοποιούν τη σύνταξη αλλά και τη σημασιολογία. Προτείνουμε μια μέθοδο μεταφοράς μάθησης που αποτελείται από τρία βήματα: αρχικά εκπαίδευση ενός γλωσσικού μοντέλου, έπειτα προσαρμογή του μοντέλου στο πρόβλημα (task) που αντιμετωπίζουμε και τέλος μεταφορά του μοντέλου αυτού σε έναν ταξινομητή για να αξιοποιήσουμε αυτές τις αναπαραστάσεις. Αναφέρουμε ότι η μέθοδος μας επιτυγχάνει 10% βελτίωση σχετικά με το βασικό μοντέλο του WASSA 2018. Επιτυγχάνουμε επίσης F1-score ίσο με 70.3%, γεγονός που μας τοποθετεί στην πρώτη τριάδα της κατάταξης του σχετικού διαγωνισμού.

Τελικά παρουσιάζουμε ένα εννοιολογικά απλό και αποτελεσματικό μοντέλο μεταφοράς μάθησης, το οποίο αντιμετωπίζει το πρόβλημα του catastrophic forgetting. Συγκεκριμένα, συνδυάζουμε την συνάρτηση βελτιστοποίησης για ένα συγκεκριμένο πρόβλημα με τη βοηθητική συνάρτηση βελτιστοποίησης του γλωσσικού μοντέλου, η οποία προσαρμόζεται κατά τη διαδικασία εκπαίδευσης. Αυτό διαφυλάσσει τη μοντελοποίηση της γλώσσας που έχει μάθει το γλωσσικό μοντέλο, ενώ επιτρέπει ταυτόχρονα αρκετές αλλαγές για να επιλυθεί το εκάστοτε πρόβλημα ταξινόμησης. Η εισαγωγή της βοηθητικής συνάρτησης του γλωσσικού μοντέλου μας επιτρέπει να ελέγχουμε απολύτως τη συνεισφορά του προεκπαιδευμένου μέρους του μοντέλου και να διασφαλίσουμε ότι η γνώση που έχει κωδικοποιήσει θα διατηρηθεί. Η προσέγγισή μας παρουσιάζει εύρωστα αποτελέσματα σε 5 διαφορετικά προβλήματα ταξινόμησης, όπου αναφέρουμε σημαντικές βελτιώσεις σε σχέση με τα βασικά μοντέλα (baselines). Η βελτίωση της απόδοσης είναι πιο φανερή όταν το σετ δεδομένων που έχει χρησιμοποιηθεί στην προεκπαίδευση ανήκει σε διαφορετικό τομέα (domain) απ' ότι το σετ δεδομένων

[1] Οι δημοσιεύσεις [1], [2], [3] έγιναν κατά την διεξαγωγή αυτής της διπλωματικής.

που έχει χρησιμοποιηθεί στην προσαρμογή (fine-tuning). Χαρακτηριστικό παράδειγμα αποτελεί το Sarcasm corpus σετ δεδομένων, μεταξύ άλλων, όπου επιτυγχάνουμε F1-σκορ 75%, μόλις 1% κάτω από το state of the art. Αξιολογούμε το μοντέλο μας σε πληθώρα διαφορετικών προβλημάτων και δείχνουμε ότι η προσέγγισή μας μπορεί να επιτύχει εντυπωσιακά αποτελέσματα ακόμα και με ελάχιστα δεδομένα εκπαίδευσης.

Λέξεις κλειδιά

μεταφορά μάθησης, γλωσσικά μοντέλα, αναγνώριση συναισθημάτων, ανάλυση συναισθήματος, μη επιβλεπόμενη μάθηση, μάθηση πολλαπλών εργασιών, βαθιά μάθηση, αναδρομικά νευρωνικά δίκτυα

Abstract

In this work [2], we address the issue of poor performance in classification tasks due to scarcity of labeled data. To yield substantial improvements in classification tasks, we leverage pretrained representations and explore transfer learning methods, both in the form of pretrained classifiers and pretrained language models. We then present a more effective and refined transfer learning approach, where we introduce an auxiliary language model loss to the transferred model. The addressed issue is crucial in deep learning and has only recently been tackled in the Natural Language Processing field, as deep neural networks typically require an extended number of training annotated examples, yet large quantities of data are often expensive and difficult to collect.

First, we propose a method for successfully utilizing a pretrained sentiment analysis classification model to reduce the test error rate on an emotion recognition classification task. Transfer learning from pretrained classifiers exploits the representation that a model has learned for one supervised setting with plenty of data to obtain competitive results on a related task where only a small dataset is available. We aim to leverage the more generic representation of the pretrained classifier to tackle the target task, building upon the intuition that knowledge of positive, negative or neutral sentiment should be beneficial for a classification in the 6 basic emotions, namely anger, joy, fear, disgust, suprise and sadness.

Next, we utilize pretrained representations from language models to address an emotion recognition classification task. A learning algorithm can use information learned in the unsupervised phase to perform better in the supervised learning stage. Specifically, pretrained word representations captured by language models are useful as they encode contextual information and model syntax and semantics. We propose a three-step transfer learning method that includes pretraining a language model, fine-tuning it on the target task and transferring the model to a classifier to leverage these representations. We show an improvement of 10% on the WASSA 2018 emotion recognition dataset baseline. We achieve a F_1-score of 70.3%, ranking in the top-3 positions of the respective competition.

Finally, we present a conceptually simple and effective transfer learning approach that addresses the problem of catastrophic forgetting. Specifically, we combine the task-specific optimization functional with an auxiliary language model objective, which is adjusted during the training process. This preserves language regularities captured by language models, while enabling sufficient adaptation for solving the target task. The introduction of the auxiliary language model loss allows us to explicitly control the weighting of the pretrained part of the model and ensure that the distilled knowledge it encodes is preserved. Our approach shows robust results on 5 different classification datasets, where we report significant boosts over the baselines. The performance improvement is more pronounced when there is a mismatch between the pretraining and target task domains, which is the case in the *Sarcasm Corpus* dataset amongst others, where we achieve a F_1-score of 75%, 1% below state-of-the-art. We evaluate our model on a variety of classification tasks and demonstrate that our approach is able to yield impressive results even on a handful of training examples.

Key words

transfer learning, language modeling, emotion recognition, sentiment analysis, unsupervised pretraining, multi-task learning, deep learning, recurrent neural networks

[2] Papers: [1], [2], [3] have been conducted under the development of this thesis.

Ευχαριστίες

Αρχικά, θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή αυτής της εργασίας Αλέξανδρο Ποταμιάνο για την καθοδήγησή του κατά τη διάρκεια της εκπόνησής της. Οι συμβουλές του με βοήθησαν να βελτιώσω την έρευνά μου και να τη δημοσιεύσω. Επιπλέον, θα ήθελα να ευχαριστήσω τον υποψήφιο διδάκτορα Χρήστο Μπαζιώτη για την υπομονή και την προθυμία του να απαντά τις ερωτήσεις μου. Οφείλω επίσης ένα μεγάλο ευχαριστώ στα παιδιά του NTUA-SLP lab που έκαναν τις άπειρες ώρες που περνούσαμε μαζί, ειδικά πριν τα deadlines, πραγματικά ευχάριστες.

Με την παρουσίαση αυτής της διπλωματικής εργασίας, κλείνει επίσημα ένας σημαντικός κύκλος της ζωής μου και ανοίγεται ένας νέος. Ευχαριστώ βαθιά τους γονείς μου, οι οποίοι πάντα με υποστήριζαν και με βοηθούσαν σε όλες τις αποφάσεις μου. Τους ευχαριστώ ιδιαίτερα διότι μου εμφύσησαν την αγάπη για τη γνώση και τις αξίες τους.

Θέλω επίσης να ευχαριστήσω τον Ανδρέα για την αδιάλειπτη παρουσία του σε όλες τις όμορφες και δυσάρεστες στιγμές, την εμπιστοσύνη του στις δυνατότητές μου και τη συνεχή παρότρυνσή του που με ενέπνεε να ακολουθήσω τους στόχους μου.

Τέλος, ευχαριστώ τους φίλους μου. Καθώς μεγαλώναμε μαζί, κάναμε ατελείωτες συζητήσεις για το μέλλον, μοιραστήκαμε τις φιλοδοξίες μας και ζήσαμε αξέχαστες στιγμές. Αισθάνομαι τυχερή και ευγνώμων για αυτά τα 6 (με κάποιους, ακόμα περισσότερα) χρόνια και ανυπομονώ να δω τη συνέχεια.

Αλεξάνδρα Χρονοπούλου,

Αθήνα, 20η Μαρτίου 2019

Περιεχόμενα

Κατάλογος πινάκων

Κατάλογος σχημάτων

Κεφάλαιο 1

Εισαγωγή

1.1 Μεταφορά Μάθησης (Transfer Learning)

Στο πεδίο της επεξεργασίας φυσικής γλώσσας (natural language processing - NLP), τα βαθιά νευρωνικά δίκτυα έχουν βελτιώσει την απόδοση των μοντέλων σε πολλά διαφορετικά προβλήματα (tasks). Ωστόσο, η εκμάθηση ενός καινούργιου μοντέλου για κάθε διαφορετικό πρόβλημα απαιτεί πληθώρα δεδομένων με ετικέτες. Για να επιτύχουν ικανοποιητικά αποτελέσματα, τα μοντέλα αυτά τυπικά χρειάζεται να εκπαιδευτούν σε εκατομμύρια δεδομένα, με ειδικές ετικέτες για κάθε υποπρόβλημα, όπως συντακτική ανάλυση [9], αυτόματη ερωτηση - απάντηση (question answering) [10, 11] και μηχανική μετάφραση [12, 13].

Ωστόσο σε πολλές πρακτικές εφαρμογές υπάρχει μικρή διαθεσιμότητα δεδομένων με ετικέτες. Σε αυτές τις περιπτώσεις η μεταφορά μάθησης προσφέρει εναλλακτική λύση, αξιοποιώντας τη γνώση που έχει αποκτήσει ένα μοντέλο επιλύοντας ένα πρόβλημα, για να αντιμετωπίσει ένα διαφορετικό, αλλά παρεμφερές, πρόβλημα. Η μεταφορά μάθησης έχει επιφέρει σημαντική πρόοδο στη βαθιά μάθηση (deep learning) [14, 8], καθώς επιτρέπει την εκπαίδευση δικτύων σε συνθήκες όπου περιορισμένα δεδομένα είναι διαθέσιμα στο σύνολο εκπαίδευσης. Η μεταφορά μάθησης συνήθως οδηγεί σε γρηγορότερη σύγκλιση και υψηλότερη απόδοση από αυτήν που θα είχε το μοντέλο, αν είχε εκπαιδευτεί μόνο σε ένα μικρό σύνολο δεδομένων. Επιπλέον, βελτιώνει τη δυνατότητα γενίκευσης, καθώς προεκπαιδευμένα μοντέλα έχουν ηθελημένα εκπαιδευτεί σε προβλήματα που επιβάλλουν στο μοντέλο να εξάγει γενικά χαρακτηριστικά, τα οποία είναι χρήσιμα σε συναφή περιεχόμενα. Επομένως, όταν το μοντέλο μεταφέρεται σε ένα νέο πρόβλημα, είναι λιγότερο πιθανό το πρόβλημα του “overfitting” στο νέο σύνολο δεδομένων εκπαίδευσης. Αυτή η γενική προ-εκπαιδευμένη αναπαράσταση χαρακτηριστικών είναι εξαιρετικά μεγάλης σημασίας στην επεξεργασία φυσικής γλώσσας (ΕΦΓ). Η μεταφορά μάθησης αντιμετωπίζει την ανάγκη για εκτενείς υπολογιστικούς πόρους και χρονοβόρα εκπαίδευση του μοντέλου.

Μια σχετική κατεύθυνση είναι αυτή της μάθησης πολλαπλών εργασιών (multi-task learning). Μέσω της ταυτόχρονης μάθησης πολλών προβλημάτων, το μοντέλο αξιοποιεί τις ομοιότητες διαφορετικών προβλημάτων, ούτως ώστε να έχει καλύτερη απόδοση σε όλα. Στη βαθιά μάθηση, συνήθως θέλουμε να μάθουμε μια αναπαράσταση χαρακτηριστικών η οποία περιέχει πολλή πληροφορία και να την χρησιμοποιήσουμε για να κάνουμε μια πρόβλεψη. Χρησιμοποιώντας το multi-task learning, μπορούμε να βελτιώσουμε τα αποτελέσματα ενός μοντέλου κατά πολύ, επιβάλλοντας του να μάθει γενικές αναπαραστάσεις. Με τον τρόπο αυτό, επιτυγχάνουμε μία τεχνητή αύξηση των δεδομένων εκπαίδευσης (implicit data augmentation), καθώς η αναπαράσταση των χαρακτηριστικών είναι προϊόν ταυτόχρονης μάθησης πάνω σε πολλά διαφορετικά σύνολα δεδομένων. Στην επεξεργασία φυσικής γλώσσας, το multi-task learning χρησιμοποιείται σε πληθώρα εφαρμογών, που περιλαμβάνουν τη μηχανική μετάφραση [15, 16, 17], τη σημασιολογική ανάλυση [18, 19, 20, 21], sequence labeling [22] και την επισημείωση μερών του λόγου [23].

1.2 Γλωσσικά Μοντέλα (Language Models)

Ο στόχος κάθε μοντέλου μεταφοράς μάθησης είναι να δημιουργήσει γενικές αναπαραστάσεις χαρακτηριστικών επιλύοντας ένα πρόβλημα και να χρησιμοποιήσει αυτή τη γνώση για να αντιμετωπίσει

ένα σχετικό πρόβλημα. Ένα πρόβλημα που είναι κατάλληλο ως μοντέλο προ-εκπαίδευσης είναι η δημιουργία ενός γλωσσικού μοντέλου.

Η δημιουργία γλωσσικών μοντέλων αποτελεί βασικό αντικείμενο της επεξεργασίας φυσικής γλώσσας, το οποίο δημιουργεί μια πιθανοτική κατανομή σε μια ακολουθία λέξεων και προσφέρει μια μοναδική αναπαράσταση κάθε εμφάνισης μιας λέξης, με βάση το περιεχόμενο στο οποίο βρίσκεται. Ένα γλωσσικό μοντέλο είναι, επομένως, ικανό να κωδικοποιεί τις λεπτές αποχρώσεις της γλώσσας, όπως και να μοντελοποιεί τη σύνταξη και τη σημασιολογία. Ένα γλωσσικό μοντέλο, επίσης, επιτρέπει την απόκτηση αναπαραστάσεων υψηλού επιπέδου και αποτελεί ένα αρχικό μοντέλο με χρήσιμη πληροφορία, το οποίο μπορεί να μεταφερθεί σε διάφορα άλλα προβλήματα της επεξεργασίας φυσικής γλώσσας.

Ουσιαστικά, ένα γλωσσικό μοντέλο λαμβάνει μία-μία κάθε λέξη μιας ακολουθίας ως είσοδο και προσπαθεί να προβλέψει την επόμενη λέξη. Ως αποτέλεσμα, δεν απαιτεί δεδομένα εκπαίδευσης με ετικέτες, τα οποία είναι δυσεύρετα. Το απλό κείμενο, ωστόσο, είναι διαθέσιμο σε μεγάλες ποσότητες για κάθε πιθανό τομέα (domain). Τα γλωσσικά μοντέλα μπορούν, επομένως, να εκπαιδευτούν σε πληθώρα δεδομένων που είναι διαθέσιμα δωρεάν. Λόγω της ικανότητάς του γλωσσικού μοντέλου να εξάγει γενικές αναπαραστάσεις λέξεων με βάση το περιεχόμενο και την μεγάλη διαθεσιμότητα δεδομένων χωρίς ετικέτες, αποτελεί πλέον ένα μοντέλο που χρησιμοποιείται ευρέως και βελτιώνει τα αποτελέσματα σε πολλά προβλήματα της επεξεργασίας φυσικής γλώσσας [24, 25, 26, 27, 28]. Τα γλωσσικά μοντέλα έχουν βελτιώσει εμφανώς τα αποτελέσματα στην εξαγωγή συμπεράσματος (natural language inference) [29], αναγνώριση οντοτήτων [30], SQuAD ερωταπόκριση [31] και ανάλυση συναισθήματος [32].

1.3 Αναγνώριση Συναισθημάτων & Ανάλυση Συναισθήματος

Η αναγνώριση συναισθήματος (emotion recognition) στο NLP είναι η διαδικασία αναγνώρισης διακριτών συναισθημάτων που εκφράζονται στο γραπτό λόγο. Η ανάλυσή τους μπορεί να θεωρηθεί ως η φυσική εξέλιξη της ανάλυσης συναισθήματος (sentiment analysis) και ένα πιο λεπτομερές μοντέλο. Χιλιάδες άρθρα έχουν γραφτεί για τις μεθόδους και εφαρμογές της αυτόματης ανάλυσης συναισθήματος. Πρόκειται, λοιπόν, για ένα σημαντικό πεδίο του NLP. Έχει φανεί εξαιρετικά χρήσιμη σε διάφορες εφαρμογές στο χώρο του μάρκετινγκ, της διαφήμισης [33, 34], των συστημάτων αυτόματης ερώτησης-απάντησης [35].

Η ανάγκη να καταλάβουμε τα συναισθήματα είναι εμφανής σε κάθε κοινωνία. Παραδείγματα εφαρμογών της αναγνώρισης συναισθημάτων μπορούν να βρεθούν στις πολιτικές επιστήμες [36], την ψυχολογία, το μάρκετινγκ [37], την επαφή ανθρώπου-υπολογιστή [38] και σε πολλές ακόμη εφαρμογές. Στο μάρκετινγκ, η αναγνώριση συναισθημάτων μπορεί να χρησιμοποιηθεί για την ανάλυση των αντιδράσεων των πελατών μιας εταιρείας σε αλλαγές προϊόντων και υπηρεσιών, για να βοηθήσει στην επιλογή του προϊόντος που πρέπει να αλλαχθεί για να βελτιώσει την άποψη των πελατών, ούτως ώστε να ενισχύσει το συναίσθημα ικανοποίησης των πελατών [39]. Η αναγνώριση συναισθήματος, επίσης, μπορεί να χρησιμοποιηθεί στην αλληλεπίδραση ανθρώπου-υπολογιστή και στα συστήματα σύστασης (recommender systems) για να παράξει αλληλεπιδράσεις ή συστάσεις βασισμένη στην συναισθηματική κατάσταση του χρήστη [40]. Αντιλαμβανόμενοι το σημαντικό ρόλο των συναισθημάτων στη διαδικασία λήψης αποφάσεων των ανθρώπων, μπορούμε να χρησιμοποιήσουμε την αυτόματη αναγνώριση συναισθημάτων προς όφελος οποιασδήποτε εταιρείας ή οργανισμού, για την αντιμετώπιση για παράδειγμα φυσικών καταστροφών. Επίσης, είναι απαραίτητη για να δημιουργήσουμε καλύτερα εργαλεία που βασίζονται στην τεχνητή νοημοσύνη, όπως chatbots.

Στο πλαίσιο της διατριβής, θα μελετήσουμε τεχνικές μεταφοράς μάθησης και multi-task learning για την επεξεργασία φυσικής γλώσσας. Προτείνουμε μια καινοτόμα μέθοδο, η οποία συνδυάζει τη μεταφορά μάθησης και το multi-task learning. Συγκεκριμένα, βασισμένοι στο ότι οι προ-εκπαιδευμένες αναπαραστάσεις από μη επιβλεπόμενα προβλήματα είναι χρησίμες για επιβλεπόμενα προβλήματα με λίγα δεδομένα εκπαίδευσης, προτείνουμε μία μέθοδο που αξιοποιεί αυτές τις προ-εκπαιδευμένες αναπαραστάσεις χαρακτηριστικών για να αντιμετωπίσει διαφορετικά προβλήματα. Προ-εκπαιδεύουμε

ένα γλωσσικό μοντέλο σε ένα γενικό σύνολο δεδομένων από το Twitter και το μεταφέρουμε σε διάφορα προβλήματα, που υπάγονται στις περιοχές της αναγνώρισης συναισθημάτων, ανάλυσης συναισθήματος, αναγνώριση σαρκασμού και ειρωνείας κλπ. Επίσης, εισάγουμε μια βοηθητική συνάρτηση κόστους από το γλωσσικό μοντέλο, για να επιτρέψουμε στο μοντέλο μας να δημιουργήσει πιο γενιές αναπαραστάσεις και να αποφύγει το overfitting. Συνδυάζοντας τεχνικές μεταφοράς μάθησης με τη συγκεκριμένη βοηθητική συνάρτηση του γλωσσικού μοντέλου, προτείνουμε ένα μοντέλο ικανό να ανταποκριθεί σε πλήθος προβλημάτων με ανταγωνιστικά αποτελέσματα.

1.4 Διάρθρωση Διπλωματικής Εργασίας

Η εργασία διαρθρώνεται ως εξής. Στο κεφάλαιο 2 παρουσιάζεται το θεωρητικό υπόβαθρο της μηχανικής μάθησης. Συγκεκριμένα, παρουσιάζονται οι βασικές έννοιες της μηχανικής μάθησης και έπειτα κλασσικές μέθοδοι ταξινόμησης και παλινδρόμησης παρουσιάζονται, όπως και σύγχρονα μοντέλα βασισμένα στα νευρωνικά δίκτυα, και ειδικά τα αναδρομικά νευρωνικά δίκτυα (recurrent neural networks - RNNs) και long short-term memory units (LSTMs). Στη συνέχεια, οι έννοιες της μεταφοράς μάθησης και του multi-task learning παρουσιάζονται, όπως επίσης και το θεωρητικό κίνητρο για τη χρήση τους. Στο κεφάλαιο 3 παρουσιάζεται το θεωρητικό υπόβαθρο της επεξεργασία φυσικής γλώσσας που απαιτείται για την κατανόηση της εργασίας. Αφού παρουσιαστούν δημοφιλή προβλήματα της επεξεργασίας φυσικής γλώσσας, παρουσιάζεται η έννοια του γλωσσικά μοντέλου, αρχικά στη μορφή n-γραμμάτων βασισμένα στη Μαρκοβιανή υπόθεση και έπειτα στη μορφή ενός αναδρομικού νευρωνικού δικτύου (RNN). Έπειτα, αναλύονται μέθοδοι μεταφοράς μάθησης που χρησιμοποιούνται σήμερα για να εκπαιδεύσουν μοντέλα στην επεξεργασία φυσικής γλώσσας. Κατόπιν, στο κεφάλαιο 4, αντιμετωπίζουμε ένα πρόβλημα αναγνώρισης συναισθημάτων και εξηγούμε τη μεθοδολογία και τα μοντέλα μεταφοράς μάθησης που υλοποιήθηκαν για να το επιλύσουμε. Αφού παρουσιαστεί η αρχιτεκτονική του μοντέλου, παρουσιάζονται τα πειραματικά αποτελέσματα και οι μελλοντικές προεκτάσεις του μοντέλου. Στη συνέχεια, στο κεφάλαιο 5, προτείνουμε μια προσέγγιση βαθιάς μάθησης βασισμένη σε προ-εκπαιδευμένες αναπαραστάσεις που μοντελοποιούνται από ένα γλωσσικό μοντέλο, το οποίο αντιμετωπίζει το πρόβλημα του catastrophic forgetting, προσθέτοντας μια βοηθητική συνάρτηση κόστους του γλωσσικού μοντέλου στο τελικό μοντέλο ταξινόμησης. Αξιοποιούμε αφηρημένες αναπαραστάσεις χαρακτηριστικών από γλωσσικά μοντέλα και τα χρησιμοποιούμε για να επιτύχουμε αξιόλογα αποτελέσματα σε προβλήματα ταξινόμησης. Έπειτα, τα αποτελέσματα των πειραμάτων μας σε 5 διαφορετικά σύνολα δεδομένων παρουσιάζονται. Τελικά, στο κεφάλαιο 6, παρουσιάζονται τα συμπεράσματα και προτείνονται μελλοντικές ιδέες ως προεκτάσεις αυτής της εργασίας.

Κεφάλαιο 2

Θεωρητικό Υπόβαθρο Μηχανικής Μάθησης

2.1 Ορισμός Μεθόδων Μηχανικής Μάθησης

Η Μηχανική Μάθηση (Machine learning - ML) είναι ένα πεδίο της Τεχνητής Νοημοσύνης (Artificial Intelligence - AI) που δημιουργεί συστήματα με την ικανότητα να μαθαίνουν αυτόματα και να βελτιώνονται χωρίς να είναι ρητά προγραμματισμένα για τον υπολογισμό ή την λύση προβλημάτων. Οι αλγόριθμοι του ML επιτρέπουν στους υπολογιστές να εκπαιδεύονται πάνω στα δεδομένα εισόδου και χρησιμοποιούν στατιστική ανάλυση για να εξάγουν τιμές οι οποίες εμπίπτουν σε ένα συγκεκριμένο εύρος. Η διαδικασία της μάθησης ξεκινά με παρατηρήσεις, που αποτελούν παραδείγματα, ή εμπειρικά αποτελέσματα ή οδηγίες, ούτως ώστε να αναγνωριστούν πρότυπα στα δεδομένα και να ληφθούν καλύτερες αποφάσεις στο μέλλον, με βάση τα παραδείγματα που διαθέτουμε. Ο πρωταρχικός σκοπός είναι να επιτρέψουμε στους υπολογιστές να μαθαίνουν αυτόματα, χωρίς ανθρώπινη παρέμβαση ή βοήθεια, και να προσαρμόζουν τις πράξεις τους κατάλληλα.

Στο ML, τα καθήκοντα ταξινομούνται γενικά σε ευρείες κατηγορίες. Οι κατηγορίες αυτές βασίζονται στον τρόπο με τον οποίο λαμβάνεται η μάθηση ή στον τρόπο με τον οποίο δίνεται ανάδραση στην εκμάθηση στο ανεπτυγμένο σύστημα. Δύο από τις πιο ευρέως υιοθετημένες μεθόδους ML είναι η *επιβλεπόμενη μάθηση*, η οποία εκπαιδεύει αλγορίθμους που βασίζονται στα δεδομένα εισόδου και εξόδου τα οποία επισημαίνονται (αποκτούν ετικέτες-labels) από τον άνθρωπο και η *μη επιβλεπόμενη μάθηση*, η οποία παρέχει τον αλγόριθμο χωρίς επισημασμένα δεδομένα, ούτως ώστε να του επιτρέψει να βρει δομή στα δεδομένα εισόδου του.

2.2 Επιβλεπόμενη Μάθηση (Supervised Learning)

Στην επιβλεπόμενη μάθηση, υπάρχουν μεταβλητές εξόδου (x) και μεταβλητή εξόδου (Y). Ο στόχος είναι η εκμάθηση μιας συνάρτησης απεικόνισης (mapping) από την είσοδο στην έξοδο μέσω ενός αλγορίθμου.

$$Y = f(X) \tag{2.1}$$

Ο στόχος είναι να προσεγγίσουμε τη συνάρτηση απεικόνισης τόσο καλά που όταν νέα δεδομένα εισόδου (x) εισάγονται στο μοντέλο, οι αντίστοιχες μεταβλητές εξόδου (Y) να μπορούν να προβλεφθούν με επιτυχία. Μπορούμε να σκεφτούμε τη διαδικασία με την οποία ένας αλγόριθμος μαθαίνει από τα δεδομένα εισόδου, όπως τη διαδικασία κατά την οποία ένας δάσκαλος επιβλέπει τη διαδικασία μάθησης. Οι σωστές απαντήσεις είναι γνωστές (ονομάζονται *ετικέτες - labels*), ο αλγόριθμος κάνει κατ' επανάληψη προβλέψεις στο σύνολο δεδομένων εκπαίδευσης και διορθώνεται από τον δάσκαλο. Η διαδικασία μάθησης σταματά όταν ο αλγόριθμος επιτύχει ένα αποδεκτό επίπεδο απόδοσης.

Τα επιβλεπόμενα προβλήματα μάθησης χωρίζονται περαιτέρω σε προβλήματα *παλινδρόμησης (regression)* και *ταξινόμησης (classification)*.

- Παλινδρόμηση: Το πρόβλημα εκτίμησης ή πρόβλεψης μιας συνεχούς ποσότητας. (το $\boldsymbol{y}$ είναι συνεχές)

- Ταξινόμηση: αφορά την ανάθεση παρατηρήσεων σε διακριτές κατηγορίες, αντί της πρόβλεψης ποσοτήτων με συνεχείς τιμές. Στην πιο απλή περίπτωση, υπάρχουν δύο πιθανές κατηγορίες; αυτή η περίπτωση είναι γνωστή ως δυαδική ταξινόμηση. (το $\boldsymbol{y}$ είναι διακριτό)

Επιβλεπόμενη Μάθηση με Νευρωνικά Δίκτυα
Στην περίπτωση των νευρωνικών δικτύων, περιοριζόμαστε στην αναζήτηση λύσεων σε συγκεκριμένες οικογένειες συναρτήσεων, π.χ. το χώρο όλων των γραμμικών συναρτήσεων με d_{in} εισόδους και d_{out} εξόδους, οι οποίες ονομάζονται *κλάσεις υποθέσεων*. Επιβάλουμε αυτόν τον περιορισμό, διότι η αναζήτηση λύσεων πάνω στον χώρο όλων των πιθανών συναρτήσεων αποτελεί ένα πολύ δύσκολο πρόβλημα. Περιορίζοντας το χώρο υποθέσεων σε μια συγκεκριμένη κλάση υποθέσεων, εισάγουμε στο μοντέλο ένα *inductive bias* (επαγωγική μεροληψία) - ένα σύνολο υποθέσεων για την μορφή της επιθυμητής λύσης. Ταυτόχρονα, διευκολύνουμε κάποιες αποδοτικές διαδικασίες στην αναζήτηση λύσεων. Μία κοινή κλάση υποθέσεων είναι ότι η λύση θα είναι γραμμική συνάρτηση υψηλών διαστασεων, π.χ. συναρτήσεις της μορφής:

$$f(\boldsymbol{x}) = \boldsymbol{x} \cdot \boldsymbol{W} + \boldsymbol{b} \tag{2.2}$$
$$\boldsymbol{x} \in \mathbb{R}^{\boldsymbol{d}_{in}}, \boldsymbol{W} \in \mathbb{R}^{\boldsymbol{d}_{in} \times \boldsymbol{d}_{out}}, \boldsymbol{b} \in \mathbb{R}^{\boldsymbol{d}_{out}}$$

όπου το διάνυσμα $\boldsymbol{x}$ είναι η είσοδος της συνάρτησης, ενώ ο πίνακας $\boldsymbol{W}$ και το διάνυσμα $\boldsymbol{b}$ είναι οι παράμετροι. Ο στόχος της εκμάθησης είναι να θέσει τις τιμές των παραμέτρων $\boldsymbol{W}$ και $\boldsymbol{b}$ έτσι ώστε η συνάρτηση να συμπεριφέρεται όπως είναι επιθυμητό, όταν της δοθούν ως δεδομένα εισόδου τα $\boldsymbol{x}_{1:k} = \boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ και αντίστοιχα επιθυμητά δεδομένα εξόδου $\boldsymbol{y}_{1:k} = \boldsymbol{y}_1, \ldots, \boldsymbol{y}_k$. Αντί λοιπόν να αναζητούμε λύσεις πάνω στο χώρο συναρτήσεων, τις αναζητούμε πάνω στο χώρο των παραμέτρων. Συχνά αναφερόμαστε στις παραμέτρους της συνάρτησης ως Θ. Για τη γραμμική περίπτωση, $\Theta = \boldsymbol{W}, \boldsymbol{b}$.

2.3 Μη Επιβλεπόμενη Μάθηση (Unsupervised Learning)

Σε άλλα προβλήματα του ML, υπάρχει μια διαφορετική τάξη καθηκόντων που αναφέρεται ως μη επιβλεπόμενη μάθηση. Στα προβλήματα αυτής της κατηγορίας, τα δεδομένα εκπαίδευσης είναι διανύσματα $\boldsymbol{x}$ τα οποία δεν έχουν αντίστοιχες ετικέτες. Επομένως, ο στόχος τη μη επιβλεπόμενης μάθησης είναι να βρίσκει μοτίβα όταν δεν υπάρχουν "σωστές απαντήσεις", ή όταν αυτές είναι αδύνατον να υπολογιστούν. Μία μεγάλη υποκατηγορία μη επιβλεπόμενων καθηκόντων είναι το πρόβλημα της ομαδοποίησης (clustering). Η ομαδοποίηση αναφέρεται στην ομαδοποίηση παρατηρήσεων με τέτοιο τρόπο ούτως ώστε τα μέλη μιας κοινής ομάδας να είναι παρόμοια το ένα με το άλλο, και να διαφέρουν σημαντικά από τα μέλη των άλλων ομάδων. Μια άλλη πολύ ενδιαφέρουσα κατηγορία μη επιβλεπόμενων καθηκόντων είναι οι τα *γεννητικά μοντέλα* (generative models). Τα μοντέλα αυτά μιμούνται τη διαδικασία δημιουργίας των δεδομένων εκπαίδευσης. Ένα καλό γεννητικό μοντέλο θα πρέπει να μπορεί να δημιουργήσει νέα δεδομένα τα οποία, αν και είναι τεχνητά, μοιάζουν με τα αυθεντικά. Αυτός ο τρόπος μάθησης είναι μη επιβλεπόμενος διότι η διαδικασία με την οποία δημιουργούνται ("γεννιούνται") τα δεδομένα δεν είναι άμεσα παρατηρήσιμη - μόνο τα ίδια τα δεδομένα είναι παρατηρήσιμα.

2.4 Παραδοσιακά Μοντέλα Μηχανικής Μάθησης

2.4.1 Συνάρτηση Κόστους (Loss Function)

Ο στόχος κάθε αλγορίθμου επιβλεπόμενης μάθησης είναι να επιστρέψει μια συνάρτηση $f()$ η οποία αντιστοιχίζει με ακρίβεια τα παραδείγματα εισόδου στις επιθυμητές ετικέτες, π.χ., μια συνάρτηση $f()$ τέτοια ώστε οι προβλέψεις $\hat{\boldsymbol{y}} = f(x)$ στα δεδομένα εκπαίδευσης να είναι ακριβείς. Για να γίνουμε πιο συγκεκριμένοι, εισάγουμε εδώ την έννοια μιας *συνάρτησης κόστους*, η οποία ποσοτικοποιεί την απώλεια (το σφάλμα) του μοντέλου που προβλέπει $\hat{\boldsymbol{y}}$ όταν η πραγματική ετικέτα είναι

$\boldsymbol{y}$. Τυπικά, η συνάρτηση κόστους $L(\hat{\boldsymbol{y}}, \boldsymbol{y})$ αναθέτει μαι αριθμητική τιμή (βαθμωτή) στην προβλεπόμενη έξοδο $\hat{\boldsymbol{y}}$ δεδομένης της πραγματικής προβλεπόμενης εξόδου $\boldsymbol{y}$. Η συνάρτηση κόστους πρέπει να είναι κάτω φραγμένη, με την ελάχιστη τιμή να επιτυγχάνεται στις περιπτώσεις όπου η πρόβλεψη είναι σωστή. Οι παράμετροι της συνάρτησης που έχει μάθει το δίκτυο θέτονται μετά με τρόπο τέτοιο ώστε να ελαχιστοποιούν την απώλεια L στα παραδείγματα εκπαίδευσης (συνήθως, ελαχιστοποιούμε το άθροισμα των απωλειών όλων των διαφορετικών παραδειγμάτων εκπαίδευσης).

Δεδομένου ενός επισημασμένου συνόλου εκπαίδευσης $(\boldsymbol{x_{1:n}}, \boldsymbol{y_{1:n}})$, μία συνάρτηση κόστους ανά δείγμα L και μια παραμετροποιημένη συνάρτηση $f(\boldsymbol{x}; \Theta)$, ορίζουμε τη συνολική απώλεια πάνω το σύνολο δεδομένων σε σχέση με τις παραμέτρους Θ ως τη μέση απώλεια πάνω σε όλα τα δεδομένα εκπαίδευσης:

$$\mathcal{L}(\Theta) = -\frac{1}{N}\sum_{i=1}^{N} L(f(\boldsymbol{x_i}; \Theta), \boldsymbol{y_i}) \tag{2.3}$$

Τα παραδείγματα εισόδου έχουν σταθερές τιμές, και οι τιμές των παραμέτρων καθορίζουν την απώλεια. Ο στόχος του αλγόριθμου μάθησης είναι να δώσει τέτοιες τιμές στις παραμέτρους Θ, ούτως ώστε η τιμή του $\mathcal{L}$ να ελαχιστοποιηθεί:

$$\hat{\Theta} = \arg\min_{\Theta} \mathcal{L}(\Theta) = \arg\min_{\Theta} \frac{1}{N}\sum_{i=1}^{N} L(f(\boldsymbol{x_i}; \Theta), \boldsymbol{y_i}). \tag{2.4}$$

Τώρα θα ορίσουμε την έννοια της *εντροπίας*. Ας υποθέσουμε ότι θέλουμε να επικοινωνήσουμε ένα σύνολο n γεγονότων από μια συγκεκριμένη κατανομή πιθανότητας p. Η εντροπία πληροφορίας είναι το μέσο ελάχιστο μέγεθος κωδικοποίησης της πληροφορίας ώστε να επικοινωνήσουμε τα γεγονότα [41]:

$$H(p) = \sum_{x} p(x) log \frac{1}{p(x)} \tag{2.5}$$

Αν η τιμή της εντροπίας είναι υψηλή (το μέγεθος κωδικοποίησης είναι κατά μέσο όρο μεγάλο), σημαίνει πως έχουμε πολλά δεδομένα εισόδου με μικρή πιθανότητα. Η εντροπία μπορεί να θεωρηθεί ένας τρόπος μέτρησης της αβεβαιότητας, εκτός από τρόπος μέτρησης της ποσότητας της πληροφορίας.

Η *cross-εντροπία* αποτελεί το μέσο ελάχιστο μέγεθος κωδικοποίησης της πληροφορίας του να επικοινωνήσουμε ένα γεγονός από μία κατανομή πιθανότητας σε μία άλλη. Ορίζεται ως:

$$H_p(q) = \sum_{x} q(x) log\left(\frac{1}{p(x)}\right) \tag{2.6}$$

Στη δυαδική περίπτωση (δύο κατηγορίες), ορίζουμε τη *δυαδική εντροπία*. Πρόκειται για την εντροπία μιας διαδικασίας Bernoulli με πιθανότητα p που μπορεί να πάρει δύο τιμές. Έστω X η τυχαία μεταβλητή που που μπορεί να πάρει μόνο δύο τιμές, 0 και 1. Αν η $probability(X = 1) = p$, τότε η $probability(X = 0) = 1 - p$ και η εντροπία ορίζεται ως:

$$\begin{aligned} H(X) &= plog\frac{1}{p} + (1-p)log\frac{1}{1-p} \\ &= -plogp - (1-p)log(1-p) \end{aligned} \tag{2.7}$$

Η *δυαδική συνάρτηση κόστους cross-εντροπίας* χρησιμοποιείται στη δυαδική ταξινόμηση με δεσμευμένες πιθανότητες εξόδου. Υποθέτουμε ότι έχουμε ένα σύνολο δεδομένων με δύο κλάσεις που έχουν τις ετικέτες 0 και 1, με τη σωστή ετικέτα $y \in \{0, 1\}$. Η έξοδος του ταξινομητή $\tilde{y}$ μετασχηματίζεται με χρήση της σιγμοειδούς (λέγεται και *λογιστική*) συνάρτηση $\sigma(x) = \frac{1}{1+e^{-x}}$ στο διάστημα [0, 1], και σχηματίζει την δεσμευμένη πιθανότητα $\tilde{y} = \sigma(\tilde{y}) = P(y = 1|x)$. Ο κανόνας πρόβλεψης είναι:

$$\text{πρόβλεψη} = \begin{cases} 0, & \text{εάν } \hat{y} < 0.5 \\ 1, & \text{εάν } \hat{y} \geq 0.5. \end{cases}$$

Το δίκτυο εκπαιδεύεται για να μεγιστοποιήσει το λογάριθμο της δεσμευμένης πιθανότητας $\log P(y = 1|\boldsymbol{x})$ για κάθε παράδειγμα του συνόλου εκπαίδευσης ($\boldsymbol{x}$, y). Η λογιστική συνάρτηση κόστους ορίζεται ως:

$$L_{\text{logistic}}(\hat{y}, y) = -y\log\hat{y} - (1-y)\log(1-\hat{y}). \tag{2.8}$$

Όταν χρησιμοποιούμε λογιστική συνάρτηση κόστους, υποθέτουμε ότι το εξωτερικό επίπεδο μετασχηματίζεται με χρήση τη σιγμοειδούς συνάρτησης.

Η *διακριτή συνάρτηση κόστους cross-εντροπίας* (αναφέρεται και ως η *αρνητική λογιστική συνάρτηση πιθανοφάνειας* (*negative log likelihood*) χρησιμοποιείται όταν είναι επιθυμητή μια πιθανοτική ερμηνεία των αποτελεσμάτων. Έστω $\boldsymbol{y} = \boldsymbol{y}_{[\mathbf{1}]}, \ldots, \boldsymbol{y}_{[\boldsymbol{n}]}$ ένα διάνυσμα που αναπαριστά την πραγματική multinomial κατανομή στις ετικέτες $\mathbf{1}, \cdots, \boldsymbol{n}$, και έστω $\hat{\boldsymbol{y}} = \hat{\boldsymbol{y}}_{[\mathbf{1}]}, \ldots, \hat{\boldsymbol{y}}_{[\boldsymbol{n}]}$ η έξοδος του γραμμικού ταξινομητή, η οποία μετασχηματίζεται από τη συνάρτηση softmax και αναπαριστά την δεσμευμένη κατανομή του να ανήκει στην κλάση ένα δείγμα στην κλάση i, $\hat{\boldsymbol{y}}_{[\boldsymbol{i}]} = P(y = 1|\boldsymbol{x})$. Τότε, η διακριτή συνάρτηση κόστους cross-εντροπίας για το n-οστό δείγμα είναι:

$$L_{\text{cross-entropy}}(\hat{\boldsymbol{y}}_{\boldsymbol{i}}, \boldsymbol{y}_{\boldsymbol{i}}) = -\boldsymbol{y}_{[\boldsymbol{i}]}\log(\hat{\boldsymbol{y}}_{[\boldsymbol{i}]}) \tag{2.9}$$

Για να βελτιστοποιήσουμε τις παραμέτρους του μοντέλου μας, θέλουμε να μεγιστοποιήσουμε την πιθανοφάνειά του, ή αλλιώς να ελαχιστοποιήσουμε το μέσο όρο της αρνητικής λογιστικής πιθανοφάνειας όλων των διαθέσιμων N δειγμάτων εκπαίδευσης. Η αντικειμενική συνάρτηση (συνάρτηση κόστους) παίρνει την εξής μορφή:

$$L_{\text{cross-entropy}}(\hat{\boldsymbol{y}}, \boldsymbol{y}) = -\frac{1}{N}\sum_{i=1}^{N} \boldsymbol{y}_{[i]}\log(\hat{\boldsymbol{y}}_{[i]}) \tag{2.10}$$

όπου N ο αριθμός των δειγμάτων εκπαίδευσης.

Κατά την εκπαίδευση νευρωνικών δικτύων (neural networks - NN), η εξίσωση 2.10 είναι ιδιαίτερα χρήσιμη. Η τιμή της συνάρτησης κόστους $L(\hat{\boldsymbol{y}}, \boldsymbol{y})$ επιτρέπει τον υπολογισμό του σφάλματος του NN ως προς τις αποφάσεις ταξινόμησης που έλαβε για τα N δείγματα. Αντί να χρησιμοποιείται ολόκληρο το σύνολο δεδομένων σε κάθε επανάληψη της διαδικασίας εκπαίδευσης, συνήθως υπολογίζουμε το σφάλμα πάνω σε υποσύνολα του συνόλου δεδομένων εκπαίδευσης (που ονομάζονται *batches*). Για να μάθει το NN τις βέλτιστες παραμέτρους, αξιοποιούμε έναν αλγόριθμο βελτιστοποίησης (optimizer) των παραμέτρων, με βάση τον υπολογισμό του ανάδελτα (gradient) $\nabla_\theta L(\hat{\boldsymbol{y}}, \boldsymbol{y})$ για την εύρεση ενός τοπικού ελάχιστου. Ο πιο δημοφιλής αλγόριθμος για τη βελτιστοποίηση των βαρών του δικτύου είναι αυτός της οπίσθιας διάδοσης (*backpropagation* [42]). Ο αλγόριθμος backpropagation στηρίζεται στον επαναλαμβανόμενο υπολογισμό των μερικών παραγώγων (gradients) κάθε επιπέδου ενός NN σε σχέση με τις παραμέτρους που χρειάζεται να ρυθμιστούν χρησιμοποιώντας τον κανόνα αλυσίδας, για να ελαχιστοποιηθεί η απώλεια. Τα βάρη του δικτύου ενημερώνονται αντίστοιχα. Το σφάλμα που υπολογίζεται από τις μερικές παραγώγους διαμορφώνει το κατά πόσο θα μεταβληθούν τα βάρη.

Στην πραγματικότητα, αυτό που προσπαθούμε να κάνουμε όταν χρησιμοποιούμε τον αλγόριθμο backpropagation είναι να προσεγγίσουμε το τοπικό ελάχιστο ενός μη-γραμμικού προβλήματος ελαχιστοποίησης. Το πρόβλημα αυτό δε μπορεί να λυθεί σε πολυωνυμικό χρόνο από κανέναν αλγόριθμο (ανήκει σε μια κατηγορία προβλημάτων που ονομάζονται NP-προβλήματα).

2.4.2 Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machines - SVMs)

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

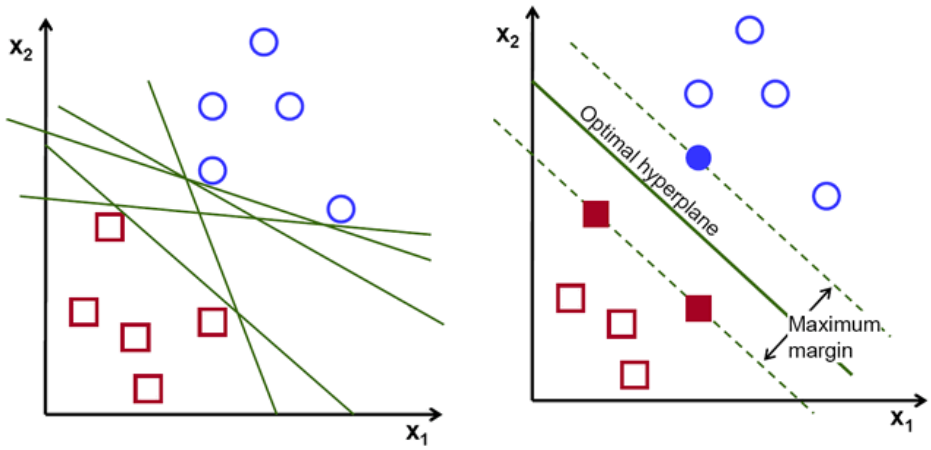
$$f(\boldsymbol{x}) = \boldsymbol{w}^{\boldsymbol{T}} \cdot \boldsymbol{\phi}(\boldsymbol{x}) + b \tag{2.11}$$

όπου $\boldsymbol{\phi}(\boldsymbol{x})$ είναι ένα μετασχηματισμός στο χώρο χαρακτηριστικών και η παράμετρος b (bias) έχει οριστεί. Το σύνολο δεδομένων εκπαίδευσης αποτελείται από N διανύσματα εισόδου $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ με

αντίστοιχες τιμές εξόδου $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N$ όπου $y_i \in \{-1, 1\}$, και νέα δείγματα $\boldsymbol{x}$ ταξινομούνται ανάλογα με το πρόσημο της $f(\boldsymbol{x})$.

Υποθέτουμε ότι τα δεδομένα εκπαίδευσης είναι γραμμικά διαχωρίσιμα στο χώρο χαρακτηριστικών, ούτως ώστε εξ ορισμού υπάρχει τουλάχιστον μία επιλογή παραμέτρων w και b τέτοια ώστε μία συνάρτηση της μορφής 2.11 ικανοποιεί την ανισότητα $f(\boldsymbol{x_i}) > 0$ για δεδομένα που έχουν $y_i = +1$ και $f(\boldsymbol{x_i}) < 0$ για δεδομένα που έχουν $y_i = -1$, έτσι ώστε $y_i f(\boldsymbol{x_i}) > 0$ για όλα τα δεδομένα εκπαίδευσης.

Ένα παράδειγμα δεδομένων εκπαίδευσης φαίνεται στο Σχήμα 2.1a, όπου τα δείγματα που ανήκουν στην πρώτη κατηγορία είναι κόκκινα και τετράγωνα, ενώ τα δείγματα που ανήκουν στην δεύτερη κατηγορία είναι μπλε και κυκλικά.



(a) Πιθανά υπερεπίπεδα διαχωρισμού.

(b) Ένας γραμμικός ταξινομητής μέγιστου περιθωρίου (maximum margin).

Σχήμα 2.1: Παράδειγμα δυαδικού διαχωρισμού δύο γραμμικά διαχωρίσιμων κλάσεων.

Υπάρχει άπειρο πλήθος πιθανών ευθειών που διαχωρίζουν τις δύο κλάσεις. Ο στόχος του αλγορίθμου SVM είναι να βρει τον πιο γενικό ταξινομητή. Με άλλα λόγια, ο αλγόριθμος SVM προσπαθεί να βρει το υπερεπίπεδο για το οποίο η ελάχιστη απόσταση μεταξύ των δύο κλάσεων (περιθώριο - margin) έχει την μέγιστη δυνατή τιμή. Το υπερεπίπεδο που ικανοποιεί την παραπάνω απαίτηση είναι το βέλτιστο υπερεπίπεδο και μπορεί να παρατηρηθεί στο Σχήμα 2.1b.

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

$$\frac{y_i f(\boldsymbol{x}_i)}{||\boldsymbol{w}||} = \frac{y_i(\boldsymbol{w}^T \boldsymbol{\phi}(\boldsymbol{x}_i) + b)}{||\boldsymbol{w}||} \tag{2.12}$$

Το περιθώριο δίνεται από την κάθετη απόσταση στο κοντινότερο σημείο x_n από το σύνολο δεδομένων, και επιθυμούμε να βελτιστοποιήσουμε τις παραμέτρους $\boldsymbol{w}$ και b για να μεγιστοποιήσουμε αυτή την απόσταση. Οπότε, το μέγιστο περιθώριο βρίσκεται λύνοντας την

$$L(\boldsymbol{w}, \mathbf{b}) = \arg\max_{\boldsymbol{w},\mathbf{b}} \left\{ \frac{1}{||\boldsymbol{w}||} \min_i \{ y_i(\boldsymbol{w}^{\mathrm{T}} \phi(\boldsymbol{x}_i) + \mathbf{b}) \} \right\} \tag{2.13}$$

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

$$L(\boldsymbol{w}) = \arg\min_{\boldsymbol{w},\mathbf{b}} \left\{ \frac{1}{2}||\boldsymbol{w}||^2 \right\} \tag{2.14a}$$

$$y_i(\boldsymbol{w}^{\mathrm{T}} \phi(\boldsymbol{x}_i) + \mathbf{b}) \geq 1, \qquad i = 1, \ldots, N. \tag{2.14b}$$

Η λύση της Εξίσωσης 2.14a δίνεται από τους πολλαπλασιαστές Lagrange [43].

2.4.3 Λογιστική Παλινδρόμηση (Logistic Regression - LR)

Σε προβλήματα ταξινόμησης, θέλουμε να καθορίσουμε την πιθανότητα μια παρατήρηση να ανήκει ή όχι σε μια συγκεκριμένη κλάση. Επομένως, επιθυμούμε να εκφράσουμε την πιθανότητα με μια τιμή μεταξύ του 0 και του 1. Ένας απλός αλγόριθμος ταξινόμησης που δημιουργεί τιμές αυτής της μορφης είναι ο ταξινομητής λογιστικής παλινδρόμησης.

Ας υποθέσουμε ότι έχουμε ένα απλό πρόβλημα δυαδικής ταξινόμησης, όπως αυτό που περιγράφηκε νωρίτερα στο ίδιο Κεφάλαιο. Έστω $\boldsymbol{x}_{i=1:N} = \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ τα διανύσματα εισόδου όπου $\boldsymbol{y}_i \in \{0, 1\}$. Η συνάρτηση ενεργοποίησης του LR ταξινομητή καθορίζεται από την εφαρμογή μιας σιγμοειδούς συνάρτησης πάνω στην γραμμική παλινδρόμηση ούτως έστε να λάβουμε την τελική απόφαση ταξινόμησης. Όπως περιγράφηκε στα **SVMs**:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{2.15}$$

Η συνάρτηση ενεργοποίησης της LR για ένα δοσμένο διάνυσμα $\boldsymbol{x}$ ορίζεται ως εξής:

$$h_{\boldsymbol{w}}(\boldsymbol{x}) = \sigma(\boldsymbol{w}^T \boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{w}^T \boldsymbol{x}}} \tag{2.16}$$

Η συνάρτηση κόστους που θέλουμε να ελαχιστοποιηθεί κατά τη διάρκεια της εκπαίδευσης είναι η εξής:

$$\frac{1}{2}||\boldsymbol{w}|| + C \sum_{i=1}^{N} log(exp(-y_i(\boldsymbol{w}^T \boldsymbol{x}_i + \mathbf{b})) + 1) \tag{2.17}$$

όπου C > 0 και b είναι οι συντελεστές που αναπαριστούν την τιμωρία (penalty) των λανθασμένων αποτελεσμάτων ταξινόμησης και την τομή του υπερεπιπέδου αντίστοιχα.

Η LR είναι ιδιαίτερα αποτελεσματική τεχνική, η οποία δεν απαιτεί εκτενείς υπολογιστικούς πόρους ή κανονικοποίηση. Αποτελεί, επομένως, ένα αξιόπιστο σημείο αναφοράς (baseline) για τα περισσότερα προβλήματα στο NLP. Ένα προφανές μειονέκτημα της μεθόδου είναι ότι δε μπορεί να λύσει μη-γραμμικά προβλήματα, καθώς το επίπεδο αποφάσεων είναι γραμμικό. Ένα σύνηθες "κόλπο" που χρησιμοποιείται για να εφαρμοστεί η LR σε ταξινόμηση με πολλές κλάσεις είναι το να θεωρούμε κάθε ζεύγος κλάσεων ως δυαδικό πρόβλημα ταξινόμησης (παίρνοντας ένα-ένα κάθε ζευγάρι κλάσεων), και να εφαρμόζουμε εκεί την λογιστική παλινδρόμηση.

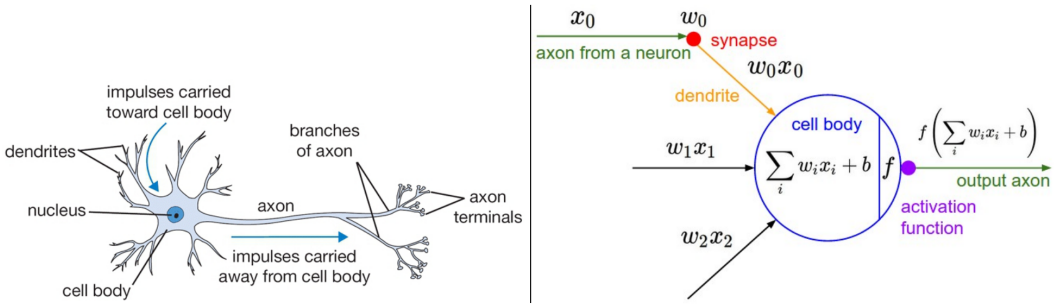
2.5 Βαθιά Νευρωνικά Δίκτυα (Deep Neural Networks)

2.5.1 Εισαγωγή

Η βαθιά μάθηση (deep learning) είναι ένα σύνολο μεθόδων μάθησης που προσπαθούν να μοντελοποιήσουν δεδομένα με περίπλοκες αρχιτεκτονικές συνδυάζοντας διαφορετικούς μη-γραμμικούς μετασχηματισμούς. Τα θεμέλια της βαθιάς μάθησης είναι τα νευρωνικά δίκτυα, τα οποία συνδυάζονται και δημιουργούν τα βαθιά νευρωνικά δίκτυα. Οι τεχνικές αυτές έχουν επιτρέψει σημαντική πρόοδο στα πεδία της αυτόματης επεξεργασίας ήχου και εικόνας, που περιλαμβάνουν την αναγνώριση χαρακτηριστικών προσώπου, την αναγνώριση φωνής, την όραση υπολογιστών, την αυτόματη επεξεργασία φυσικής γλώσσας, την ταξινόμηση κειμένου. Υπάρχει πληθώρα πρακτικών εφαρμογών. Ένα εντυπωσιακό παράδειγμα είναι το πρόγραμμα AlphaGo, το οποίο έμαθε να παίζει το παιχνίδι "go" με μεθόδους βαθιάς μάθησης, κερδίζοντας τον παγκόσμιο πρωταθλητή το 2016.

2.5.2 Τεχνητά Νευρωνικά Δίκτυα

Ένα Τεχνητό Νευρωνικό Δίκτυο (Artificial Neural Network - ANN) είναι ένα υπολογιστικό μοντέλο εμπνευσμένο από τη βιολογία, το οποίο δημιουργεί μοτίβα βασισμένο στη δομή και τη λειτουργία των νευρώνων που υπάρχουν στον ανθρώπινο εγκέφαλο. Το πεδίο των ANNs είχε αρχικά προσπαθήσει να μοντελοποιήσει βιολογικά νευρωνικά συστήματα, αλλά έχει παρεκκλίνει από τότε και έχει γίνει ένα αντικείμενο έρευνας από μηχανικούς. Ο λόγος είναι ότι μέσω των ANNs επιτυγχάνονται καλά αποτελέσματα στα διάφορα προβλήματα μηχανικής μάθησης. Επομένως, θα αναφέρουμε πρώτα εισαγωγικά μια σύντομη περιγραφή των βιολογικών συστημάτων που επηρέασαν κατά μεγάλο βαθμό το ML (βαθιά μάθηση).



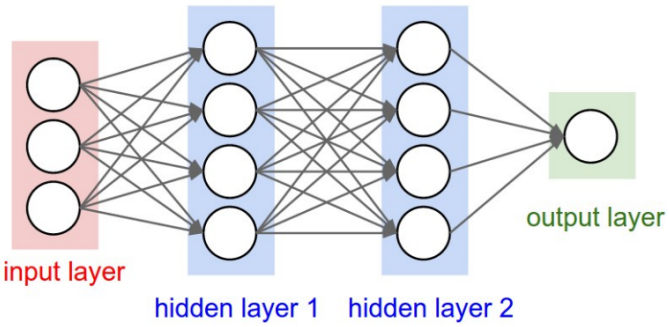
Σχήμα 2.2: Ένας βιολογικός νευρώνας (αριστερά) και ο μαθηματικός του συμβολισμός (δεξιά). Σχήμα από [4].

Η βασική υπολογιστική μονάδα του μυαλού είναι ο *νευρώνας*. Υπάρχουν δισεκατομμύρια νευρώνες στο ανθρώπινο νευρικό σύστημα. Στο Σχήμα 2.2 φαίνεται η σύγκριση μεταξύ ενός βιολογικού νευρώνα και του μαθηματικού του συμβολισμού. Κάθε νευρώνας λαμβάνει σήματα εισόδου από τους *δενδρίτες* και παράγει σήματα εξόδου πάνω στον *άξονά* του. Ο άξονας συνδέει μέσω συνάψεων τους δενδρίτες των νευρώνων. Στο υπολογιστικό μοντέλο του νευρώνα, τα σήματα που ταξιδεύουν κατά μήκος των αξόνων (π.χ. x_0) αλληλεπιδρούν πολλαπλασιαστικά (π.χ. w_0x_0) με τους δενδρίτες του άλλου νευρώνα βάσει της δύναμης της σύναψης (π.χ. w_0). Η ιδέα είναι ότι οι δυνάμεις των συνάψεων (ή αλλιώς βάρη w) μαθαίνονται και ελέγχουν την δύναμη της επιρροής του ενός νευρώνα στον άλλο. Στο βασικό μοντέλο, οι δενδρίτες μεταφέρουν το σήμα στο σώμα του κυττάρου, όπου όλα αθροίζονται. Αν το τελικό άθροισμα έχει τιμή μεγαλύτερη από ένα συγκεκριμένο όριο, ο νευρώνας ενεργοποιείται, στέλνοντας σήμα κατά μήκος του άξονα. Στο υπολογιστικό μοντέλο, υποθέτουμε ότι μας ενδιαφέρει μόνο η συχνότητα που στέλνονται αυτά τα σήματα. Επομένως, μοντελοποιούμε το ρυθμό αποστολής σημάτων του νευρώνα με μια *συνάρτηση ενεργοποίησης* (activation function) f, η οποία αναπαριστά τη συχνότητα αποστολής των σημάτων αυτών στον άξονα. Μία συνηθισμένη επιλογή συνάρτησης ενεργοποίησης είναι η σιγμοειδής συνάρτηση σ, μιας και παίρνει εισόδους με πραγματικές τιμές και τις περιορίζει στο διάστημα $[0, 1]$.

Για να μάθουν πολύπλοκες μη-γραμμικές συναρτήσεις, αρχιτεκτονικές που συνδυάζουν διαφορετικούς τεχνητούς νευρώνες μπορούν να σχεδιαστούν και υλοποιηθούν. Τέτοιες αρχιτεκτονικές ονομάζονται πολυεπίπεδα αντίληπτρα (Multi-Layer Perceptrons - MLPs).

Κάθε νευρωνικό δίκτυο δημιουργείται από τα εξής επίπεδα, όπως φαίνεται και στο Σχήμα 2.3:

- *Επίπεδο εισόδου (Input layer)*: Το επίπεδο αυτό λαμβάνει τα δεδομένα εισόδου. Παρέχει πληροφορίες από τον έξω κόσμο στο δίκτυο χωρίς περαιτέρω υπολογισμούς. Οι κόμβοι απλά περνούν την πληροφορία στο κρυφό επίπεδο.
- *Κρυφά επίπεδα (Hidden layer/s)*: Μέσω αυτού του επιπέδου, η είσοδος υπόκειται σε επεξεργασία και εξάγονται τα χαρακτηριστικά της. Όσο κινούμαστε προς ανώτερα κρυμμένα επίπεδα, εξάγονται χαρακτηριστικά ανώτερου σημασιολογικού περιεχομένου.
- *Επίπεδο εξόδου (Output layer)*: Μετά την επεξεργασία των δεδομένων, λαμβάνεται η απόφαση από το δίκτυο σε αυτό το επίπεδο.



Σχήμα 2.3: Ένα Νευρωνικό Δίκτυο 3 επιπέδων. Σχήμα από [4].

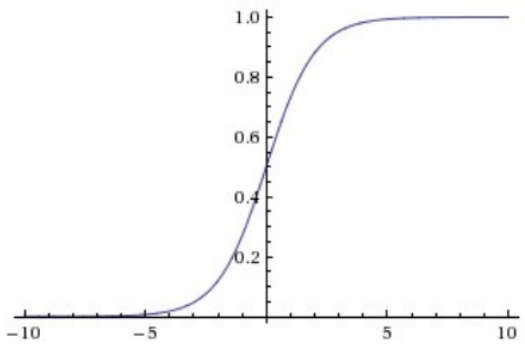
Συνάρτηση Ενεργοποίησης (Activation function)
Οι συναρτήσεις ενεργοποίησης είναι κόμβοι που αποφασίζουν αν ένας νευρώνας πρέπει ή όχι να ενεργοποιηθεί. Αυτό το υπολογίζουν χρησιμοποιώντας μη-γραμμικές συναρτήσεις. Ένα νευρωνικό δίκτυο χρειάζεται να προσθέσει μη-γραμμικές συναρτήσεις για να έχει ακριβή αποτελέσματα. Οι συναρτήσεις που χρησιμοποιούνται συνήθως ως συναρτήσεις ενεργοποίησης είναι:

- *Σιγμοειδής (Sigmoid).*

  Η σιγμοειδής μη-γραμμικότητα έχει τη μαθηματική μορφή:

  $$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{2.18}$$

  παίρνει μια πραγματική τιμή και την περιορίζει στο διάστημα μεταξύ 0 και 1. Η γραφική παράσταση της συνάρτησης φαίνεται στο Σχήμα 2.4.



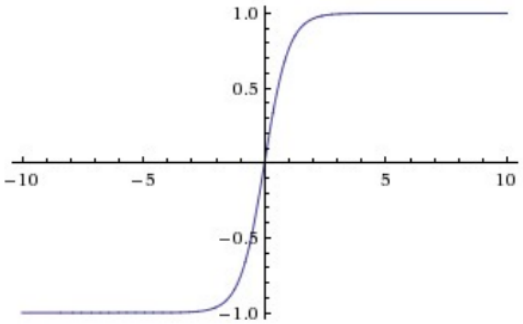
Σχήμα 2.4: Η σιγμοειδής συνάρτηση.

  Ωστόσο, η σιγμοειδής συνάρτηση έχει δύο μεγάλα μειονεκτήματα: 1) όταν είναι πολύ κοντά στο 0 ή στο 1, η τιμή του gradient πλησιάζει το 0. Με άλλα λόγια, το gradient "εξαφανίζεται" και η διαδικασία εκπαίδευσης σταματά. 2) Η έξοδος της σιγμοειδούς δεν είναι κεντραρισμένη στο 0. Οπότε, καθώς τα δεδομένα που έρχονται στο νευρώνα έχουν πάντα θετικές τιμές, το gradient των βαρών θα είναι είτε πάντα θετικό, ή πάντα αρνητικό και μια ανεπιθύμητη εναλλαγή τους εισάγεται στο δίκτυο.

- *Υπερβολική εφαπτομένη (Hyperbolic tangent - tanh).*

  Η συνάρτηση tanh περιορίζει την τιμή ενός πραγματικού αριθμού στο διάστημα [-1, 1], όπως φαίνεται στο Σχήμα 2.5. Το πλεονέκτημα αυτής της συνάρτησης είναι ότι οι τιμές της tanh είναι κεντραρισμένες στο 0, το οποίο βοηθά τον επόμενο νευρώνα στη διαδικασία της διάδοσης. Η συνάρτηση ενεργοποίησης tanh είναι η εξής:

  $$f(x) = tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{2.19}$$



Σχήμα 2.5: Η συνάρτηση tanh.

Η συνάρτηση tanh είναι απλά μια κλιμακωμένη σιγμοειδής, όπου $tanh(x) = 2\sigma(2x) - 1$. Η tanh προφανώς έχει κέντρο το 0. Παρόλα αυτά, το πρόβλημα απότομης μείωσης του gradient εξακολουθεί να υπάρχει.

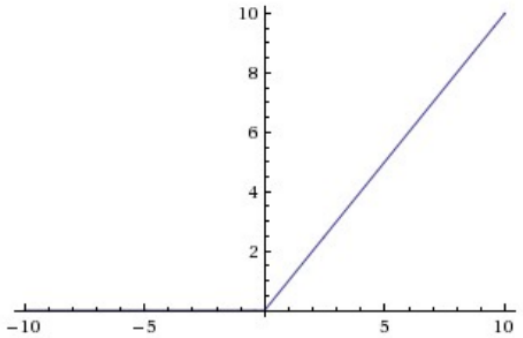
- *Rectified Linear Unit (ReLU).*

  Η ReLU είναι μία από τις πιο ευρέως χρησιμοποιούμενες συναρτήσεις ενεργοποίησης. Έχει την εξής μορφή:

  $$f(x) = x^+ = \max(0, x) \tag{2.20}$$

  Με άλλα λόγια, η συνάρτηση ενεργοποίησης έχει όριο το 0.

  Η ReLU δεν έχει υπολογιστικά ακριβές πράξεις (όπως εκθέτες) και επιπλέον συγκλίνει γρηγορότερα. Ταυτόχρονα, έχει χαμηλή πυκνότητα (sparsity), γεγονός που είναι επιθυμητό, καθώς για κάθε αρνητική είσοδο, η συνάρτηση δεν ενεργοποιείται. Αυτό σημαίνει πως μόνο νευρώνες που πιθανόν επεξεργάζονται ενδιαφέρουσες πτυχές του προβλήματος ενεργοποιούνται. Αποφεύγει επίσης το πρόβλημα κορεσμού (απότομης μείωσης), λόγω της γραμμικής μορφής της. Ωστόσο, η ReLU μπορεί εύκολα να κάνει τους νευρώνες να "κολλήσουν". Αν τα gradients έχουν αρνητικές τιμές, τότε η έξοδος είναι πάντα 0 και ο νευρώνας νεκρώνεται.



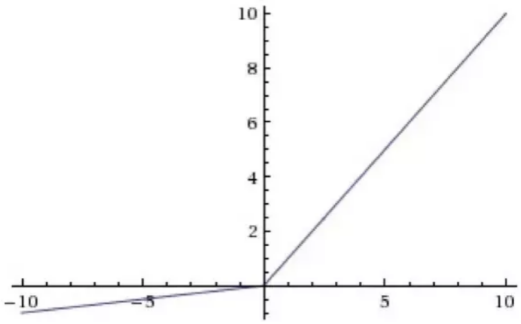
Σχήμα 2.6: Η συνάρτηση ReLU

- *Leaky ReLU.*

  Αντί η συνάρτηση να μηδενίζεται όταν x < 0, η leaky ReLU ξεπερνάει το μειονέκτημα της ReLU, καθώς επιτρέπει μια μικρή αρνητική τιμή ξατά τη διάρκεια της οπίσθιας διάδοσης. Αυτή η λειτουργία περιγράφεται από την εξής σχέση:

$$\text{f(x)} = \begin{cases} x, & \text{αν } x > 0 \\ ax, & \text{διαφορετικά} \end{cases}$$

Όπου a μια σταθερά με μικρή τιμή (ενδεικτικά, $a = 0.01$). Οπότε, η ReLU αντιμετωπίζει το πρόβλημα της "νεκρής ReLU" (dying ReLU).



Σχήμα 2.7: Η συνάρτηση leaky ReLU.

Κανονικοποίηση (Regularization)
Η Εξίσωση 2.4 προσπαθεί να ελαχιστοποιήσει την απώλεια και μπορεί να οδηγήσει σε *overfitting* στα δεδομένα εκπαίδευσης. Το overfitting είναι πιθανό να συμβεί όταν έχουμε ένα τόσο πλούσιο (περίπλοκο) μοντέλο, που δεν πλησιάζουμε απλώς την επιθυμητή συνάρτηση, αλλά μοντελοποιούμε και τον θόρυβο. Αυτό οδηγεί, φυσικά, σε ένα μοντέλο που δε μπορεί να γενικεύσει στα δεδομένα ελέγχου.

Η κανονικοποίηση αποτελεί ένα τρόπο αντιμετώπισης αυτής της ανεπιθύμητης συμπεριφοράς. Για να αντιμετωπίσουμε πιθανή απώλεια της ικανότητας γενίκευσης, επιβάλλουμε περιορισμούς στην επιτρεπτή μορφή της λύσης. Συγκεκριμένα, η συνάρτηση κόστους παίρνει την ακόλουθη μορφή:

$$\hat{\Theta} = \arg\min_{\Theta} \mathcal{L}(\Theta) = \arg\min_{\Theta} \frac{1}{N} \sum_{i=1}^{N} L(f(\boldsymbol{x_i}; \Theta), \boldsymbol{y_i}) + \lambda R(\Theta). \tag{2.21}$$

Ο όρος κανονικοποίησης πρέπει να λαμβάνει υπόψη του τις τιμές των παραμέτρων και να αξιολογεί την πολυπλοκότητά τους. Στόχος είναι να έχουμε παραμέτρους που αντιστοιχούν σε μικρή απώλεια και μικρή πολυπλοκότητα. Αυτό που στην ουσία προσπαθεί να επιτύχει η κανονικοποίηση είναι να μειώσει τη χωρητικότητα του μοντέλου, ή αλλιώς να τιμωρήσει τα περίπλοκα μοντέλα και να ευνοήσει τα απλούστερα.

Το λ πρέπει να οριστεί εμπειρικά, με βάση την απόδοση ταξινόμησης πάνω στα δεδομένα επαλήθευσης (development set) και ονομάζεται *υπερπαράμετρος* (*hyperparameter*). Οι όροι κανονικοποίησης R υπολογίζουν τις νόρμες των πινάκων των παραμέτρων και προτιμούν λύσεις με μικρές νόρμες. Οι πιο κοινές νόρμες κανονικοποίησης είναι:

- L_2 *νόρμα*:

$$R_{L_2}(\boldsymbol{W}) = ||\boldsymbol{W}||_2^2 = \sum_{i,j} (\boldsymbol{W}_{[i,j]})^2. \tag{2.22}$$

  Ο όρος R παίρνει τη μορφή μιας τυπικής Ευκλείδειας νόρμας (L_2-νόρμα) των παραμέτρων, προσπαθώντας να κρατήσει το άθροισμα των τετραγώνων των τιμών τους χαμηλό. Τα βάρη του δικτύου που έχουν μεγάλες τιμές $\boldsymbol{W}_{[i,j]}$ θα τιμωρηθούν, καθώς θεωρούνται μη πιθανά. Για την L_2 νόρμα συχνά συναντάται στη βιβλιογραφία ο όρος *weight decay*. Όπως μπορούμε να παρατηρήσουμε από την Εξίσωση 2.22, τα μεγάλα βάρη τιμωρούνται πολύ, ενώ τα μικρά βάρη δεν επηρεάζονται σχεδόν καθόλου.

- L_1 *νόρμα*:

$$R_{L_1}(\boldsymbol{W}) = ||\boldsymbol{W}||_1 = \sum_{i,j} |\boldsymbol{W}_{[i,j]}|. \tag{2.23}$$

Η L_1 νόρμα τιμωρεί ομοιόμορφα χαμηλές και υψηλές τιμές και προσπαθεί να μειώσει όλες τις μη-μηδενικές παραμέτρους, ούτως ώστε να τείνουν στο μηδέν. Επομένως, επιβάλει χαμηλή πυκνότητα (sparsity) στο μοντέλο. Η L_1 νόρμα συναντάται επίσης στη βιβλιογραφία ως *lasso* [44].

- *Dropout:*

  Μία αποτελεσματική τεχνική αποφυγής του overfitting στα δεδομένα εκπαίδευσης είναι το *dropout* [45, 46]. Το dropout έχει σχεδιαστεί για να μην επιτρέπει στο δίκτυο να βασίζεται σε συγκεκριμένα βάρη κατά τη μάθηση. Εισάγει τυχαιότητα στο μοντέλο, καθώς απενεργοποιεί τυχαία κάποιους νευρώνες σε ένα δίκτυο (ή σε ένα συγκεκριμένο επίπεδο του δικτύου), κατά τη διάρκεια της εκπαίδευσης. Το dropout αποτελεί πολύ σημαντικό παράγοντα που τα νευρωνικά δίκτυα επιτυγχάνουν εξαιρετικά αποτελέσματα.

Βελτιστοποίηση (Optimization)

Για να εκπαιδεύσουμε το μοντέλο, πρέπει να λύσουμε το πρόβλημα βελτιστοποίησης της Εξίσωσης 2.21. Μία συνήθης λύση είναι να χρησιμοποιήσουμε κάποια μέθοδο βασισμένη στα gradients. Το *gradient* ενός συγκεκριμένο σημείο είναι η κλίση της εφαπτομένης της συνάρτησης σε εκείνο το σημείο. Έχει κατεύθυνση προς τη μεγαλύτερη αύξηση της συνάρτησης. Οι μέθοδοι που βασίζονται στο gradient (κλίση) προσπαθούν να ελαχιστοποιήσουν την αντικειμενική συνάρτηση $\mathcal{L}(\Theta)$ υπολογίζοντας επανειλημμένα μια εκτίμηση της κόστους $\mathcal{L}$ στα δεδομένα εκπαίδευσης, υπολογίζοντας τα gradients των παραμέτρων Θ του μοντέλου σε σχέση με την εκτίμηση κόστους και ενημερώνει τις παραμέτρους στην αντίθετη κατεύθυνση της κλίσης της αντικειμενικής συνάρτησης.

Ο αλγόριθμος Απότομης Καθόδου (gradient descent - GD) είναι ένας από τους πιο δημοφιλείς αλγορίθμους για βελτιστοποίηση στα νευρωνικά δίκτυα. Υπολογίζει την κλίση της συνάρτησης κόστους ως προς τις παραμέτρους θ για το σύνολο δεδομένων. Ο ρυθμός εκμάθησης (learning rate - η) αποτελεί υπερπαράμετρο που ελέγχει το βαθμό που θα προσαρμοστούν οι παράμετροι του μοντέλου σε σχέση με το gradient της συνάρτησης κόστους. Ο GD ορίζεται ως:

$$\theta = \theta - \eta \nabla_\theta J(\theta) \tag{2.24}$$

Ο Στοχαστικός αλγόριθμος Απότομης Καθόδου (Stochastic Gradient Descent - SGD) [47] αντιθέτως ενημερώνει τις παραμέτρους για κάθε παράδειγμα εκπαίδευσης $\boldsymbol{x}_i$ με ετικέτα $\boldsymbol{y}_i$:

$$\theta = \theta - \eta \nabla_\theta J(\theta; x_i; y_i) \tag{2.25}$$

Ο GD εκτελεί πλεονάζοντες υπολογισμούς για μεγάλα σύνολα δεδομένων, καθώς επανυπολογίζει gradients για παρόμοια παραδείγματα πριν από κάθε ενημέρωση παραμέτρων. Ο SGD αποφεύγει αυτούς τους πλεονάζοντες υπολογισμούς εκτελώντας μια ενημέρωση τη φορά. Επομένως, είναι συχνά πολύ ταχύτερος και μπορεί να χρησιμοποιηθεί για να μάθει online.

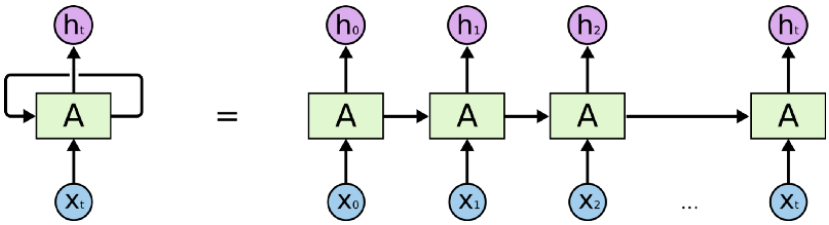
Ο απλός gradient descent, ωστόσο, δεν εγγυάται καλή σύγκλιση. Ο ρυθμός εκμάθησης πρέπει να οριστεί κατάλληλα, διότι μια μικρή τιμή οδηγεί σε αργή σύγκλιση, ενώ μια πολύ μεγάλη τιμή μπορεί να αποτρέψει τη σύγκλιση και να οδηγήσει τη συνάρτηση κόστους να παρουσιάζει διακυμάνσεις γύρω από το ελάχιστο, ή ακόμα και να αποκλίνει. Επιπλέον, ο ίδιος ρυθμός εκμάθησης εφαρμόζεται σε όλες τις ενημερώσεις παραμέτρων. Αν τα δεδομένα μας έχουν χαμηλή πυκνότητα και τα χαρακτηριστικά έχουν πολύ διαφορετικές συχνότητες, είναι πιθανό ότι δε θέλουμε να τα ενημερώνουμε όλα στον ίδιο βαθμό. Αντιθέτως, μπορεί να θέλουμε να εκτελούμε μεγαλύτερες ενημερώσεις στα χαρακτηριστικά που παρατηρούμε σπάνια. Τελικά, μια πρόκληση που παρουσιάζεται όταν προσπαθούμε να ελαχιστοποιήσουμε μη-κυρτές συναρτήσεις κόστους (και η οποία είναι συνήθης στα νευρωνικά δίκτυα) είναι να αποφύγουμε να παγιδευτούμε σε τοπικά ελάχιστα που δεν είναι βέλτιστα.

Υπάρχουν σήμερα πολλοί διαφορετικοί αλγόριθμοι βελτιστοποίησης που αποφεύγουν τα παραπάνω προβλήματα. Υπάρχουν αλγόριθμοι με αυτόματη ρύθμιση του ρυθμού εκμάθησης, όπως ο Adagrad [48], ο Adadelta [49] και ο Adam [50], ο οποίος χρησιμοποιείται ευρέως στο NLP με νευρωνικά δίκτυα.

Οπίσθια διάδοση (Backpropagation)
Για να ελαχιστοποιήσουμε τη συνάρτηση κόστους $J(\theta)$ ενός νευρωνικού δικτύου χρησιμοποιώντας βέλτιστο σύνολο τιμών για τις παραμέτρους θ (βάρη, συμβολίζονται επίσης με w), πρέπει να υπολογίσουμε το gradient. Αν και τα μαθηματικά του υπολογισμού του gradient σε νευρωνικά ακολουθούν απλώς τον κανόνα αλυσίδας της, μπορούν να γίνουν πολύπλοκα και να οδηγήσουν σε σφάλματα για πολύπλοκα δίκτυα. Ευτυχώς, τα gradients μπορούν να υπολογιστούν αποδοτικά με τον αλγόριθμο οπίσθιας διάδοσης *backpropagation* [51, 52]. Το backpropagation υπολογίζει συστηματικά της παραγώγους μιας περίπλοκης μαθηματικής έκφρασης χρησιμοποιώντας τον κανόνα αλυσίδας και αποθηκεύοντας τα ενδιάμεσα αποτελέσματα. Ένα νευρωνικό δίκτυο μπορεί να απεικονιστεί με ένα κατευθυνόμενο γράφο, όπου κάθε κόμβος αντιστοιχεί σε ένα συγκεκριμένο βάρος του δικτύου. Για να ανανεώσουμε τα βάρη του δικτύου μετά από κάθε υπολογισμό της συνάρτησης κόστους, αποδεικνύεται βασική μια έκφραση για την μερική παράγωγο $\frac{\partial J}{\partial w}$ της συνάρτησης κόστους J σε σχέση με οποιοδήποτε βάρος w του δικτύου. Παρατηρώντας τις τιμές των μερικών παραγώγων, αποκτούμε διαίσθηση σχετικά με την ευαισθησία της συνάρτησης κόστους ως προς αυτές τις παραμέτρους. Τα gradients αποτελούν μέτρο του πόσο καλά συμπεριφέρεται το δίκτυο και μας βοηθούν να προσαρμόσουμε (fine-tune) κατάλληλα τα βάρη, για να μοντελοποιήσουμε σωστά τα δεδομένα μας.

2.5.3 Αναδρομικά Νευρωνικά Δίκτυα (Recurrent Neural Networks - RNNs)

Τα Αναδρομικά Νευρωνικά Δίκτυα (Recurrent Neural Networks - RNNs) αποτελούν ένα ισχυρό και εύρωστο τύπο νευρωνικών δικτύων, τα οποία είναι ιδιαιτέρως χρήσιμα λόγω της εσωτερικής τους μνήμης. Οι συνδέσεις μεταξύ των μονάδων σε ένα RNN δημιουργούν ένα κατευθυνόμενο γράφω σε μία ακολουθία. Αυτό επιτρέπει στο δίκτυο να παρουσιάζει δυναμική χρονική συμπεριφορά για μια χρονική ακολουθία. Τα RNNs χρησιμοποιούν την εσωτερική τους κατάσταση (μνήμη) για να επεξεργαστούν ακολουθιακές εισόδους στο δίκτυο. Διαισθητικά, τα RNNs έχουν την ικανότητα να θυμούνται σημαντικές πληροφορίες σχετικά με την είσοδο που έχουν δεχθεί, γεγονός που τους επιτρέπει να κάνουν ακριβείς προβλέψεις για τα δεδομένα που θα ακολουθήσουν. Όπως μπορεί κανείς να παρα-



Σχήμα 2.8: Ένα βασικό αναδρομικό νευρωνικό δίκτυο. Πηγή: http://colah.github.io

τηρήσει στο Σχήμα 2.9, τα βασικά RNNs είναι κόμβοι οργανωμένα σε διαδοχικά επίπεδα. Το RNN πρώτα παίρνει το x_0 από την ακολουθία εισόδου και εξάγει h_0 (κρυφή κατάσταση). Η κρυφή κατάσταση h_0 μαζί με το x_1 αποτελούν την είσοδο για το επόμενο βήμα. Αντίστοιχα, η h_1 μαζί με το x_2 αποτελούν την είσοδο για το επόμενο βήμα και ούτω καθεξής. Άρα, το RNN θυμάται το περιεχόμενο της εισόδου που έχει ήδη δεχθεί κατά τη διάρκεια της εκπαίδευσης.

Επομένως, για κάθε χρονική στιγμή t, οι εξισώσεις που περιγράφουν τη λειτουργία του RNN είναι:

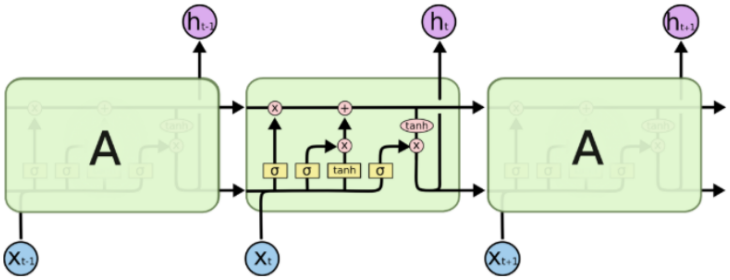
$$h_t = f_h(W_{hh}h_{t-1} + W_{hx}x_t + b_h) \tag{2.26}$$
$$y_t = f_y(W_{yh}h_t + b_y) \tag{2.27}$$

όπου h_t η κρυφή κατάσταση στη χρονική στιγμή t, x_t το διάνυσμα εισόδου στη χρονική στιγμή t, y_t το διάνυσμα εξόδου στη χρονική στιγμή t, b_h το bias (μεροληψία) για το h, b_y το bias για το y και f_x, f_h οι συναρτήσεις ενεργοποίησης για x και h αντίστοιχα. Υπάρχουν τρεις διαφορετικοί πίνακες βαρών: W_{hx} (βάρη από την είσοδο προς το κρυφό επίπεδο), W_{hh} (βάρη από το κρυφό στο κρυφό επίπεδο), και W_{yh} (βάρη από το κρυφό επίπεδο προς το επίπεδο εξόδου).

Αμφίδρομα RNNs (Bi-directional RNNs)
Όπως αναφέρθηκε παραπάνω, τα RNNs συλλαμβάνουν πληροφορίες για τα ακολουθιακά δεδομένα που έχουν δεχθεί μέχρι τη χρονική στιγμή t και τις κωδικοποιούν στην κρυφή τους κατάσταση. Ωστόσο, είναι επίσης πιθανό να λαμβάνουν πιο πολλή πληροφορία διαβάζοντας μια δεδομένη ακολουθία ανάποδα, για να μπορούν να πραγματοποιήσουν ακριβέστερες προβλέψεις. Άρα, ένα αμφίδρομο RNN δημιουργείται με τον εξής τρόπο:

Κωδικοποιούμε την ακολουθία εισόδου από την αρχή ως το τέλος (εμπρόσθιο - forward RNN) αλλά και την ακολουθία από το τέλος ως την αρχή (οπίσθιο - backward RNN). Έπειτα, συνδυάζουμε την κρυφή κατάσταση των δύο RNN για να βρούμε την κρυφή κατάσταση για κάθε χρονική στιγμή. Συγκεκριμένα, υπολογίζουμε ξεχωριστά την κρυφή κατάσταση του εμπρόσθιου RNN $\overrightarrow{h_t}$ τη χρονική στιγμή t αλλά και την αντίστοιχη κρυφή κατάσταση του οπίσθιου RNN $\overleftarrow{h_t}$ και τις συνενώνουμε για να υπολογίσουμε την τελική κρυφή κατάσταση σε κάθε χρονική στιγμή. Επομένως, η κρυφή κατάσταση στη χρονική στιγμή t είναι απλά η συνένωση των δύο διανυσμάτων: $h_t = \overrightarrow{h_t} || \overleftarrow{h}_{T-t}$. Το ίδιο ισχύει και για όλες τις $T+1$ χρονικές στιγμές της ακολουθίας εισόδου.

Δίκτυο μακράς βραχυπρόθεσμης μνήμης (Long Short-Term Memory - LSTM) unit
Θεωρητικά, τα RNNs μπορούν να διατηρήσουν πληροφορία από συνδέσεις μεγάλων αποστάσεων μεταξύ των ακολουθιών εισόδου. Το πρόβλημα των απλών RNNs είναι υπολογιστικό: κατά τη διάρκεια εκπαίδευσης ενός RNN με τον αλγόριθμο οπίσθιας διάδοσης, τα gradients που διαδίδονται προς τα πίσω είναι πιθανό να πάρουν πολύ μικρές τιμές (που τείνουν στο μηδέν) ή να "εκραγούν" (να τείνουν στο άπειρο), επειδή οι υπολογισμοί που γίνονται κατά τη διαδικασία αυτή χρησιμοποιούν αριθμούς πεπερασμένης ακρίβειας.
Τα δίκτυα LSTM networks (που προτάθηκαν από [53]) ξεπερνούν σε ένα βαθμό αυτά τα προβλήματα προφυλάσσοντας τις συνδέσεις μεγάλων αποστάσεων ανάμεσα σε λέξεις και διαγράφουν πληροφορίες που δεν είναι σημαντικές από την πύλη του κυττάρου (cell gate) μέσω του επιπέδου της πύλης λήθης (forget gate).



Σχήμα 2.9: Δομή ενός κυττάρου LSTM.

Δοσμένης μίας ακολουθίας $x_1, x_2, ..., x_t, ..., x_n$ διανυσμάτων μιας ακολουθίας εισόδου μήκους n, για το διάνυσμα x_t, με εισόδους h_{t-1} και c_{t-1}, τα h_t και c_t υπολογίζονται ως εξής:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{2.28}$$
$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{2.29}$$
$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{2.30}$$

$$u_t = tanh(W_u x_t + U_u h_{t-1} + b_u) \tag{2.31}$$
$$c_t = f_t \odot c_{t-1} + i_t \odot u_t \tag{2.32}$$
$$h_t = o_t \odot tanh(c_t) \tag{2.33}$$

- **Πύλη λήθης (Forget gate)** (f_t): Αυτή η πύλη αποφασίζει ποιες πληροφορίες πρέπει να διαγραφούν ή να διατηρηθούν. Πληροφορίες από την προηγούμενη κρυφή κατάσταση h_{t-1} και

πληροφορίες από την τωρινή είσοδο x_t περνούν από μία σιγμοειδή συνάρτηση ενεργοποίησης. Οι τιμές βρίσκονται στο διάστημα [0, 1]. Όσο πιο κοντά είναι στο 0, είναι πιο πιθανό να διαγραφούν, ενώ όσο πιο κοντά είναι στο 1, να διατηρηθούν.

- **Πύλη εισόδου (Input gate)** (i_t): Η προηγούμενη κρυφή κατάσταση και η τωρινή είσοδο περνούν από τη σιγμοειδή συνάρτηση. Η κρυφή κατάσταση και η τωρινή είσοδος περνούν επίσης στη συνάρτηση tanh για να λάβουν τιμές ανάμεσα στο -1 και το 1 (u_t). Τελικά, η έξοδος της tanh πολλαπλασιάζεται με την έξοδο της σιγμοειδούς ($i_t \odot u_t$). Η σιγμοειδής φιλτράρει τις σημαντικές πληροφορίες της tanh.

- **Κατάσταση κυττάρου (Cell state)** (c_t): Η κατάσταση κυττάρου πολλαπλασιάζεται με το διάνυσμα λήθης. Η πράξη αυτή ενέχει την πιθανότητα να απορριφθούν τιμές αν πολλαπλασιαστούν με τιμές που τείνουν στο 0. Έπειτα, παίρνουμε την έξοδο της πύλης εισόδου και τις αθροίζουμε, οπότε λαμβάνουμε νέες τιμές για την κατάσταση κυττάρου, οι οποίες είναι πιο σχετικές με το πρόβλημά μας.

- **Πύλη εξόδου (Output gate)** (o_t): Η πύλη εξόδου αποφασίζει ποια θα είναι η επόμενη κρυφή κατάσταση. Καθώς η κρυφή κατάσταση περιέχει πληροφορίες για τις προηγούμενες εισόδους, χρησιμοποιείται επίσης για προβλέψεις. Αρχικά, η προηγούμενη κρυφή κατάσταση και η τωρινή είσοδος περνούν στην σιγμοειδή. Έπειτα, η αλλαγμένη κατάσταση του κυττάρου περνάει στην συνάρτηση tanh. Πολλαπλασιάζουμε την έξοδο της tanh με τη σιγμοειδή έξοδο ($o_t \odot tanh(c_t)$) για να αποφασίσουμε ποιες πληροφορίες θα πρέπει να διατηρήσει η κρυφή κατάσταση. Η έξοδος είναι η νέα κρυφή κατάσταση. Η νέα κατάσταση κυττάρου και η νέα κρυφή κατάσταση προχωρούν και στην επόμενη χρονική στιγμή.

Μηχανισμός Προσοχής (Self-Attention mechanism)
Η βασική ιδέα πίσω από τον μηχανισμό προσοχής είναι ότι δεν συνεισφέρουν όλα τα διανύσματα μιας ακολουθίας εξίσου στην έννοια που εκφράζεται από τη συνολική είσοδο. Οπότε, το μοντέλο δεν πρέπει να χρησιμοποιεί όλα τα διανύσματα εξίσου για να κάνει μία πρόβλεψη, αλλά να εστιάζει στα τμήματα της εισόδου που περιέχουν τις πιο σχετικές πληροφορίες για ένα συγκεκριμένο πρόβλημα. Για να υλοποιήσουμε αυτή τη προσέγγιση, χρησιμοποιούμε ένα μηχανισμό προσοχής (attention mechnaism) [54, 13] για να βρούμε τη σχετική σημασία κάθε διανύσματος εισόδου μιας ακολουθίας. Προκειμένου να εστιάσουμε στα διανύσματα που περιέχουν την πιο σημαντική πληροφορία, θέτουμε ένα βάρος a_i στο κρυφό βήμα που αντιστοιχεί σε κάθε διάνυσμα h_i. Υπολογίζουμε την πεπερασμένη αναπαράσταση r όλης της ακολουθίας εισόδου, ως το σταθμισμένο άθροισμα όλων των κρυφών καταστάσεων.

$$e_i = tanh(W_h h_i + b_h), \quad e_i \in [-1, 1] \tag{2.34}$$

$$a_i = \frac{exp(e_i)}{\sum_{t=1}^{T} exp(e_t)}, \quad \sum_{i=1}^{T} a_i = 1 \tag{2.35}$$

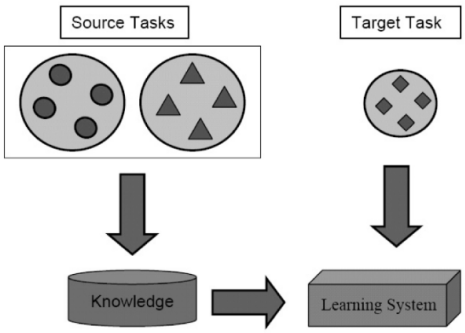
$$r = \sum_{i=1}^{T} a_i h_i \tag{2.36}$$

όπου W_h και b_h είναι τα βάρη του επιπέδου προσοχής.

2.6 Μεταφορά Μάθησης (Transfer Learning - TL)

Κίνητρο
Στη μηχανική μάθηση (και ειδικά στη βαθιά μάθηση - deep learning), αντιμετωπίζουμε ένα πολύ σημαντικό πρόβλημα. Αυτό είναι το γεγονός ότι τα δίκτυα που επιλύουν περίπλοκα προβλήματα

απαιτούν τεράστιες ποσότητες δεδομένων. Ωστόσο, η απόκτηση αυτών των δεδομένων για τα επιβλεπόμενα μοντέλα είναι συχνά ανέφικτη λόγω χρονικών ή υπολογιστικών περιορισμών. Επιπλέον, τα μοντέλα που έχουν εκπαιδευτεί σε μικρά, ειδικά σύνολα δεδομένων έχουν χειρότερη απόδοση όταν χρησιμοποιούνται για να αντιμετωπίσουν ένα διαφορετικό πρόβλημα, το οποίο μπορεί να είναι σχετικά παρεμφερές με το πρόβλημα στο οποίο έχουν εκπαιδευτεί.



Σχήμα 2.10: Μέθοδος μεταφοράς μάθησης (transfer learning).

Ο στόχος της μεταφοράς μάθησης είναι να βελτιώσει την εκμάθηση του προβλήματος-στόχου (*target* task) αξιοποιώντας γνώση από το πρόβλημα-πηγή (*source* task), όπως φαίνεται στο Σχήμα 2.11.

Ορισμός

Στη μεταφορά μάθησης χρησιμοποιούνται οι έννοιες του *τομέα* (domain) και i (task). Ένας τομέας D αποτελείται από ένα χώρο χαρακτηριστικών X και μια περιθώρια (marginal) κατανομή πιθανότητας $P(X)$ στο χώρο χαρακτηριστικών όπου $X = x_1, \ldots, x_n \in X$. Δεδομένου ενός τομέα $D = \{X, P(X)\}$, ένα πρόβλημα T αποτελείται από το χώρο ετικετών Y και μια δεσμευμένη κατανομή πιθανότητας $P(Y|X)$, η οποία συνήθως είναι αποτέλεσμα μάθησης από τα δεδομένα εκπαίδευσης $\{x_i, y_i\}$, με $x_i \in X$ και $y_i \in Y$.

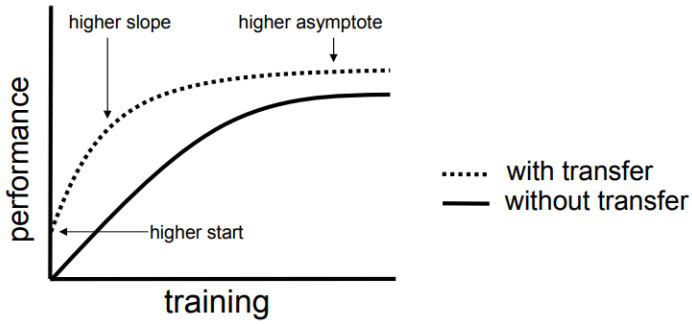
Δεδομένου ενός τομέα-πηγής (source domain) D_S, ένα αντίστοιχο πρόβλημα-πηγή (source task) T_S, αλλά και ένα τομέα-στόχο (target domain) D_T και ένα πρόβλημα-στόχο T_T, η μεταφορά μάθησης αποσκοπεί στο να μας επιτρέψει να μάθουμε την δεσμευμένη κατανομή πιθανότητας του στόχου $P(Y_T|X_T)$ στον D_T με τις πληροφορίες που έχουν συλλεχθεί από τους D_S και T_S, όπου $D_S \neq D_T$ ή $T_S \neq T_T$.

Ποιοτική ανάλυση

Υπάρχουν τρεις τρόποι με τους οποίους συνήθως η μεταφορά μάθησης βελτιώνει τη διαδικασία εκπαίδευσης, οι οποίοι φαίνονται στο Σχήμα 2.11. Πρώτον, η αρχική απόδοση που επιτυγχάνεται στο target task χρησιμοποιώντας μόνο τη γνώση που έχει μεταφερθεί από το source task, προτού εκπαιδευτεί παραπάνω, σε σχέση με την αρχική απόδοση ενός τυχαία αρχικοποιημένου μοντέλου. Δεύτερον, ο χρόνος που χρειάζεται για να εκπαιδευτεί πλήρως το μοντέλο στο target tasks δεδομένης της γνώσης που έχει μεταφερθεί, σε σχέση με το χρόνο που χρειάζεται για να το μάθει εξαρχής. Τρίτον, το τελικό επίπεδο απόδοσης που επιτυγχάνεται στο target task σε σχέση με το τελικό επίπεδο χωρίς μεταφορά μάθησης. [5]

Η περίπτωση της μη-επιβλεπόμενης προεκπαίδευσης (unsupervised pretraining)

Μία συγκεκριμένη περίπτωση μεταφοράς μάθησης είναι όταν το source task είναι μη επιβλεπόμενο και το target task είναι επιβλεπόμενο. Αυτή η περίπτωση έχει ιδιαίτερο ενδιαφέρον, καθώς πολύ συχνά έχουμε διαθέσιμες μεγάλες ποσότητες μη επιβλεπόμενων δεδομένων εκπαίδευσης, αλλά πολύ λίγα δεδομένα εκπαίδευσης με ετικέτες. Η εκπαίδευση με επιβλεπόμενες τεχνικές στο επισημασμένο υποσύνολο πολλές φορές οδηγεί σε overfitting. Αποκτώντας ποιοτικές αναπαραστάσεις από τα μη επιβλεπόμενα δεδομένα, το μοντέλο μας μπορεί να έχει καλύτερη απόδοση στο πρόβλημα επιβλεπόμενης μάθησης που αντιμετωπίζουμε [55].



Σχήμα 2.11: Οι 3 τρόποι με τους οποίους η μεταφορά μάθησης βελτιώνει την μάθηση, σύμφωνα με τους Torrey et al. [5].

Αυτή η περίπτωση μεταφοράς μάθησης ονομάζεται *μη-επιβλεπόμενη προεκπαίδευση (unsupervised pretraining)*. Αυτή η διαδικασία αποτελεί παράδειγμα του πώς μια αναπαράσταση που έχει δημιουργηθεί από το μοντέλο, όταν αυτό αντιμετωπίζει ένα συγκεκριμένο πρόβλημα (μη επιβλεπόμενο) μπορεί κάποιες φορές να είναι χρήσιμη για ένα άλλο πρόβλημα (επιβλεπόμενο). Ονομάζεται *προεκπαίδευση (pretraining)*, επειδή αποτελεί μόνο το πρώτο βήμα προτού ένας αλγόριθμος εκπαίδευσης εφαρμοστεί για να προσαρμόσει (*fine-tune*) όλα τα επίπεδα μαζί. Ως προς το πρόβλημα επιβλεπόμενης μάθησης, μπορεί να θεωρηθεί ένας όρος κανονικοποίησης και αρχικοποίησης των παραμέτρων.

- **Κανονικοποίηση**: Είναι πιθανό ότι η προεκπαίδευση αρχικοποιεί ένα βαθύ νευρωνικό δίκτυο σε μία περιοχή που θα ήταν αλλιώς απροσπέλαστη - για παράδειγμα, μια περιοχή που περιτριγυρίζεται από περιοχές όπου η συνάρτηση κόστους εναλλάσσεται τόσο πολύ από το ένα παράδειγμα στο άλλο που μπορεί να υπολογιστεί μόνο μια εκτίμηση του gradient που περιέχει πολύ θόρυβο.

- **Αρχικοποίηση παραμέτρων (Parameter initialization)**: Η προεκπαίδευση, στις περισσότερες περιπτώσεις, βελτιώνει την απόδοση στο επιβλεπόμενο πρόβλημα. Η βασική ιδέα είναι πως κάποια χαρακτηριστικά που είναι χρήσιμα για την επίλυση του μη επιβλεπόμενου προβλήματος είναι επίσης χρήσιμα και για τη ν επίλυση του επιβλεπόμενου προβλήματος.

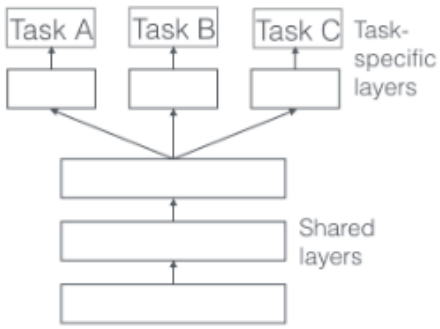
2.7 Μάθηση Πολλαπλών Εργασιών (Multi-task Learning)

Στη μάθηση πολλαπλών εργασιών (multi-task learning - MTL)[56], έχουμε πολλά συσχετιζόμενα προβλήματα στα οποία θέλουμε να κάνουμε ταυτόχρονα προβλέψεις. Θα θέλαμε να αξιοποιήσουμε την πληροφορία σε ένα από τα προβλήματα ούτως ώστε να βελτιώσουμε την ακρίβεια (accuracy) στα υπόλοιπα προβλήματα. Στη βαθιά μάθηση, η ιδέα είναι να έχουμε διαφορετικά δίκτυα που μοιράζονται μέρος της δομής τους και έχουν κάποιες κοινές παραμέτρους. Έτσι, ένας κοινός πυρήνας πρόβλεψης (η κοινή δομή) επηρεάζεται από όλα τα προβλήματα, και δεδομένα εκπαίδευσης για ένα από τα προβλήματα μπορεί να αξιοποιηθούν για τη βελτίωση των προβλέψεων των υπολοίπων.

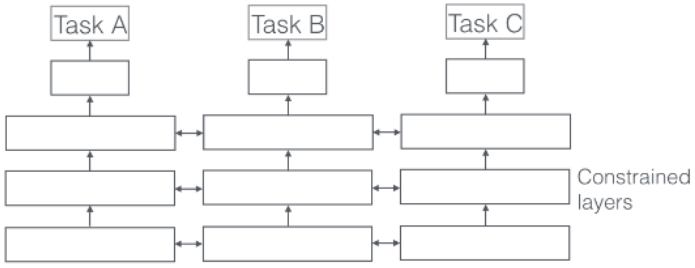
Στη βαθιά μάθηση με νευρωνικά δίκτυα, το MTL γίνεται τυπικά είτε με παραμέτρους που είναι αυστηρά κοινές (hard parameter sharing) ή είναι εν μέρει κοινές (soft parameter sharing).

Αυστηρά κοινές παράμετροι (hard parameter sharing)
Γενικά υλοποιείται με το να είναι κοινά όλα τα κρυμμένα επίπεδα για όλα τα διαφορετικά προβλήματα που ο αλγόριθμος προσπαθεί να μάθει ταυτόχρονα, ενώ υπάρχουν ξεχωριστά επίπεδα εξόδου (ένα διαφορετικό για κάθε πρόβλημα).

Το hard parameter sharing μειώνει τον κίνδυνο του overfitting. Η λογική πίσω από αυτό είναι ότι όσα περισσότερα προβλήματα προσπαθεί το μοντέλο να μάθει να λύνει ταυτόχρονα, τόσο πιο πολύ αναγκάζεται να βρει μια κοινή αναπαράσταση που είναι αποτελεσματική για όλα τα προβλήματα. Οπότε, τόσο μειώνεται η πιθανότητα να κάνει το μοντέλο overfit πάνω στο αρχικό πρόβλημα.



Σχήμα 2.12: Αυστηρά κοινές παράμετροι (hard parameter sharing) για MTL σε βαθιά νευρωνικά δίκτυα.[6].



Σχήμα 2.13: Εν μέρει κοινές παράμετροι (soft parameter sharing) για MTL σε βαθιά νευρωνικά δίκτυα. [6].

Εν μέρει κοινές παράμετροι (soft parameter sharing)
Από την άλλη πλευρά, στο soft parameter sharing, κάθε πρόβλημα έχει το δικό του μοντέλο και τις δικές του παραμέτρους. Η απόσταση ανάμεσα στις παραμέτρους του μοντέλου κανονικοποιείται για να ωθήσει τις παραμέτρους να είναι παρόμοιες μεταξύ τους [57].

Κεφάλαιο 3

Θεωρητικό Υπόβαθρο Επεξεργασίας Φυσικής Γλώσσας

3.1 Ορισμός Επεξεργασίας Φυσικής Γλώσσας (Natural Language Processing)

Η Επεξεργασία Φυσικής Γλώσσας (Natural Language Processing - NLP) αποτελεί ένα τομέα της επιστήμης υπολογιστών, της τεχνητής νοημοσύνης (που αποκαλείται επίσης μηχανική μάθηση) και της γλωσσολογίας που ασχολείται με τις αλληλεπιδράσεις μεταξύ ηλεκτρονικών υπολογιστών και ανθρώπινων (φυσικών) γλωσσών. Είναι η διαδικασία του υπολογιστή κατά τη οποία εξάγει σημαντικές πληροφορίες από την φυσική γλώσσα που δέχεται ως είσοδο και/ή παράγει φυσική γλώσσα ως έξοδο. Είναι η ανάλυση της ανθρώπινης γλώσσας βάσει της σημασιολογίας και διάφορων τεχνικών ανάλυσης [58]. Το NLP αποσκοπεί στο να προσδιορίσει τον υπολογιστικό μηχανισμό που απαιτείται ώστε να παρουσιάζει διάφορες μορφές γλωσσικής συμπεριφοράς. Αποσκοπεί επίσης στο σχεδιασμό, την υλοποίηση και την αξιολόγηση συστημάτων που επεξεργάζονται τις φυσικές γλώσσες για πρακτικές εφαρμογές. Το NLP βρίσκεται στην τομή της γλωσσολογίας και της επιστήμης υπολογιστών και ασχολείται με τις υπολογιστικές πλευρές της ανθρώπινης γλώσσας. Κύριος σκοπός του είναι να δημιουργήσει προγράμματα που επεξεργάζονται τις λέξεις και τα κείμενα της φυσικής γλώσσας. Οι κύριες πτυχές του NLP είναι:

- Απόκτηση Πληροφορίας (Information Retrieval - IR): Αφορά την αποθήκευση, αναζήτηση και ανάκτηση πληροφοριών σε έγγραφα κειμένου. Είναι ένας τομέας της επιστήμης υπολογιστών που σχετίζεται αρκετά με τις βάσεις δεδομένων και βασίζεται σε κάποιες NLP μεθόδους.
- Μηχανική Μετάφραση (Machine Translation - MT): Σχετίζεται με την αυτόματη μετάφραση από μία φυσική γλώσσα σε μία άλλη [59].
- Γλωσσολογική Ανάλυση (Language Analysis): Αφορά την ανάλυση μιας πρότασης εισόδου για την κατασκευή ενός συντακτικού δένδρου και την περαιτέρω ανάλυση του συναισθήματος που γίνεται για να προσδιοριστούν οι σημαντικές λέξεις σε μία πρόταση.

Τα επίπεδα της γλώσσας στα οποία επικεντρώνεται το NLP είναι:

- Φωνολογία (Phonology): Σχετίζεται με την ερμηνεία των φωνητικών ήχων και την αντιστοιχία τους σε λέξεις.
- Μορφολογία (Morphology): Σχετίζεται με την συνθετική φύση των λέξεων, που δημιουργούνται από μορφήματα - τις μικρότερες μονάδες που εμπεριέχουν νόημα για το NLP.
- Λεξιλογικό (Lexical): Σε αυτό το επίπεδο οι άνθρωποι, όπως και τα NLP συστήματα, ερμηνεύουν το νόημα κάθε λέξης. Διάφοροι τύποι επεξεργασίας συμβάλλουν στην κατανόηση του φυσικού λόγου σε επίπεδο λέξεων - ο πρώτος είναι το να γίνει επισήμανση μερών του λόγου για κάθε λέξη. Κατά τη διάρκεια της επεξεργασίας, οι λέξεις που έχουν πάνω από μία αντιστοιχία σε μέρη του λόγου λαμβάνουν ως ετικέτα την πιο πιθανή αντιστοιχία με βάση το περιεχόμενο στο οποίο βρίσκονται.
- Συντακτικό (Syntactic): Επικεντρώνεται στην ανάλυση μιας πρότασης ως προς τη γραμματική της δομή.

- Σημασιολογικό (Semantic): Καθορίζει τις πιθανές έννοιες μιας πρότασης. Επικεντρώνεται στις αλληλεπιδράσεις μεταξύ των διαφόρων εννοιών των λέξεων που δομούν μία πρόταση. Μπορεί να περιλαμβάνει σημασιολογική αποσαφήνιση των λέξεων που κατά τη διάρκεια της επισήμανσης μερών του λουο έλαβαν πολλαπλές ετικέτες.
- Πραγματολογικό (Pragmatic): Στοχεύει στην κατανόηση και εξήγηση του πώς εμπεριέχεται το νόημα σε πολλά έγγραφα, χωρίς στην πραγματικότητα να αναφέρεται ρητά σε αυτά. Απαιτεί την κατανόηση των προθέσεων των ομιλητών.

3.2 Εφαρμογές του NLP

Το NLP περιέχει θεωρητικά εργαλεία και πρακτικές υλοποιήσεις για πληθώρα εφαρμογών. Στην πραγματικότητα, κάθε εφαρμογή που χρησιμοποιεί κείμενο μπορεί δυνητικά να χρησιμοποιηθεί από το NLP. Οι πιο δημοφιλείς εφαρμογές του NLP είναι οι ακόλουθες:

- Απόκτηση πληροφορίας και Αναζήτηση στο ίντερνετ IR & Web Search): η επιστήμη της αναζήτησης σε έγγραφα για ανάκτηση μεταδεδομένων, αλλά και η αναζήτηση σε βάσεις δεδομένων και τον Παγκόσμιο Ιστό.
- Εξαγωγή Πληροφοριών (Information Extraction - IE): επικεντρώνεται στην αναγνώριση, επισήμανση, και εξαγωγή δομημένων πληροφοριών από μεγάλες συλλογές εγγράφων.
- Περίληψη Κειμένου (Summarization): η διαδικασία αξιολόγησης των πιο σημαντικών πληροφοριών από μία πηγή, για να παραχθεί μια συνοπτική έκδοση.
- Ερωταπόκριση (Question Answering - QA): Η απάντηση από τα έγγραφα στην εξαχθείσα ή παραγόμενη απάντηση.
- Μηχανική Μετάφραση (Machine Translation - MT): η χρήση λογισμικού υπολογιστών για να μεταφραστεί κείμενο ή ομιλία από τη μία φυσική γλώσσα στην άλλη αυτόματα.
- Αναγνώριση Φωνής & Σύνθεση (Speech Recognition & Synthesis): η εξαγωγή της γραπτής αναπαράστασης μιας προφορικής ομιλίας.
- Παραγωγή Κειμένου (Text Generation): Μία μέθοδος για τη δημιουργία φυσικών προτάσεων από “λέξεις-κλειδιά”.
- Κατανόηση και Παραγωγή Φυσικής Γλώσσας (Natural Language Understanding and Generation - NLU, NLG): Ένα NLG σύστημα λειτουργεί σαν μεταφραστής που μετατρέπει μία αναπαράσταση για γλώσσα υπολογιστών σε μία αναπαράσταση για φυσική γλώσσα.

3.2.1 Ανάλυση Συναισθήματος (Sentiment Analysis)

Η ανάλυση συναισθήματος, που αναφέρεται επίσης ως εξόρυξη απόψεων, είναι ο τομέας έρευνας που αναλύει τις απόψεις, τα συναισθήματα, τις αξιολογήσεις, τις συμπεριφορές των ανθρώπων για διάφορα προϊόντα, υπηρεσίες, οργανισμούς, άλλα άτομα, ζητήματα, και γεγονότα. Αντιπροσωπεύει ένα μεγάλο χώρο προβλημάτων [60]. Η ανάλυση συναισθήματος, στην απλούστερή της μορφή, στοχεύει στην αναγνώριση *θετικού, ουδέτερου* ή *αρνητικού* συναισθήματος από κείμενο.

Οι απόψεις παίζουν σημαντικό ρόλο σε όλες σχεδόν τις ανθρώπινες δραστηριότητες, καθώς επηρεάζουν τις συμπεριφορές μας. Οποτεδήποτε καλούμαστε να πάρουμε μία απόφαση, θέλουμε να γνωρίζουμε τις απόψεις των άλλων. Στην πραγματικότητα, οι επιχειρήσεις και οι οργανισμοί πάντα θέλουν να γνωρίζουν τις απόψεις των καταναλωτών και του κοινού σχετικά με τα προϊόντα και τις υπηρεσίες τους. Οι καταναλωτές όμως επίσης θέλουν να γνωρίζουν τις απόψεις των χρηστών ενός προϊόντος πριν το αγοράσουν, αλλά και τις απόψεις των άλλων για διάφορους υποψήφιους πολιτικούς, προτού αποφασίσουν την ψήφο τους στις βουλευτικές εκλογές. Στο παρελθόν, όταν ένας οργανισμός ή μία

επιχείρηση χρειαζόταν την άποψη του κοινού ή των καταναλωτών της, αναγκαζόταν να διενεργήσει έρευνες. Η εξόρυξη των απόψεων του κοινού και των καταναλωτών, πλέον, αποτελεί αντικείμενο έρευνας και προϊόν, καθώς είναι σημαντικό για το μάρκετινγκ, τις δημόσιες σχέσεις και τις προεκλογικές εκστρατείες.

Με την εκρηκτική αύξηση χρηστών των μέσων μαζικής ενημέρωσης (social media), που περιλαμβάνουν αξιολογήσεις προϊόντων, συζητήσεις σε φόρουμ, μπλογκ, σχόλια και δημοσιεύσεις σε Twitter, Facebook κ.λ.π. στο ίντερνετ, τόσο οι άνθρωποι ατομικά όσο και οι οργανισμοί τα χρησιμοποιούν όλο και περισσότερο για τη λήψη αποφάσεων. Για ένα οργανισμό, δεν είναι πια απαραίτητο να διεξάγει έρευνες για να συλλέξει απόψεις χρηστών, διότι αυτές οι πληροφορίες βρίσκονται πια στο ίντερνετ και είναι δημόσιες. Ωστόσο, η εύρεση και παρακολούθηση απόψεων στο ίντερνετ, καθώς και η αξιολόγηση των πληροφοριών που περιέχουν συνεχίζει να είναι ένα δύσκολο πρόβλημα, λόγω της πληθώρας ιστοσελίδων και χρηστών. Κάθε ιστοσελίδα συνήθως περιέχει τεράστιο όγκο κειμένου, οπό τον οποίο δεν είναι πάντα εύκολο να εξαχθούν συμπεράσματα. Ο μέσος αναγνώστης δυσκολεύεται να αναγνωρίσει σχετικές μεταξύ τους ιστοσελίδες και να εξάγει περιληπτική έκδοση των απόψεων που εμπεριέχουν. Είναι χρήσιμα, τότε, τα συστήματα αυτόματης ανάλυσης συναισθήματος.

Τα επίπεδα ανάλυσης συναισθήματος σε έγγραφα είναι:

- Επίπεδο εγγράφου (document level): Ο στόχος είναι να ταξινομηθεί η άποψη που εκφράζει ολόκληρο το έγγραφο ως θετική, αρνητική, ή ουδέτερη [61, 62]. Για παράδειγμα, δεδομένης μιας κριτικής προϊόντος, το σύστημα καθορίζει κατά πόσο η κριτική αντικατοπτρίζει θετική ή αρνητική άποψη για το προϊόν. Το πρόβλημα αυτό είναι ευρέως γνωστό ως ταξινόμηση συναισθήματος σε επίπεδο εγγράφου (document-level sentiment classification). Σε αυτό το επίπεδο ανάλυσης, υποθέτουμε ότι κάθε έγγραφο εκφράζει άποψη πάνω σε μία συγκεκριμένη οντότητα (π.χ. ένα συγκεκριμένο προϊόν). Οπότε, δεν μπορεί να εφαρμοστεί σε έγγραφα που αξιολογούν ή συγκρίνουν πολλαπλές οντότητες.

- Επίπεδο πρότασης (sentence level): Ο στόχος εδώ είναι να ταξινομηθεί κάθε πρόταση ως έκφραση θετικής, αρνητικής, ή ουδέτερης άποψης. Ουδέτερη άποψη συνήθως σημαίνει ότι δεν εκφέρεται καμία άποψη. Αυτό το επίπεδο ανάλυσης είναι στενά συνδεδεμένο με την ταξινόμηση υποκειμενικότητας (subjectivity classification) [63], η οποία διαχωρίζει τις *αντικειμενικές* προτάσεις (που εκφράζουν πραγματική πληροφορία) από τις *υποκειμενικές* προτάσεις (που εκφράζουν υποκειμενικές απόψεις).

- Επίπεδο οντότητας και απόψης (entity and aspect level): Τόσο το επίπεδο εγγράφου όσο και το επίπεδο πρότασης δεν ανακαλύπτουν ακριβώς τι άρεσε και δεν άρεσε στους ανθρώπους. Το επίπεδο άποψης διενεργεί πιο λεπτομερή ανάλυση. Αντί να ασχολείται με τις δομές της γλώσσας (έγγραφα, παράγραφοι, προτάσεις, φράσεις), ασχολείται με την ίδια την άποψη. Βασίζεται στην ιδέα ότι μία άποψη σχηματίζεται από ένα συναίσθημα (θετικό ή αρνητικό) και ένα στόχο-αντικείμενο (της άποψης). Το να αναγνωριστεί η άποψη χωρίς το αντικείμενο στο οποίο αναφέρεται δεν είναι χρήσιμο.

3.2.2 Αναγνώριση Συναισθημάτων (Emotion Recognition)

Η αναγνώριση συναισθημάτων από έγγραφα είναι ένα πρόσφατο πεδίο έρευνας που συνδέεται στενά με την ανάλυση συναισθήματος. Η αναγνώριση συναισθημάτων προσπαθεί να αναγνωρίσει συναισθήματα από την γραπτή τους έκφραση. Τα κύρια συναισθήματα που αποσκοπεί να αναγνωρίσει είναι θυμός, αηδία, φόβος, χαρά, λύπη και έκπληξη.

Το ίντερνετ περιέχει τεράστια συλλογή εγγράφων που με τη σειρά τους περιέχουν μεγάλες ποσότητες κειμένου. Οι πηγές κειμένου που είναι χρήσιμες για την αναγνώριση συναισθημάτων είναι οι κριτικές προϊόντων, άρθρα εφημερίδων, ανάλυση χρηματιστηρίου, προσωπικά blog, περιοδικά, ιστοσελίδες κοινωνικής δικτύωσης, φόρουμ, κριτικές, πολιτικά debate; στην πραγματικότητα, οπουδήποτε οι άνθρωποι εκφράζουν και μοιράζονται τις απόψεις τους ελεύθερα. Ανάμεσα στα πολλά προβλήματα με τα οποία ασχολείται η αυτόματη αναγνώριση κειμένου, η αυτόματη αναγνώριση συναισθημάτων

είναι ένα από τα πιο ταχέως αναπτυσσόμενα και ενδιαφέροντα. Μιας και το συναίσθημα είναι σημαντικό για να κατανοήσουμε την ανθρώπινη εμπειρία και επικοινωνία, τα συναισθήματα έχουν μελετηθεί από τις επιστήμες της ψυχολογίας και της συμπεριφορικής ανάλυσης. Μέσω της αναγνώρισης κειμένου, μπορούμε πλέον αυτόματα να αναγνωρίζουμε και ταξινομούμε τα συναισθήματα στο γραπτό λόγο; ωστόσο, οι μεοδοι δεν είναι πάντα συνεπείς και υπάρχουν πολλές προκλήσεις ακόμα στη σύγκριση διάφορων προσεγγίσεων.

3.3 Γλωσσικό Μοντέλο (Language Modeling)

Τα γλωσσικά μοντέλα (language models - LMs) υπολογίζουν την πιθανότητα της εμφάνισης ενός αριθμού λέξεων σε μια συγκεκριμένη ακολουθία. Η πιθανότητα μίας ακολουθίας T λέξεων $\{w_1, w_2, \ldots, w_T\}$ ορίζεται ως $P(w_1, w_2, \ldots, w_T)$. Ο κύριος στόχος ενός γλωσσικού μοντέλου είναι να δημιουργήσει ικανοποιητική πιθανοτική πληροφορία ούτως ώστε οι πιο πιθανές ακολουθίες λέξεων να έχουν μεγαλύτερη πιθανότητα εμφάνισης.

$$P(w_1, w_2, \ldots, w_T) = \prod_{i=1}^{i=T} P(w_i | w_1, w_2, \ldots, w_{i-1}) \tag{3.1}$$

3.3.1 Γλωσσικά Μοντέλα n-λέξεων

Το πιο απλό μοντέλο που αναθέτει πιθανότητες σε προτάσεις και ακολουθίες λέξεων είναι το *γλωσσικό μοντέλο n-λέξεων*. Στο LM n-λέξεων (n-gram LM), σπάμε τη διαδικασία πρόβλεψης μίας ακολουθίας λέξεων $w_1, w_2, \ldots, w_T$ στην πρόβλεψη μίας λέξης κάθε φορά. Για να το υπολογίσουμε, σπάμε την πιθανότητα της ακολουθίας με χρήση του κανόνα αλυσίδας:

$$P(w_1, w_2, \ldots, w_T) = P(w_1)P(w_2|w_1)\ldots P(w_T|w_1, w_2, \ldots, w_{T-1}) \tag{3.2}$$

Αλυσίδα Markov
Η πιθανότητα του LM $P(w_1, w_2, \ldots, w_T)$ είναι το γινόμενο των πιθανοτήτων των λέξεων, δεδομένου του ιστορικού των προηγούμενων λέξεων. Η ιδέα του LM n-λέξεων είναι ότι αντί να υπολογίζουμε την πιθανότητα μία λέξης, δεδομένου όλου του ιστορικού των προηγούμενων λέξεων, μπορούμε να προσεγγίσουμε το ιστορικό αυτό χρησιμοποιώντας μόνο ένα συγκεκριμένο μικρό αριθμό από τις πιο πρόσφατες λέξεις. Οπότε, για να μπορούμε να εκτιμήσουμε την κατανομή πιθανότητας των λέξεων, περιορίζουμε το ιστορικό σε n λέξεις:

$$P(w_T|w_1, w_2, \ldots, w_{T-1}) \approx P(w_T|w_{T-n}, \ldots, w_{T-2}, w_{T-1}) \tag{3.3}$$

Το μοντέλο που περιγράφεται από την Εξίσωση 3.3, όπου υπολογίζουμε τη πιθανότητας της ακολουθίας βάσει μόνο ενός περιορισμένου ιστορικού λέξεων ονομάζεται *αλυσίδα Markov* και το πλήθος των προηγούμενων καταστάσεων (στην περίπτωσή μας, λέξεων) αποτελεί την τάξη του μοντέλου.

Σύμφωνα με την *εικασία Markov*, ένας περιορισμένος αριθμός προηγούμενων λέξεων είναι υπεύθυνος για την πιθανότητα της επόμενης λέξης. Ενώ η εικασία Markov για την k-οστή τάξη είναι προφανώς λανθασμένη για οποιοδήποτε k (οι προτάσεις μπορούν να έχουν πάρα πολύ μακρινές εξαρτήσεις μεταξύ τους), παρόλα αυτά παράγει ικανοποιητικά αποτελέσματα στη δημιουργία γλωσσικού μοντέλου, ακόμα και για μικρές τιμές του k. Υπήρξε η κυρίαρχη προσέγγιση στη δημιουργία γλωσσικού μοντέλου για δεκαετίες.

Συνήθως, ο πραγματικός αριθμός λέξεων του ιστορικού βασίζεται στο πόσα δεδομένα εκπαίδευσης είναι διαθέσιμα. Στις περισσότερες περιπτώσεις χρησιμοποιούνται γλωσσικά μοντέλα 3-λέξεων (*trigrams*), τα οποία χρησιμοποιούν ιστορικό των δύο προηγούμενων λέξεων για να προβλέψουν την επόμενη. Απαντώνται επίσης στη βιβλιογραφία γλωσσικά μοντέλα 2-λέξεων (*bigrams*), μίας λέξης(*unigrams*), και πολλών άλλων τάξεων.

Εκτίμηση (Estimation)

Για την πιο συνηθισμένη περίπτωση του LM 3-λέξεων, η εκτίμηση των πιθανοτήτων πρόβλεψης λέξεων που είναι της μορφής $P(w_3|w_1, w_2)$ είναι απλή. Μετράμε πόσο συχνά, στο σύνολο δεδομένων εκπαίδευσης, η ακολουθία w_1, w_2 ακολουθείται από την λέξη w_3, σε σχέση με το πόσο συχνά ακολουθείται από οποιαδήποτε άλλη λέξη. Μέσω της εκτίμησης μέγιστης πιθανοφάνειας (maximum likelihood estimation - MLE), υπολογίζουμε:

$$P(w_3|w_1, w_2) = \frac{\text{count}(w_1, w_2, w_3)}{\sum_w \text{count}(w_1, w_2, w)} \tag{3.4}$$

Σύγχυση (Perplexity)

Υπάρχουν πολλές μετρικές για να αξιολογήσουμε ένα LM. Πολλές φορές ένα LM αξιολογείται από την απόδοσή του σε ένα πρόβλημα, όπως το κατά πόσο βελτιώνει την ποιότητα μετάφρασης σε ένα αυτόματο σύστημα μετάφρασης [64].

Μία πιο διαισθητική αξιολόγηση ενός LM είναι χρησιμοποιώντας την έννοια της σύγχυσης (perplexity) σε προτάσεις που δεν έχει ήδη μοντελοποιήσει. Το perplexity είναι μια έννοια της θεωρίας πληροφορίας που μετράει πόσο καλά ένα πιθανοτικό μοντέλο προβλέπει ένα δείγμα. Μικρές τιμές του perplexity δείχνουν καλύτερο LM. Δεδομένου ενός εγγράφου από n λέξεις $w_1, w_2, \ldots, w_n$ και τη συνάρτηση ενός LM που αναθέτει πιθανότητα εμφάνισης κάθε λέξης βάσει του ιστορικού της, το perplexity ενός LM σε σχέση με το έγγραφο είναι:

$$2^{-\frac{1}{n}\sum_{i=1}^{n} \log_2 \text{LM}(w_i|w_{1:i-1})} \tag{3.5}$$

Τα γλωσσικά μοντέλα (LMs) που μοντελοποιούν καλά τη γλώσσα θα αναθέσουν ψηλές τιμές στις πιθανότητες των γεγονότων του εγγράφου, οδηγώντας σε μικρές τιμές του perplexity. Η μετρική perplexity αποτελεί ένα καλό δείκτη της ποιότητας του γλωσσικού μοντέλου. Προκύπτει από ένα συγκεκριμένο έγγραφο, οπότε έχει νόημα να συγκρίνουμε perplexities που έχουν εκπαιδευτεί και αξιολογηθεί πάνω στο ίδιο έγγραφο.

3.3.2 Νευρωνικά Γλωσσικά Μοντέλα (Neural Language Models)

Για να ξεπεράσουμε τα προβλήματα των παραδοσιακών γλωσσικών μοντέλων, προτάθηκε η χρήση μη-γραμμικών νευρωνικών δικτύων. Αυτά επιτρέπουν τον υπολογισμό δεσμευμένης πιθανότητας πάνω σε ένα έγγραφο οποιουδήποτε μεγέθους, με μόνο γραμμική αύξηση του πλήθους των παραμέτρων.

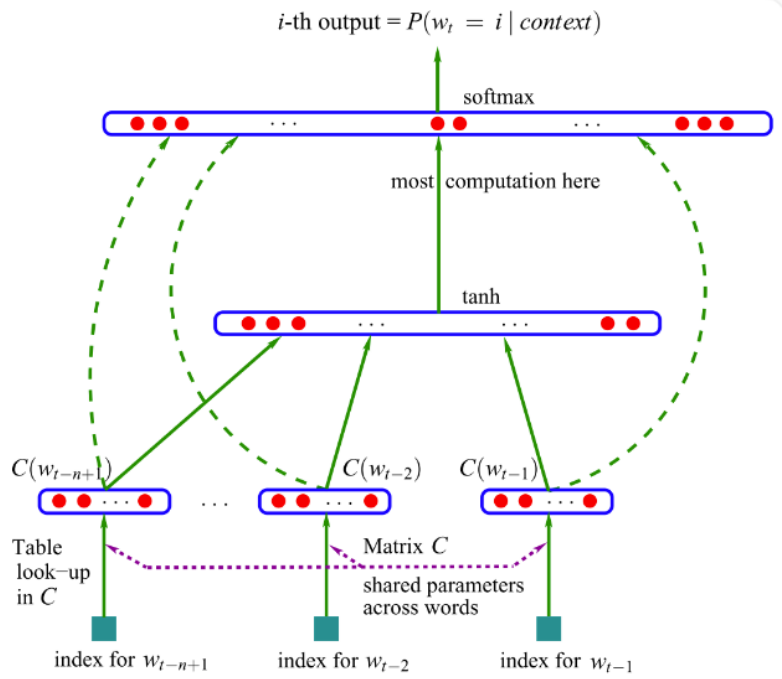
Το νευρωνικό πιθανοτικό γλωσσικό μοντέλο προτάθηκε από [7]. Το μοντέλο αυτό λαμβάνει ως είσοδο τις διανυσματικές αναπαραστάσεις που εμπεριέχονται σε ένα "παράθυρο" λέξεων που περιέχει τις n προηγούμενες λέξεις, που αναζητούνται σε έναν πίνακα C. Αυτά τα διανύσματα είναι πλέον γνωστά ως διανύσματα λέξεων (word embeddings). Αυτά τα διανύσματα λέξεων ($C(w) \in \mathbb{R}^{d_w}$) συνενώνονται και δίνονται ως είσοδος σε ένα κρυφό επίπεδο, η έξοδος του οποίου περνά στο επίπεδο ταξινόμησης softmax. Επομένως:

$$\begin{aligned} \boldsymbol{x} &= [C(w_1); C(w_2); \ldots; C(w_n)] \\ \hat{y} = P(w_i|w_{1:k}) = LM(w_{1:k}) &= \text{softmax}(\boldsymbol{h}\boldsymbol{W^2} + \boldsymbol{b^2}) \\ \boldsymbol{h} &= g(\boldsymbol{x}\boldsymbol{W^1} + \boldsymbol{b^1}) \\ \boldsymbol{x} &= [C(w_1); C(w_2); \ldots; C(w_n)] \\ C(w) &= \boldsymbol{E}_{[w]} \end{aligned} \tag{3.6}$$

όπου $w_i \in V$, $\boldsymbol{E} \in \mathbb{R}^{|V| \times d_w}$, $\boldsymbol{W^1} \in \mathbb{R}^{n \cdot d_w \times d_{\text{hid}}}$, $\boldsymbol{b^1} \in \mathbb{R}^{d_{\text{hid}}}$, $\boldsymbol{W^2} \in \mathbb{R}^{d_{\text{hid}} \times |V|}$, $\boldsymbol{b^2} \in \mathbb{R}^{|V|}$.

Το V είναι ένα πεπερασμένο λεξιλόγιο. Το μέγεθος του λεξιλογίου |V|, κυμαίνεται ανάμεσα σε 1,000 και 1,000,000 λέξεις, με το πιο συνηθισμένο μέγεθος να είναι περίπου 70,000 μοναδικές λέξεις.

Πρόσφατα, τα εμπρόσθια νευρωνικά δίκτυα αντικαταστάθηκαν από αναδρομικά νευρωνικά δίκτυα (RNNs) [65] και LSTMs [54] για τη δημιουργία γλωσσικών μοντέλων.



Σχήμα 3.1: Ένα εμπρόσθιο νευρωνικό γλωσσικό μοντέλο (a feed-forward neural network language model). [7]

3.4 Μεταφορά Μάθησης (Transfer Learning - TL)

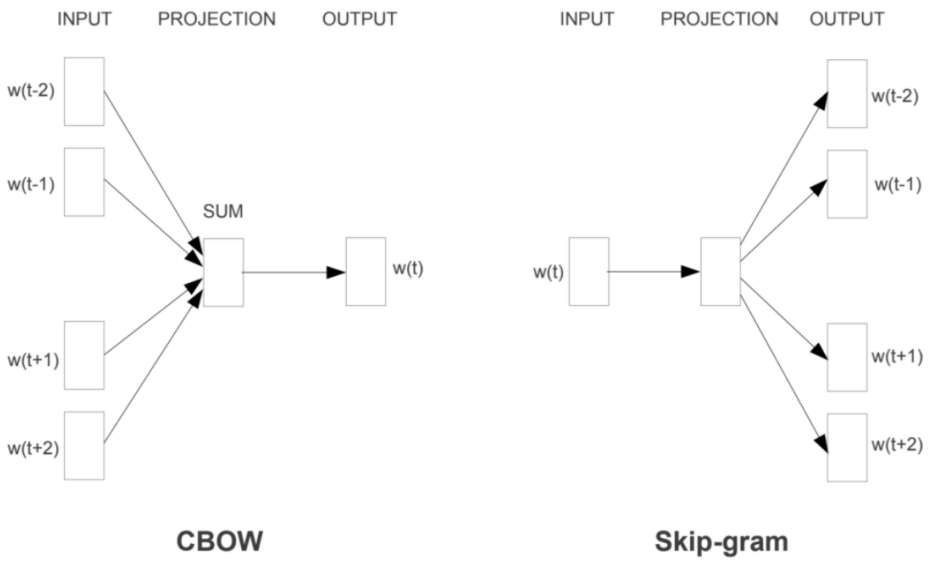
Τα περισσότερα NLP μοντέλα, σήμερα, βασίζονται σε κατενημημένες προεκπαιδευμένες αναπαραστάσεις λέξεων, όπως το word2vec [8] και το GloVe [66] για να αρχικοποιήσουν το επίπεδο εισόδου (embedding layer). Ενώ τέτοια προεκπαιδευμένα διανύσματα λέξεων είναι ικανά να μοντελοποιήσουν τις σημασιολογικές ομοιότητες των λέξεων, έχουν περιορισμούς που δεν τους επιτρέπουν να μοντελοποιήσουν φαινόμενα πολυσημίας, ή εναλλαγής κειμένου, μεταφορική χρήση της γλώσσας κλπ. Επομένως, δεν είναι ικανά να μοντελοποιήσουν όλες τις λεπτές πτυχές και έννοιες της φυσικής γλώσσας. Για να αντιμετωπίσουν αυτό το πρόβλημα, έχουν προταθεί προεκπαιδευμένες αναπαραστάσεις από γλωσσικά μοντέλα, οι οποίες δίνουν μια καλή αναπαράσταση του περιεχομένου (context) [25, 32], και αναθέτουν σε κάθε λέξη ένα διάνυσμα το οποίο είναι διαφορετικό κάθε φορά (ακόμα και για την ίδια λέξη) και εξαρτάται από το "περιβάλλον" στο οποίο τη βρίσκουν.

3.4.1 Διανύσματα Λέξεων (Word Embeddings)

Η ιδέα πίσω από τα διανύσματα λέξεων που έχουν δημιουργηθεί με μη επιβλεπόμενη μάθηση είναι ότι θα θέλαμε τα διανύσματα παρόμοιων λέξεων να έχουν κοντινές μεταξύ τους τιμές. Ενώ η ομοιότητα λέξεων είναι δύσκολο να προσδιοριστεί και εξαρτάται από το εκάστοτε πρόβλημα, οι σύγχρονες προσεγγίσεις αντλούν έμπνευση από την *κατανεμημένη υπόθεση (distributional hypothesis)* [67], υποστηρίζοντας ότι οι λέξεις έχουν παρόμοιο νόημα όταν εμφανίζονται σε παρόμοιο περιεχόμενο. Δηλαδή, το word2vec επιχειρεί να παράγει κατανεμημένες αριθμητικές αναπαραστάσεις λέξεων, οι οποίες κωδικοποιούν την ομοιότητα των λέξεων. Πολλές διαφορετικές μέθοδοι δημιουργούν επιβλεπόμενα παραδείγματα εκπαίδευσης, στόχος των οποίων είναι είτε να προβλέψουν τη λέξη βάσει του περιεχομένου, ή να προβλέψουν το περιεχόμενο βάσει της λέξης. Το πιο σημαντικό σύνολο προεκπαιδευμένων διανυσμάτων λέξεων είναι το *word2vec*. Το word2vec αποτελεί μια προσέγγιση του γλωσσικού μοντέλου, εφαρμοσμένη σε ένα πεπερασμένο παράθυρο από λέξεις.

Word2vec [8]
Το word2vec είναι ένα νευρωνικό δίκτυο 2 επιπέδων που επεξεργάζεται το κείμενο. Στόχος του είναι

να ανακατασκευάζει το γλωσσολογικό περιεχόμενο των λέξεων. Ενώ το word2vec δεν είναι βαθύ νευρωνικό δίκτυο, λαμβάνει ως είσοδο ένα μεγάλο έγγραφο και παράγει ένα χώρο διανυσμάτων υψηλών διαστάσεων (τυπικά στην τάξη των εκατοντάδων), όπου κάθε μοναδική λέξη του εγγράφου αντιστοιχίζεται σε ένα διάνυσμα στο χώρο αυτό. Τα διανύσματα λέξεων μπορούν έπειτα να χρησιμοποιηθούν από βαθιά νευρωνικά δίκτυα. Είναι τοποθετημένα έτσι σε αυτό το διανυσματικό χώρο ούτως ώστε τα διανύσματα λέξεων με παρόμοιες έννοιες να βρίσκονται κοντά μεταξύ τους. Το word2vec είναι ένα ιδιαίτερα φτηνό υπολογιστικά μοντέλο πρόβλεψης για την εκμάθηση διανυσμάτων λέξεων από απλό κείμενο.



Σχήμα 3.2: Μοντέλα εκπαίδευσης του word2vec. Πηγή: [8].

Αν τροφοδοτηθεί με αρκετά δεδομένα και περιεχόμενο, το word2vec μπορεί να κάνει εξαιρετικά ακριβείς προβλέψεις σχετικά με το νόημα μίας λέξης, με βάση τις εμφανίσεις της στο παρελθόν. Αυτές οι προβλέψεις μπορούν να χρησιμοποιηθούν για να κωδικοποιήσουν τη σχέση αυτής της λέξης με τις υπόλοιπες λέξεις, ή για να ομαδοποιηθούν διάφορα έγγραφα και να ταξινομηθούν ανάλογα με το θέμα τους. Αυτές οι ομαδοποιήσεις χρησιμοποιούνται για έρευνα, ανάλυση συναισθήματος και συστήματα σύστασης (recommender systems) σε πολλούς τομείς, όπως η ακαδημαϊκή έρευνα, η νομική αναζήτηση, το ηλεκτρονικό εμπόριο και η διαχείριση πελατειακών σχέσεων.

Το word2vec μπορεί να δημιουργηθεί με δύο διαφορετικές προσεγγίσεις, το *Continuous Bag-of-Words* (CBOW) και το *Skip-Gram*, όπως φαίνεται στο Σχήμα 3.3. Όταν το διάνυσμα χαρακτηριστικών που έχει ανατεθεί σε μία λέξη δε μπορεί να χρησιμοποιηθεί για να προβλέψει έγκυρα το περιεχόμενο αυτής της λέξης, οι συνιστώσες του διανύσματος προσαρμόζονται. Το περιεχόμενο κάθε λέξης στο κείμενο λειτουργεί όπως ένας "δάσκαλος" που στέλνει σήματα λάθους, ούτως ώστε να προσαρμοστεί εκ νέου το διάνυσμα χαρακτηριστικών. Τα διανύσματα λέξεων που έχουν παρόμοιο περιεχόμενο τοποθετούνται κοντά μεταξύ τους, με αντίστοιχη προσαρμογή των τιμών των διανυσμάτων τους.

- *Continuous Bag of Words (CBOW)*

  Ας υποθέσουμε ότι θέλουμε να προβλέψουμε τη λέξη w_i, τότε η είσοδος του μοντέλου μας είναι $w_{i-2}, w_{i-1}, w_{i+1}, w_{i+2}$ (οι προηγούμενες και επόμενες λέξεις από τη λέξη που θέλουμε να προβλέψουμε). Η έξοδος του νευρωνικού δικτύου τότε θα είναι w_i. Άρα, μπορεί να σκεφτεί κανείς το CBOW μοντέλο ως εκμάθηση διανυσμάτων λέξεων με εκπαίδευση ενός μοντέλου να προβλέπει μία λέξη βάσει του περιεχομένου.

- *Skip-Gram*

  Αυτό το μοντέλο είναι ουσιαστικά αντίθετο από το CBOW, μιας και σε αυτή τη περίπτωση η είσοδος του μοντέλου είναι η λέξη w_i και η έξοδος πρέπει να είναι $w_{i-2}, w_{i-1}, w_{i+1}, w_{i+2}$.

Στόχος είναι, επομένως, η εκμάθηση διανυσμάτων λέξεων με εκπαίδευση ενός μοντέλου να προβλέπει το περιεχόμενο βάσει μίας λέξης.

Αν και η κατανεμημένη υπόθεση προσφέρει μία ελκυστική πλατφόρμα για να βρίσκουμε τις ομοιότητες μεταξύ λέξεων, αναπαριστώντας τις σύμφωνα με το περιεχόμενο στο οποίο τις βρίσκουμε, έχει κάποιους εγγενείς περιορισμούς, οι οποίοι πρέπει να λαμβάνονται υπόψη όταν χρησιμοποιούμε τις συγκεκριμένες αναπαραστάσεις. Ο πιο σημαντικός, που έχει εξεταστεί διεξοδικά σε αυτή τη διατριβή, είναι η απουσία περιεχομένου.

Απουσία περιεχομένου (Lack of context)
Οι κατανεμημένες προσεγγίσεις ουσιαστικά προσθέτουν τις διανυσματικές αναπαραστάσεις μιας λέξης, που μπορεί να προέρχονται από διαφορετικά περιεχόμενα, και υπολογίζουν έτσι το διάνυσμα αυτής της λέξης. Το αποτέλεσμα δεν έχει συγκεκριμένο περιεχόμενο (context-free), ή αλλιώς είναι ανεξάρτητο του περιεχομένου. Ένα προφανές πρόβλημα που παρατηρείται είναι ότι οι πολύσημες (που έχουν πολλές έννοιες) λέξεις δε μπορούν να αναπαρασταθούν σωστά.

Παράθυρο γειτονικών λέξεων (Window of surrounding words)
Ένας άλλος περιορισμός που προκύπτει όταν μαθαίνουμε διανύσματα λέξεων βάσει ενός μικρού παράθυρου από γειτονικές λέξεις είναι ότι κάποιες φορές καταλήγουμε να δίνουμε το ίδιο σχεδόν διάνυσμα σε αντίθετες λέξεις (π.χ. "καλός" και "κακός") [68]. Αυτό προφανώς δημιουργεί προβλήματα στην ανάλυση συναισθήματος [69]. Κάποιες φορές, αυτά τα διανύσματα ομαδοποιούν σημασιολογικά παρόμοιες λέξεις που έχουν αντίθετο συναισθηματικό περιεχόμενο. Αυτό οδηγεί το μοντέλο ταξινόμησης για ανάλυση συναισθήματος να είναι ανίκανο να αναγνωρίσει τις αντίθετες έννοιες, άρα περιορίζεται η απόδοσή του.

3.4.2 Προεκπαιδευμένες Αναπαραστάσεις Λέξεων από Γλωσσικά Μοντέλα

Τα γλωσσικά μοντέλα προτάθηκαν πρόσφατα για τη δημιουργία αναπαραστάσεων λέξεων αξιοποιώντας το περιεχόμενο. Οι παραδοσιακές μέθοδοι δημιουργίας διανυσμάτων λέξεων, όπως το word2vec, παίρνουν υπόψη όλες τις προτάσεις στις οποίες βρίσκεται μία συγκεκριμένη λέξη για να δημιουργήσουν μια ενιαία διανυσματική αναπαράσταση της εν λόγω λέξης. Ωστόσο, μία λέξη μπορεί να έχει τελείως διαφορετική έννοια σε διαφορετικά περιεχόμενα. Μιας και η δημιουργία γλωσσικού μοντέλου επιτρέπει τον υπολογισμό των από κοινού πιθανοτήτων σε μία ακολουθία λέξεων, αποτελεί ένα διαισθητικά λογικό τρόπο αναπαράστασης του περιεχομένου μιας συγκεκριμένης λέξης. Τα προεκπαιδευμένα γλωσσικά μοντέλα κωδικοποιούν πληροφορία για το περιεχόμενο και χαρακτηριστικά υψηλού επιπέδου της γλώσσας, μοντελοποιώντας τη σύνταξη και τη σημασιολογία. Επομένως, επιτρέπουν στα NLP μοντέλα να διασαφηνίζουν το νόημα και να καταλαβαίνουν τη σωστή έννοια κάθε λέξης. Η αναπαράσταση χαρακτηριστικών που δημιουργείται έτσι περιέχει ποιοτική γνώση μιας συγκεκριμένης γλώσσας .

Επιπλέον, το LM (γλωσσικό μοντέλο) δημιουργείται από μη επισημασμένα δεδομένα, τα οποία μπορούν να βρεθούν εύκολα και δεν απαιτούν υπολογιστικό κόπο. Αποτελεί, επομένως, ένα ελκυστικό πρόβλημα, καθώς μπορεί να αξιοποιήσει μεγάλα έγγραφα κειμένου που είναι διαθέσιμα στο ίντερνετ. Για παράδειγμα, το μοντέλο Universal Language Model Fine-Tuning (ULMFiT) [25], που είναι πολύ δημοφιλής τρόπος δημιουργίας αναπαραστάσεων λέξεων από LMs, προεκπαιδεύτηκε στην αγγλική Wikipedia.

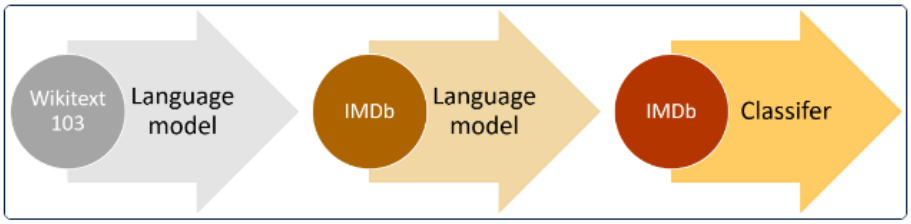
Οι προεκπαιδευμένες αναπαραστάσεις από LMs μπορούν να χρησιμοποιηθούν είτε σαν πρόσθετα χαρακτηριστικά ([32] ή μπορούν να προσαρμοστούν (fine-tune) κατευθείαν στο πρόβλημα-στόχο (target-task) [25, 24]. Στην πρώτη περίπτωση, είναι απαραίτητο να δημιουργηθούν αρχιτεκτονικές συγκεκριμένες για το εν λόγω πρόβλημα, στις οποίες να προστεθούν οι προεκπαιδευμένες αναπραστάσεις ως επιπλέον χαρακτηριστικά. Στη δεύτερη περίπτωση, που χρησιμοποιείται για παράδειγμα στο μοντέλο Generative Pre-trained Transformer (OpenAI GPT) [26], πολύ λιγότερες παράμετροι απαιτούνται και το μοντέλο μπορεί να εκπαιδευτεί στα προβλήματα-στόχους απλά με το να προσαρμόζει τις ήδη εκμαθημένες παραμέτρους. Επιπλέον, στην περίπτωση του OpenAI Transformer, τα

LMs που χρησιμοποιούνται είναι μονής κατεύθυνσης (uni-directional). Τα διανύσματα ELMo [32] και BERT [24], ωστόσο, βασίζονται σε αμφίδρομα LMs, για να ενισχύσουν τις αναπαραστάσεις χαρακτηριστικών που έχουν μάθει, ώστε να επιλύσουν με μεγαλύτερη ακρίβεια διάφορα προβλήματα.

Τώρα θα παρουσιάσουμε με συντομία δύο μεθόδους αξιοποίησης γλωσσικών αναπαραστάσεων από τα γλωσσικά μοντέλα για να επιλύσουμε προβλήματα ταξινόμησης, οι οποίες ονομάζονται ULMFiT και ELMo.

1) Universal Language Model Fine-Tuning [25]

Σε αυτή την προσέγγιση, εισάγεται μία αποτελεσματική μέθοδος μεταφοράς μάθησης που μπορεί να εφαρμοστεί σε κάθε πρόβλημα στο NLP. Η προτεινόμενη προσέγγιση βασίζεται στην προεκπαίδευση ενός γενικού LM σε ένα μη επισημασμένο σύνολο δεδομένων (π.χ. Wikipedia), την προσαρμογή του στο σύνολο δεδομένων-στόχο και την μεταφορά του σε ένα μοντέλο ταξινόμησης και την περαιτέρω εκπαίδευση του ως τη σύγκλιση.



Σχήμα 3.3: Τα τρία βήματα του ULMFiT.

Αρχικά, οι συγγραφείς εκπαιδεύουν ένα state-of-the-art LM (AWD-LSTM [70]) στο Wiki-103 με λεξιλόγιο περίπου 300,000 λέξεων. Η ιδέα είναι ότι ένα σύνολο δεδομένων σαν του ImageNet θα πρέπει να χρησιμοποιηθεί και στη φυσική γλώσσα, ούτως ώστε να μοντελοποιήσει τις γενικές ιδιότητές της.

Έπειτα, συνεχίζουν να το εκπαιδεύουν (fine-tune) πάνω στο μικρό σύνολο δεδομένων ταξινόμησης που έχουν (target task). Προτείνουν ένα εξεζητημένο τρόπο προσαρμογής των βαρών του δικτύου στο target task που στηρίζεται στη μεροληπτική προσαρμογή (discriminative fine-tuning) και τον κεκλιμένα τριγωνικό ρυθμό εκμάθησης (slanted triangular learning rate).

Μεροληπτική προσαρμογή (discriminative fine-tuning)

Προτείνεται μία καινοτόμα μέθοδος προσαρμογής που ονομάζεται *discriminative fine-tuning*, η οποία βασίζεται στην υπόθεση ότι διαφορετικά επίπεδα ενός νευρωνικού δικτύου μοντελοποιούν διαφορετικά είδη πληροφορίας [71], άρα δεν θα πρέπει να εκπαιδεύονται στον ίδιο βαθμό. Οπότε, αντί να χρησιμοποιούν SGD (2.25), χωρίζουν τις παραμέτρους θ σε $\{\theta^1, \ldots, \theta^L\}$, όπου το θ^l περιέχει τις παραμέτρους του μοντέλου στο l-οστό επίπεδο και L είναι ο αριθμός των επιπέδων του μοντέλου. Αντίστοιχα, ο ρυθμός εκμάθησης κάθε επιπέδου δίνεται από τα $\{\eta^1, \ldots, \eta^L\}$, όπου η^l είναι ο ρυθμός εκμάθησης του l-οστού επιπέδου. Άρα, η ανανέωση SGD με μεροληπτική προσαρμογή έχει την εξής μορφή:

$$\theta_t^l = \theta_{t-1}^l - \eta^l \nabla_{\theta^l} J(\theta) \tag{3.7}$$

Κεκλιμένα τριγωνικός ρυθμός εκμάθησης (slanted triangular learning rate)

Προτείνεται επίσης ο κεκλιμένα τριγωνικός ρυθμός εκμάθησης (*slanted triangular learning rate - STLR*). Πρόκειται για μια μέθοδο που πρώτα αυξάνει γραμμικά το ρυθμό εκμάθησης και μετά το μειώνει γραμμικά. Η ιδέα είναι ότι για να προσαρμοστούν οι παράμετροι του LM στα χαρακτηριστικά ενός συγκεκριμένου συνόλου χαρακτηριστικών, είναι επιθυμητό το μοντέλο να συγκλίνει γρήγορα σε μία κατάλληλη περιοχή του χώρου παραμέτρων στην αρχή και μετά να βελτιώνει τις παραμέτρους. Το STLR έχει την ακόλουθη μορφή:

$$cut = [T \cdot cut\_frac] \tag{3.8}$$

$$\text{p} = \begin{cases} \frac{t}{cut}, & \text{εάν } t < cut \\ 1 - \frac{t-cut}{cut \cdot (\frac{1}{cut\_frac} - 1)}, & \text{αλλιώς} \end{cases} \tag{3.9}$$

$$\eta_t = \eta_{max} \cdot \frac{1 + p \cdot (ratio - 1)}{ratio} \tag{3.10}$$

όπου T ο αριθμός των εποχών εκπαίδευσης, cut_frac είναι το κλάσμα της συνάρτησης των εποχών κατά τις οποίες αυξάνεται ο ρυθμός εκμάθησης, cut είναι η εποχή στην οποία ο ρυθμός εκμάθησης ξεκινά να μειώνεται, p το κλάσμα του αριθμού εποχών όπου ο ρυθμός αυξανόταν και μειωνόταν αντίστοιχα. Το $ratio$ καθορίζει πόσο μικρότερος είναι ο ελάχιστος ρυθμός εκμάθησης από τον μέγιστο ρυθμό εκμάθησης η_{max}, και η_t είναι ο ρυθμός εκμάθησης στην εποχή t. Το STLR αποτελεί παραλλαγή του τριγωνικού ρυθμού μάθησης [72].

Τελικά, μεταφέρεται το προεκπαιδευμένο LM σε ένα μοντέλο ταξινόμησης, και αυξάνεται με δύο πρόσθετα γραμμικά blocks. Επίσης χρησιμοποιείται η τεχνική του *gradual unfreezing* για να αποφευχθεί το overfitting, το οποίο βασίζεται στο να εκπαιδεύεται αρχικά μόνο το επίπεδο ταξινόμησης για μία εποχή, και έπειτα στη δεύτερη εποχή να εκπαιδεύεται επίσης το κρυφό επίπεδο, και ούτω καθεξής, μέχρι όλα τα επίπεδα να εκπαιδεύονται.

Η προτεινόμενη μέθοδος επιτυγχάνει ανταγωνιστική απόδοση, μειώνοντας το σφάλμα στα δεδομένα ελέγχου σε πληθώρα προβλημάτων ταξινόμησης, Απαιτεί, όμως, προσεκτική προσαρμογή στην εκπαίδευση (fine-tuning) και χρησιμοποιεί περίπλοκες τεχνικές για να προσαρμόσει ικανοποιητικά την κατανομή πιθανότητας των λέξεων του source task στο target task.

2) Embeddings from Language Models (ELMo) [32]

Τα διανύσματα ELMo είναι βαθιές αναπαραστάσεις λέξεων που μοντελοποιούν το περιεχόμενο και προσφέρουν αναπαραστάσεις υψηλού επιπέδου για τη γλώσσα, μοντελοποιώντας περίπλοκα χαρακτηριστικά της χρήσης λέξεων και προσαρμόζοντας τα σε διαφορετικά λεξιλογικά περιεχόμενα. Τα διανύσματα εξάγονται από ένα αμφίδρομο LSTM που εκπαιδεύεται με συνάρτηση κόστους γλωσσικού μοντέλου σε ένα τεράστιο σύνολο δεδομένων. Οι αναπαραστάσεις του ELMo αποτελούν συνάρτηση όλων των εσωτερικών επιπέδου του αμφίδρομου γλωσσικού μοντέλου. Χρησιμοποιούνται συνενωμένες με προεκπαιδευμένα διανύσματα λέξεων (όπως τα word2vec, GloVe) για να αυξήσουν την πληροφορία που μοντελοποείται και οδηγούν σε state-of-the-art αποτελέσματα σε διάφορα προβλήματα ταξινόμησης.

Δεδομένου ότι για κάθε λεκτική μονάδα (token) t_k, $\boldsymbol{x}_k^{LM}$ είναι μία αναπαράσταση ανεξάρτητη του περιεχομένου (δημιουργείται είτε με διανύσματα λεκτικών μονάδων είτε με συνελικτικά νευρωνικά δίκτυα (convolutional neural networks - CNN) πάνω στους χαρακτήρες), ένα αμφίδρομο LSTM L επιπέδων (2.5.3) υπολογίζει $2L + 1$ αναπαραστάσεις της μορφής:

$$\begin{aligned} R_k &= \{\boldsymbol{x}_k^{LM}, \overrightarrow{\boldsymbol{h}}_{k,j}^{LM}, \overleftarrow{\boldsymbol{h}}_{k,j}^{LM} |\ j = 1, \ldots, L\} \\ &= \boldsymbol{h}_{k,j}^{LM} |\ j = 0, \ldots, L\} \end{aligned} \tag{3.11}$$

Τα διανύσματα ELMo είναι ένα σταθμισμένο άθροισμα των ενδιάμεσων επιπέδων του προεκπαιδευμένου γλωσσικού μοντέλου, ανάλογα με το πρόβλημα για το οποίο χρησιμοποιούνται:

$$\text{ELMo}_k^{task} = E(R_k; \Theta^{task}) = \gamma^{task} \sum_{j=0}^{L} s_j^{task} \boldsymbol{h}_{k,j}^{LM} \tag{3.12}$$

όπου s^{task} είναι τα βάρη κανονικοποιημένα με softmax και η βαθμωτή παράμετρος γ^{task} επιτρέπει στο μοντέλο να κλιμακώνει όλο το διάνυσμα ELMo.

Τα διανύσματα ELMo προστίθενται σε 6 σύνολα δεδομένων που περιλαμβάνουν ερώτηση-απόκριση (question answering), γλωσσική συνεπαγωγή (textual entailment), σημασιολογική επισημείωση (semantic role labeling), αναγνώριση οντοτήτων (named entity extraction) και ανάλυση συναισθήματος και επιτυγχάνουν state-of-the-art αποτελέσματα σε όλα. Ωστόσο, το πώς θα σταθμιστεί το άθροισμα των ELMo διανυσμάτων απαιτεί πολλή προσοχή για κάθε διαφορετικό πρόβλημα. Επομένως, η προτεινόμενη μέθοδος, αν και είναι πάρα πολύ αποτελεσματική, έχει κάποιους περιορισμούς και απαιτεί συγκεκριμένη αρχιτεκτονική για κάθε πρόβλημα (task).

Κεφάλαιο 4

Ensemble Νευρωνικών Μεθόδων Μεταφοράς Μάθησης για Ταξινόμηση Συναισθημάτων

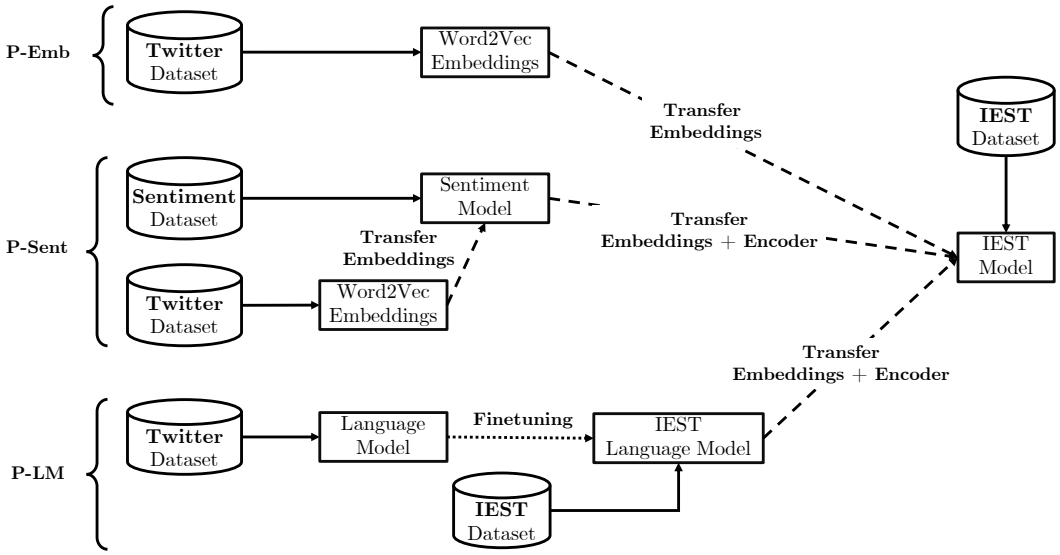
4.1 Εισαγωγή

Η αναγνώριση συναισθημάτων (emotion recognition) αποτελεί ένα ιδιαίτερα ενδιαφέρον αντικείμενο μελέτης της επεξεργασίας φυσικής γλώσσας (NLP), καθώς η γλώσσα συνήθως αντικατοπτρίζει την συναισθηματική κατάσταση του ατόμου. Επομένως, είναι φυσικό να μελετούμε τα ανθρώπινα συναισθήματα για να καταλάβουμε πώς αντικατοπτρίζονται στο γραπτό λόγο. Το πρόβλημα της αυτόματης αναγνώρισης συναισθημάτων από κείμενο είναι πολύ σημαντικό για την ακαδημαϊκή κοινότητα. Ορίζεται συνήθως ως η ταξινόμηση λέξεων, φράσεων ή εγγράφων σε έναν αριθμό από προεπιλεγμένες συναισθηματικές κατηγορίες ή διαστάσεις. Σε κάποιες περιπτώσεις, προβλήματα παλινδρόμησης μπορούν επίσης να οριστούν, όπως για παράδειγμα το πρόβλημα της πρόβλεψης του βαθμού στον οποίο ένα συναίσθημα εκφράζεται στο κείμενο [73].

Τα μέσα κοινωνικής δικτύωσης (social media) και ειδικά υπηρεσίες micro-blogging όπως το Twitter, έχουν γίνει αντικείμενο εκτενούς έρευνας από την κοινότητα του NLP. Η γλώσσα που χρησιμοποιείται στα social media αλλάζει συνεχώς και αναπτύσσεται ενσωματώνοντας νέες συντακτικές και σημασιολογικές δομές, όπως η χρήση των emoji και των hashtag, αλλά και αρκτικόλεξων. Επίσης ενσωματώνει την καθομιλουμένη ("slang"), μετατρέποντας την αυτόματη επεξεργασία φυσικής γλώσσας σε ακόμα πιο απαιτητικό πρόβλημα. Επιπλέον, η ανάλυση τέτοιου περιεχομένου αξιοποιεί τη μεγάλη διαθεσιμότητα συνόλων δεδομένων που προσφέρονται στο Twitter, ικανοποιώντας την ανάγκη της βαθιάς μάθησης για τεράστιο όγκο δεδομένων για εκπαίδευση των βαθιών νευρωνικών δικτύων. Η αναγνώριση συναισθημάτων είναι εξαιρετικά ενδιαφέρουσα στα social media, διότι έχει πλήθος πρακτικών εφαρμογών, όπως η ανίχνευση της κοινής γνώμης για πολιτικές τάσεις [74, 75, 76], παρακολούθηση του χρηματιστηρίου [77, 78], παρακολούθηση της γνώμη για προϊόντα [79], ακόμα και αναγνώριση επικοινωνίας που σχετίζεται με αυτοκτονίες [80].

Παλιότερα, η αναγνώριση συναισθημάτων, όπως τα περισσότερα προβλήματα στο NLP, αντιμετωπιζόταν με παραδοσιακές μεθόδους που περιλάμβαναν την μη αυτόματη εξαγωγή χαρακτηριστικών από λεξικά συναισθημάτων [81, 82, 83] που μετά δίνονται ως είσοδοι σε ταξινομητές, όπως ο Naive Bayes και τα SVMs [84, 85, 86]. Ωστόσο, τα βαθιά νευρωνικά δίκτυα επιτυγχάνουν βελτιωμένη απόδοση σε σχέση με τις παραδοσιακές μεθόδους, χάρη στην ικανότητά τους να μαθαίνουν πιο γενικά χαρακτηριστικά από μεγάλα σύνολα δεδομένων, παράγοντας state-of-the-art αποτελέσματα στην αναγνώριση συναισθημάτων και τη ανάλυσή τους (emotion recognition and sentiment analysis) [87, 88, 89].

Σε αυτό το κεφάλαιο, παρουσιάζουμε τη δουλειά μας, στην οποία προσεγγίζουμε ένα πρόβλημα ταξινόμησης σε πολλές κλάσεις. Το πρόβλημα έγκειται στην πρόβλεψη του συναισθήματος μίας πρότασης που έχει δημοσιευτεί στο Twitter (tweet), όταν η συναισθηματικά φορτισμένη λέξη έχει αφαιρεθεί από το tweet. Οι λέξεις που έχουν αφαιρεθεί (*trigger-words*) περιλαμβάνουν όρους όπως "sad", "happy", "disgusted", "surprised", "angry", "afraid" και τα συνώνυμά τους. Δεδομένου ενός tweet, καλούμαστε να προβλέψουμε το συναίσθημα που εμπεριέχει, ανάμεσα στις ακόλουθες κατηγορίες: θυμός, αηδία, φόβος, χαρά, λύπη, έκπληξη.
Προσεγγίζουμε αυτό το πρόβλημα (Implicit Emotion recognition Shared Task - IEST [90]) χρησιμοποιώντας προεκπαιδευμένες αναπαραστάσεις χαρακτηριστικών από μοντέλα που έχουν εκπαιδευτεί



Σχήμα 4.1: Επισκόπηση των μεθόδων μεταφοράς μάθησης του μοντέλου μας.

σε διαφορετικά προβλήματα. Η μεταφορά μάθησης (transfer learning - TL) αποτελεί ένα τρόπο να αρχικοποιούμε τα μοντέλα μας σε μία περιοχή που τα βοηθά να έχουν καλύτερη απόδοση στο συγκεκριμένο πρόβλημα που αντιμετωπίζουμε. Το προτεινόμενο μοντέλο μας αξιοποιεί 3 διαφορετικές μεθόδους TL από προεκπαιδευμένα μοντέλα: διανύσματα λέξεων, ένα νευρωνικό μοντέλο ταξινόμησης για ανάλυση συναισθήματος, γλωσσικά μοντέλα.

4.2 Σχετική Βιβλιογραφία

Η μεταφορά μάθησης (transfer learning - TL) χρησιμοποιεί γνώση από το πρόβλημα στο οποίο έχει ήδη εκπαιδευτεί για να βελτιώσει την απόδοση σε ένα σχετικό πρόβλημα μειώνοντας τα απαιτούμενα δεδομένα εκπαίδευσης [5, 91]. Στην όραση υπολογιστών (computer vision), το TL χρησιμοποιείται για να αντιμετωπίσει το γεγονός ότι συχνά περιορισμένα επισημασμένα δεδομένα εκπαίδευσης είναι διαθέσιμα για μερικές κατηγορίες προβλημάτων. Το επιτυγχάνει αυτό προσαρμόζοντας τους ταξινομητές που έχουν εκπαιδευτεί σε άλλες κατηγορίες προβλημάτων [92]. Παραδείγματα εφαρμογής του TL υπάρχουν στην αναγνώριση προσώπων [93], την οπτική ερώτηση-απόκριση (visual question-answering) [94], όπου χαρακτηριστικά της εικόνας εκπαιδευμένα στο ImageNet [14] και διανύσματα λέξεων που έχουν εκπαιδευτεί σε τεράστιο όγκο δεδομένων συνδυάζονται.

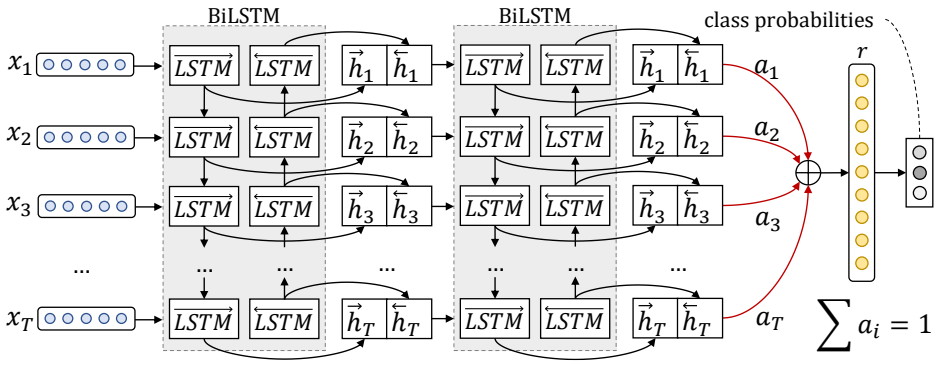
Με την ίδια λογική, οι μέθοδοι του TL έχουν επίσης εφαρμοστεί στο NLP. Τα προεκπαιδευμένα διανύσματα λέξεων [95, 66] έχουν γίνει κύριες συνιστώσες των περισσότερων αρχιτεκτονικών. Οι προεκπαιδευμένες αναπαραστάσεις λέξεων έχουν βελτιώσει την απόδοση των μοντέλων για συνεπαγωγή (entailment) [96], ανάλυση συναισθήματος [97], σύνοψη κειμένου [98], και ερώτηση-απόκριση [99].

Ωστόσο, για να αποκτήσουμε διανύσματα λέξεων υψηλής ποιότητας, πρέπει να εισάγουμε βαθιές αναπαραστάσεις λέξεων με βάση του περιεχόμενο. Αυτές οι αναπαραστάσεις πρέπει να αντιμετωπίζουν την πρόκληση της μοντελοποίησης περίπλοκων χαρακτηριστικών της χρήσης λέξεων, όπως η σύνταξη και η σημασιολογία, και να τα χρησιμοποιούν κατάλληλα για διαφορετικά γλωσσολογικά περιεχόμενα. Τέτοια διανύσματα μπορούν να διαχειριστούν την πολυσημία και να κωδικοποιήσουν τις λεπτές αποχρώσεις της γλώσσας.

Πρόσφατα, προσεγγίσεις που αξιοποιούν προεκπαιδευμένα γλωσσικά μοντέλα έχουν έρθει στο προσκήνιο, καθώς μπορούν να μάθουν την σύνθεση της γλώσσας, να κωδικοποιήσουν τις αλληλεξαρτήσεις των λέξεων που βρίσκονται σε μεγάλη απόσταση και τα χαρακτηριστικά που εξαρτώνται από το περιεχόμενο. Για παράδειγμα, τα διανύσματα λέξεων ELMo [32] και ULMFiT [100] πετυχαίνουν state-of-the-art αποτελέσματα σε πληθώρα NLP προβλημάτων. Η δουλειά μας εμπνέεται κυρίως

από το ULMFiT, το οποίο επεκτείνουμε στο Twitter.

4.3 Προτεινόμενο Μοντέλο



Σχήμα 4.2: Το προτεινόμενο μοντέλο αποτελείται από ένα αμφίδρομο LSTM 2 επιπέδων (bi-LSTM) με ένα μηχανισμό προσοχής. Όταν το μοντέλο αρχικοποιείται με προεκπαιδευμένα γλωσσικά μοντέλα, χρησιμοποιούμε LSTM μονής κατεύθυνσης (uni-directional) αντί για αμφίδρομα.

Όλα τα μοντέλα μας που βασίζονται στο TL μοιράζονται την ίδια αρχιτεκτονική: Ένα LSTM 2 επιπέδων με ένα μηχανισμό προσοχής. Φαίνεται στο Σχήμα 4.2.

Επίπεδο εισόδου (Embedding Layer)

Η είσοδος του δικτύου είναι ένα μήνυμα από Twitter, το οποίο χειριζόμαστε σαν μία ακολουθία από λέξεις. Χρησιμοποιούμε ένα επίπεδο εισόδου (embedding layer) για να προβάλουμε τις λέξεις $w_1, w_2, ..., w_N$ σε ένα διανυσματικό χώρο χαμηλών διαστάσεων R^W, όπου W το μέγεθος του embedding layer και N το πλήθος των λέξεων σε ένα tweet.

Επίπεδο LSTM

Ένα LSTM παίρνει ως είσοδο μία ακολουθία από διανύσματα λέξεων και παράγει επισημάνσεις των λέξεων (word annotations) $h_1, h_2, ..., h_N$, όπου h_i η κρυφή κατάσταση τη χρονική στιγμή i, συνοψίζοντας όλες τις πληροφορίες της πρότασης μέχρι το w_i. Χρησιμοποιούμε ένα αμφίδρομο LSTM για να πάρουμε επισημάνσεις λέξεων που συνοψίζουν τις πληροφορίες και στις δύο κατευθύνσεις. Ένα αμφίδρομο LSTM αποτελείται από ένα εμπρόσθιο (forward) $\overrightarrow{f}$, το οποίο αναλύει την πρόταση από την λέξη w_1 ως την w_N και ένα οπίσθιο (backward) $\overleftarrow{f}$ που την αναλύει από την λέξη w_N ως την w_1. Αποκτούμε την τελική επισήμανση για κάθε λέξη h_i, συνενώνοντας (concatenating) τις επισημάνσεις και στις δύο κατευθύνσεις, $h_i = \overrightarrow{h_i} \parallel \overleftarrow{h_i}, \quad h_i \in R^{2L}$, όπου $\parallel$ σηματοδοτεί την πράξη συνένωσης και L το μέγεθος του κάθε LSTM. Όταν το μοντέλο αρχικοποιείται με προεκπαιδευμένα γλωσσικά μοντέλα, χρησιμοποιούμε LSTM μονής κατεύθυνσης (uni-directional) αντί για αμφίδρομα.

Επίπεδο προσοχής (Attention Layer)

Για να υπογραμμίσουμε τη συνεισφορά των λέξεων που εμπεριέχουν την περισσότερη πληροφορία, ενισχύουμε το LSTM μας με ένα μηχανισμό προσοχής, που αναθέτει ένα βάρος a_i σε κάθε επισήμανση λέξης h_i. Υπολογίζουμε την αναπαράσταση r όλου του μηνύματος εισόδου, ως το σταθμισμένο άθροισμα όλων των επισημάνσεων λέξεων.

$$e_i = tanh(W_h h_i + b_h), \quad e_i \in [-1, 1] \tag{4.1}$$

$$a_i = \frac{exp(e_i)}{\sum_{t=1}^{T} exp(e_t)}, \quad \sum_{i=1}^{T} a_i = 1 \tag{4.2}$$

$$r = \sum_{i=1}^{T} a_i h_i, \quad r \in R^{2L} \tag{4.3}$$

όπου W_h και b_h τα βάρη του δικτύου προσοχής.

Επίπεδο εξόδου (output Layer)
Χρησιμοποιούμε την αναπαράσταση r ως ένα διάνυσμα χαρακτηριστικών για ταξινόμηση και το δίνουμε ως είσοδο σε ένα softmax επίπεδο ταξινόμησης με L νευρώνες, το οποίο δίνει ως έξοδο μια κατανομή πιθανότητας σε όλες τις κλάσεις p_c, όπως περιγράφεται στην Εξίσωση 4.4:

$$p_c = \frac{e^{Wr+b}}{\sum_{i\in[1,L]}(e^{W_i r+b_i})} \tag{4.4}$$

όπου W και b τα βάρη και το bias του δικτύου αντίστοιχα.

4.3.1 Διανύσματα Λέξεων (Word Embeddings)

Στην πρώτη προσέγγιση, χρησιμοποιούμε *word2vec* διανύσματα λέξεων για να αρχικοποιήσουμε το embedding layer του δικτύου μας. Τα βάρη του embedding layer παραμένουν παγωμένα κατά τη διάρκεια της εκπαίδευσης. Τα *word2vec* έχουν εκπαιδευτεί σε 550,000,000 προτάσεις από το Twitter, με αρνητική δειγματοληψία (negative sampling) ίση με 5 και ελάχιστο πλήθος λέξεων 20, χρησιμοποιώντας την υλοποίηση του Gensim [101]. Το λεξιλόγιο που προκύπτει περιέχει 800, 000 λέξεις.

4.3.2 Προεκπαιδευμένος Ταξινομητής

Στη δεύτερη προσέγγιση, εκπαιδεύουμε αρχικά ένα μοντέλο ανάλυσης συναισθημάτων στο *Sent 17* σύνολο δεδομένων, χρησιμοποιώντας την αρχιτεκτονική που περιγράφεται στην ενότητα 4.3. Το embedding layer του δικτύου αρχικοποιείται με τα προεκπαιδευμένα διανύσματα λέξεων. Έπειτα, προσαρμόζουμε (fine-tune) το δίκτυο στο IEST σύνολο δεδομένων, αντικαθιστώντας το τελευταίο επίπεδο (επίπεδο ταξινόμησης) του προηγούμενου δικτύου με ένα νέο επίπεδο ταξινόμησης για το συγκεκριμένο πρόβλημα.

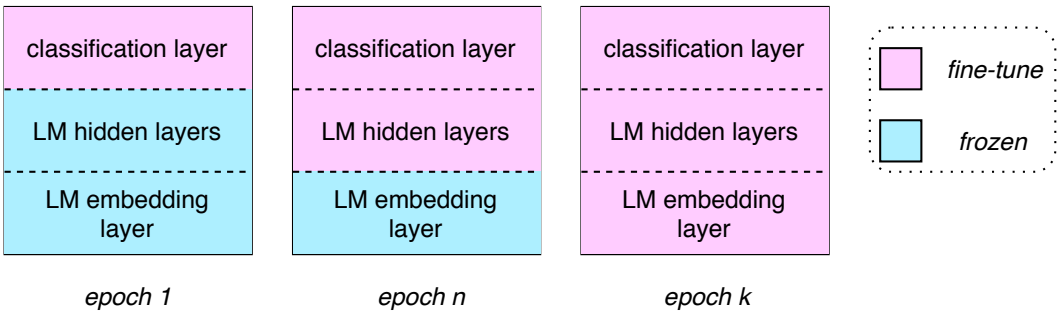
4.3.3 Προεκπαιδευμένο Γλωσσικό Μοντέλο

Στην τρίτη προσέγγιση, ακολουθούμε τα εξής βήματα: (1) εκπαιδεύουμε ένα γλωσσικό μοντέλο (LM) σε ένα σύνολο δεδομένων από το Twitter, (2) προσαρμόζουμε το LM στο πρόβλημα που έχουμε και τελικά, (3) μεταφέρουμε το embedding και το LSTM επίπεδο του LM, προσθέτουμε ένα μηχανισμό προσοχής και ένα επίπεδο ταξινόμησης (επίπεδο εξόδου) και εκπαιδεύουμε το μοντέλο στο πρόβλημα που έχουμε.

Προεκπαίδευση του LM (LM Pretraining)
Συλλέγουμε τρία σύνολα δεδομένων από το Twitter, όπως περιγράφεται στην Ενότητα 4.4.1 και για κάθε σύνολο, εκπαιδεύουμε ένα LM. Σε κάθε σύνολο δεδομένων χρησιμοποιούμε τις 50,000 πιο συχνές λέξεις σαν λεξιλόγιο. Καθώς η βιβλιογραφία σχετικά με τη μεταφορά LM είναι περιορισμένη, ιδιαίτερα στον τομέα του Twitter, στοχεύουμε να προσεγγίσουμε τα επιθυμητά χαρακτηριστικά του προεκπαιδευμένου LM. Για να το κάνουμε αυτό, η συνεισφορά μας σε αυτό το ερευνητικό πεδίο είναι η διεξαγωγή πειραμάτων με ένα corpus σχετικό με το task μας (EmoCorpus), ένα γενικό corpus (GenCorpus) και μία μίξη και των δύο (EmoCorpus+).

Προσαρμογή του LM (LM Fine-tuning)
Αυτό το βήμα είναι σημαντικό επειδή, παρά την ποικιλία των δεδομένων γενικού περιεχομένου που χρησιμοποιούμε στην προεκπαίδευση, τα δεδομένα του target task πιθανότατα θα έχουν πολύ διαφορετική κατανομή.

Επομένως, προσαρμόζουμε τα τρία προεκπαιδευμένα LMs στο IEST σύνολο δεδομένων, ακολουθώντας δύο προσεγγίσεις. Η πρώτη είναι από fine-tuning, σύμφωνα με το οποίο όλα τα επίπεδα του νευρωνικού δικτύου εκπαιδεύονται ταυτόχρονα. Η δεύτερη είναι μια απλοποιημένη αλλά παρόμοια προσέγγιση με αυτή του *gradual unfreezing*, που προτάθηκε από [100], την οποία θα συμβολίσουμε ως *Sequential Unfreezing* (SU). Σύμφωνα με αυτή τη μέθοδο, αφού μεταφέρουμε τα βάρη των embedding και LSTM επιπέδων, επιτρέπουμε μόνο στο τελικό επίπεδο να εκπαιδεύεται για $n-1$ εποχές. Στην n-οστή εποχή, αφήνουμε και τα δύο LSTM επίπεδα να εκπαιδευτούν. Επιτρέπουμε στο μοντέλο να κάνει fine-tune, ως την εποχή $k-1$. Τελικά, στην εποχή k, επιτρέπουμε επίσης στο embedding layer να εκπαιδευτεί ως τη σύγκλιση. Με άλλα λόγια, διεξάγουμε πειράματα με ζεύγη αριθμών {n, k}, όπου n υποδεικνύει την εποχή που θα επιτρέψουμε στα LSTM επίπεδα να εκπαιδευτούν και k η εποχή που θα αφήσουμε και το embedding layer να εκπαιδευτεί. Το να εκτελέσουμε απλό fine-tuning, στην περίπτωση μας, ενέχει τον κίνδυνο του catastrophic forgetting, ή αλλιώς την απότομη απώλεια της γνώσης επίλυσης ενός προηγούμενου προβλήματος, μιας και οι πληροφορίες που είναι σχετικές στο εν λόγω πρόβλημα ενσωματώνονται. Οπότε, για να αποφύγουμε μία τέτοια κατάσταση, αφήνουμε το μοντέλο να εκπαιδευτεί σταδιακά, ξεκινώντας από το τελικό επίπεδο, που είναι το πιο ειδικό, και καταλήγοντας μετά από κάποιες εποχές να εκπαιδεύουμε και τα πιο γενικά επίπεδα (LSTM layer, embedding layer), μέχρι που όλα τα επίπεδα εκπαιδεύονται. Η μέθοδός μας φαίνεται στο Σχήμα 4.3.



Σχήμα 4.3: Sequential unfreezing. Αφήνουμε τα κρυφά επίπεδα του LM να εκπαιδευτούν στην εποχή n, και αφήνουμε το επίπεδο εισόδου (embedding layer) του LM να εκπαιδευτεί στην εποχή k.

Μεταφορά του LM (LM Transfer)

Αυτό είναι το τελευταίο βήμα της TL προσέγγισής μας. Έχουμε τώρα αρκετά LMs από το δεύτερο βήμα της διαδικασίας. Μεταφέρουμε τα βάρη των embedding και LSTM layers σε ένα τελικό ταξινομητή. Πειραματιζόμαστε και πάλι τόσο με απλές όσο και με περίπλοκες τεχνικές προσαρμογής (fine-tuning), για να προσδιορίσουμε ποια είναι πιο χρήσιμη για το δικό μας πρόβλημα.

Επιπλέον, εισάγουμε μία μέθοδο συνένωσης (*concatenation method*), η οποία προσπαθεί να αξιοποιήσει τη συσχέτιση του γλωσσικού μοντέλου με το εν λόγω task. Χρησιμοποιούμε προεκπαιδευμένα LMs για να αξιοποιήσουμε το γεγονός ότι το task μας είναι στην ουσία άσκηση συμπλήρωσης (cloze test). Σε ένα LM, η δεσμευμένη πιθανότητα εμφάνισης κάθε λέξης εξαρτάται από τις λέξεις που έχουν προηγηθεί, $P(w_t|w_1,\ldots,w_{t-1})$. Σε LMs που υλοποιούνται με RNNs, η πιθανότητα αυτή κωδικοποιείται στην κρυφή κατάσταση του RNN, $P(w_t|h_{t-1})$. Επομένως, συνενώνουμε (concatenate) την κρυφή κατάσταση του LSTM, ακριβώς πριν την λέξη που έχει αφαιρεθεί, $h_{implicit}$, με την έξοδο του μηχανισμού προσοχής, r:

$$r' = r \parallel h_{implicit}, \quad h_i \in R^{2L} \tag{4.5}$$

όπου L είναι απλά το μέγεθος κάθε LSTM, και το δίνουμε έπειτα ως είσοδο στο επίπεδο ταξινόμησης (output layer). Με τον τρόπο αυτό, προφυλάσσουμε τις πληροφορίες που έχουν κωδικοποιηθεί για την λέξη που έχει αφαιρεθεί.

4.3.4 Ensembling

Συνδυάζουμε τις προβλέψεις των 3 TL μοντέλων μας με σκοπό να αυξήσουμε την ικανότητα γενίκευσης του τελικού ταξινομητή. Για αυτό το λόγο, χρησιμοποιούμε μια προσέγγιση με διανύσματα λέξεων, μια προσέγγιση με ένα προεκπαιδευμένο μοντέλο ανάλυσης συναισθήματος, και μια προσέγγιση με ένα προεκπαιδευμένο LM. Χρησιμοποιούμε δύο τρόπους για ensembling, τον αστάθμιστο μέσο όρο (unweighted average) και την ψήφο της πλειοψηφίας (majority voting).

Αστάθμιστος Μέσος Όρος (Unweighted Average - UA)
Σε αυτή την προσέγγιση, η τελική πρόβλεψη εκτιμάται από τον αστάθμιστο μέσο όρο των posterior πιθανοτήτων για όλα τα διαφορετικά μοντέλα. Η τελική πρόβλεψη p για ένα παράδειγμα εκπαίδευσης εκτιμάται από:

$$p = \arg\max_c \frac{1}{C} \sum_{i=1}^{M} \vec{p_i}, \quad p_i \in \mathbb{R}^C \tag{4.6}$$

όπου C ο αριθμός των κλάσεων, M ο αριθμός των διαφορετικών μοντέλων, $c \in \{1, ..., C\}$ υποδηλώνει μία κλάση και $\vec{p_i}$ το πιθανοτικό διάνυσμα που υπολογίζεται από το μοντέλο $i \in \{1, ..., M\}$ χρησιμοποιώντας τη συνάρτηση softmax.

Ψήφος Πλειοψηφίας (Majority Voting - MV)
Η προσέγγιση αυτή (majority voting - MV) μετράει τις ψήφους - επιλογές όλων των διαφορετικών μοντέλων και επιλέγει την κλάση που παίρνει τις περισσότερες ψήφους. Σε σύγκριση με την UA, η MV επηρεάζεται λιγότερο από τις αποφάσεις ενός δικτύου. Ωστόσο, αυτή η προσέγγιση δεν λαμβάνει υπόψη της καθόλου τις πληροφορίες που έχουν εξαχθεί από τα μοντέλα μειοψηφίας. Για ένα task με C κλάσεις και M διαφορετικά μοντέλα, η πρόβλεψη για ένα συγκεκριμένο παράδειγμα υπολογίζεται με τον εξής τρόπο:

$$\begin{aligned} v_c &= \sum_{i=1}^{M} F_i(c) \\ p &= \arg\max_{c \in \{1,...,C\}} v_c \end{aligned} \tag{4.7}$$

όπου το v_c συμβολίζει τις ψήφους για την κλάση c από όλα τα μοντέλα, F_i είναι η απόφαση του i-οστού μοντέλου, η οποία είναι είτε 1 ή 0 σε σχέση με το αν ένα μοντέλο ταξινομήθηκε στην κλάση c ή όχι και p είναι η τελική πρόβλεψη.

4.4 Πειράματα & Αποτελέσματα

4.4.1 Πειραματικό Σύνολο Δεδομένων

Εκτός από το IEST σύνολο δεδομένων, χρησιμοποιούμε ένα σύνολο δεδομένων από τον SemEval για ταξινόμηση συναισθήματος (*Sent 17*) καθώς και άλλα μη επισημασμένα σύνολα δεδομένων για τα γλωσσικά μοντέλα.

Σύνολο δεδομένων από Twitter χωρίς ετικέτες (labels)
Μαζέψαμε ένα σύνολο δεδομένων που αποτελείται από 550 εκατομμύρια αγγλικά tweet από το 2014 ως το 2017. Αυτό το σύνολο δεδομένων χρησιμοποιήθηκε για τον υπολογισμό στατιστικών μεταξύ λέξεων και προεπεξεργασία του κειμένου μας, όπως και για την εκπαίδευση των *word2vec* διανυσμάτων λέξεων που περιγράφονται στην ενότητα 4.3.1.

Για την εκπαίδευση των LMs, που περιγράφεται στην ενότητα 4.3.3, χρησιμοποιήσαμε τρία υποσύνολα του corpus αυτού. Το πρώτο περιέχει 2 εκατομμύρια (2M) tweets που περιέχουν όλα συναισθηματικά φορτισμένες λέξεις. Για να δημιουργήσουμε αυτό το σύνολο δεδομένων, επιλέξαμε tweets που περιλάμβαναν μία από τις 6 κατηγορίες συναισθημάτων του προβλήματος που θέλουμε να επιλύσουμε (*θυμός, αηδία, φόβος, χαρά, λύπη* και *έκπληξη*) ή συνώνυμα. Διασφαλίσαμε ότι το σύνολο δεδομένο περιέχει ίδιο αριθμό προτάσεων σε κάθε κατηγορία (balanced) συνδυάζοντας περίπου 350,000 tweets από κάθε κατηγορία. Το δεύτερο κομμάτι έχει 5,000,000 tweets, τυχαία επιλεγμένα από το αρχικό corpus. Στοχεύσαμε στη δημιουργία ενός γενικού υποσυνόλου δεδομένων, για να εστιάσουμε στις δομικές σχέσεις των λέξεων, αντί για το συναισθηματικό τους περιεχόμενο. Το τρίτο κομμάτι δημιουργήθηκε από τα δύο προηγούμενα. Συνδυάσαμε το 2M corpus με το 2,000,000 tweets από το γενικό corpus, δημιουργώντας ένα τελικό σύνολο δεδομένων 4,000,000 tweets (4M). Συμβολίζουμε τα τρία σύνολα δεδομένων (corpora) ως *EmoCorpus* (2M), *EmoCorpus+* (4M) και *GenCorpus* (5M).

Σύνολο Δεδομένων Ανάλυσης Συναισθήματος
Χρησιμοποιούμε το σύνολο δεδομένων του SemEval17 Task4A (Sent17) [102] για να εκπαιδεύσουμε το μοντέλο ταξινόμησης συναισθήματος, όπως περιγράφεται στην ενότητα 4.3.2. Το σύνολο δεδομένων δημιουργήθηκε από μηνύματα από Twitter επισημασμένα με το είδος του συναισθήματος που εκφράζουν (*θετικό, αρνητικό, ουδέτερο*). Το σύνολο δεδομένων εκπαίδευσης (training set) περιέχει 56,000 tweets και το σύνολο δεδομένων επαλήθευσης (validation set) περιέχει 6,000 tweets.

4.4.2 Πειραματική Διάταξη

Εκπαίδευση
Χρησιμοποιούμε τον Adam [103] για να βελτιστοποιήσουμε τα δίκτυά μας με mini-batches μεγέθους 64 και "ψαλιδίζουμε" τη νόρμα των gradients (norm clipping) [104] στο 0.5, ως ένα επιπλέον μέτρο ασφάλειας για το πρόβλημα της απότομης αύξησης των gradients (exploding gradients). Χρησιμοποιήσαμε επίσης PyTorch [105] και Scikit-learn [106].

Υπερπαράμετροι
Για όλα τα μοντέλα μας, χρησιμοποιούμε την ίδια αρχιτεκτονική με LSTM 2 επιπέδων και μηχανισμό προσοχής. (Ενότητα 4.3). Όλες οι υπερπαράμετροι που χρησιμοποιήθηκαν φαίνονται στον Πίνακα 4.1.

Layer	P-Emb	P-Sent	P-LM
Embedding	300	300	400
Embedding noise	0.1	0.1	0.1
Embedding dropout	0.2	0.2	0.2
LSTM size	400	400	600/800
LSTM dropout	0.4	0.4	0.4

Πίνακας 4.1: Υπερπαράμετροι των μοντέλων μας.

4.4.3 Αποτελέσματα

Baselines
Στον Πίνακα 4.3 συγκρίνουμε την προτεινόμενη TL προσέγγιση με δύο ισχυρά baselines: (1) ένα μοντέλο Bag-of-Words (BoW) με TF-IDF βάρη και (2) ένα Bag-of-Embeddings (BoE) μοντέλο, όπου αντλούμε τις *word2vec* αναπαραστάσεις των λέξεων σε ένα tweet και υπολογίζουμε την αναπαράστασή του tweet ως το κεντροειδές των *word2vec* ανπαραστάσεων των λέξεων από τις οποίες αποτελείται. Τόσο τα χαρακτηριστικά από το *BoW* όσο και τα χαρακτηριστικά από το *BoE* δίνονται έπειτα ως είσοδος σε ένα γραμμικό SVM ταξινομητή, με ρυθμισμένη την παράμετρο $C = 0.6$.

Μοντέλα P-Emb και P-Sent (4.3.1, 4.3.2)
Αξιολογούμε τα μοντέλα *P-Emb* και *P-Sent* χρησιμοποιώντας τόσο αμφίδρομα όσο και LSTM απλής κατεύθυνσης. Το F1 score των ensembling μοντέλων μας φαίνεται στον Πίνακα 4.3. Όπως ήταν αναμενόμενο, τα αμφίδρομα LSTM μοντέλα έχουν καλύτερη απόδοση.

LM Fine-tuning	LM Transfer			
	Simple fine-tuning	Sequential Unfreezing	Concat.	F1
Simple fine-tuning	✓			67.2
	✓		✓	66.7
		✓		67.6
		✓	✓	67.3
Sequential Unfreezing	✓			67.3
	✓		✓	66.7
		✓		67.8
		✓	✓	**68.2**

Πίνακας 4.2: Αποτελέσματα του γλωσσικού μοντέλου (P-LM), εκπαιδευμένου στο EmoCorpus. Η πρώτη στήλη αναφέρεται στην διαδικασία προσαρμογής που ακολουθείται στο βήμα (*LM Fine-tuning*), ενώ η δεύτερη στήλη περιγράφει τον τρόπο εκπαίδευσης στο βήμα της μεταφοράς του LM (*LM Transfer*). Με Concat. συμβολίζουμε τη μέθοδο συνένωσης (*concatenation method*).

P-LM (4.3.3)
Για τα πειράματα που διεξάγουμε με προεκπαιδευμένα LMs, αποσκοπούμε στο να μεταφέρουμε όχι απλά το πρώτο επίπεδο του δικτύου μας, αλλά ολόκληρο το μοντέλο, για να μοντελοποιήσουμε υψηλού επιπέδου χαρακτηριστικά της γλώσσας. Όπως αναφέρθηκε παραπάνω, υπάρχουν τρία διακριτά βήματα σχετικά με τη διαδικασία εκπαίδευσης αυτή της TL προσέγγισης: (1) *Προεκπαίδευση του LM - LM pretraining*: εκπαιδεύουμε 3 LMs στα EmoCorpus, EmoCorpus+ και GenCorpus σύνολα δεδομένων αντίστοιχα, (2) *Προσαρμογή του LM - LM fine-tuning*: προσαρμόζουμε (fine-tune) τα LMs στο σύνολο δεδομένων IEST με 2 διαφορετικούς τρόπους. Ο πρώτος είναι να εκπαιδεύουμε ταυτόχρονα όλα τα επίπεδα (simple fine-tuning), ενώ ο δεύτερος είναι η τεχνική sequential unfreezing (SU) που προτείνουμε. (3) *LM transfer*: Έχουμε τώρα 6 LMs, προσαρμοσμένα στο σύνολο δεδομένων IEST. Μεταφέρουμε τα βάρη τους στον τελικό ταξινομητή μας στα 6 βασικά συναισθήματα, προσθέτοντας ένα μηχανισμό προσοχής στα LSTM επίπεδα και διεξάγουμε ξανά πειράματα με τις μεθόδους fine-tuning και *concatenation*, που προτάθηκαν στην ενότητα 4.3.3.

Όταν εκπαιδεύεται στο EmoCorpus, το P-LM μοντέλο έχει F1-score 68.2%. Όταν εκπαιδεύεται στο EmoCorpus+, το F1-score αποκτά τιμή ίση με 68.0%. Τελικά, όταν εκπαιδεύεται στο GenCorpus, αποκτά F1-score ίσο με 67.5%. Αν και το EmoCorpus περιέχει λιγότερα παραδείγματα εκπαίδευσης, τα *P-LM*s που εκπαιδεύτηκαν σε αυτό μαθαίνουν να κωδικοποιούν πιο χρήσιμες πληροφορίες για το πρόβλημα που αντιμετωπίζουμε.

Στον Πίνακα 4.2 παρουσιάζουμε όλους τους πιθανούς συνδυασμούς όταν μεταφέρουμε το *P-LM* στο σύνολο δεδομένων του IEST. Παρατηρούμε ότι η τεχνική του sequential unfreezing δίνει συνεχώς καλύτερα αποτελέσματα από την τεχνική του simple fine-tuning. Επίσης συγκρίνουμε την προσέγγιση που μας δίνει γενικά τα καλύτερα αποτελέσματα, δηλαδή την *SU + Concat.*, με *P-LM*s εκπαιδευμένα σε τρία διαφορετικά σύνολα δεδομένων από Twitter.

Το μοντέλο μας είναι ensemble των μοντέλων με την καλύτερη απόδοση. Συγκεκριμένα, αξιοποιούμε τα εξής μοντέλα: (1) μεταφορά μάθησης (TL) από προεκπαιδευμένα διανύσματα λέξεων, (2) TL από προεκπαιδευμένο ταξινομητή ανάλυσης συναισθήματος, (3) TL από 3 διαφορετικά LMs, που εκπαιδεύτηκαν σε 2M, 4M and 5M tweets αντίστοιχα. Χρησιμοποιούμε αστάθμιστο μέσο όρο για το ensembling των καλύτερων μοντέλων μας από όλες τις προαναφερθείσες προσεγγίσεις. Τα τελικά αποτελέσματά μας φαίνονται στον Πίνακα 4.3.

Model	F1
Bag of Words (BoW)	60.1
Bag of Embeddings (BoE)	60.5
Ensembling (UA) P-Emb + P-Sent	68.4
Ensembling (UA) P-Sent + P-LM	69.5
Ensembling (UA) P-Emb + P-LM	70.1
Ensembling (MV) All	70.0
Ensembling (UA) All	**70.2**

Πίνακας 4.3: Αποτελέσματα των πειραμάτων. Τα *BoW* και *BoE* είναι τα baseline μοντέλα μας, ενώ *P-Emb*, *P-Sent* και *P-LM* είναι οι τρεις προσεγγίσεις TL που προτείνουμε. Το *UA* συμβολίζει τον αστάθμιστο μέσο όρο (Unweighted Average) και το *MV* την ψήφο πλειοψηφίας (Majority Voting).

4.4.4 Συζήτηση

Όπως φαίνεται στον Πίνακα 4.3, παρατηρούμε ότι όλα τα προτεινόμενα μοντέλα πετυχαίνουν πολύ καλύτερη απόδοση από τα baselines με μεγάλη διαφορά. Σε ό,τι αφορά το ensembling, και οι δύο προσεγγίσεις, *MV* and *UA*, απέφεραν παρόμοια βελτίωση της απόδοσης σε σχέση με τα μεμονωμένα μοντέλα. Παρατηρούμε ότι προσθέτοντας την πρόβλεψη του *P-LM* μοντέλου στο ensemble μοντέλο έχουμε τη μεγαλύτερη βελτίωση. Αυτό μπορεί να ερμηνευτεί ως ότι τα *P-LMs* κωδικοποιούν περισσότερες πτυχές της πληροφορίας από ότι οι υπόλοιπες προσεγγίσεις.

Επιλέον διενεργούμε μια ανάλυση των αποτελεσμάτων μας, για να βρούμε τι βοήθησε περισσότερο στην τελική απόδοση του μοντέλου. Τα αποτελέσματα φαίνονται στον Πίνακα 4.4. Παρατηρούμε ότι όταν και τα 3 μοντέλα εκπαιδεύονται με ένα LSTM μονής κατεύθυνσης και το ίδιο πλήθος παραμέτρων, το *P-LM* έχει καλύτερη απόδοση τόσο από το *P-Emb* όσο και από το*P-Sent* μοντέλο. Όπως ήταν αναμενόμενο, όταν χρησιμοποιούμε αμφίδρομο LSTM, τα αποτελέσματα βελτιώνονται για τα μοντέλα *P-Emb* και *P-Sent*. Είναι πιθανό ότι η ενισχυμένη αναπαράσταση χαρακτηριστικών που κωδικοποιεί το αμφίδρομο επίπεδο επιτρέπει στο μοντέλο να κάνει καλύτερες προβλέψεις για το συναίσθημα μίας πρότασης. Υποθέτουμε ότι το *P-LM* με αμφίδρομα προεκπαιδευμένα γλωσσικά μοντέλα θα είχε καλύτερη απόδοση και από τα δύο. Επιπλέον, καταλήγουμε ότι τόσο το sequential unfreezing όσο και η μέθοδος συνένωσης (concatenation method) ενισχύουν την απόδοση της *P-LM* προσέγγισης. Σε ό,τι αφορά το sequential unfreezing, επιτρέπει σταδιακή προσαρμογή των επιπέδων του μοντέλου στο πρόβλημα που θέλουμε να επιλύσουμε. Έτσι, διατηρεί τη γνώση που είναι κωδικοποιημένη στο source task και προσαρμόζει σταδιακά τα βάρη των επιπέδων, ξεκινώντας από το επίπεδο εξόδου (task-specific) και προχωρώντας στα κρυφά επίπεδα και τελικά στο embedding layer (που είναι το πιο γενικό).

TL Model	F1	TL Model	F1	TL Model	F1
P-Emb	66.8	P-Sent	67.1	P-LM	67.5
P-Emb + bidir.	**68.4**	P-Sent + bidir.	**67.4**	P-LM + SU	67.9
P-Emb + SU	65.4	P-Sent + SU	66.5	P-LM + SU + Concat.	**68.2**
P-Emb + SU + Concat.	66.4	P-Sent + SU + Concat.	66.8		

Πίνακας 4.4: Ανάλυση συνεισφοράς των προτεινόμενων προσεγγίσεων μεταφοράς μάθησης, δηλαδή των *P-Emb*, *P-Sent* και *P-LM*. Με *SU* συμβολίζεται το Sequential Unfreezing, *bidir.* το αμφίδρομο LSTM, *Concat.* η μέθοδος συνένωσης.

4.5 Συμπεράσματα

Σε αυτό το paper, περιγράφουμε τις μεθόδους βαθιάς μάθησης για την ταξινόμηση tweets ανάλογα με το συναίσθημα που εκφράζουν, έχοντας αφαιρέσει την συναισθηματικά φορτισμένη λέξη από κάθε tweet. Η προτεινόμενη μέθοδος βασίζεται σε ένα ensemble από διαφορετικές τεχνικές μεταφοράς μάθησης (transfer learning). Παρέχουμε εμπειρικά αποτελέσματα όπου φαίνεται πως η χρήση χαρακτηριστικών υψηλού επιπέδου από κείμενο, όπως αυτά που κωδικοποιούν τα γλωσσικά μοντέλα (language models), οδηγούν σε καλύτερη απόδοση. Στο μέλλον, σχεδιάζουμε να διεξάγουμε πειράματα με μοντέλα σε επίπεδο φωνημάτων (subword-level), καθώς έχουν δείξει ότι είναι ικανά να προσπεράσουν το πρόβλημα των λέξεων που βρίσκονται στα δεδομένα ελέγχου, ενώ δε βρίσκονται στα δεδομένα εισόδου (out-of-vocabulary) [107, 108], το οποίο είναι πολύ εμφανές στο Twitter. Επιπλέον, θα θέλαμε να εξερευνήσουμε άλλες προσεγγίσεις μεταφοράς μάθησης.

Τέλος, μοιραζόμαστε τον κώδικα που χρησιμοποιήθηκε για την εκπαίδευση των μοντέλων μας [1], για να επιτρέψουμε και ενθαρρύνουμε την αναπαραγωγή των αποτελεσμάτων μας και την διεξαγωγή σχετικών πειραμάτων σε αυτόν τον τομέα.

[1] /github.com/alexandra-chron/wassa-2018

Κεφάλαιο 5

Μεταφορά Μάθησης από Γλωσσικά Μοντέλα με χρήση Βοηθητικής Συνάρτησης Κόστους

5.1 Εισαγωγή

Στην Επεξεργασία Φυσικής Γλώσσας (NLP), οι κατανεμημένες αναπαραστάσεις από προεκπαιδευμένα διανύσματα λέξεων όπως το word2vec [8] και το GloVe [66] αποτελούν πλέον συνήθη τρόπο αρχικοποίησης των μοντέλων βαθιάς μάθησης. Τα προεκπαιδευμένα διανύσματα λέξεων μπορούν να μοντελοποιήσουν τις ομοιότητες μεταξύ λέξεων και είναι χρήσιμες για τη σημασιολογική κατανόηση του κειμένου. Η μεταφορά πληροφοριών από μη επιβλεπόμενα δεδομένα εκπαίδευσης έχει δειχθεί ότι βελτιώνει την απόδοση σε σχέση με την τυχαία αρχικοποίηση σε πλήθος προβλημάτων του NLP, όπως η κατάτμηση σε μέρη του λόγου, αναγνώριση οντοτήτων στα ελληνικά [66] και ερώτηση - απόκριση [109]. Ωστόσο, το περιεχόμενο δε μπορεί να μοντελοποιηθεί κατάλληλα από τα προεκπαιδευμένα διανύσματα λέξεων, καθώς συνήθως αναθέτουν ένα διάνυσμα σε κάθε λέξη. Οι πολύσημες λέξεις, επομένως, δεν αναπαρίστανται σωστά και το νόημα μιας λέξης μπορεί πολλές φορές να διαφεύγει, δεδομένου ότι λαμβάνει ένα μοναδικό διάνυσμα σε όλες τις εμφανίσεις της για αναπαράσταση.

Οι προεκπαιδευμένες αναπαραστάσεις λέξεων από γλωσσικά μοντέλα (language μοντέλα - LMs) έχουν γίνει πρόσφατα δημοφιλείς στο NLP. Τα προεκπαιδευμένα LMs κωδικοποιούν πληροφορίες για το περιεχόμενο και μοντελοποιούν τα υψηλού επιπέδου χαρακτηριστικά της γλώσσας, μοντελοποιώντας τη σύνταξη και τη σημασιολογία, παράγοντας state-of-the-art αποτελέσματα σε ένα μεγάλο εύρος προβλημάτων, όπως named entity recognition [27], αυτόματη μηχανική μετάφραση [110] και ταξινόμηση κειμένου [25]. Αναθέτουν διαφορετικό διάνυσμα σε κάθε εμφάνιση μιας συγκεκριμένης λέξης, βάσει των γειτονικών της λέξεων. Άρα, προσαρμόζονται στην αλλαγή τομέα (domain) και παρέχουν πλούσιες αναπαραστάσεις ως προς το περιεχόμενο (contextual embeddings).

Ωστόσο, όταν τα contextual embeddings από LMs χρησιμοποιούνται ως πρόσθετα χαρακτηριστικά (π.χ ELMo [32]), για να πάρουμε καλά αποτελέσματα απαιτούνται ειδικές αρχιτεκτονικές για κάθε υπο-πρόβλημα (task). Ταυτόχρονα, οι προσεγγίσεις που βασίζονται στην προσαρμογή (fine-tuning) ενός LM σε ένα task (π.χ. ULMFiT [25]) εξαρτώνται από την προεκπαίδευση ενός μοντέλου σε εκτενές λεξιλόγιο και από τη χρήση πολύπλοκων τεχνικών προσαρμογής του ρυθμού εκμάθησης. Προτείνουμε μία απλή και αποτελεσματική προσέγγιση για μεταφορά μάθησης, η οποία αξιοποιεί contextual embeddings από LMs και δεν απαιτεί περίπλοκο σχεδιασμό της προσαρμογής του ρυθμού εκμάθησης. Αρχικά εκπαιδεύουμε το LM σε ένα σύνολο δεδομένων από το Twitter και έπειτα μεταφέρουμε τα βάρη του. Προσθέτουμε ένα αναδρομικό επίπεδο (RNN layer) και ένα επίπεδο ταξινόμησης. Το μοντέλο αυτό εκπαιδεύεται για ταξινόμηση με μια βοηθητική συνάρτηση κόστους του LM, η οποία μας επιτρέπει να ελέγχουμε ευθέως τη συνεισφορά του προεκπαιδευμένου μέρος του μοντέλου και να διασφαλίζουμε ότι η γνώση που έχει αποκτήσει διατηρείται.

Οι συνεισφορές αυτού του paper συνοψίζονται στα παρακάτω σημεία:

- Δείχνουμε ότι η μεταφορά μάθησης (transfer learning) από LMs επιτυγχάνει ανταγωνιστικά αποτελέσματα, ενώ είναι διαισθητικά απλή και υπολογιστικά εφικτή.
- Αντιμετωπίζουμε το πρόβλημα του catastrophic forgetting, προσθέτοντας μια βοηθητική συνάρτηση κόστους του LM και χρησιμοποιώντας μια μέθοδο unfreezing.

- Τα αποτελέσματα δείχνουν ότι η προσέγγισή μας φέρνει αποτελέσματα αντίστοιχα με πιο πολύπλοκες μεθόδους μεταφοράς μάθησης.

5.2 Σχετική Βιβλιογραφία

Η μη επιβλεπόμενη προεκπαίδευση έχει παίξει σημαντικό ρόλο στα βαθιά νευρωνικά δίκτυα, καθώς αναπαραστάσεις που έχουν δημιουργηθεί με εκπαίδευση σε ένα task μπορούν να είναι χρήσιμες για ένα άλλο task. Στο NLP, τα προεκπαιδευμένα διανύσματα λέξεων [95, 66] χρησιμοποιούνται ευρέως, βελτιώνοντας την απόδοση σε διάφορα tasks, όπως κατάτμηση σε συντακτικά μέρη του λόγου [111] και ερώτηση - απόκριση [109].

Στοχεύοντας στο να οδηγήσουν τα μοντέλα να μάθουν από δεδομένα χωρίς ετικέτες (unlabeled data) οι Dai et al. [28] χρησιμοποιούν μη επιβλεπόμενες αντικειμενικές συναρτήσεις όπως το autoencoding μιας ακολουθίας και το γλωσσικό μοντέλο για να λάβουν ποιοτικές προεκπαιδευμένες αναπαραστάσεις. Οι Ramachandran et al. [110] επίσης προεκπαιδεύουν ζεύγη encoder-decoder χρησιμοποιώντας LMs και προσαρμόζοντάς τα σε ένα συγκεκριμένο task. Τα διανύσματα ELMo [32] δημιουργούνται από αμφίδρομα νευρωνικά LMs επιπέδου χαρακτήρων, βελτιώνοντας τα αποτελέσματα σε πληθώρα προβλημάτων ως πρόσθετες αναπαραστάσεις χαρακτηριστικών.

Στην ίδια κατεύθυνση το μοντέλο ULMFiT [25] επιδεικνύει εντυπωσιακά αποτελέσματα σε πολλά προβλήματα χρησιμοποιώντας προεκπαιδευμένα LMs. Το προτεινόμενο μοντέλο απαιτεί τρία διακριτά βήματα, που περιλαμβάνουν προεκπαίδευση ενός LM, προσαρμογή του στο σύνολο δεδομένων του task ταξινόμησης με μια περίπλοκη διαδικασία προσαρμογής του ρυθμού εκμάθηση, και μεταφορά του σε ένα μοντέλο ταξινόμησης.

Το Η Μάθηση Πολλαπλών Εργασιών (Multi-Task Learning) με αυστηρά κοινές παραμέτρους (hard parameter sharing) [112] στα νευρωνικά δίκτυα έχει αποδειχθεί αποτελεσματικό σε πλήθος tasks [113]. Πρόσφατα, εναλλακτικές προσεγγίσεις προτάθηκαν, σύμφωνα με τις οποίες τα μοντέλα μοιράζονται παραμέτρους μόνο στα κατώτερα (πιο γενικά) δίκτυα [114]. Εισάγοντας επισημειωμένα μέρη του λόγου (part of speech tags) στα κατώτερα επίπεδα του δικτύου, το προτεινόμενο μοντέλο πετυχαίνει εύρωστα αποτελέσματα σε προβλήματα chunking και Combinatory Category Grammar (CCG) super tagging. Η βοηθητική συνάρτηση κόστους του LM που προτείνουμε ακολουθεί αυτή τη προσέγγιση και προσπαθεί να βελτιώσει την απόδοση του υψηλότερου επιπέδου ταξινόμησης.

5.3 Προτεινόμενο Μοντέλο

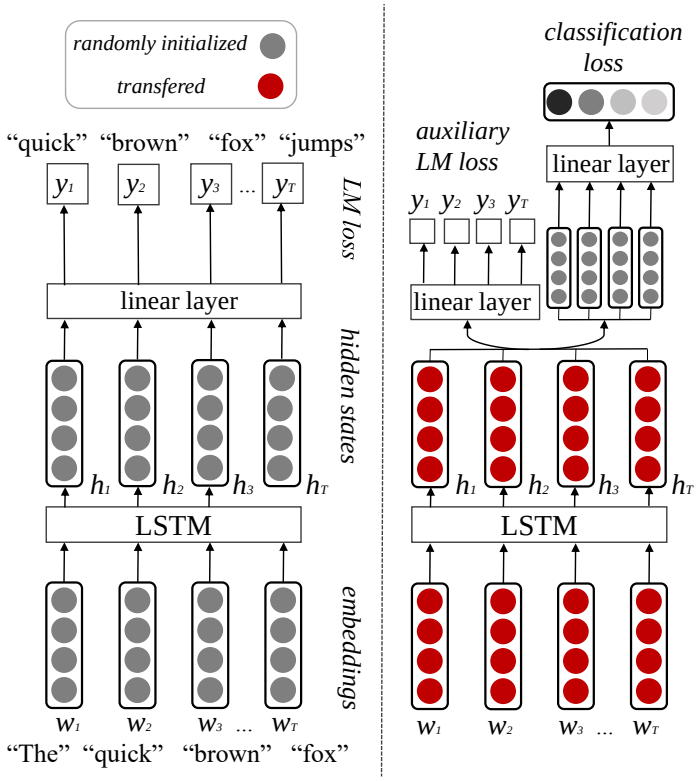
Προτείνουμε το **SiATL**, που συμβολίζει τη **Si**ngle-step **A**uxiliary loss **T**ransfer **L**earning (μεταφορά μάθησης με βοηθητική συνάρτηση σε ένα βήμα). Αρχικά εκπαιδεύουμε ένα LM. Έπειτα, μεταφέρουμε τα βάρη του σε ένα μοντέλο ταξινόμησης και προσθέτουμε σε αυτό και ένα αναδρομικό επίπεδο (recurrent layer). Επίσης χρησιμοποιούμε μια βοηθητική LM συνάρτηση κόστους για να αποφύγουμε το catastrophic forgetting.

5.3.1 Μη Επιβλεπόμενη Προεκπαίδευση (Unsupervised Pretraining)

Εκπαιδεύουμε ένα γλωσσικό μοντέλο (LM) σε επίπεδο λέξεων, που αποτελείται από ένα embedding LSTM επίπεδο [53], 2 κρυφά LSTM επίπεδα και ένα επίπεδο εξόδου. Θέλουμε να ελαχιστοποιήσουμε την αρνητική συνάρτηση λογαριθμικής πιθανοφάνειας του LM:

$$L(\hat{p}) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T^n}\log\hat{p}(x_t^n|x_1^n,...,x_{t-1}^n) \tag{5.1}$$

όπου $\hat{p}(x_t^n|x_1^n,...,x_{t-1}^n)$ η κατανομή της $t-$οστής λέξης στην n-οστή πρόταση, δεδομένου ότι $t-1$ λέξεις βρίσκονται πριν από αυτήν στην ακολουθία και με N τον τελικό αριθμό προτάσεων.



Σχήμα 5.1: Εποπτική αναπαράσταση του προτεινόμενου μοντέλου. Μεταφέρουμε το προεκπαιδευμένο LM και προσθέτουμε ένα πρόσθετο αναδρομικό επίπεδο και μια βοηθητική αντικειμενική συνάρτηση του LM.

5.3.2 Μεταφορά & Βοηθητική Συνάρτηση Κόστους

Μεταφέρουμε τα δίκτυα από το προεκπαιδευμένο μοντέλο και προσθέτουμε ένα LSTM με μηχανισμό προσοχής [115, 13].

Για να προσαρμόσουμε τη συνεισφορά του προεκπαιδευμένου μοντέλου στο νέο task, εισάγουμε μία βοηθητική συνάρτηση κόστους του LM κατά τη διάρκεια της εκπαίδευσης. Η κοινή συνάρτηση κόστους είναι το σταθμισμένο άθροισμα του κόστους ταξινόμησης L_{task} και του κόστους του βοηθητικού LM L_{LM}, όπου γ η παράμετρος στάθμισης που επιτρέπει προσαρμογή στο target task και ταυτόχρονα διατηρεί τις χρήσιμες πληροφορίες από το αρχικό task. Συγκεκριμένα:

$$L = L_{task} + \gamma L_{LM} \tag{5.2}$$

5.3.3 Εκθετική Μείωση της Παραμέτρου Στάθμισης της Κοινής Συνάρτησης Κόστους

Ένα πλεονέκτημα του προτεινόμενου μοντέλου είναι ότι η συνεισφορά του LM μπορεί να ελεγχθεί σε κάθε εποχή εκπαίδευσης. Στις πρώτες λίγες εποχές, το LM πρέπει να συνεισφέρει περισσότερο στην κοινή συνάρτηση κόστους του SiATL ούτως ώστε το καινούργιο αναδρομικό επίπεδο (RNN layer) να προσαρμοστεί στην κατανομή των λέξεων. Αφού η γνώση του προεκπαιδευμένου LM μεταφερθεί σε ένα νέο τομέα, η συνάρτηση κόστους ταξινόμησης είναι πιο σημαντική και η παράμετρος στάθμισης(γ) θα πρέπει να πάρει μικρότερες τιμές. Σε αυτό το paper, χρησιμοποιούμε εκθετική μείωση για το γ όσο περνούν οι εποχές εκπαίδευσης.

5.3.4 Σταδιακή Προσαρμογή (Sequential Unfreezing)

Αντί να εκπαιδεύουμε όλα τα επίπεδα ταυτόχρονα στο τελικό μοντέλο, προτείνουμε τη σταδιακή εκπαίδευσή τους με την τεχνική sequential unfreezing, που προτάθηκε από Chronopoulou et al. [2]. Αρχικά προσαρμόζουμε (fine-tune) μόνο το καινούργιο, τυχαία αρχικοποιημένο LSTM και το επίπεδο εξόδου για $n-1$ εποχές. Στην n-στή εποχή, επιτρέπουμε και στο προεκπαιδευμένο κρυφό επίπεδο να εκπαιδευτεί. Αφήνουμε το μοντέλο να εκπαιδεύεται έτσι μέχρι την εποχή $k-1$. Τελικά, στην εποχή k, επιτρέπουμε επίσης στο embedding επίπεδο να εκπαιδευτεί (και άρα σε όλο το δίκτυο). Το δίκτυο εκπαιδεύεται ως τη σύγκλιση. Οι τιμές των n και k υπολογίζονται με προσαρμογή υπερπαραμέτρων (hyperparameter tuning). Θεωρούμε το sequential unfreezing σημαντικό, διότι ελαχιστοποιεί τον κίνδυνο του overfitting σε μικρά σύνολα δεδομένων.

5.3.5 Βελτιστοποιητές (Optimizers)

Χρησιμοποιούμε SGD για το προεκπαιδευμένο LM με μικρό ρυθμό εκμάθησης, για να διατηρήσουμε τις πληροφορίες για το περιεχόμενο και να αποφύγουμε το catastrophic forgetting. Ωστόσο, θέλουμε το επιπλέον LSTM και το επίπεδο ταξινόμησης να εκπαιδεύονται γρήγορα για να προσαρμοστούν στο target task, άρα σε αυτή την περίπτωση χρησιμοποιούμε τον βελτιστοποιητή Adam [50].

5.4 Πειράματα & Αποτελέσματα

5.4.1 Πειραματικό Σύνολο Δεδομένων

Για να εκπαιδεύσουμε το LM, μαζεύουμε ένα σύνολο δεδομένων με 20 εκατομμύρια μηνύματα στο Twitter στα αγγλικά, αποκτώντας προσεγγιστικά 2 εκατομμύρια μοναδικές λέξεις. Χρησιμοποιούμε τις 70,000 πιο συχνά εμφανιζόμενες λέξεις ως λεξιλόγιο. Αξιολογούμε το μοντέλο μας σε 5 σύνολα δεδομένων: *Sent17* για ανάλυση συναισθήματος [102], *PsychExp* για αναγνώριση συναισθημάτων [116], *Irony18* για αναγνώριση ειρωνείας [117], *SCv1* και *SCv2* για αναγνώριση σαρκασμού [118, 119]. Περισσότερες λεπτομέρειες για τα σύνολα δεδομένων μπορούν να βρεθούν στον Πίνακα 5.1.

Σύνολο Δεδομένων	Τομέας	# κλάσεων	# δεδομένων εκπαίδευσης	# δεδομένων ελέγχου
Irony18	Tweets	4	3834	784
Sent17	Tweets	3	49570	12284
SCv2	Debate Forums	2	1000	2260
SCv1	Debate Forums	2	1000	995
PsychExp	Experiences	7	1000	6480

Πίνακας 5.1: Σύνολα δεδομένων για τα προβλήματα ταξινόμησης.

5.4.2 Πειραματική Διάταξη

Για να προεπεξεργαστούμε τα μηνύματα στο Twitter (tweets), χρησιμοποιούμε τη βιβλιοθήκη *Ekphrasis* [89]. Για τα γενικά σύνολα δεδομένων, χρησιμοποιούμε τη βιβλιοθήκη NLTK [120]. Για το Neural Bag-of-Words (NBoW) βασικό μοντέλο (baseline), χρησιμοποιούμε *word2vec* [95] διανύσματα λέξεων 300 διαστάσεων. Για τα νευρωνικά μοντέλα, χρησιμοποιούμε ένα LM με μέγεθος embedding επιπέδου ίσο με 400, 2 κρυφά επίπεδα, 1000 νευρώνες ανά επίπεδο, dropout 0.2 και μέγεθος batch 32. Προσθέτουμε γκαουσιανό θόρυβο μεγέθους 0.01 στο επίπεδο embedding. "Ψαλιδίζουμε" τα gradients το 5, ως πρόσθετο μέτρο ασφάλειας για το πρόβλημα των "exploding" gradients. Για κάθε νευρωνικό δίκτυο ταξινόμησης, προσθέτουμε στο προεκπαιδευμένο LM ένα επίπεδο LSTM μεγέθους 100 με μηχανισμό προσοχής και επίπεδο ταξινόμησης Softmax. Για την ανάπτυξη του μοντέλου μας χρησιμοποιούμε PyTorch [105] και Scikit-learn [121].

	Irony18	Sent17	SCv2	SCv1	PsychExp
BoW	43.7	61.0	65.1	60.9	25.8
NBoW	45.2	63.0	61.1	51.9	20.3
P-LM	42.7 ± 0.6	61.2 ± 0.7	69.4 ± 0.4	48.5 ± 1.5	38.3 ± 0.3
P-LM + su	41.8 ± 1.2	62.1 ± 0.8	69.9 ± 1.0	48.4 ± 1.7	38.7 ± 1.0
P-LM + aux	45.5 ± 0.9	65.1 ± 0.6	72.6 ± 0.7	55.8 ± 1.0	40.9 ± 0.5
SiATL (P-LM + aux + su)	**47.0** ± 1.1	**66.5** ± 0.2	**75.0** ± 0.7	**56.8** ± 2.0	**45.8** ± 1.6
ULMFiT (Wiki-103)	23.6 ± 1.6	60.5 ± 0.5	**68.7** ± 0.6	**56.6** ± 0.5	21.8 ± 0.3
ULMFiT (Twitter)	**41.6** ± 0.7	**65.6** ± 0.4	67.2 ± 0.9	44.0 ± 0.7	**40.2** ± 1.1
State of the art	53.6	68.5	76.0	69.0	57.0
	[122]	[123]	[124]	[125]	

Πίνακας 5.2: Ανάλυση πάνω στα σύνολα δεδομένων ταξινόμησης. Παρουσιάζεται η μέση τιμή μετά από 5 εκτελέσεις του πειράματος με τυπική απόκλιση. Με *BoW* συμβολίζουμε το Bag of Words, *NBoW* το Neural Bag of Words. Το *P-LM* είναι ένα μοντέλο ταξινόμησης αρχικοποιημένο με το προεκπαιδευμένο LM, *su* για τη μέθοδο sequential unfreezing και *aux* για τη βοηθητική συνάρτηση κόστους του LM. Σε όλες τις περιπτώσεις, η μετρική που χρησιμοποιούμε είναι το F_1.

5.5 Αποτελέσματα & Συζήτηση

Baselines και Σύγκριση

Ο Πίνακας 5.2 συνοψίζει τα αποτελέσματά μας. Οι πρώτες δύο γραμμές δείχνουν με λεπτομέρεια την απόδοση των Bag-of-Words (BoW) and Neural Bag-of-Words (NBoW) μοντέλων. Παρατηρούμε ότι όταν αρκετά δεδομένα είναι διαθέσιμα (π.χ. *Sent17*), τα baselines δίνουν αξιοπρεπή αποτελέσματα. Έπειτα, τα αποτελέσματα για το γενικό ταξινομητή που αρχικοποιείται με ένα προεκπαιδευμένο LM (P-LM) φαίνονται με και χωρίς sequential unfreezing, ακολουθούμενα από τα αποτελέσματα του προτεινόμενου μοντέλου SiATL. Το SiATL επίσης συγκρίνεται με το αρκετά σχετικό μοντέλο ULMFiT (εκπαιδευμένο στο Wiki-103 ή στο Twitter) και το state-of-the-art αποτέλεσμα σε κάθε task. Το ULMFiT επίσης προσαρμόζει ένα LM για tasks ταξινόμησης. Το προτεινόμενο SiATL επιτυγχάνει σταθερά καλύτερη απόδοση από τα baselines, τη μέθοδο P-LM και το ULMFiT σε όλα τα σύνολα δεδομένων. Αν και δεν εφαρμόζουμε κάποιο περίπλοκο μηχανισμό προσαρμογής του ρυθμού εκμάθησης και περιοριζόμαστε σε προεκπαίδευση στο Twitter (και όχι σε κάποιο πιο γενικό σύνολο δεδομένων όπως η Wikipedia), σημειώνουμε καλύτερα αποτελέσματα τόσο στα 2 Twitter σύνολα δεδομένων όσο και στα 3 γενικά.

Βοηθητική Συνάρτηση Κόστους του LM

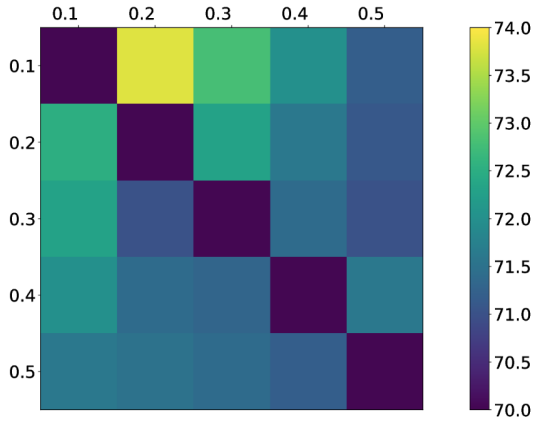
Η επίδραση της βοηθητικής αντικειμενικής συνάρτησης υπογραμμίζεται σε πολύ μικρά σύνολα δεδομένων, όπως το *SCv1*, όπου οδηγεί σε μια εντυπωσιακή βελτίωση της απόδοσης (κατά 7%). Υποθέτουμε ότι όταν ο ταξινομητής απλά αρχικοποιείται με το προεκπαιδευμένο LM, σύντομα αντιμεωπίζει πρόβλημα overfitting, καθώς το λεξιλόγιο του target task είναι πολύ περιορισμένο. Η βοηθητική LM συνάρτηση κόστους, ωστόσο, επιτρέπει λεπτές προσαρμογές του μοντέλου στο πρόβλημα ταξινόμησης που δίνεται.

Εκθετική Μείωση του γ

Για την εύρεση του βέλτιστου διαστήματος γ, παρατηρούμε εμπειρικά ότι η εκθετική μείωση του γ στο μισό της αρχικής του τιμής κατά τη διάρκεια των εποχών εκπαίδευσης παρέχει βέλτιστα αποτελέσματα στα προβλήματα ταξινόμησης. Ένα heatmap του γ φαίνεται στο Σχήμα 5.2. Παρατηρούμε το γ πρέπει να λαμβάνει μικρές τιμές, για να κλιμακώσει τη συνεισφορά της συνάρτησης κόστους του LM στην ίδια τάξη μεγέθους με τη συνάρτηση κόστους του προβλήματος ταξινόμησης.

Σταδιακή Προσαρμογή (Sequential Unfreezing)

Τα αποτελέσματα δείχνουν πως η τεχνική του sequential unfreezing είναι καίρια για την προτεινόμενη



Σχήμα 5.2: Heatmap της επίδοσης του γ στην μετρική F_1 στα δεδομένα ελέγχου *SCv2*. Ο οριζόντιος άξονας παρουσιάζει την αρχική τιμή του γ και ο οριζόντιος την τελική του τιμή.

μέθοδο, καθώς επιτρέπει στο LM να προσαρμοστεί στην κατανομή των λέξεων του target task. Η βελτίωση της επίδοσης μπορεί να φανεί πιο έντονα όταν υπάρχει αναντιστοιχία μεταξύ του τομέα στον οποίο εκπαιδεύεται το LM και σε αυτόν του προβλήματος ταξινόμησης, πχ, στα tasks των οποίων τα δεδομένα εκπαίδευσης δεν ανήκουν στον τομέα του Twitter. Ειδικά για το *PsychExp* και *SCv2*, η τεχνική sequentially unfreezing προσφέρει σημαντική αύξηση του F_1.

Αριθμός Δεδομένων Εκπαίδευσης

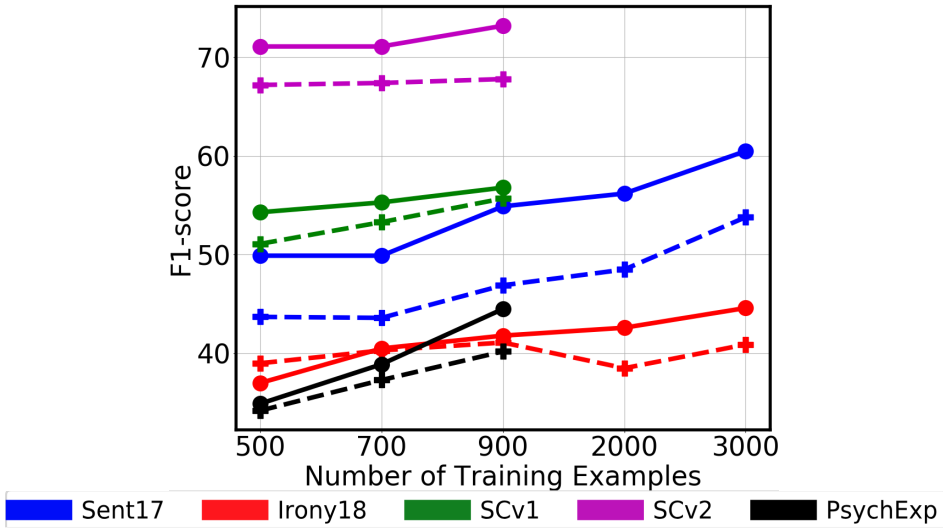
Η μεταφορά μάθησης είναι ιδιαίτερα χρήσιμη όταν περιορισμένα δεδομένα εκπαίδευσης είναι διαθέσιμα. Παρατηρούμε ότι στο μεγαλύτερο σύνολο δεδομένων που χρησιμοποιούμε *Sent17*, το SiATL έχει καλύτερη απόδοση από το ULMFi, αλλά μόνο κατά ένα πολύ μικρό περιθώριο (όπως μπορεί να φανεί στον Πίνακα 5.2), ενώ στο μικρό σύνολο δεδομένων του *SCv2*, η απόδοση του SiATL είναι κατά πολύ καλύτερη, με τιμή κοντά σε αυτήν του state-of-the-art μοντέλου [124]. Καθώς το *Sent17* είναι μακράν το μεγαλύτερο σύνολο δεδομένων μας (με προσεγγιστικά 60,000 παραδείγματα), δεν βελτιώνεται σημαντικά από τη μεταφορά μάθησης. Η επίδραση της προτεινόμενης μεθόδους SiATL υπογραμμίζεται, ωστόσο, στο *SCv2*, το οποίο αποτελείται από μόλις 3,000 παραδείγματα. Άρα, η προσέγγισή μας είναι κατάλληλη για tasks με πολύ μικρό αριθμό δειγμάτων.

Επίσης, η απόδοση του SiATL σε σχέση με το ULMFiT ως προς το μέγεθος του συνόλου εκπαίδευσης μπορεί να παρατηρηθεί στο Σχήμα 5.3. Αξίζει να σημειωθεί ότι το προτεινόμενο μοντέλο επιτυγχάνει ανταγωνιστικά αποτελέσματα με λιγότερα από 1000 δείγματα εκπαίδευσης για τα *Irony18, SCv2, SCv1* και *PsychExp* σύνολα δεδομένων, δείχνοντας την ευρωστία του SiATL ακόμα και όταν εκπαιδεύεται σε ελάχιστα δεδομένα. Υποθέτουμε ότι το προεκπαιδευμένο μέρος του μοντέλου παρέχει μία αναπαράσταση λέξεων με πολλή πληροφορία. Είναι σημαντικό να σημειωθεί πως αν και τα *SCv2, SCv1* σύνολα δεδομένων δεν ανήκουν στον τομέα του Twitter, το μοντέλο μας προσαρμόζεται στη διαφορετική κατανομή λέξεων εύκολα και επιδεικνύει πολύ καλή απόδοση με μόλις 500 δείγματα εκπαίδευσης.

Τελικά, η απόδοση του SiATL στο *Sen17* είναι ενθαρρυντική. Το μοντέλο επιτυγχάνει 61% F_1 με 3000 δείγματα εκπαίδευσης, που συνιστούν μετά βίας το 10% του πλήρους συνόλου εκπαίδευσης (49,000 δείγματα).

5.6 Συμπεράσματα & Μελλοντικές Προεκτάσεις

Εισάγουμε το SiATL, ένα απλό και αποτελεσματικό μοντέλο μεταφοράς μάθησης για προβλήματα ταξινόμησης σε κείμενο. Η προσέγγισή μας βασίζεται σε προεκπαίδευση ενός LM και μεταφορά των βαρών του σε έναν ταξινομητή με ένα επιπρόσθετο αναδρομικό επίπεδο. Το μοντέλο εκπαιδεύεται με χρήση μίας συνάρτησης κόστους ταξινόμησης σε συνδυασμό με μία βοηθητική συνάρτηση κόστους



Σχήμα 5.3: Αποτελέσματα της προτεινόμενης μεθόδου (SiATL) (o) και του ULMFiT (+) για διαφορετικά σύνολα δεδομένων ως συνάρτηση του αριθμού των παραδειγμάτων εκπαίδευσης.

του LM. Το SiATL αποφεύγει το catastrophic forgetting. Τα πειράματα σε διάφορα tasks ταξινόμησης οδηγούν σε αποτελέσματα πολλές φορές συγκρίσιμα με το state-of-the-art, επιδεικνύοντας την αποτελεσματικότητα της προσέγγισής μας. Η μέθοδός μας συνεχώς επιτυγχάνει καλύτερα αποτελέσματα από πιο περίπλοκες τεχνικές μεταφοράς μάθησης, όπως το ULMFiT.

Στο μέλλον, σχεδιάζουμε να εξερευνήσουμε περαιτέρω αναπαραστάσεις του περιεχομένου από γλωσσικά μοντέλα (LMs). Ένα πρώτο βήμα σε αυτή τη διεύθυνση θα ήταν να επεκτείνουμε τη προσέγγισή μας χρησιμοποιώντας αμφίδρομο μοντέλο, μιας και τα αμφίδρομα γλωσσικά μοντέλα έχουν δείξει ότι κωδικοποιούν πιο βαθιά πληροφορία για δεδομένη είσοδο. Επίσης, θα θέλαμε να διεξάγουμε πειράματα σε διαφορετικούς τομείς και να πειραματιστούμε με ένα μοντέλο που δρα σε επίπεδο subword (μορφημάτων). Έτσι, το πρόβλημα που προκύπτει με τις λέξεις εκτός λεξιλογίου (out-of-vocabulary) μπορεί να αποφευχθεί, το οποίο είναι ιδιαίτερα εμφανές στο Twitter.

Κεφάλαιο 6

Συμπεράσματα και Μελλοντικές Προεκτάσεις της Εργασίας

Σε αυτή την εργασία, ερευνούμε νευρωνικές μεθόδους μεταφοράς μάθησης σε συνθήκες βαθιάς μάθησης για την βελτίωση των αποτελεσμάτων σε πληθώρας πεδίων, όπως η αναγνώριση συναισθημάτων, η ανάλυση συναισθήματος και συναφή προβλήματα ταξινόμησης. Τα μοντέλα που προτείνουμε επιτρέπουν την εκπαίδευση βαθιών νευρωνικών δικτύων με περιορισμένα δεδομένα εκπαίδευσης και την επίτευξη ανταγωνιστικής απόδοσης. Στην αναγνώριση συναισθημάτων, η μεταφορά ενός ταξινομητή ανάλυσης συναισθήματος σε ένα μοντέλο αναγνώρισης των 6 βασικών συναισθημάτων μπορεί να συνεισφέρει στη δημιουργία συστημάτων τα οποία είναι ικανά να αντιλαμβάνονται και να αναλύουν τα ανθρώπινα συναισθήματα καθώς και τις ανθρώπινες συμπεριφορές. Επιπλέον, οι γλωσσικές αναπαραστάσεις που μπορούν να προκύψουν με την εκπαίδευση γλωσσικών μοντέλων αποτελούν διανύσματα ικανά να μοντελοποιήσουν το περιεχόμενο. Κωδικοποιούν τα υψηλού (αφηρημένου) επιπέδου χαρακτηριστικά της γλώσσας και μπορούν να αναπαραστήσουν τη σύνταξη αλλά και τη σημασιολογία. Στα πλαίσια αυτής της εργασίας, προτείνουμε ένα νέο τρόπο προσαρμογής προεκπαιδευμένων αναπαραστάσεων από γλωσσικά μοντέλα σε ένα μοντέλο ταξινόμησης, με την ταυτόχρονη εισαγωγή μιας βοηθητικής συνάρτησης κόστους από το γλωσσικό μοντέλο. Προτείνουμε λοιπόν δύο διαφορετικές προσεγγίσεις, η μία εκ των οποίων βασίζεται σε προεκπαιδευμένο ταξινομητή και η άλλη σε προεκπαιδευμένο γλωσσικό μοντέλο και αξιοποιούμε την αφηρημένη αναπαράσταση χαρακτηριστικών που έχουν, μεταφέροντας την σε προβλήματα (tasks) όπου υπάρχει έλλειψη επισημασμένων (labeled) δεδομένων.

Αρχικά, υλοποιούμε ένα μοντέλο βαθιάς μάθησης για την ταξινόμηση προτάσεων στα βασικά συναισθήματα, σε περιπτώσεις όπου η συναισθηματικά φορτισμένη λέξη δεν είναι παρούσα στο κείμενο. Διεξάγουμε πειράματα με χρήση διαφορετικών προσεγγίσεων μεταφοράς μάθησης. Συγκεκριμένα, χρησιμοποιούμε προεκπαιδευμένα διανύσματα λέξεων, ένα προεκπαιδευμένο μοντέλο ανάλυσης συναισθήματος, αλλά και προεκπαιδευμένα γλωσσικά μοντέλα. Οι πρώτες δύο προσεγγίσεις βασίζονται σε αμφίδρομα LSTM μοντέλα πολλαπλών επιπέδων με ένα μηχανισμό προσοχής, ενώ η τρίτη χρησιμοποιεί ένα LSTM μοντέλο πολλαπλών επιπέδων μονής κατεύθυνσης (uni-directional). Χρησιμοποιούμε ένα μοντέλο ensembling για να καθορίσουμε ποιες προσεγγίσεις συνεισφέρουν περισσότερο στην απόδοση του μοντέλου. Αξιοποιώντας την αναπαράσταση χαρακτηριστικών ενός μοντέλου ανάλυσης συναισθήματος για το πρόβλημα της ταξινόμησης στα βασικά συναισθήματα, εξετάζουμε εάν μία αναπαράσταση που προέκυψε με επιβλεπόμενη μέθοδο μπορεί να μεταφερθεί σε ένα συναφές σύνολο δεδομένων ικανοποιητικά χωρίς overfitting. Η ιδέα είναι ότι, θεωρώντας ότι το αρχικό task (sentiment analysis - ανάλυση συναισθήματος) είναι πιο γενικό από το τελικό (emotion recognition - αναγνώριση συναισθημάτων), η αναπαράστασή του μπορεί να περιέχει σημαντικές πληροφορίες που να είναι χρήσιμες για το μοντέλο του τελικού task, οδηγώντας έτσι σε απόδοση κοντά στο state-of-the-art.

Έπειτα, προτείνουμε ένα καινούργιο, διαισθητικά απλό μοντέλο μεταφοράς μάθησης, το οποίο αξιοποιεί αναπαραστάσεις από γλωσσικά μοντέλα, τις μεταφέρει σε ένα μοντέλο ταξινόμησης και προσθέτει σε αυτό μία βοηθητική αντικειμενική συνάρτηση του γλωσσικού μοντέλου. Καθώς το πρόβλημα που αντιμετωπίζουμε συχνά όταν προσαρμόζουμε ένα γλωσσικό μοντέλο σε ένα μικρό σύνολο δεδομένων με ετικέτες είναι το overfitting, το οποίο προκαλείται από το catastrophic forgetting, προτείνουμε μια σαφή προσέγγιση που το αντιμετωπίζει ευθέως. Το κίνητρο που οδήγησε στην προσθήκη μιας βοηθητικής συνάρτησης κόστους σχετίζεται κατά μεγάλο βαθμό με τη μάθηση πολλαπλών εργασιών (multi-task learning). Επειδή το μοντέλο μας έχει εκπαιδευτεί σε γλωσσική μοντελοποίηση,

έχει κωδικοποιήσει μία αναπαράσταση χαρακτηριστικών που είναι γενική και αντιπροσωπεύει την κατανομή των λέξεων σε ένα συγκεκριμένο τομέα. Για να διατηρηθεί αυτή η γνώση, εκπαιδεύουμε σταδιακά τα επίπεδα του γλωσσικού μοντέλου στο πρόβλημα ταξινόμησης που αντιμετωπίζουμε. Με αντίστοιχη συλλογιστική, χρησιμοποιούμε επίσης δύο διαφορετικούς βελτιστοποιητές. Ο πρώτος χρησιμοποιείται για το προεκπαιδευμένο μέρος του μοντέλου και δεν επιτρέπει γρήγορη προσαρμογή στο τελικό task, ενώ ο δεύτερος χρησιμοποιείται στο καινούργιο LSTM επίπεδο που προσθέτουμε για την ταξινόμηση και οδηγεί σε ταχεία μεταβολή των βαρών αυτού του επιπέδου με τρόπο που να προσαρμόζεται στο τελικό task. Προκειμένου να δείξουμε την επίδραση της προσέγγισής μας με μεταφορά μάθησης, ελέγχουμε το σύστημά μας σε 5 διαφορετικά σύνολα δεδομένων, 3 εκ των οποίων ανήκουν σε διαφορετικό τομέα από ότι το corpus του γλωσσικού μοντέλου (*Sarcasm Corpus 1*, *Sarcasm Corpus 2* και *PsychExp*).

Στο μέλλον, μπορούμε να βελτιώσουμε το προτεινόμενο μοντέλο μεταφοράς μάθησης με την προσθήκη άλλων βοηθητικών συναρτήσεων κόστους, εκτός αυτής του γλωσσικού μοντέλου. Είναι πιθανό να συνεισέφερε ένα task το οποίο να κωδικοποιεί τη σημασιολογία, όπως η ερώτηση-απόκριση αλλά και η πρόβλεψη της επόμενης πρότασης. Με αυτό τον τρόπο, θα πετυχαίναμε την εκπαίδευση ακριβέστερων γλωσσικών αναπαραστάσεων, που θα εφαρμόζονταν σε διάφορα απαιτητικά NLP tasks, όπως η εξαγωγή συμπεράσματος (natural language inference) και η επισημείωση ακολουθιών (sequence tagging).

Ένας περιορισμός της προτεινόμενης προσέγγισης είναι ότι το sequential unfreezing απαιτεί προσεκτική επιλογή υπερπαραμέτρων. Για να αποφύγουμε την λεπτομερή προσαρμογή των υπερπαραμέτρων, σχεδιάζουμε να μοντελοποιήσουμε τόσο την αντικειμενική συνάρτηση του γλωσσικού μοντέλου όσο και του μοντέλου ταξινόμησης σαν δύο παίκτες που εμπλέκονται σε ένα min-max πρόβλημα. Εμπνεόμενοι από τη θεωρία παιγνίων, θα μπορούσαμε τότε να διατυπώσουμε το πρόβλημα ως ανταγωνιστικό (adversarial), σε μία μικρή περιοχή γύρω από το τοπικό ελάχιστο του κόστους του προεκπαιδευμένου γλωσσικού μοντέλου. Τότε, θα ήταν εφικτή η εύρεση ανταγωνιστικών μεθόδων εκπαίδευσης. Για παράδειγμα, θα μπορούσαμε να χρησιμοποιήσουμε τον βελτιστοποιητή optimistic Adam [126], ο οποίος προσφέρει τη δυνατότητα μείωσης των ταλαντώσεων εκπαίδευσης που προκαλεί ο Adam γύρω από το τοπικό ελάχιστο.

Μία ακόμα μελλοντική κατεύθυνση θα ήταν ο πειραματισμός με εκπαίδευση ανταγωνιστικού τομέα [127]. Ένα σημαντικό πρόβλημα στη μεταφορά μάθησης είναι ότι όταν η αρχική και η τελική κατανομή πιθανότητας των δύο μοντέλων διαφέρουν σημαντικά, γεγονός που συμβαίνει συχνά, το προεκπαιδευμένο μοντέλο δεν είναι ικανό να προσαρμοστεί στο τελικό πρόβλημα (target task). Επομένως, μια ιδέα θα ήταν να δημιουργήσουμε μια κοινή αναπαράσταση χαρακτηριστικών για διαφορετικούς τομείς (domains), αντικαθιστώντας τη βοηθητική συνάρτηση κόστους του γλωσσικού μοντέλου με μία συνάρτηση κόστους ταξινόμησης τομέα. Στόχος της συγκεκριμένης θα ήταν να προβλέψει τον τομέα μιας δοσμένης πρότασης (είτε source είτε target). Αντί να ελαχιστοποιούμε το συγκεκριμένο κόστος, σχεδιάζουμε να το μεγιστοποιήσουμε, ενώ ταυτόχρονα θα ελαχιστοποιούμε το κόστος ταξινόμησης στο δοσμένο task. Αυτό στηρίζεται στην ιδέα ότι αν ένα μοντέλο δεν είναι ικανό να διαχωρίσει αν μία πρόταση ανήκει στο source ή στο target task, τότε είναι πιθανό να έχει μάθει μια γενική και εύρωστη αναπαράσταση χαρακτηριστικών που μπορεί να χρησιμοποιηθεί και στους δύο τομείς, αποφεύγοντας το catastrophic forgetting.

Εν κατακλείδι, σχεδιάζουμε να εισάγουμε πληροφορία σε επίπεδο subword (μορφημάτων) στα γλωσσικά μοντέλα και να διεξάγουμε πειράματα με αμφίδρομες αναδρομικές αρχιτεκτονικές. Επίσης σχεδιάζουμε να εκπαιδεύσουμε γλωσσικά μοντέλα σε γενικά σύνολα δεδομένων και να εξετάσουμε τεχνικές προσαρμογής της μείωσης της παραμέτρου γ κατά τη διάρκεια της εκπαίδευσης.

Βιβλιογραφία

[1] S. M. Mohammad, F. Bravo-Marquez, M. Salameh, and S. Kiritchenko, "Semeval-2018 Task 1: Affect in tweets," in *Proceedings of International Workshop on Semantic Evaluation (SemEval-2018)*, New Orleans, LA, USA, 2018.

[2] A. Chronopoulou, A. Margatina, C. Baziotis, and A. Potamianos, "Ntua-slp at iest 2018: Ensemble of neural transfer methods for implicit emotion classification," *arXiv preprint arXiv:1809.00717*, 2018.

[3] A. Chronopoulou, C. Baziotis, and A. Potamianos, "An Embarrassingly Simple Approach for Transfer Learning from Pretrained Language Models," *arXiv e-prints*, p. arXiv:1902.10547, Feb 2019.

[4] A. Karpathy, "Convolutional neural networks for visual recognition." [Online]. Available: http://cs231n.github.io/neural-networks-1/

[5] L. Torrey and J. Shavlik, "Transfer learning," in *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*, 2010, pp. 242–264.

[6] S. Ruder, "An overview of multi-task learning in deep neural networks," *arXiv preprint arXiv:1706.05098*, 2017.

[7] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," *Journal of machine learning research*, vol. 3, no. Feb, pp. 1137–1155, 2003.

[8] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[9] D. Andor, C. Alberti, D. Weiss, A. Severyn, A. Presta, K. Ganchev, S. Petrov, and M. Collins, "Globally normalized transition-based neural networks," *arXiv preprint arXiv:1603.06042*, 2016.

[10] J. Weston, A. Bordes, S. Chopra, A. M. Rush, B. van Merriënboer, A. Joulin, and T. Mikolov, "Towards ai-complete question answering: A set of prerequisite toy tasks," *arXiv preprint arXiv:1502.05698*, 2015.

[11] K. M. Hermann, T. Kocisky, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, and P. Blunsom, "Teaching machines to read and comprehend," in *Advances in Neural Information Processing Systems*, 2015, pp. 1693–1701.

[12] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey *et al.*, "Google's neural machine translation system: Bridging the gap between human and machine translation," *arXiv preprint arXiv:1609.08144*, 2016.

[13] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proceedings of the International Conference on Learning Representations*, San Diego, California, 2015.

[14] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," 2009.

[15] D. Dong, H. Wu, W. He, D. Yu, and H. Wang, "Multi-task learning for multiple language translation," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, vol. 1, 2015, pp. 1723–1732.

[16] B. Zoph and K. Knight, "Multi-source neural translation," *arXiv preprint arXiv:1601.00710*, 2016.

[17] M. Johnson, M. Schuster, Q. V. Le, M. Krikun, Y. Wu, Z. Chen, N. Thorat, F. Viégas, M. Wattenberg, G. Corrado *et al.*, "Google's multilingual neural machine translation system: Enabling zero-shot translation," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 339–351, 2017.

[18] H. Peng, S. Thomson, and N. A. Smith, "Deep multitask learning for semantic dependency parsing," *arXiv preprint arXiv:1704.06855*, 2017.

[19] K. Zhao and L. Huang, "Joint syntacto-discourse parsing and the syntacto-discourse treebank," *arXiv preprint arXiv:1708.08484*, 2017.

[20] S. Swayamdipta, S. Thomson, C. Dyer, and N. A. Smith, "Frame-semantic parsing with softmax-margin segmental rnns and a syntactic scaffold," *arXiv preprint arXiv:1706.09528*, 2017.

[21] J. Guo, W. Che, H. Wang, and T. Liu, "Exploiting multi-typed treebanks for parsing with deep multi-task learning," *arXiv preprint arXiv:1606.01161*, 2016.

[22] I. Augenstein, S. Ruder, and A. Søgaard, "Multi-task learning of pairwise sequence classification tasks over disparate label spaces," *arXiv preprint arXiv:1802.09913*, 2018.

[23] E. Enguehard, Y. Goldberg, and T. Linzen, "Exploring the syntactic abilities of rnns with multi-task learning," *arXiv preprint arXiv:1706.03542*, 2017.

[24] 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.

[25] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," in *Proceedings of the Annual Meeting of the ACL*, Melbourne, Australia, 2018, pp. 328–339.

[26] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," 2018.

[27] M. Peters, W. Ammar, C. Bhagavatula, and R. Power, "Semi-supervised sequence tagging with bidirectional language models," in *Proceedings of the Annual Meeting of the ACL*, Vancouver, Canada, 2017, pp. 1756–1765. [Online]. Available: http://aclweb.org/anthology/P17-1161

[28] A. M. Dai and Q. V. Le, "Semi-supervised sequence learning," in *Proceedings of the Advances in Neural Information Processing Systems*, 2015, pp. 3079–3087.

[29] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning, "A large annotated corpus for learning natural language inference," *arXiv preprint arXiv:1508.05326*, 2015.

[30] E. F. Sang and F. De Meulder, "Introduction to the conll-2003 shared task: Language-independent named entity recognition," *arXiv preprint cs/0306050*, 2003.

[31] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "Squad: 100,000+ questions for machine comprehension of text," *arXiv preprint arXiv:1606.05250*, 2016.

[32] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," in *Proceedings of the Conference of the NAACL:HLT*, New Orleans, Louisiana, 2018, pp. 2227–2237. [Online]. Available: http://aclweb.org/anthology/N18-1202

[33] G. Qiu, X. He, F. Zhang, Y. Shi, J. Bu, and C. Chen, "Dasa: dissatisfaction-oriented advertising based on sentiment analysis," *Expert Systems with Applications*, vol. 37, no. 9, pp. 6182–6191, 2010.

[34] X. Jin, Y. Li, T. Mah, and J. Tong, "Sensitive webpage classification for content advertising," in *Proceedings of the 1st international workshop on Data mining and audience intelligence for advertising*. ACM, 2007, pp. 28–33.

[35] V. Stoyanov, C. Cardie, and J. Wiebe, "Multi-perspective question answering using the opqa corpus," in *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2005, pp. 923–930.

[36] J. N. Druckman and R. McDermott, "Emotion and the framing of risky choice," *Political Behavior*, vol. 30, no. 3, pp. 297–321, 2008.

[37] R. P. Bagozzi, M. Gopinath, and P. U. Nyer, "The role of emotions in marketing," *Journal of the academy of marketing science*, vol. 27, no. 2, p. 184, 1999.

[38] S. Brave and C. Nass, "Emotion in human-computer interaction," *The human-computer interaction handbook: fundamentals, evolving technologies and emerging applications*, pp. 81–96, 2003.

[39] N. Gupta, M. Gilbert, and G. D. Fabbrizio, "Emotion detection in email customer care," *Computational Intelligence*, vol. 29, no. 3, pp. 489–505, 2013.

[40] S. Voeffray, "Emotion-sensitive human-computer interaction (hci): State of the art-seminar paper," *Emotion Recognition*, pp. 1–4, 2011.

[41] C. Olah, "Visual information theory." [Online]. Available: https://colah.github.io/posts/2015-09-Visual-Information/

[42] R. Hecht-Nielsen, "Theory of the backpropagation neural network," in *Neural networks for perception*. Elsevier, 1992, pp. 65–93.

[43] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 2011.

[44] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[45] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv preprint arXiv:1207.0580*, 2012.

[46] 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.

[47] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of COMPSTAT'2010*. Springer, 2010, pp. 177–186.

[48] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, no. Jul, pp. 2121–2159, 2011.

[49] M. D. Zeiler, "Adadelta: an adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.

[50] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proceedings of the International Conference on Learning Representations*, 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

[51] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," California Univ San Diego La Jolla Inst for Cognitive Science, Tech. Rep., 1985.

[52] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

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

[54] A. Graves, "Generating sequences with recurrent neural networks," *arXiv preprint arXiv:1308.0850*, 2013.

[55] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*. MIT press Cambridge, 2016, vol. 1.

[56] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.

[57] L. Duong, T. Cohn, S. Bird, and P. Cook, "Low resource dependency parsing: Cross-lingual parameter sharing in a neural network parser," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, vol. 2, 2015, pp. 845–850.

[58] S. Dhuria, "Natural language processing: An approach to parsing and semantic analysis," *International Journal of New Innovations in Engineering and Technology*, 2015.

[59] C. D. Manning, C. D. Manning, and H. Schütze, *Foundations of statistical natural language processing*. MIT press, 1999.

[60] B. Liu, "Sentiment analysis and opinion mining," *Synthesis lectures on human language technologies*, vol. 5, no. 1, pp. 1–167, 2012.

[61] P. D. Turney, "Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews," in *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 2002, pp. 417–424.

[62] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: sentiment classification using machine learning techniques," in *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*. Association for Computational Linguistics, 2002, pp. 79–86.

[63] J. M. Wiebe, R. F. Bruce, and T. P. O'Hara, "Development and use of a gold-standard data set for subjectivity classifications," in *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*. Association for Computational Linguistics, 1999, pp. 246–253.

[64] Y. Goldberg, "Neural network methods for natural language processing," *Synthesis Lectures on Human Language Technologies*, vol. 10, no. 1, pp. 1–309, 2017.

[65] T. Mikolov, M. Karafiát, L. Burget, J. Černockỳ, and S. Khudanpur, "Recurrent neural network based language model," in *Eleventh Annual Conference of the International Speech Communication Association*, 2010.

[66] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation." in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Doha, Qatar, 2014, pp. 1532–1543.

[67] Z. S. Harris, "Distributional structure," *Word*, vol. 10, no. 2-3, pp. 146–162, 1954.

[68] R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, and C. D. Manning, "Semi-supervised recursive autoencoders for predicting sentiment distributions," in *Proceedings of the conference on empirical methods in natural language processing*. Association for Computational Linguistics, 2011, pp. 151–161.

[69] X. Wang, Y. Liu, S. Chengjie, B. Wang, and X. Wang, "Predicting polarities of tweets by composing word embeddings with long short-term memory," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, vol. 1, 2015, pp. 1343–1353.

[70] S. Merity, N. S. Keskar, and R. Socher, "Regularizing and optimizing lstm language models," *arXiv preprint arXiv:1708.02182*, 2017.

[71] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Advances in neural information processing systems*, 2014, pp. 3320–3328.

[72] L. N. Smith, "Cyclical learning rates for training neural networks," in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2017, pp. 464–472.

[73] S. M. Mohammad and F. Bravo-Marquez, "Wassa-2017 shared task on emotion intensity," *arXiv preprint arXiv:1708.03700*, 2017.

[74] F. Pla and L.-F. Hurtado, "Political tendency identification in twitter using sentiment analysis techniques," in *Proceedings of COLING 2014, the 25th international conference on computational linguistics: Technical Papers*, 2014, pp. 183–192.

[75] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe, "Predicting elections with twitter: What 140 characters reveal about political sentiment." *Icwsm*, vol. 10, no. 1, pp. 178–185, 2010.

[76] W. Li and H. Xu, "Text-based emotion classification using emotion cause extraction," *Expert Systems with Applications*, vol. 41, no. 4, pp. 1742–1749, 2014.

[77] J. Si, A. Mukherjee, B. Liu, Q. Li, H. Li, and X. Deng, "Exploiting topic based twitter sentiment for stock prediction," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, vol. 2, 2013, pp. 24–29.

[78] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," *Journal of computational science*, vol. 2, no. 1, pp. 1–8, 2011.

[79] W. Chamlertwat, P. Bhattarakosol, T. Rungkasiri, and C. Haruechaiyasak, "Discovering consumer insight from twitter via sentiment analysis." *J. UCS*, vol. 18, no. 8, pp. 973–992, 2012.

[80] P. Burnap, W. Colombo, and J. Scourfield, "Machine classification and analysis of suicide-related communication on twitter," in *Proceedings of the 26th ACM conference on hypertext & social media*. ACM, 2015, pp. 75–84.

[81] F. Å. Nielsen, "A new anew: Evaluation of a word list for sentiment analysis in microblogs," *arXiv preprint arXiv:1103.2903*, 2011.

[82] S. M. Mohammad and P. D. Turney, "Emotions evoked by common words and phrases: Using mechanical turk to create an emotion lexicon," in *Proceedings of the NAACL HLT 2010 workshop on computational approaches to analysis and generation of emotion in text*. Association for Computational Linguistics, 2010, pp. 26–34.

[83] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," *CS224N Project Report, Stanford*, vol. 1, no. 12, 2009.

[84] J. Bollen, H. Mao, and A. Pepe, "Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena." *Icwsm*, vol. 11, pp. 450–453, 2011.

[85] S. M. Mohammad, S. Kiritchenko, and X. Zhu, "NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets," *arXiv preprint arXiv:1308.6242*, 2013.

[86] S. Kiritchenko, X. Zhu, and S. M. Mohammad, "Sentiment analysis of short informal texts," *Journal of Artificial Intelligence Research*, vol. 50, pp. 723–762, 2014.

[87] J. Deriu, M. Gonzenbach, F. Uzdilli, A. Lucchi, V. D. Luca, and M. Jaggi, "Swisscheese at semeval-2016 task 4: Sentiment classification using an ensemble of convolutional neural networks with distant supervision," in *Proceedings of the 10th international workshop on semantic evaluation*, no. EPFL-CONF-229234, 2016, pp. 1124–1128.

[88] P. Goel, D. Kulshreshtha, P. Jain, and K. K. Shukla, "Prayas at emoint 2017: An ensemble of deep neural architectures for emotion intensity prediction in tweets," in *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, 2017, pp. 58–65.

[89] C. Baziotis, N. Pelekis, and C. Doulkeridis, "Datastories at semeval-2017 task 4: Deep lstm with attention for message-level and topic-based sentiment analysis," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, Vancouver, Canada, 2017, pp. 747–754. [Online]. Available: http://aclweb.org/anthology/S17-2126

[90] R. Klinger, O. De Clercq, S. M. Mohammad, and A. Balahur, "Iest: Wassa-2018 implicit emotions shared task," *arXiv preprint arXiv:1809.01083*, 2018.

[91] S. J. Pan, Q. Yang *et al.*, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, no. 10, pp. 1345–1359, 2010.

[92] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2014, pp. 1717–1724.

[93] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 1701–1708.

[94] A. Agrawal, J. Lu, S. Antol, M. Mitchell, C. L. Zitnick, D. Parikh, and D. Batra, "Vqa: Visual question answering," *Int. J. Comput. Vision*, vol. 123, no. 1, pp. 4–31, May 2017.

[95] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proceedings of the Advances in Neural Information Processing Systems*, 2013, pp. 3111–3119.

[96] S. R. Bowman, C. Potts, and C. D. Manning, "Recursive neural networks can learn logical semantics," *arXiv preprint arXiv:1406.1827*, 2014.

[97] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the 2013 conference on empirical methods in natural language processing*, 2013, pp. 1631–1642.

[98] R. Nallapati, B. Zhou, C. Gulcehre, B. Xiang *et al.*, "Abstractive text summarization using sequence-to-sequence rnns and beyond," *arXiv preprint arXiv:1602.06023*, 2016.

[99] M. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi, "Bidirectional attention flow for machine comprehension," *arXiv preprint arXiv:1611.01603*, 2016.

[100] J. Howard and S. Ruder, "Fine-tuned language models for text classification," *CoRR*, vol. abs/1801.06146, 2018. [Online]. Available: http://arxiv.org/abs/1801.06146

[101] R. Řehůřek and P. Sojka, "Software Framework for Topic Modelling with Large Corpora," in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, Valletta, Malta, May 2010, pp. 45–50, `http://is.muni.cz/publication/ 884893/en`.

[102] S. Rosenthal, N. Farra, and P. Nakov, "Semeval-2017 task 4: Sentiment analysis in twitter," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, Vancouver, Canada, 2017, pp. 502–518.

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

[104] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks." *ICML (3)*, vol. 28, pp. 1310–1318, 2013.

[105] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017. [Online]. Available: https://openreview.net/forum?id=BJJsrmfCZ

[106] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, and others, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[107] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," *arXiv preprint arXiv:1508.07909*, 2015.

[108] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *arXiv preprint arXiv:1607.04606*, 2016.

[109] C. Xiong, V. Zhong, and R. Socher, "Dynamic coattention networks for question answering," 2016.

[110] P. Ramachandran, P. Liu, and Q. Le, "Unsupervised pretraining for sequence to sequence learning," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Copenhagen, Denmark, 2017, pp. 383–391. [Online]. Available: http://aclweb.org/anthology/D17-1039

[111] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *Journal of Machine Learning Research*, pp. 2493–2537, 2011.

[112] R. Caruana, "Multitask learning: A knowledge-based source of inductive bias," in *Machine Learning: Proceedings of the Tenth International Conference*, 1993, pp. 41–48.

[113] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the International Conference on Machine learning*, 2008, pp. 160–167.

[114] A. Søgaard and Y. Goldberg, "Deep multi-task learning with low level tasks supervised at lower layers," in *Proceedings of the Annual Meeting of the ACL*, Berlin, Germany, 2016, pp. 231–235.

[115] Z. Lin, M. Feng, C. N. d. Santos, M. Yu, B. Xiang, B. Zhou, and Y. Bengio, "A structured self-attentive sentence embedding," *arXiv preprint arXiv:1703.03130*, 2017.

[116] H. G. Wallbott and K. R. Scherer, "How universal and specific is emotional experience? evidence from 27 countries on five continents," *Information (International Social Science Council)*, no. 4, pp. 763–795, 1986. [Online]. Available: https://doi.org/10.1177/053901886025004001

[117] C. Van Hee, E. Lefever, and V. Hoste, "Semeval-2018 task 3: Irony detection in english tweets," in *Proceedings of The 12th International Workshop on Semantic Evaluation (SemEval-2018)*, New Orleans, Louisiana, 2018, pp. 39–50.

[118] S. Oraby, V. Harrison, L. Reed, E. Hernandez, E. Riloff, and M. A. Walker, "Creating and characterizing a diverse corpus of sarcasm in dialogue," in *Proceedings of the SIGDIAL 2016 Conference, The 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, 2016, pp. 31–41. [Online]. Available: http://aclweb.org/anthology/W/W16/W16-3604.pdf

[119] S. Lukin and M. Walker, "Really? well. apparently bootstrapping improves the performance of sarcasm and nastiness classifiers for online dialogue," in *Proceedings of the Workshop on Language Analysis in Social Media*, Atlanta, Georgia, June 2013, pp. 30–40. [Online]. Available: http://www.aclweb.org/anthology/W13-1104

[120] E. Loper and S. Bird, "Nltk: The natural language toolkit," in *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*, 2002, pp. 63–70. [Online]. Available: https://doi.org/10.3115/1118108.1118117

[121] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *Journal of machine learning research*, pp. 2825–2830, 2011.

[122] C. Baziotis, A. Nikolaos, P. Papalampidi, A. Kolovou, G. Paraskevopoulos, N. Ellinas, and A. Potamianos, "Ntua-slp at semeval-2018 task 3: Tracking ironic tweets using ensembles of word and character level attentive rnns," in *Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018)*, New Orleans, Louisiana, 2018, pp. 613–621. [Online]. Available: http://aclweb.org/anthology/S18-1100

[123] M. Cliche, "Bb_twtr at semeval-2017 task 4: Twitter sentiment analysis with cnns and lstms," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, Vancouver, Canada, 2017, pp. 573–580. [Online]. Available: http://aclweb.org/anthology/S17-2094

[124] S. Ilic, E. Marrese-Taylor, J. A. Balazs, and Y. Matsuo, "Deep contextualized word representations for detecting sarcasm and irony," in *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, 2018, pp. 2–7. [Online]. Available: https://aclanthology.info/papers/W18-6202/w18-6202

[125] B. Felbo, A. Mislove, A. Søgaard, I. Rahwan, and S. Lehmann, "Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Copenhagen, Denmark, 2017, pp. 1615–1625. [Online]. Available: http://aclweb.org/anthology/D17-1169

[126] C. Daskalakis, A. Ilyas, V. Syrgkanis, and H. Zeng, "Training gans with optimism," *arXiv preprint arXiv:1711.00141*, 2017.

[127] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2096–2030, 2016.

Παράρτημα Α

Συντομογραφίες

(AI): Artificial Intelligence
(ANN): Artificial Neural Network
(Bi-LSTM): Bi-directional LSTM
(BoW): Bag-of-Words
(CBOW): Continuous Bag-Of-Words
(CNN): Convolutional Neural Network
(CV): Computer Vision
(DL): Deep Learning
(DNNs): Deep Neural Networks
(GPU): Graphical Processor Unit
(GD): Gradient Descent
(IE): Information Extraction
(LM): Language Model
(LR): Logistic Regression classifier
(LSTM): Long Short-Term Memory unit
(ML): Machine Learning
(MT): Machine Translation
(MTL): Multi-Task learning
NBoW: Neural Bag-of-Words
(NLP): Natural Language Processing
(POS): Part-Of-Speech
(QA): Question Answering
(RNNs): Recurrent Neural Networks
(SGD): Stochastic Gradient Descent
(SVM): Support Vector Machine classifier
(TL): Transfer Learning