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

Decoupling Emergent Strategies in Task-Oriented Negotiation Dialogue Systems

Διπλωματική Εργασία

του

Γεράσιμου Σ. Χατζούδη

Επιβλέπων: Ποταμιάνος Αλέξανδρος
Κατσαμάνης Αθανάσιος

Εργαστήριο Επεξεργασίας Φωνής και Φυσικής Γλώσσας
Αθήνα, Ιούλιος 2020

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Σημάτων, Ελέγχου και Ρομποτικής
Εργαστήριο Επεξεργασίας Φωνής και Φυσικής Γλώσσας

Decoupling Emergent Strategies in Task-Oriented Negotiation Dialogue Systems

Διπλωματική Εργασία

του

Γεράσιμου Σ. Χατζούδη

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

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 17η Ιουλίου, 2020.

.....	.....	.....
Ποταμιάνος Αλέξανδρος	Κατσαμάνης Αθανάσιος	Τζαφέστας Κωνσταντίνος
Αν. Καθηγητής Ε.Μ.Π.	Κύριος Ερευνητής Ι.Ε.Λ., Ε.Κ. ΑΘΗΝΑ	Αν. Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούλιος 2020

.....

Γεράσιμος Σ. Χατζούδης
Διπλωματούχος Ηλεκτρολόγος Μηχανικός
και Μηχανικός Υπολογιστών Ε.Μ.Π.

Περίληψη

Μία από τις σπουδαιότερες προκλήσεις της Τεχνητής Νοημοσύνης είναι η δημιουργία συστημάτων που μπορούν να συνδιαλέγονται με τον άνθρωπο. Αυτό το ζήτημα μελετάται από τον κλάδο της Επεξεργασίας Φυσικής Γλώσσας που γνώρισε μεγάλη άνθιση κυρίως χάρη στη ραγδαία εξέλιξη της Μηχανικής Μάθησης και των Βαθιών Νευρωνικών Δικτύων. Στην παρούσα διπλωματική εργασία, ενστερνιστήκαμε την άποψη οτι η γλώσσα συγκροτείται ως ένα αυστηρό μέσο επικοινωνίας πρακτόρων που αλληλεπιδρούν εντός ενός περιβάλλοντος με σκοπό την ικανοποίηση κάποιας ανάγκης τους. Για το λόγο αυτό, εστιάζουμε σε διαλογικά προβλήματα προκαθορισμένου σκοπού. Ένα από τα δυσκολότερα είναι αυτό της διαπραγμάτευσης διότι απαιτεί γλωσσική επάρκεια και συλλογιστική ικανότητα.

Το υπό μελέτη πρόβλημα Deal Or No Deal εισάγεται από τη Facebook και εστιάζει στη διαπραγμάτευση δύο πρακτόρων με δεδομένο πλήθος αντικειμένων μέσω γραπτού λόγου. Το πρόβλημα αναλύεται σε ένα πρόβλημα Ταξινόμησης και σε ένα Γλωσσικής Μοντελοποίησης. Αρχικά, αναπαράξαμε και τροποποιήσαμε τον διαθέσιμο κώδικα της υλοποίησης. Στη συνέχεια, θεωρήσαμε πως η ακρίβεια είναι η κατάλληλη μετρική του προβλήματος ταξινόμησης και μελετήσαμε παραδοσιακές μεθόδους μοντέλων μηχανικής μάθησης. Έπειτα και στα δύο προβλήματα, αποδεικνύουμε πως η χρήση Μεταφοράς Μάθησης και συγκεκριμένα των Transformers BERT και GPT-2 προσδίδουν εξαιρετικά αποτελέσματα και παρουσιάζουμε κάποιες μεθόδους παραγωγής διαλόγων αξιοποιώντας το βελτιστοποιημένο GPT-2.

Η πολιτική λήψης αποφάσεων και στρατηγικών ως τώρα αγνοούνταν ή λαμβάνονταν υπόψη σε μεταγενέστερο στάδιο. Θεωρούμε πως η απόφαση και η ενέργεια προϋπάρχει του μηνύματος και για αυτό σχεδιάσαμε και υλοποιήσαμε ένα περιβάλλον προσομοίωσης αυτόματων διαπραγματευτών που εκπαιδεύονται με χρήση Ενισχυτικής Μάθησης. Συγκεκριμένα, διαμορφώνουμε τις συμπεριφορές των πρακτόρων, εξετάζουμε το βέλτιστο τρόπο απεικόνισης της διαπραγματευτικής πληροφορίας, δημιουργούμε συστήματα που επιχειρούν να εκμεταλλευτούν συγκεκριμένες συμπεριφορές των "αντιπάλων", μελετάμε το δίλημμα Εξερεύνησης-Αξιοποίησης και παρουσιάζουμε πώς η διαπραγμάτευση μεταξύ άπληστων πρακτόρων μπορεί να οδηγήσει σε μία κοινωνική ισορροπία. Τέλος, ένα από τα σημαντικότερα προβλήματα της Μηχανικής Μάθησης είναι η οπτικοποίηση και η ερμηνευσιμότητα των αποτελεσμάτων. Προς αυτή την κατεύθυνση, προτείνεται και υλοποιείται μια μέθοδος αναγνώρισης και οπτικοποίησης των στρατηγικών των αυτόματων διαπραγματευτών.

Λέξεις Κλειδιά

διάλογος, διαπραγμάτευση, μηχανική μάθηση, νευρωνικά δίκτυα, ενισχυτική μάθηση, οπτικοποίηση, στρατηγική, μεταφορά μάθησης

Abstract

One of the major challenges of Artificial Intelligence is to create systems that can communicate with humans. This issue is being studied by the highly flourishing Physical Language Processing industry, mainly thanks to the rapid evolution of Machine Learning and Deep Neural Networks. In this dissertation, we have embraced the view that language is constituted as a rigorous means of communicating agents who interact in an environment in order to satisfy their need. For this reason, we are focusing on task-oriented dialogue problems of a predetermined purpose. One of the most difficult is that of negotiation because it requires language competence and reasoning.

The problem under consideration Deal Or No Deal is introduced by Facebook and focuses on multi-issue bargaining between two agents through a communication channel. The problem is broken down into a Classification problem and a Language Modeling problem. Initially, we reproduced and modified the available implementation code. We then considered that accuracy is the most appropriate metric of the classification problem and we studied traditional machine learning models. Then on both problems, we demonstrate that the use of Transfer Learning and in particular Transformers BERT and GPT-2, deliver excellent results and we present some methods of generating dialogs utilizing the fine-tuned GPT-2.

Decision-making and strategy has so far been ignored or taken into account at a later stage. We believe that the decision and the action precede the message and that is why we designed and implemented an environment for simulating automated negotiators trained using Reinforcement Learning. Specifically, we shape the behavior of agents, we look at the best way to portray the negotiating information, we create systems that attempt to exploit specific attitudes of "opponents", we study the Exploration-Exploitation Trade-Off and we present how negotiation between greedy agents can lead in a social equilibrium. Finally, one of the most important problems of Machine Learning is the visualization and the interpretability of the results. To this end, To this end, a method of identifying and visualizing the strategies of automated negotiators is proposed and implemented.

Keywords

dialogue, negotiation, machine learning, neural networks, reinforcement learning, visualization, strategy, transfer learning

Ευχαριστίες

Νιώθω την ανάγκη να ευχαριστήσω τον καθηγητή μου κ. Ποταμιάνο Αλέξανδρο που μου έδωσε την ευκαιρία αν εκπονήσω την παρούσα διπλωματική εργασία πλάι στον κ. Κατσαμάνη Αθανάσιο υπό τη στέγη του εργαστηρίου. Θα ήθελα να ευχαριστήσω τον κ. Ποταμιάνο γιατί μέσα από τα μαθήματά του έθεσε τόσο ψηλά τον ακαδημαϊκό πήχη που όταν χρειάστηκε να εργαστώ στον χώρο μπορούσα με ταπεινότητα να αρχίζω να μιλάω την "ίδια γλώσσα". Είμαι τυχερός που εργάστηκα στην Innoetics όχι μόνο επειδή πρωτοπορεί στον κλάδο και αποτελεί μία από τις πιο πετυχημένες start-up στην Ελλάδα. Είμαι τυχερός που γνώρισα τους ανθρώπους της γιατί από εκεί ξεκινούν όλα και αυτό μάλλον έχει σημασία στο τέλος της μέρας.

Ακούγοντας τα καλύτερα λόγια για τον κ. Κατσαμάνη τόσο από την Innoetics όσο και από τον φίλο Γιώργο Παρασκευόπουλο που βοηθάει σε απίστευτο βαθμό πολλούς προπτυχιακούς φοιτητές, ήρθαμε σε επαφή και ξεκινήσαμε την συνεργασία. Νιώθω τυχερός που τα καλά λόγια επιβεβαιώθηκαν αμέσως και διανθίστηκαν μέσα στην ακαδημαϊκή χρονιά, διότι με δημιουργική σκέψη και πάντοτε με καλοσύνη και ευγένεια, μού έδωσε την ελευθερία, την κατεύθυνση και το κίνητρο να αναζητήσω τη γνώση όπου κι αν βρίσκεται και να χαρώ όλη την πορεία της διπλωματικής.

Θέλω να ευχαριστήσω με όλη μου την ψυχή τον μέντορά μου κ. Βασίλη Μακιό που με τιμάει με την φιλία του και παίζει καθοριστικό ρόλο στη ζωή μου. Ακόμα, θέλω να ευχαριστήσω την παρέα του TEDxNTUA. Ξεκινήσαμε σαν ονειροπόλοι, γίναμε φίλοι και στο τέλος της μέρας αφήσαμε κάτι μεγάλο, μια δημιουργική διέξοδο εν καιρώ κρίσης για όλη την κοινότητα του Πολυτεχνείου. Τώρα σειρά έχουν οι επόμενοι και αυτούς πρέπει να στηρίζουμε.

Θα ήθελα να ευχαριστήσω την οικογένειά μου που με έχει στηρίξει κυριολεκτικά στα πάντα. Η αγάπη μου για αυτούς τους ανθρώπους δεν αποτυπώνεται με δυο λέξεις. Είμαι ευλογημένος. Ευλογημένος νιώθω και για την ευρύτερη οικογένειά μου, τους κολλητούς μου φίλους και τους ανθρώπους που έχουν σταθεί πλάι μας και συνεχίζουν να στέκονται. Η διπλωματική είναι αφιερωμένη σε όλους αυτούς τους ανθρώπους, στην οικογένεια!

Φέτος στο μυαλό μου ένιωθα ένα μεγάλο χρέος. Δεν εργαζόμουν μόνο για μένα. Εργαζόμουν για τον συνεργάτη μου και τον καθηγητή μου που ήθελα να αποδείξω οτι μπορώ να τα καταφέρω και να τους κάνω περήφανους. Εργαζόμουν και για όλη την οικογένεια, συγγενείς και φίλους, που με στήριξαν χωρίς να το γνωρίζω μεταξύ άλλων και σε ένα μικρό ατυχές συμβάν. Δεν θέλησαν ποτέ να μου πουν ποιοι είναι. Όλο το έτος ένιωθα ένα ανώτερο χρέος και σε περιόδους στασιμότητας, αυτό το χρέος μού έδινε κίνητρο: να ανταποδώσω την πίστη τους με σκληρή δουλειά. Η διπλωματική είναι αφιερωμένη σε αυτούς τους ανθρώπους που είναι πάντοτε στις χαρές και τις λύπες μαζί. Στο μαζί, λοιπόν, με αγάπη!

Γεράσιμος Σ. Χατζούδης

Περιεχόμενα

Σχήματα

Πίνακες

Κεφάλαιο 1

Εισαγωγή

Μία από τις σπουδαιότερες προκλήσεις της Τεχνητής Νοημοσύνης είναι η δημιουργία συστημάτων που μπορούν να συνδιαλέγονται με τον άνθρωπο. Μολονότι η ύπαρξη ρομπότ και οι περιρρέουσες φιλοσοφικές αναζητήσεις σχετικά με το αν μπορούν να σχηματίσουν μορφή ευφυίας εμφανίζονται αρκετά νωρίτερα, το 1950 ο Alan Turing καταθέτει στην επιστημονική κοινότητα τη δημοσίευση με τίτλο "Computing Machinery and Intelligence" [121]. Εκεί εισάγει την έννοια του Turing τεστ, όπου προκειμένου να εξετάσει το αν "σκέφτονται οι μηχανές", δημιουργεί το παιχνίδι της μίμησης (imitation game) στο οποίο ένας παίκτης προσπαθεί μέσω ερωταπαντήσεων σε γραπτό λόγο να ξεχωρίσει αν πίσω από ένα δωμάτιο βρίσκεται ένας άνθρωπος ή μία μηχανή. Παρόλο που το ζήτημα της ευφυίας είναι αρκετά σύνθετο, επιθυμούμε να θίξουμε οτι η γλώσσα έχει περίοπτη θέση στο χώρο της Τεχνητής Νοημοσύνης και η μελέτη της αποτελεί μεγάλη πρόκληση στην εν λόγω κοινότητα αλλά και ευρέως. Η μελέτη της γλώσσας πραγματοποιείται από τον κλάδο της Επεξεργασίας Φυσικής Γλώσσας (Natural Language Processing - NLP) που γνώρισε μεγάλη άνθιση κυρίως χάρη της ραγδαίας εξέλιξης της Μηχανικής Μάθησης (Machine Learning) και των Βαθιών Νευρωνικών Δικτύων (Deep Neural Networks). Οι αιτίες εντοπίζονται στο μεγάλο εύρος διαθέσιμων δεδομένων και στην όλο και μεγαλύτερη υπολογιστική ισχύ.

1.1 Μικρή Τομή στη Φιλοσοφία της Γλώσσας

Πολλοί, σήμερα, διατείνονται οτι η χρήση όλο και μεγαλύτερων γλωσσικών μοντέλων που εκπαιδεύονται σε όλο και μεγαλύτερους όγκους δεδομένων, μπορούν να "λύσουν" το πρόβλημα της γλώσσας. Η πεποίθηση βασίζεται στο γεγονός οτι όσο μεγαλύτερα είναι τα μοντέλα τόσο περισσότερα διαφορετικά χαρακτηριστικά της γλώσσας εντοπίζονται στα στρώματά τους, ειδικότερα αν τα δεδομένα είναι πλούσια σε γλωσσικά χαρακτηριστικά. Κατά την εκπόνηση της παρούσας διπλωματικής εργασίας ενστερνιστήκαμε την άποψη οτι η γλώσσα δημιουργείται και συγκροτείται ως ένα μέσο επικοινωνίας μεταξύ σαφώς καθορισμένων πρακτόρων που προσπαθούν μέσω της μετάδοσης ενός μηνύματος να κοινοποιήσουν και να ικανοποιήσουν κάποια ανάγκη τους εντός ενός μεταβαλλόμενου περιβάλλοντος. Αποδομώντας το εν λόγω ζήτημα, η αρχιτεκτονική δομή τόσο των πρακτόρων όσο και του περιβάλλοντος καθορίζουν πλήρως τις συμπεριφορές που θα αναδυθούν. Οι συμπεριφορές αυτές εικάζεται οτι είναι άμεσα συνυφασμένες με την ικανοποίηση των αναγκών των πρακτόρων και οι ανάγκες τους είναι με τη σειρά τους άρρηκτα συνδεδεμένες τόσο με τη δομή των πρακτόρων όσο και με τη δομή του περιβάλλοντος. Στην ουσία, ο ισχυρισμός έγκειται στο οτι η γλώσσα "γεννάται"

από τη στιγμή που "γεννάται" η ανάγκη για επικοινωνία του πιο αρχέγονου ενστίκτου, αυτού της επιβίωσης. Η γλώσσα αλλάζει, βέβαια, από περιβάλλον σε περιβάλλον αφενός λόγω των έμβιων ή άβιων όντων (δεν θα μπορούσε άλλωστε να δοθεί κάποιο όνομα π.χ. για ένα ζώο το οποίο εμπειρικά δεν συναντάται σε έναν τόπο), αλλά και λόγω των συνθηκών που εντοπίζονται σε αυτό. Για παράδειγμα, οι γλώσσες πολιτισμών που τυγχάνει να βρίσκονται σε εύκρατες γεωγραφικές ζώνες, εμφανίζουν μια ευρύτερη ποικιλομορφία διότι οι πράκτορες που έμεναν σε αυτές μπορούσαν να ικανοποιήσουν ευκολότερα τις βασικές τους ανάγκες και συνεπώς χρειαζόταν να ονοματίσουν πιο σύνθετες έννοιες που περιγράφουν διαδικασίες ή καταστάσεις ιεραρχικά υψηλότερων αναγκών. Παρατηρείται πως από γλώσσα σε γλώσσα ακόμα και η απόδοση μιας λέξης για το ίδιο αντικείμενο μπορεί να είναι διαφορετική και να αποδίδει ένα εντελώς διαφορετικό νόημα. Προς επίρρωση αυτού, η λέξη "άγαλμα" συνειρμικά οδηγεί στις λέξεις αγαλλίαση ή αγάλλομαι ενώ η λέξη "statue" οδηγεί σε μία πιο στατική και άψυχη προσέγγιση.

Έχοντας κατά νου τη σπουδαιότητα της ανάγκης, που πηγάζει από την αρχιτεκτονική δομή ενός πράκτορα, και επιδιώκεται να ικανοποιηθεί σε ένα δεδομένο περιβάλλον, δίνεται έμφαση σε προβλήματα τα οποία έχουν ένα καθορισμένο "τέλος" (με την έννοια του σκοπού). Επιπρόσθετα, θα μπορούσε να θεωρήσει κανείς πως μία από τις σπουδαιότερες ανθρώπινες ανακαλύψεις είναι αυτή του σχηματισμού γραπτού λόγου διότι με αυτό τον τρόπο μπορεί η γλώσσα, από μία άυλη και στιγμιαία μορφή, να αποτυπώνεται σε μια μορφή ικανή να μεταδίδεται στα χρόνια. Ιδιαίτερο ενδιαφέρον παρουσιάζεται στις τελευταίες δεκαετίες όπου με τη χρήση του Ίντερνετ όλο και περισσότερη είναι η πληροφορία που αποθηκεύεται και μεταδίδεται σε μορφή κειμένου. Δεδομένων των υπαρχόντων πόρων, ο στόχος του μηχανικού εστιάζει στην αξιοποίηση αυτών με σκοπό τη δημιουργία εργαλείων ή υπηρεσιών που λύνουν κάποιο πρόβλημα.

1.2 Διαλογικά Συστήματα

Για τους λόγους που αναφέρθηκαν παραπάνω εστιάζουμε σε προβλήματα καθορισμένου σκοπού (task-oriented). Τέτοια μπορεί να είναι το κλείσιμο ενός εισιτηρίου σε κάποιο θέατρο ή η κράτηση ενός τραπεζιού σε ένα εστιατόριο. Παρόλα αυτά, υπάρχει και μια άλλη κατηγορία διαλογικών συστημάτων (chatbots) που δεν εστιάζει στην επίτευξη ενός συγκεκριμένου στόχου αλλά αποσκοπεί ακόμα και στην τέρψη του χρήστη. Μολονότι και σε αυτή την περίπτωση χρησιμοποιείται μια μετρική που προσδιορίζει την επιτυχία ή μη των συστημάτων αυτών, θίγεται το ζήτημα πως και σε αυτές τις περιπτώσεις υπάρχει ένας ξεκάθαρος σκοπός, ο οποίος για παράδειγμα θα μπορούσε να είναι η αύξηση των ενδορφινών στον εγκέφαλο. Δηλαδή, ακόμα και η πιο θεωρητικά ασήμαντη συνομιλία δεν είναι αίολη αλλά αντιθέτως υποβόσκει ένας σκοπός ακόμα κι αν η ενέργεια γίνεται ασυναίσθητα. Σε αυτές τις περιπτώσεις, το φάσμα των πιθανών διαλόγων και η θεματολογία είναι αρκετά μεγάλη. Μία έξυπνη μέθοδος είναι αυτή που χρησιμοποιήθηκε από τον Weizenbaum στην ELIZA [136], το πρώτο διαλογικό σύστημα που φέρει την ιδιότητα του ροτζεριανού ψυχοθεραπευτή. Κατά αυτή τη μέθοδο ψυχοθεραπείας, αν ένας ασθενής πει "Έκανα μια μεγάλη βόλτα με την βάρκα", ο ψυχίατρος μπορεί να αποκριθεί "Μίλησέ μου για τις βάρκες". Αν αυτή η συζήτηση δεν λάμβανε χώρα σε ένα ιατρείο, κατά πάσα πιθανότητα δεν θα έστεκε νοηματικά. Το συμπέρασμα είναι πως πρέπει να επιλέγουμε κατάλληλα το περιβάλλον στο οποίο υλοποιούμε τα διαλογικά συστήματα.

1.3 Ενισχυτική Μάθηση

Η Ενισχυτική Μάθηση (Reinforcement Learning) είναι ένας σπουδαίος κλάδος της μηχανικής μάθησης που ειδικά τα τελευταία χρόνια με τις σπουδαίες εφαρμογές της θεωρείται ένας ανερχόμενος κλάδος. Χρησιμοποιείται από τον κλάδο της ρομποτικής ή τη χρήση συστημάτων που παίζουν σκάκι ή go από μόνα τους μέχρι σε προσομοιώσεις που μοντελοποιούν την κοινωνική συμπεριφορά των πρακτόρων. Η διαφοροποίησή της με τις υπόλοιπες σύγχρονες κατηγορίες μηχανικής μάθησης έγκειται στο γεγονός ότι στοχεύει στη βέλτιστη λήψη αποφάσεων και επιπλέον αποτελεί ένα πλαίσιο το οποίο διαισθητικά αποτελεί μια αφαιρετική αναπαράσταση του ίδιου μας του κόσμου. Πράκτορες αλληλεπιδρούν με το περιβάλλον και λαμβάνουν ανταμοιβές προκειμένου να πετύχουν κάποιο σκοπό. Σε αυτό το σημείο, θα πρέπει να τονιστεί μια άποψη που σχηματίστηκε κατά τη διάρκεια της διπλωματικής. Κάθε πράκτορας θέτει τους δικούς του στόχους και το περιβάλλον δεν αποδίδει ποτέ ανταμοιβές, αλλά αλλάζει απλά καταστάσεις. Οι ανταμοιβές σχετίζονται με το πώς αντιλαμβάνεται και αξιολογεί ο πράκτορας τη μεταβολή του περιβάλλοντος.

1.4 Διαπραγμάτευση - Αντικείμενα Μελέτης

Όπως προείπαμε, θα μελετήσουμε ένα πρόβλημα καθορισμένου σκοπού. Ίσως από τα δυσκολότερα προβλήματα τους πεδίου αυτού είναι το πρόβλημα της διαπραγμάτευσης διότι απαιτεί τόσο μια γλωσσική επάρκεια όσο και μια συλλογιστική ικανότητα. Το πρόβλημα έχει ενδιαφέρον τόσο από θεωρητική σκοπιά (θεωρία παιγνίων) όσο και από πρακτική σκοπιά. Αν και μελετάται ένα πρόβλημα δύο πρακτόρων, μπορεί εύκολα να γενικευτεί και σε περισσότερους πράκτορες, αναμένοντας βέβαια και πιο σύνθετες συμπεριφορές. Στις περισσότερες περιπτώσεις αλληλεπίδρασης ανθρώπου-μηχανής, η σχέση τους είναι είτε ανταγωνιστική (π.χ., στο σκάκι ή στο Go) είτε συνεργατική (π.χ., οικιακοί βοηθοί). Το ενδιαφέρον σε αυτή την περίπτωση είναι οτι το διαλογικό σύστημα βρίσκεται κάπου ενδιάμεσα των δύο κατηγοριών και ανάλογα με το πώς θέλουμε να αντιμετωπίσει τον χρήστη μπορούμε να το ρυθμίσουμε κατάλληλα. Παρατηρούνται δύο αντίρροπες κατευθύνσεις προς τη μελέτη του προβλήματος της διαπραγμάτευσης. Η πρώτη σχετίζεται με την παραγωγή γλώσσας μέσω της ανάγκης ικανοποίησης του προβλήματος (bottom-up) και η δεύτερη σχετίζεται με τη χρησιμοποίηση υπαρχόντων γλωσσικών μοντέλων τα οποία εστιάζουν στο συγκεκριμένο πρόβλημα (top-down). Η πρώτη περίπτωση αποτελεί μια εξαιρετικά ενδιαφέρουσα επιλογή, η οποία ταυτίζεται αρκετά με τις ιδέες που έχουν παρατεθεί ως τώρα. Η δυσκολία της έγκειται στο κομμάτι της υλοποίησης και της ποικιλομορφίας της γλώσσας. Δεδομένης της έντονης ανάπτυξης του κλάδου της Επεξεργασίας Φυσικής Γλώσσας, δίνεται έμφαση στη δεύτερη κατεύθυνση και γίνεται χρήση των πιο σύγχρονων μοντέλων που παρατηρούνται στη βιβλιογραφία ως τώρα.

1.4.1 Deal Or No Deal

Το υπό μελέτη πρόβλημα Deal Or No Deal εισάγεται από τη Facebook και εστιάζει στη διαπραγμάτευση δύο πρακτόρων με δεδομένο πλήθος αντικειμένων μέσω ενός καναλιού επικοινωνίας σε γραπτό λόγο. Σε ένα πρόβλημα διαπραγμάτευσης, θα πρέπει το σύστημα να δέχεται ένα κείμενο, να κατανοεί σε τι διαμοιρασμό αντικειμένων αντιστοιχεί η είσοδος, να επεξεργάζεται την κατάλληλη απόφαση και εν τέλει να παράγει γλώσσα προκειμένου να μεταδώσει το μήνυμά της ενέργειας που θα λάβει. Σε μια πρώτη προσέγγιση το πρόβλημα αναλύεται σε δύο υποπροβλήματα: (1) στο Πρόβλημα Ταξινόμησης και (2) στο Πρόβλημα

της Γλωσσικής Μοντελοποίησης. Σημειώνεται ακόμα πως χρησιμοποιείται ο κώδικας της Facebook για το εν λόγω πρόβλημα [142]. Αρχικά, αναπαράξαμε τα αποτελέσματα και τροποποιήσαμε τον διαθέσιμο κώδικα υλοποίησης, επαυξάνοντάς τον με άλλα παραδοσιακά μοντέλα μηχανικής μάθησης και με χρήση προ-εκπαιδευμένων διανυσμάτων λέξεων. Τα αποτελέσματα παρουσιάζουν ανεπαίσθητες αποκλίσεις.

Παρόλα αυτά, στο πρόβλημα της Ταξινόμησης εντοπίστηκε οτι χρησιμοποιείται μια μετρική αξιολόγησης που δεν συνάδει με τη σπουδαιότητα επίλυσης του προβλήματος. Επιλέχθηκε αντί για τη μετρική της σύγχυσης στο πρόβλημα ταξινόμησης να χρησιμοποιηθεί η μετρική της ακρίβειας. Με τη μετρική της ακρίβειας θεωρούμε πως εστιάζουμε στην ουσία του ζητήματος, την σωστή κατανόηση της πρότασης που κάνει ένας πράκτορας στο μοντέλο μας. Υλοποιήσαμε, λοιπόν, αρχικά ορισμένους παραδοσιακούς ταξινομητές μηχανικής μάθησης και συγκρίναμε τα αποτελέσματά τους. Στη συνέχεια, κάναμε χρήση του Transformer BERT όπου τα αποτελέσματα ήταν εξαιρετικά καλύτερα.

Όσον αφορά τη γλωσσική μοντελοποίηση, επιδιώξαμε και σε αυτό το πρόβλημα να ερευνήσουμε αν η χρήση Μεταφοράς Μάθησης μπορεί να προσδώσει καλύτερα αποτελέσματα. Συγκεκριμένα, χρησιμοποιήθηκε ο Transformer GPT-2 και τα αποτελέσματα εμφανίζονται αρκετά καλύτερα. Στη συνέχεια, κάναμε χρήση του εκπαιδευμένου και βελτιστοποιημένου, πάνω στο σύνολο διαλόγων μας, GPT-2 προκειμένου να παράξουμε γλώσσα με τη μορφή διαλόγων. Συγκεκριμένα, παρουσιάζονται 4 τεχνικές παραγωγής κειμένου: (1) greedy decoding, (2) beam search, (3) top-k και (4) top-p (nucleus) sampling.

1.4.2 Αυτόματα Συστήματα Διαπραγμάτευσης

Έως τώρα η πολιτική λήψης αποφάσεων και η σχεδίαση στρατηγικών αγνοούνταν ή λαμβάνονταν υπόψη σε μεταγενέστερο στάδιο. Θεωρούμε, βέβαια, πως η απόφαση και η ενέργεια προϋπάρχει του μηνύματος και για αυτό σχεδιάσαμε και υλοποιήσαμε ένα περιβάλλον προσομοίωσης αυτόματης διαπραγμάτευσης το οποίο είναι απεξαρτημένο από τα γλωσσικά χαρακτηριστικά και εστιάζει σε χαμηλού επιπέδου αναπαραστάσεις των ενεργειών. Οι πράκτορες εκπαιδεύονται με τη χρήση Ενισχυτικής Μάθησης και συγκεκριμένα με τη μέθοδο Q-learning. Με αυτή τη μελέτη, διαμορφώνουμε τις συμπεριφορές των πρακτόρων, εξετάζουμε το βέλτιστο τρόπο απεικόνισης της διαπραγματευτικής πληροφορίας, δημιουργούμε συστήματα που επιχειρούν να εκμεταλλευτούν συγκεκριμένες συμπεριφορές των "αντιπάλων", μελετάμε το δίλημμα Εξερεύνησης-Αξιοποίησης και παρουσιάζουμε πώς η διαπραγμάτευση μεταξύ άπληστων πρακτόρων μπορεί να οδηγήσει σε μία κοινωνική ισορροπία.

Ένα επίσης μεγάλο ζήτημα της Μηχανικής Μάθησης είναι η Οπτικοποίηση και η Αξιολόγηση των Αποτελεσμάτων που συνήθως εξάγονται από τα Νευρωνικά Δίκτυα ή από πιο σύνθετα μοντέλα. Τα αποτελέσματα αν και σε πολλές περιπτώσεις είναι εξαιρετικά καλά, δεν είναι εύκολα ερμηνεύσιμα, το οποίο πολλές φορές μπορεί ναι είναι προβληματικό. Για παράδειγμα, έστω οτι θέλουμε να λάβουμε μια σοβαρή ιατρική απόφαση για την υγεία μας και έχουμε ένα σύστημα το οποίο προτείνει μια διαδικασία Α τη στιγμή που ο γιατρός μπορεί να προτείνει μια διαδικασία Β. Ποιον εμπιστευόμαστε; Και οι δύο μπορεί να σφάλλουν. Από τη στιγμή που το σύστημα προτείνει τη διαδικασία Α, δεν θα ήταν πολύ χρησιμότερο αν γνωρίζαμε τον λόγο που κατέληξε σε αυτή την απόφαση; Σε αυτή τη διπλωματική θα εστιάσουμε και σε αυτό το κομμάτι, όπου οπτικοποιούμε στο παίγνιο διαπραγμάτευσης, τις πιθανές κινήσεις ενός πράκτορα μαζί με αντίστοιχες αξίες των ενεργειών αυτών. Αυτή μέθοδος είναι εύκολα επεκτάσιμη και σε πολυπρακτορικά περιβάλλοντα. Επιπλέον, παρέχεται η δυνατότητα δημιουργίας rollouts.

1.5 Διάρθρωση Διπλωματικής Εργασίας

Στο Κεφάλαιο 2 γίνεται μια εισαγωγή στο θεωρητικό υπόβαθρο της Μηχανικής Μάθησης και των Βαθιών Νευρωνικών Δικτύων. Στη συνέχεια, τα Κεφάλαια 3 και 4 περιγράφουν τις βασικές ιδέες της Επεξεργασίας Φυσικής Γλώσσας. Πιο συγκεκριμένα, το Κεφάλαιο 3 μελετά τις Διανυσματικές Αναπαραστάσεις Λέξεων ενώ το Κεφάλαιο 4 μελετά τα Γλωσσικά Μοντέλα. Στη συνέχεια, στο Κεφάλαιο 5 γίνεται μια εκτενής εισαγωγή του κλάδου της Ενισχυτικής Μάθησης και στο Κεφάλαιο 6 παρουσιάζονται τα διαθέσιμα διαλογικά συστήματα και το πρόβλημα της διαπραγμάτευσης. Στο Κεφάλαιο 7, μελετάται η χρήση End-to-End Task-oriented Διαλογικών Συστημάτων στο πρόβλημα διαπραγμάτευσης Deal Or No Deal που εισάγεται από τη Facebook και προτείνονται βελτιώσεις των συστημάτων στα υποπροβλήματα που ορίζονται. Ακόμα, στο Κεφάλαιο 8, μελετάται η low-level ενέργεια στο πρόβλημα της διαπραγμάτευσης που σχετίζεται με τη λήψη μιας απόφασης και προτείνεται ένας τρόπος οπτικοποίησης των στρατηγικών που οι πράκτορες μαθαίνουν μέσω προσομοιώσεων trial-and-error. Τέλος, στο Κεφάλαιο 9 πραγματοποιείται μια αποτίμηση των εννοιών και υλοποιήσεων που παρουσιάστηκαν και προτείνονται κάποιες μελλοντικές προεκτάσεις του προβλήματος.

Μέρος Ι

Εισαγωγικές Έννοιες

Κεφάλαιο 2

Θεωρητικό Υπόβαθρο Μηχανικής Μάθησης

Στο Κεφάλαιο αυτό πραγματοποιείται μια εισαγωγή στο Θεωρητικό Υπόβαθρο της Μηχανικής Μάθησης (Machine Learning). Στα Κεφάλαια 2.1 και 2.2 παρουσιάζεται ο ορισμός και προσδιορίζονται οι κατηγορίες της Μηχανικής Μάθησης αντίστοιχα. Στο Κεφάλαιο 2.3 πραγματοποιείται μια εισαγωγή στα Παραδοσιακά Μοντέλα Μηχανικής Μάθησης και στο Κεφάλαιο 2.4 μελετώνται τα Νευρωνικά Δίκτυα και η Βαθιά Μάθηση. Ένα βιβλίο στο οποίο βασιστήκαμε και προτείνεται για μια εισαγωγή στη Βαθιά Μάθηση είναι το "Deep Learning" των Goodfellow, Bengio και Courville [33].

2.1 Ορισμός

Η Μηχανική Μάθηση (Machine Learning - ML) αποτελεί μέρος του κλάδου της Τεχνητής Νοημοσύνης (Artificial Intelligence - AI) και ορίστηκε για πρώτη φορά το 1959 από τον Άρθουρ Σάμουελ ως το "πεδίο μελέτης που δίνει στους υπολογιστές την ικανότητα να μαθαίνουν, χωρίς να έχουν ρητά προγραμματιστεί". Ένας επίσημος ορισμός δόθηκε λίγο αργότερα, το 1997, από τον Τομ Μίτσελ όπου προτείνει πως "Ένα πρόγραμμα υπολογιστή λέγεται ότι μαθαίνει από εμπειρία E ως προς μια κλάση εργασιών T και ένα μέτρο επίδοσης P, αν η επίδοσή του σε εργασίες της κλάσης T, όπως αποτιμάται από το μέτρο P, βελτιώνεται με την εμπειρία E" [81]. Η Μηχανική Μάθηση, λοιπόν, εστιάζει στο σχεδιασμό και την υλοποίηση αλγορίθμων και συστημάτων που εκπαιδεύονται και βελτιστοποιούνται αυτόματα. Η εκπαίδευση ενός μοντέλου πραγματοποιείται πάνω σε ένα σύνολο δεδομένων εισόδου και χρησιμοποιώντας στατιστική ανάλυση, αυτό βελτιστοποιείται προκειμένου να καταφέρει αποδοτικότερα είτε να αναγνωρίζει πρότυπα στις παρατηρήσεις είτε να λαμβάνει καλύτερες αποφάσεις.

2.2 Κατηγορίες Μηχανικής Μάθησης

Ο κλάδος της Μηχανικής Μάθησης διαχωρίζεται σε τρεις ευρείες κατηγορίες ανάλογα με τον τρόπο που πραγματοποιείται η διαδικασία της εκμάθησης και μεταδίδεται το σήμα ανατροφοδότησης στο υπό εκπαίδευση σύστημα. Η πρώτη κατηγορία ονομάζεται Επιβλεπόμενη Μάθηση

(Supervised Learning) και κατά την εκπαίδευση παρατίθενται μαζί με τα διαθέσιμα δεδομένα εισόδου και οι αντίστοιχες επισημασμένες έξοδοι (ετικέτες - labels). Η επισήμανση αυτή συνήθως πραγματοποιείται από ανθρώπους και συνεπώς η διαδικασία είναι αρκετά χρονοβόρα και από κάθε άποψη δαπανηρή. Η δεύτερη, λοιπόν, κατηγορία που δεν φέρει επισημασμένα δεδομένα αλλά επιχειρεί να ανακαλύψει δομές στις υπάρχουσες παρατηρήσεις ονομάζεται Μη Επιβλεπόμενη Μάθηση (Unsupervised Learning). Η τρίτη και τελευταία κατηγορία καλείται Ενισχυτική Μάθηση (Reinforcement Learning) και αντιμετωπίζει το υπό εκπαίδευση σύστημα ως έναν πράκτορα που αλληλεπιδρώντας με ένα δυναμικό περιβάλλον αποσκοπεί να μεγιστοποιήσει ένα κέρδος που ορίζεται από μία Συνάρτηση Ανταμοιβών (Reward Function). Συγκεντρωτικά, η πρώτη κατηγορία προσπαθεί να δημιουργήσει ένα μοντέλο που απεικονίζει δεδομένα εισόδου σε δεδομένα εξόδου, η δεύτερη επιδιώκει να αναγνωρίσει μια υποβόσκουσα δομή στα δεδομένα εισόδου και η τρίτη αποσκοπεί στη βέλτιστη λήψη αποφάσεων.

Σημειώνεται, επιπλέον, πως δύο ενδιαφέρουσες κατηγορίες που συναντώνται συχνά στη σχετική βιβλιογραφία είναι αυτή της Ημι-Επιβλεπόμενης Μάθησης (Semi-Supervised Learning) και της Μετα-Εκπαίδευσης (Meta Learning). Η πρώτη αποτελεί μια μίξη της Επιβλεπόμενης και Μη Επιβλεπόμενης Μάθησης, όπου κάποια δεδομένα είναι επισημασμένα ενώ τα περισσότερα δεν είναι και η δεύτερη εστιάζει στο πώς μπορεί ένα σύστημα να μάθει πώς να μαθαίνει.

2.2.1 Επιβλεπόμενη Μάθηση

Στην Επιβλεπόμενη Μάθηση έχουμε στη διάθεσή μας ένα σύνολο δεδομένων εισόδου X και τις αντίστοιχες επισημασμένες εξόδους y. Σκοπός είναι η εκμάθηση ενός μοντέλου το οποίο θα αντιστοιχεί τις εισόδους X στις εξόδους y. Συμβολίζουμε με $f(\cdot)$ τη συνάρτηση απεικόνισης (mapping) που περιγράφηκε.

$$X \to Y \text{ ή αλλιώς } y = f(X) \tag{2.1}$$

Το πρόβλημα έγκειται, συνεπώς, στην εύρεση ή προσέγγιση της κατάλληλης συνάρτησης απεικόνισης. Οι κατηγορίες προβλημάτων που παρατηρούνται στην Επιβλεπόμενη Μάθηση προκύπτουν από τη διαφορετική φύση των εξόδων y. Αν η έξοδος y είναι διακριτή τότε γίνεται λόγος για προβλήματα **Ταξινόμησης (Classification)** ενώ αν η έξοδος y είναι συνεχής τότε γίνεται λόγος για προβλήματα **Παλινδρόμησης (Regression)**. Στα προβλήματα Ταξινόμησης, δεδομένης κάποιας εισόδου το μοντέλο θα πρέπει να την ταξινομήσει σε κάποια κατηγορία ενώ στα προβλήματα Παλινδρόμησης το μοντέλο θα πρέπει να αποδώσει μια συνεχή τιμή ως έξοδο.

2.2.2 Μη Επιβλεπόμενη Μάθηση

Σε αντίθεση με την προηγούμενη κατηγορία, τα δεδομένα σε αυτή την περίπτωση δεν παρέχονται μαζί με αντίστοιχες ετικέτες (labels). Η Μη Επιβλεπόμενη Μάθηση επιχειρεί να ανακαλύψει την κατανομή των δεδομένων ή δομές που υποβόσκουν σε αυτά. Τα προβλήματα που συνήθως εντοπίζονται σε αυτή την κατηγορία είναι τα προβλήματα της **Ομαδοποίησης (Clustering)**. Στα προβλήματα αυτά, επιχειρείται να ομαδοποιηθούν τα δεδομένα εισόδου σε ορισμένες κατηγορίες-ομάδες ανάλογα με κάποια προκαθορισμένη μετρική απόστασης. Μια άλλη ενδιαφέρουσα κατηγορία προβλημάτων Μη Επιβλεπόμενης Μάθησης είναι αυτή των **Γεννητικών Μοντέλων (Generative Models)**. Μιμούμενα τα δεδομένα εκπαίδευσης, τα γεννητικά μοντέλα μπορούν εν συνεχεία να παράξουν νέα δεδομένα τα οποία διαφέρουν από αυτά της εκπαίδευσης.

2.2.3 Ενισχυτική Μάθηση

Η Ενισχυτική Μάθηση διαφέρει από τις προηγούμενες δύο κατηγορίες διότι εστιάζει στη βέλτιστη λήψη αποφάσεων. Εν συντομία, ένας πράκτορας αλληλεπιδρά με το περιβάλλον του προκειμένου να επιτύχει τη μεγιστοποίηση κέρδους που αποδίδεται από μια συνάρτηση ανταμοιβής. Ενδελεχής μελέτη της Ενισχυτικής Μάθησης παρουσιάζεται στην Ενότητα 5.

2.3 Παραδοσιακά Μοντέλα Μηχανικής Μάθησης

Στο πρόβλημα της Επιβλεπόμενης Μάθησης, σύμφωνα με την Εξίσωση 2.1, επιθυμούμε να αντιστοιχήσουμε ένα σύνολο δεδομένων εισόδου $\{x, x \in R^{d_{in}}\}$ σε ένα σύνολο δεδομένων εξόδου $\{y, y \in R^{d_{out}}\}$. Το πρόβλημα ανάγεται στον προσδιορισμό της κατάλληλης συνάρτησης $f(\cdot)$. Το ερώτημα που τίθεται είναι τί είδους συνάρτηση θα μπορούσαμε να χρησιμοποιήσουμε. Απλουστεύοντας προς το παρόν το πρόβλημα, γίνεται η θεώρηση οτι η συνάρτηση $f(\cdot)$ είναι μια γραμμική συνάρτηση της μορφής

$$f(x) = xW + b \tag{2.2}$$

όπου $W \in R^{d_{in} \times d_{out}}$ και $b \in R^{d_{out}}$. Πραγματοποιείται, λοιπόν, η κλάση υπόθεσης οτι η συνάρτηση $f(\cdot)$ είναι γραμμική εισάγοντας στο μοντέλο επαγωγική μεροληψία και ο στόχος είναι η εύρεση των κατάλληλων παραμέτρων $\theta = \{W, b\}$.

2.3.1 Συνάρτηση Κόστους

Έστω οτι εισάγεται η είσοδος x στο μοντέλο. Η έξοδος αντιστοιχεί στην πρόβλεψη που κάνει το μοντέλο μας και συμβολίζεται ως

$$\hat{y} = f(x) \tag{2.3}$$

Η ποσοτικοποίηση του σφάλματος μεταξύ της προβλεπόμενης εξόδου $\hat{y}$ και της πραγματικής εξόδους y γίνεται με τη χρήση της Συνάρτησης Κόστους $L(y, \hat{y})$. Η Συνάρτηση Κόστους αναθέτει μια βαθμωτή τιμή και θα πρέπει να είναι φραγμένη προς τα κάτω που σημαίνει οτι όσο χαμηλότερη η τιμή του σφάλματος τόσο καλύτερη είναι η πρόβλεψη. Δεδομένου ενός επισημασμένου σετ δεδομένων $(x_{1:n}, y_{1:n})$, το σφάλμα κατά μήκος όλου του συνόλου ορίζεται ως

$$L(\theta) = -\frac{1}{N} \sum_{i=1}^{N} L(f(x_i; \theta), y_i) \tag{2.4}$$

Στόχος, λοιπόν, είναι η εύρεση των βέλτιστων παραμέτρων $\hat{\theta}$ οι οποίες ελαχιστοποιούν το συνολικό σφάλμα

$$\hat{\theta} = \arg\min_{\theta} L(\theta) = \arg\min_{\theta} -\frac{1}{N} \sum_{i=1}^{N} L(f(x_i; \theta), y_i) \tag{2.5}$$

2.3.2 Μηχανές Διανυσμάτων Υποστήριξης

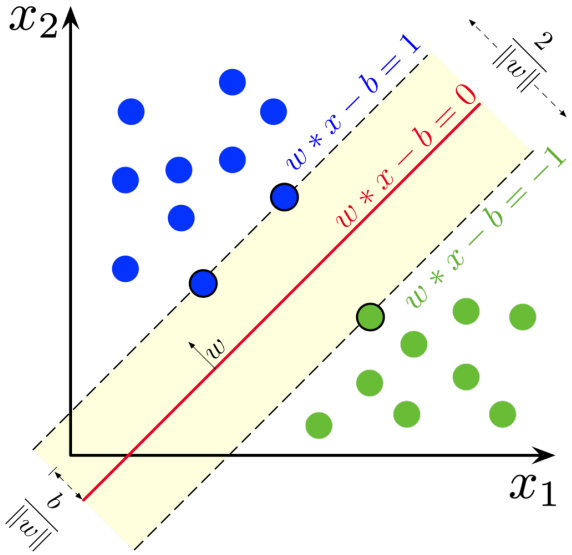
Έστω οτι έχουμε δύο γραμμικά διαχωρίσιμες κλάσεις και θέλουμε να σχεδιάσουμε ένα γραμμικό ταξινομητή. Στόχος μας είναι να σχεδιάσουμε ένα υπερεπίπεδο της μορφής

$$f(x) = w^T x + b \tag{2.6}$$

όπου $x \in R^d$ τα προς εκπαίδευση δεδομένα και $w \in R^d$, $b \in R^d$ οι παράμετροι του ταξινομητή. Ακόμα, συμβολίζουμε με $y \in \{-1, 1\}$ τις ετικέτες των δύο κλάσεων.

Το πλήθος των υπερεπιπέδων που μπορούν να ταξινομήσουν όλα τα δεδομένα εκπαίδευσης x με επιτυχία είναι άπειρο. Παρόλα αυτά, η επιλογή θα πρέπει να γίνει με τέτοιο τρόπο έτσι ώστε ο ταξινομητής να είναι αποδοτικός ως προς τη γενίκευση δεδομένων εκτός του συνόλου εκμάθησης. Το βέλτιστο υπερεπίπεδο (optimal hyperplane) είναι αυτό που μεγιστοποιεί το περιθώριο (margin) ανάμεσα στις δύο κλάσεις. Σε αυτό το σημείο σημειώνεται πως η απόσταση ενός σημείου x από το υπερεπίπεδο δίνεται από την σχέση

$$z = \frac{|f(x)|}{\|w\|} \tag{2.7}$$



Σχήμα 2.1: Μηχανές Διανυσμάτων Υποστήριξης (SVM). Wikipedia

Δεδομένου οτι οι δύο κλάσεις είναι γραμμικά διαχωρίσιμες, μπορούν να επιλεγούν παράμετροι w και b τέτοιοι ώστε

$$f(x) > 0 \ \forall x \text{ με ετικέτα } y = +1 \tag{2.8}$$
$$f(x) < 0 \ \forall x \text{ με ετικέτα } y = -1 \tag{2.9}$$

Κλιμακώνοντας το w μπορούμε να απαιτήσουμε η $f(x)$ στα πλησιέστερα σημεία των δύο κλάσεων να είναι $+1$ και -1 για κάθε κλάση αντίστοιχα. Με αυτή την απαίτηση το μέγιστο μήκος περιθωρίου γίνεται $\frac{2}{\|w\|}$ και έχουμε

$$f(x) \geq 1 \ \forall x \text{ με ετικέτα } y = +1 \tag{2.10}$$
$$f(x) \leq -1 \ \forall x \text{ με ετικέτα } y = -1 \tag{2.11}$$

Το πρόβλημα πλέον ανάγεται στην εύρεση των παραμέτρων w και b ούτως ώστε να ελαχιστοποιείται η συνάρτηση

$$J(w, b) = \frac{2}{\|w\|} \tag{2.12}$$

$$\text{υπό τους περιορισμούς } y_i(w^T x_i + b) \geq 1, \ i = 1, 2, \cdots, N \tag{2.13}$$

Η λύση του τετραγωνικού προβλήματος γραμμικού προγραμματισμού (Linear Programming) δίνεται με τη χρήση πολλαπλασιαστών Lagrange.

2.3.3 Λογιστική Παλινδρόμηση (Logistic Regression)

Η Λογιστική Παλινδρόμηση (Logistic Regression) χρησιμοποιείται στην επίλυση γραμμικών προβλημάτων ταξινόμησης. Η διαφοροποίησή της μεταξύ άλλων απλών ταξινομητών είναι πως υπολογίζει την πιθανότητα ενός δείγματος εισόδου $x \in R^d$ να ανήκει σε μία κλάση. Έστω οτι μελετάμε ένα πρόβλημα δυαδικής ταξινόμησης με κλάσεις $y = \{0, 1\}$. Στο διάνυσμα της εξόδου μιας συνάρτησης $f(\cdot)$ εφαρμόζεται η λογιστική ή αλλιώς σιγμοειδής συνάρτηση που συμπιέζει τις τιμές του διανύσματος στο πεδίο (0,1).

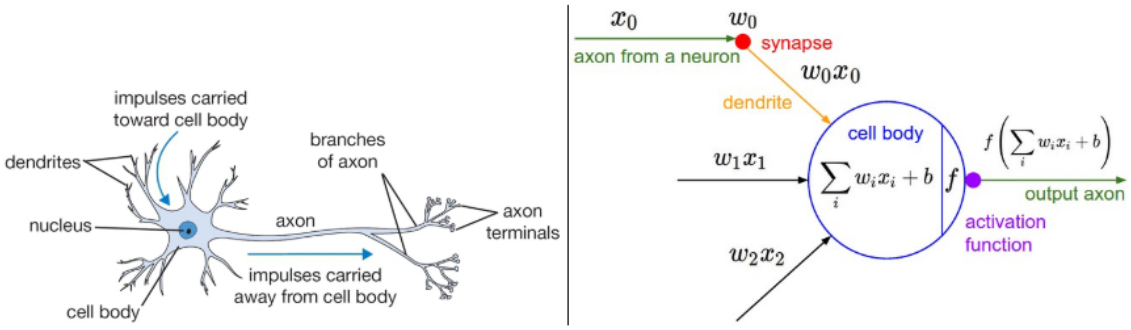
$$P(y = 1|x) = \frac{1}{1 + e^{-f(x)}} \tag{2.14}$$

όπου η $f(x)$ μπορεί να είναι μια γραμμική συνάρτηση της μορφής 2.6. Αν η πιθανότητα αυτή είναι μεγαλύτερη του 0.5 τότε το δείγμα x κατηγοριοποιείται στην πρώτη κλάση. Εναλλακτικά ταξινομείται στη δεύτερη με πιθανότητα $P(y = 0|x) = 1 - P(y = 1|x)$. Η συνάρτηση κόστους που καλούμαστε να ελαχιστοποιήσουμε είναι η συνάρτηση κόστους διεντροπίας (cross-entropy) που ορίζεται από τη σχέση

$$J(w) = -[ylog(P(y = 1|x)) + (1 - y)log(1 - P(y = 1|x))] \tag{2.15}$$

2.4 Νευρωνικά Δίκτυα

Η Βαθιά Μάθηση (Deep Learning) ανήκει στον κλάδο της Μηχανικής Μάθησης και εστιάζει στην εξαγωγή προτύπων από τα δεδομένα με τη χρήση νευρωνικών δικτύων (neural networks). Τα Τεχνητά Νευρωνικά Δίκτυα (Artificial Neural Networks) αποτελούν υπολογιστικά συστήματα βασισμένα σε μια αφαιρετική αναπαράσταση των νευρώνων του ανθρώπινου εγκεφάλου. Η χρήση και η ικανότητα βελτιστοποίησης των παραμέτρων των νευρωνικών δικτύων προσέφεραν στον κλάδο της Βαθιάς Μάθησης σπουδαία αποτελέσματα σε ποικίλες εφαρμογές, κυρίως λόγω του οτι η αρχιτεκτονική αυτή μπορεί να προσεγγίσει μη-γραμμικές συναρτήσεις.



Σχήμα 2.2: Βιολογικός Νευρώνας (Αριστερά) και η Αφαιρετική Υπολογιστική του Αναπαράσταση (Δεξιά) [51]

Βασική υπολογιστική μονάδα του εγκεφάλου αποτελεί ο νευρώνας. Ένας νευρώνας δέχεται ένα ερέθισμα μέσω των δενδριτών και παράγει ένα σήμα εξόδου στον άξονά του. Ο άξονάς του με τη σειρά του συνδέεται μέσω συνάψεων με δενδρίτες άλλων νευρώνων. Στο υπολογιστικό μοντέλο, το σήμα των αξόνων που διέρχεται από τις συνάψεις όπου και πολλαπλασιάζεται με κάποιο βάρος. Τα νέα σήματα έχοντας δεχθεί αυτή την επεξεργασία διέρχονται στο σώμα του

νευρώνα όπου και αθροίζονται και το αποτέλεσμα διέρχεται από μία συνάρτηση ενεργοποίησης (activation function) η οποία αναπαριστά τη συχνότητα των σημάτων αυτών στον άξονα.

Πολλοί νευρώνες συνδεδεμένοι μεταξύ τους δημιουργούν αρχιτεκτονικές που ονομάζονται πολυεπίπεδα αντίληπτρα (multi-layer perceptrons). Ένα νευρωνικό δίκτυο αποτελείται από το επίπεδο της εισόδου (input layer), από ένα ή περισσότερα κρυφά επίπεδα (hidden layer) και από ένα επίπεδο εξόδου (output layer). Όσον αφορά τις συναρτήσεις ενεργοποίησης (activation functions) συνήθως χρησιμοποιούνται η σιγμοειδής συνάρτηση, η υπερβολική εφαπτομένη, η Rectified Linear Unit (ReLU) ή η Leaky ReLU.

Το πρόβλημα που συναντάται συχνά είναι αυτό του overfitting και συμβαίνει όταν η συνάρτηση που βελτιστοποιούμε εκπαιδεύεται τόσο πολύ πάνω στα δεδομένα της εκπαίδευσης, όπου μοντελοποιεί και τον θόρυβο αυτών και δεν επιτυγχάνει καλές γενικεύσεις. Ένας τρόπος να αντιμετωπιστεί αυτό το πρόβλημα είναι μέσω της Κανονικοποίησης (Regularization).

$$\hat{\theta} = \arg\min_{\theta} L(\theta) = \arg\min_{\theta} -\frac{1}{N}\sum_{i=1}^{N} L(f(x_i;\theta), y_i) + \lambda R(\theta) \tag{2.16}$$

Η υπερπαράμετρος (hyperparameter) λ δεν είναι δεδομένη αλλά εξαρτάται από το πρόβλημα και οι όροι κανονικοποίησης $R(\theta)$ υπολογίζουν νόρμες των πινάκων των παραμέτρων. Δύο νόρμες που χρησιμοποιούνται είναι η L1 και L2 νόρμες όπως αυτές περιγράφονται από της Εξισώσεις 2.17 και 2.18 αντίστοιχα.

$$R_{L_1}(W) = \|W\|_1 = \sum_{i,j} |W_{[i,j]}| \tag{2.17}$$

$$R_{L_2}(W) = \|W\|_2^2 = \sum_{i,j} (W_{[i,j]})^2 \tag{2.18}$$

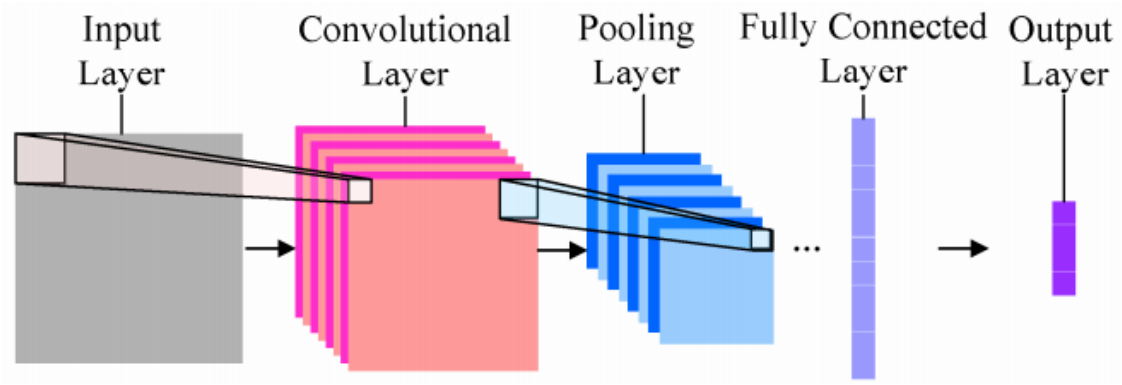
Εναλλακτικοί τρόποι κανονικοποίησης είναι η τεχνική Dropout [116] που απενεργοποιεί τυχαία κάποιους νευρώνες σε ένα δίκτυο και η τεχνική Early Stopping όπου η εκμάθηση σταματά τη στιγμή που το Σφάλμα του σετ αξιολόγησης σταματά να μειώνεται.

Για να κάνουμε εκμάθηση στο νευρωνικό δίκτυο χρειάζεται να λύσουμε το πρόβλημα βελτιστοποίησης που περιγράφεται από την Εξίσωση 2.16. Ενδεικτικά κάποιοι προτεινόμενοι τρόποι είναι ο Αλγόριθμος Καθόδου (Gradient Descent), ο Στοχαστικός Αλγόριθμος Απότομης Καθόδου (Stochastic Gradient Descent) [10], ο Adagrad [24], ο Adadelta [143] και ο Adam [54]. Για να ελαχιστοποιηθεί η Συνάρτηση Κόστους χρειάζεται να υπολογιστούν οι παράγωγοι και να ενημερωθεί το δίκτυο. Μια αποτελεσματική μέθοδος είναι η Οπίσθια Διάδοση (Backpropagation) [103], [59], όπου οι παράγωγοι υπολογίζονται με συστηματικό τρόπο και αποτελεσματικά με τη χρήση του κανόνα της αλυσίδας.

2.4.1 Συνελικτικά Νευρωνικά Δίκτυα

Μια σημαντική κατηγορία Δικτύων, εμπνευσμένη από τον κλάδο της Νευροεπιστήμης, είναι τα Συνελικτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks), τα οποία αποτελούν δίκτυα εμπρόσθιας διάδοσης που βασίζουν τη λειτουργία τους στη διαδικασία της συνέλιξης. Η συνέλιξη ορίζεται ως

$$s(t) = x(t) * w(t) = \int x(h)w(t-h)dh \tag{2.19}$$



Σχήμα 2.3: Η δομή ενός Συνελτικού Νευρωνικού Δικτύου [92]

όπου η συνάρτηση w ονομάζεται φίλτρο (filter ή kernel). Βέβαια, στις υλοποιήσεις της Μηχανικής Μάθησης χρησιμοποιείται η διακριτή συνέλιξη

$$s(t) = x(t) * w(t) = \sum_{h} x(t+h)w(h) \tag{2.20}$$

Χρησιμοποιείται ένα σύνολο από φίλτρα τα οποία είναι μικρότερα σε μέγεθος από την είσοδο προκειμένου να εντοπίσουν τοπικές δομές. Έπειτα από τη διαδικασία της συνέλιξης, χρησιμοποιείται μια μη-γραμμική συνάρτηση ενεργοποίησης (πχ. ReLU) και στη συνέχεια λαμβάνει χώρα μια υποδειγματοληψία με τη μέθοδο pooling. Συνήθως, γίνεται χρήση του max pooling όπου επιλέγεται μόνο η μέγιστη τιμή εντός μιας περιοχής και του average pooling όπου η τιμή εντός μιας περιοχής προκύπτει από το μέσο όρο των τιμών της. Τέλος, οι τιμές που προκύπτουν συνδέονται συχνά με ένα πλήρως συνδεδεμένο στρώμα νευρώνων.

2.5 Σύνοψη

Στο Κεφάλαιο αυτό, πραγματοποιήθηκε μια Εισαγωγή στις Βασικές Αρχές και το Θεωρητικό Υπόβαθρο της Μηχανικής Μάθησης και των Βαθιών Νευρωνικών Δικτύων. Το κεφάλαιο αυτό αποτελεί τον ακρογωνιαίο λίθο της διπλωματικής εργασίας καθώς τα σύγχρονα μοντέλα που θα μελετήσουμε στη συνέχεια βασίζονται στις βασικές αρχές που περιγράψαμε. Στις επόμενες ενότητες, γίνεται μια εισαγωγή στο θεωρητικό υπόβαθρο της Επεξεργασίας Φυσικής Γλώσσας (ΕΦΓ). Συγκεκριμένα, στο Κεφάλαιο 3 μελετάμε τις Διανυσματικές Αναπαραστάσεις Λέξεων και στο Κεφάλαιο 4 γίνεται μια εισαγωγή στα μοντέλα της ΕΦΓ.

Κεφάλαιο 3

Διανυσματικές Αναπαραστάσεις Λέξεων

Σε αυτό το κεφάλαιο μελετώνται οι τρόποι με τους οποίους αναπαρίστανται οι λέξεις προκειμένου να χρησιμοποιηθούν οι αναπαραστάσεις αυτές στην είσοδο των μοντέλων που θα αναφερθούν στα επόμενα κεφάλαια και αποσκοπούν στην επίλυση πληθώρας προβλημάτων της **Επεξεργασίας Φυσικής Γλώσσας (ΕΦΓ)**. Για λόγους πληρότητας αναφέρεται οτι πολύ πριν την οργανωμένη σύσταση του κλάδου της Επεξεργασίας Φυσικής Γλώσσας, κυριαρχούσε η ιδέα της χρήσης ατομικών συμβόλων για τον προσδιορισμό κάθε λέξης. Στη μελέτη αυτή, ζητείται η δημιουργία ενός διανύσματος για κάθε μία λέξη που υπάρχει στο λεξικό. Ο στόχος είναι οι διανυσματικές αναπαραστάσεις να προσδίδουν μια αίσθηση ομοιότητας ή διαφοροποίησης ανάμεσα σε λέξεις που είναι όμοιες ή άσχετες αντίστοιχα μεταξύ τους. Από την στιγμή που οι λέξεις μετατρέπονται σε διανύσματα, σε αυτά μπορούν να προσδοθούν ιδιότητες, χρησιμοποιώντας μετρικές απόστασης όπως είναι οι Jaccard, Cosine και Euclidean. Τα διανύσματα που καλούμαστε να δημιουργήσουμε συναντώνται στην βιβλιογραφία με τους όρους **word vectors ή word embeddings**.

Κρίνεται σκόπιμο να περιγραφούν οι δύο διαφορετικές σημασιολογικές προσεγγίσεις: (α) η συμβολική σημασιολογία (**denotional semantics**) και (β) η διανεμητική σημασιολογία (**distributional semantics**). Η συμβολική σημασιολογία αντιμετωπίζει τις λέξεις ως ξεχωριστά σύμβολα, δημιουργεί πιο αραιές αναπαραστάσεις και δεν προσδίδει κάποια αίσθηση ομοιότητας ή ανομοιότητας μεταξύ των λέξεων (**localist representation**). Αντιθέτως, η διανεμητική σημασιολογία δημιουργεί λεξικές αναπαραστάσεις ανάλογα με το περιεχόμενο (**context**) που εμφανίζεται κάποια λέξη. Είναι πιο πυκνές αναπαραστάσεις και λόγω της φύσης τους συγκρατούν ομοιότητες ή ανομοιότητες μεταξύ των λέξεων.

Στην αγγλική γλώσσα, εκτιμάται οτι υπάρχουν 13 εκατομμύρια διαφορετικά tokens. Παρόλα αυτά, το νόημα μεταξύ κάποιων λέξεων δεν είναι τόσο διαφορετικό. Για παράδειγμα, η σχέση μεταξύ των λέξων "child" και "children" είναι πολύ πιο στενή συγκριτικά με αυτή των λέξεων "child" και "embeddings". Αν συμβολίσουμε ως V το λεξικό μας, δηλαδή τις λέξεις που κατέχουμε στην γνώση μας και $|V|$ το πλήθος των λέξεων που υπάρχουν στο λεξικό αυτό, αναζητούμε, κατά προτίμηση, έναν "γλωσσικό" χώρο διαστάσεων $|V|\text{x}N$ όπου $N \ll 13^6$. Έτσι, σε κάθε λέξη θα αντιστοιχεί ένα διάνυμα μεγέθους $1\text{x}N$. Ιδανικά, κάθε μία από τις N διαστάσεις θα θέλαμε να πληροί διαφορετική σημασιολογική ή και συντακτική ιδιότητα (πχ. κάποια διάσταση να σχετίζεται με τον ενικό ή πληθυντικό και κάποια άλλη με ενεστώτα-αόριστο κλπ).

Αρχικά, με τον όρο λεξικό V καλείται μια δομή που περιέχει όλες τις λέξεις που υπάρχουν σε μια γλώσσα και συνήθως κάθε μία από αυτές έρχεται σε μια αντιστοιχία με κάποια άλλη οντότητα. Στη μελέτη αυτή, ένα λεξικό μπορεί να αποτελείται από μια λέξη και το αντίστοιχο διάνυσμα λέξης που την περιγράφει.

Στις Ενότητες 3.1 και 3.2 παρουσιάζονται κάποιες απλές προσεγγίσεις αναπαράστασης των λέξεων: ως δείκτες σε λεξικά και ως one-hot διανύσματα αντίστοιχα. Έπειτα, παρουσιάζονται κάποιες SVD Μέθοδοι (Ενότητα 3.3) και στη συνέχεια γίνεται η μετάβαση σε iteration-based μεθόδους, όπως είναι το Word2Vec (Ενότητα 3.4). Τέλος, αναφερόμαστε στα Διανύσματα Glove (Ενότητα 3.5) και στις contextual διανυσματικές αναπαραστάσεις λέξεων (Ενότητα 3.6).

3.1 Λέξεις ως Δείκτες στο Λεξικό

Η πρώτη αφελής ιδέα είναι η χρησιμοποίηση κάποιων δεικτών που αναφέρονται στα σημεία που είναι αποθηκευμένη κάθε λέξη στο λεξικό μας. Έστω οτι η λέξη "δημιουργία" αντιστοιχεί στον δείκτη $i^{\text{δημιουργία}} = 42$ και η λέξη "κόσμος" αντιστοιχεί στον δείκτη $i^{\text{κόσμος}} = 12$. Αν εξαιρέσουμε το γεγονός οτι δεν επιτυγχάνουμε τα προαναφερθέντα χαρακτηριστικά που επιθυμούμε, δημιουργούνται ανισοτικές σχέσεις μεταξύ των λέξεων. Στο συγκεκριμένο παράδειγμα, ισχύει οτι $i^{\text{δημιουργία}} > i^{\text{κόσμος}}$ και προσδίδουμε μια σημασιολογική σχέση η οποία δεν αποδίδει την πραγματική συσχέτιση των λέξεων.

3.2 One-Hot Vectors

Ο όρος "one-hot" προέρχεται από τον κλάδο της Ψηφιακής Σχεδίασης Κυκλωμάτων, όπου για τον εντοπισμό ενός σήματος χρειαζόμαστε μία ενεργή στάθμη και όλες τις υπόλοιπες ανενεργές. Ομοίως, κάθε λέξη απεικονίζεται ως ένα διάνυσμα μεγέθους $\mathbb{R}^{|V|\text{x}1}$, όπου όλες οι $|V|$ διαστάσεις λαμβάνουν την τιμή 0 εκτός από μιας που λαμβάνει την τιμή 1 και η θέση στην οποία λαμβάνει αυτή την τιμή αντιστοιχεί στον δείκτη στον οποίο είναι αποθηκευμένη στο λεξικό. Για παράδειγμα, σε ένα λεξικό $V = \{\text{δημιουργία, κόσμος, τεχνητή, νοημοσύνη}\}$ με $|V| = 4$, δημιουργούνται τα ακόλουθα one-hot διανύσματα:

$$w^{\text{δημιουργία}} = \begin{bmatrix}1\\0\\0\\0\end{bmatrix}, w^{\text{κόσμος}} = \begin{bmatrix}0\\1\\0\\0\end{bmatrix}, w^{\text{τεχνητή}} = \begin{bmatrix}0\\0\\1\\0\end{bmatrix}, w^{\text{νοημοσύνη}} = \begin{bmatrix}0\\0\\0\\1\end{bmatrix}$$

Είναι εμφανές οτι αν μεγαλώσουν οι διαστάσεις του λεξικού μας, τόσο πιο μεγάλες θα γίνουν και οι λεξικές αναπραστάσεις. Ο πίνακας που για κάθε λέξη περιέχει και την αντίστοιχη αναπαράσταση είναι διαστάσεων $|V|\text{x}|V|$. Αυτός ο πίνακας θα είναι της παρακάτω μορφής,

$$\begin{bmatrix}w^{\text{δημιουργία}^\top}\\w^{\text{κόσμος}^\top}\\w^{\text{τεχνητή}^\top}\\w^{\text{νοημοσύνη}^\top}\end{bmatrix} = \begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix}$$

Αυτός ο πίνακας είναι αρκετά αραιός και αν αναλογιστούμε οτι υπάρχει περίπτωση να χρησιμοποιήσουμε λεξικά πολύ μεγάλου μεγέθους για κάθε ένα διάνυσμα θα έχουμε πάρα πολύ

άχρηστη και ασχρησιμοποίητη πληροφορία (0) και μόνο ένα 1. Σπαταλάμε μεγάλο μέρος της μνήμης με αυτό τον τρόπο. Εκτός αυτου, τα διανύσματα των λέξεων δεν υποδηλώνουν κάποια εξάρτηση μεταξύ τους γιατί είναι ανεξάρτητα μεταξύ τους ανά δύο. Ενδεικτικά,

$$(w^{\text{τεχνητή}})^\top w^{\text{νοημοσύνη}} = (w^{\text{τεχνητή}})^\top w^{\text{κόσμος}} = 0$$

3.3 SVD Μέθοδοι

Οι SVD μέθοδοι καλούνται να λύσουν τα παραπάνω προβλήματα. Καλούνται, έτσι, διότι χρησιμοποιούν τη μέθοδο Singular Value Decomposition προκειμένου να μειώσουν τις διαστάσεις [31]. Τα διαφορετικά είδη των SVD μεθόδων προκύπτουν κατά βάση από τον πίνακα στον οποίο επιλέγει η κάθε μέθοδος να εφαρμόσει μείωση διαστάσεων.

3.3.1 Word-Document Matrix

Έστω οτι έχουμε M πλήθος αρχείων κειμένου. Αρχικοποιούμε με μηδενικά έναν πίνακα X διαστάσεων $|V|\text{x}M$. Κάθε γραμμή αντιστοιχεί σε μια συγκεκριμένη λέξη και κάθε στήλη σε κάθε ένα από τα M έγγραφα. Για κάθε αρχείο j, διαπερνάμε όλο το κείμενο και κάθε φορά που συναντάμε κάποια λέξη, έστω i αυξάνουμε κατά μια μονάδα την τιμή $X_{i,j}$ του πίνακα. Ουσιαστικά, δημιουργούμε έναν μετρητή λέξεων ανά αρχείο. Με αυτόν τον τρόπο έχουμε δημιουργήσει τον ζητούμενο πίνακα. Παρόλα αυτά, πρακτικά τα αρχεία μπορεί να είναι πάρα πολλά και ως αποτέλεσμα ο πίνακας να είναι πολύ μεγάλος. Στη συνέχεια παρουσιάζεται μια καλύτερη προσέγγιση.

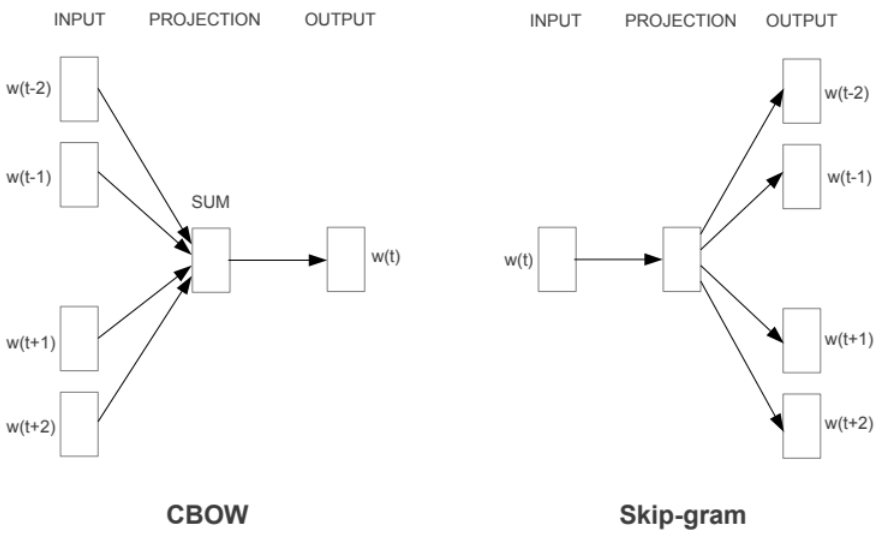
3.3.2 Window based Co-occurence matrix

Σε αυτή την περίπτωση ο πίνακας X αποθηκέυει co-occurences γειτονικών λέξεων εντός ενός προκαθορισμένου παραθύρου (affinity matrix).

3.4 Iteration based Methods - Word2Vec

Μια διαφορετική αντιμετώπιση του προβλήματος αναγνωρίζει οτι ο υπολογισμός και η αποθήκευση ενός τεράστιου όγκου πληροφορίας σχετικά με τις συσχετίσεις των λέξεων σε τεράστια αρχεία κειμένων είναι χρονοβόρος και κοστοβόρος. Εστιάζει, λοιπόν, στον υπολογισμό μέσω επαναληπτικών διαδικασιών (iterations) και προσδιορίζει την πιθανότητα εμφάνισης μιας λέξης, δεδομένου του περιεχομένου (context) στο οποίο βρίσκεται, ένα βήμα τη φορά. Στη βιβλιογραφία παρατηρούνται διάφορες προσεγγίσεις [8] [15] αλλά αυτή που στιγμάτισε τον κλάδο της Επεξεργασίας Φυσικής Γλώσσας είναι το Word2Vec του Mikolov [77]. Για την καλύτερη κατανόησή του παρουσιάζονται τα τέσσερα δομικά του στοιχεία. Το Continuous Bag of Words (CBOW) και το skip-gram αποτελούν τους δύο προτεινόμενους αλγορίθμους ενώ το Negative Sampling [80] και Hierarchical Softmax αποτελούν τις δύο προτεινόμενες μεθόδους εκπαίδευσης.

Όπως παρατηρείται στο Σχήμα 3.1, ο αλγόριθμος CBOW δεδομένου του περιεχομένου (context) μιας λέξης, προσπαθεί να προβλέψει τη κεντρική αυτή λέξη. Αντιθέτως, στο Skip-gram μοντέλο, δεδομένης μιας κεντρικής λέξης επιχειρείται να προβλεφθεί η κατανομή των



Σχήμα 3.1: Οι δύο προτεινόμενοι αλγόριθμοι Word2Vec: CBOW (Αριστερά) και Skip-gram (Δεξιά). Η εικόνα πάρθηκε από το [77].

λέξεων που συνιστούν το περιεχόμενο (context) της κεντρικής λέξης. Επιπλέον, το Negative Sampling βασίζεται στη δειγματοληψία "αρνητικών" παραδειγμάτων ενώ το Hierarchical Softmax προτείνει μια αποδοτική δενδρική δομή για τον υπολογισμό των πιθανοτήτων των λέξεων του λεξικού.

3.5 Glove

Σε αντίθεση με τις παραπάνω μεθόδους το Glove [93] είναι βασισμένο σε ένα μοντέλο που προβλέπει την πιθανότητα εμφάνισης μιας λέξης j που εμφανίζεται στο περιεχόμενο (context) μιας λέξης i. Η εκμάθηση πραγματοποιείται με μία συνάρτηση κόστους ελαχίστων τετραγώνων. Επιπλέον, με αυτό τον τρόπο χρησιμοποιούνται όλα τα στατιστικά των κειμένων και επιτυγχάνεται η δημιουργία ενός διανυσματικού χώρου λέξεων που έχει συγκρατήσει αρκετά σημαντική πληροφορία των νοημάτων των λέξεων.

3.6 Contextual Word Embeddings

Τα διανύσματα λέξεων που έχουν περιγραφεί ως τώρα είναι στατικά. Παρόλα αυτά, πολλές είναι οι φορές που μπορεί να χρησιμοποιηθεί μια συγκεκριμένη λέξη σε διαφορετικές προτάσεις και να αποδίδει ένα εντελώς διαφορετικό νόημα. Για παράδειγμα, στην πρόταση "κάνετε διπλό κλικ με το ποντίκι σας", η λέξη ποντίκι έχει διαφορετική σημασία από το ζώο. Κρίνεται, λοιπόν, ανεπαρκές να χρησιμοποιηθεί η ίδια στατική αναπαράσταση για αυτή τη λέξη. Σχηματίστηκε, λοιπόν, η άποψη οτι το περιεχόμενο είναι αυτό που καθορίζει τη λέξη και η ίδια λέξη σε διαφορετικό περιεχόμενο θα πρέπει να αναπαρίσταται με διαφορετικό τρόπο. Μια γνωστή κατηγορία Contextual Word Embeddings είναι το ELMo [95]. Σημειώνεται πως τέτοιου είδους διανύσματα μπορούν να προκύψουν μέσω γλωσσικών μοντέλων, αναδρομικών δικτύων και Transformers.

3.7 Σύνοψη

Στο Κεφάλαιο αυτό μελετήσαμε τους διαφορετικούς τρόπους με τους οποίους μπορούμε να αναπαραστήσουμε τις λέξεις. Η κατάλληλη διανυσματικής αναπαράσταση των λέξεων παίζει σπουδαίο ρόλο καθώς αυτά τα διανύσματα εισάγονται στα μοντέλα που περιγράφονται στο επόμενο κεφάλαιο.

Κεφάλαιο 4

Γλωσσικά Μοντέλα (Language Models)

Σε πληθώρα προβλημάτων Επεξεργασίας Φυσικής Γλώσσας καλούμαστε να υπολογίσουμε την πιθανότητα εμφάνισης μιας συγκεκριμένης πρότασης. Χρειαζόμαστε, λοιπόν, την δημιουργία ενός μοντέλου το οποίο θα υπολογίζει την πιθανότητα εμφάνισης μιας ακολουθίας πεπερασμένου αριθμού λέξεων που σχηματίζει κάποια πρόταση. Αυτά τα μοντέλα ονομάζονται Γλωσσικά Μοντέλα (Language Models). Τα γλωσσικά μοντέλα, επιπλέον, μπορούν να χρησιμοποιηθούν και στην παραγωγή προτάσεων. Η μαθηματική μοντελοποίηση του προβλήματος παρουσιάζεται στην εξίσωση (4.1).

$$P(w_1, ..., w_m) = \prod_{i=1}^{m} P(w_i | w_1, ..., w_{i-1}) \tag{4.1}$$

Επιθυμούμε, λοιπόν, το μοντέλο που θα δημιουργήσουμε να δίνει μικρή πιθανότητα σε προτάσεις που δεν συνηθίζουμε να χρησιμοποιούμε ή είναι συντακτικά λανθασμένες και αντιθέτως να δίνει μεγαλύτερη πιθανότητα σε προτάσεις οι οποίες είναι πιο σύνηθες να συναντήσουμε. Για παράδειγμα, η πρόταση "πολύ την εγώ τεχνητή αγαπώ νοημοσύνη" αναμένουμε να έχει μικρότερη πιθανότητα από την πρόταση "εγώ αγαπώ πολύ την τεχνητή νοημοσύνη". Σε αυτό το σημείο, αξίζει να τονίσουμε οτι επειδή μια πρόταση μπορεί να μην εμφανίζεται συχνά δεν σημαίνει οτι αυτή είναι λανθασμένη ή δεν μπορεί να σταθεί στον προφορικό ή γραπτό λόγο! Το Γλωσσικό Μοντέλο, λοιπόν, θα αποδώσει μικρή πιθανότητα και χαμηλή αξιολόγηση ως προς την ορθότητα της πρότασης.

Στο κεφάλαιο αυτό θα μελετηθούν διαφορετικά είδη Γλωσσικών Μοντέλων και συγκεκριμένα θα αναλύσουμε και θα αξιολογήσουμε τα ακόλουθα μοντέλα. Συγκεκριμένα, θα μελετήσουμε το n-gram γλωσσικό μοντέλο (ενότητα 4.1) και στην συνέχεια το πρώτο μεγάλης κλίμακας γλωσσικό μοντέλο βαθιάς μάθησης (ενότητα 4.2). Έπειτα, εισάγουμε τα Recurrent Neural Networks (ενότητα 4.3), τη δομική μονάδα Gated Recurrent Unit (ενότητα 4.4) και τα Long-Short Term Memories (ενότητα 4.6). Εισάγουμε τη μετρική της Σύγχυσης (ενότητα 4.5) και στην ενότητα 4.7 μελετώνται οι Transformers όπου δίνεται έμφαση στο BERT (ενότητα 4.7.4) και στο GPT-2 (ενότητα 4.7.5).

4.1 n-γραμματικά Γλωσσικά Μοντέλα

Με τον όρο n-γραμμα καλούμε κάθε ακολουθία n διαδοχικών όρων. Πολλές φορές συναντάμε τους όρους δίγραμμα που αντιστοιχεί σε μια ακολουθία δύο διαδοχικών όρων (π.χ. "καλός μαθητής") και τρίγραμμα που αντιστοίχως περιγράφει μια ακολουθία τριών διαδοχικών όρων (π.χ. "βαθιά νευρωνικά δίκτυα"). Ανάλογα με την επιλογή του n, ονομάζουμε και το αντίστοιχο μοντέλο n-γραμματικό. Ερμηνεύοντας αυτή την επιλογή, αυτό που πραγματικά επιχειρούμε είναι να κρατήσουμε κάποια στατιστικά στοιχεία για τις συνεμφανίσεις λέξεων εντός ενός παραθύρου. Οι ακόλουθες δύο εξισώσεις υποδηλώνουν την σχέση δι-γραμματικών και τρι-γραμματικών γλωσσικών μοντέλων.

$$P(w_2|w_1) = \frac{count(w_1, w_2)}{count(w_1)} \tag{4.2}$$

$$P(w_3|w_1, w_2) = \frac{count(w_1, w_2, w_3)}{count(w_1, w_2)} \tag{4.3}$$

Συνεπώς, η πιθανότητα εμφάνισης της επόμενης λέξης δεδομένων των συμφραζομένων εντός ενός παραθύρου καθορίζεται από την επιλογή του n. Όμως, ένα τέτοιο μικρό παράθυρο δεν μπορεί να συμπεριλάβει το περιεχόμενο της πρότασης στην ολότητά του. Για παράδειγμα, έστω οτι έχουμε την πρόταση "Ήθελα να παίξω μπάσκετ και έτσι πήγα στο _____". Λογικά κάποιος θα συμπλήρωνε την λέξη "γήπεδο". Αν χρησιμοποιούσαμε ένα τρι-γραμμικό μοντέλο, αυτό θα έπρεπε να προβλέψει την ζητούμενη λέξη δεχόμενο ως δεδομένα την πρόταση "πήγα στο _____". Είναι εμφανές πως αυτή η μοντελοποίηση δεν είναι βέλτιστη.

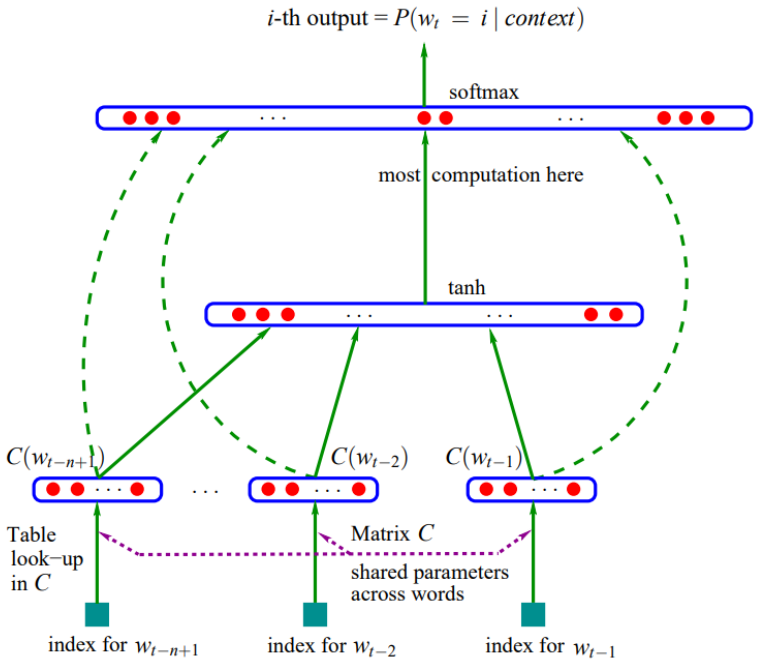
Δημιουργούνται, ακόμα, και δύο επιπρόσθετα προβλήματα. Το πρώτο πρόβλημα αφορά το γεγονός οτι ο αριθμητής πχ. στην εξίσωση 3 μπορεί να είναι μηδέν επειδή είναι πολύ πιθανό οι λέξεις w_1, w_2 και w_3 να μην έχουν εμφανιστεί ποτέ διαδοχικά. Όμως, αυτό έχει ως συνέπεια να μηδενίζεται και η πιθανότητα να εμφανιστεί η λέξη w_3 μετά από την διαδοχική εμφάνιση των λέξεων w_1 και w_2. Συνήθως, το ζήτημα αυτό επιλύεται με μεθόδους που αποκαλούνται smoothing ή discounting, οι οποίες επιδιώκουν να προσδώσουν κάποια μικρή πιθανότητα στην εμφάνιση λέξεων [27]. Επίσης, η εξίσωση 4.3 μπορεί να μην ορίζεται διότι αν οι λέξεις w_1 και w_2 δεν εμφανιστούν ποτέ διαδοχικά που είναι πιθανό μηδενίζουν τον παρονομαστή. Συνήθως, επιλύεται με την μέθοδο backoff [56]. Σε αυτή την κατηγορία προβλημάτων, η αύξηση του παραθύρου συμβάλλει στην αύξηση των ζητημάτων που περιγράφηκαν. Η δεύτερη κατηγορία προβλημάτων αφορά την αποθήκευση στην μνήμη. Ο όγκος των δεδομένων που χρειάζεται να αποθηκευτεί σχετίζεται με το πλήθος των n-γραμμάτων που συναντώνται στην συλλογή κειμένων. Προφανώς, όσο αυξάνεται το n, δηλαδή το παράθυρο, τόσο περισσότερη πληροφορία πρέπει να αποθηκεύσει το μοντέλο.

Μια τυπική τιμή για το μέγεθος του παραθύρου είναι $n \leq 5$. Χρειάζεται να αντισταθμιστούν τα προβλήματα που προκύπτουν από την επιλογή ενός μικρού παραθύρου με την ικανότητα αυτού να συμπεριλαμβάνει τα συμφραζόμενα (context) της πρότασης.

4.2 Window-based Neural Language Model

Η κατάρα της διαστατικότητας (curse of dimensionality) είναι ένα σύνηθες ζήτημα στον κλάδο των Βαθιάς Μάθησης και εν γένει της Αναγνώρισης Προτύπων. Αυτό το πρόβλημα

επεδίωξε να αντισταθμίσει ο Bengio [8] με την χρήση νευρωνικών δικτύων για τη δημιουργία Γλωσσικών Μοντέλων. Το προτεινόμενο μοντέλο επιδιώκει συγχρόνως να μάθει αφενός μια κατανεμημένη αναπαράσταση λέξεων και αφετέρου μια συνάρτηση πιθανότητας για λεξικές ακολουθίες.



Σχήμα 4.1: Αρχιτεκτονική Window-based Neural Language Model [8]

Ως περιεχόμενο της πρότασης, και σε αυτή την περίπτωση χρησιμοποιείται ένα παράθυρο n λέξεων. Για να προβλέψουμε την επόμενη λέξη, λοιπόν, εισάγουμε στο μοντέλο τις n λέξεις που προηγήθηκαν. Αυτές αναζητούνται σε έναν πίνακα C οι παράμετροι του οποίου είναι κοινές για όλες τις λέξεις του παραθύρου. Με αυτόν τον τρόπο, κωδικοποιούμε τις λέξεις, οι αντίστοιχες αναπαραστάσεις των οποίων αναγράφονται ως $C(w_{t-n+1})$, $C(w_{t-2})$ και $C(w_{t-1})$ και πλέον καλούνται word embeddings ($C(w) \in \mathbb{R}^{d_w}$). Αυτά τα διανύσματα συνενώνονται σειριακά, και περνούν σε ένα κρυφό στρώμα, η έξοδος του οποίου οδηγείται σε ένα επίπεδο ταξινόμησης softmax. Η λειτουργία του μοντέλου μπορεί να περιγραφεί με τις εξισώσεις 4.4 και 4.5.

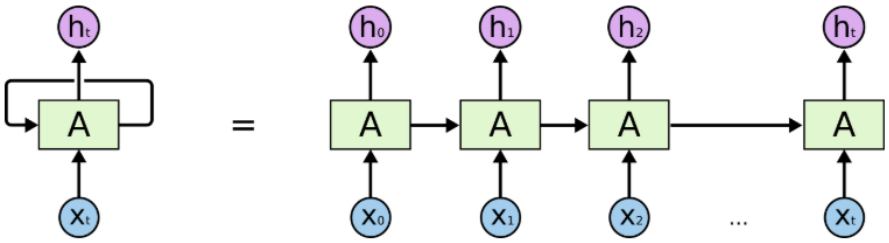
$$\boldsymbol{x} = [C(w_{t-n+1}); C(w_{t-2}); ...; C(w_{t-1})] \tag{4.4}$$

$$\hat{y} = softmax(W^{(2)} tanh(W^{(1)} x + b^{(1)}) + W^{(3)} x + b^{(3)}) \tag{4.5}$$

4.3 Recurrent Neural Networks (RNNs)

Τα εμπρόσθια νευρωνικά μοντέλα, όπως ήταν αυτό που περιγράψαμε στην προηγούμενη ενότητα, αντικαταστάθηκαν από αναδρομικά νευρωνικά δίκτυα (RNNs) διότι τα τελευταία δεν περιορίζονται σε ένα μικρό παράθυρο λέξεων αλλά μπορούν να χρησιμοποιήσουν όλες τις λέξεις της ακολουθίας [79]. Κάθε λέξη εισάγεται σε διαφορετική χρονική στιγμή. Συνεπώς,

υπάρχουν τόσες χρονικές στιγμές (timesteps) όσο είναι και το μέγεθος της πρότασης ή της ακολουθίας που εισάγουμε.



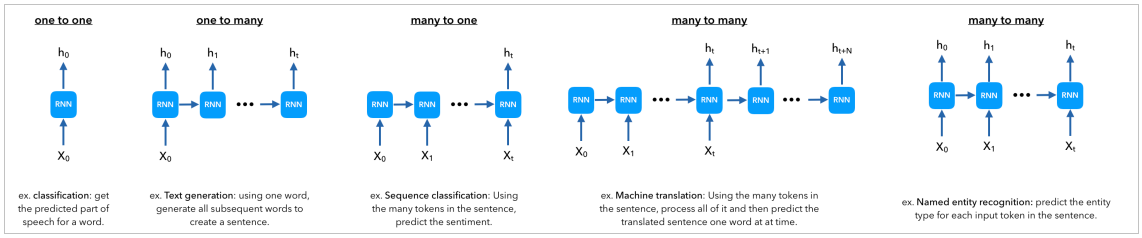
Σχήμα 4.2: Ένα αναδρομικό νευρωνικό δίκτυο (RNN)[88]

Θα μελετήσουμε τις εξισώσεις την χρονική στιγμή t:

$$h_t = tanh(W_{hh}h_{t-1} + W_{xh}X_t + b_h), \text{ όπου} \tag{4.6}$$

$W_{hh} \in \mathbb{R}^{HXH}$ είναι ο πίνακας βαρών των hidden units (H είναι το μέγεθος των κρυφών διαστάσεων), $h_{t-1} \in \mathbb{R}^{NXH}$ είναι η κρυφή κατάσταση της προηγούμενης χρονικής στιγμής (N είναι το μέγεθος του batch), $W_{xh} \in \mathbb{R}^{EXH}$ είναι ο πίνακας βαρών των στοιχείων της εισόδου (E είναι οι διαστάσεις των embeddings), $X_t \in \mathbb{R}^{NXE}$ είναι η είσοδος την χρονική στιγμή t, $b_h \in \mathbb{R}^{HX1}$ είναι το hidden units bias και τέλος h_t είναι η έξοδος του Αναδρομικού Νευρωνικού Δικτύου την χρονική στιγμή t.

Η στιγμή $t = 0$ είναι η μοναδική χρονική στιγμή όπου το δίκτυο δέχεται μία μόνο είσοδο. Ως h_{t-1} θα μπορούσαμε να χρησιμοποιήσουμε ένα μηδενικό διάνυσμα (χωρίς εκμάθηση) ή κάποια αρχικοποιημένη τιμή (με εκμάθηση). Να τονιστεί οτι τα Αναδρομικά Νευρωνικά Δίκτυα χρησιμοποιούνται σε πληθώρα προβλημάτων που ανήκουν στην οικογένεια της Επεξεργασίας Φυσικής Γλώσσας. Όπως, φαίνεται στην παρακάτω εικόνα, συνήθως λαμβάνουμε διαφορετικές εξόδους του Δικτύου μας για διαφορετικά προβλήματα.



Σχήμα 4.3: Εφαρμογές Αναδρομικών Νευρωνικών Δικτύων [84]

Οι παράμετροι που θα πρέπει να μάθει το Αναδρομικό Νευρωνικό δίκτυο είναι οι πίνακες W_{hh} και W_{xh} (και b_h). Συνεπώς, συγκριτικά με τις παραδοσιακές τεχνικές, πλέον επιτυγχάνουμε να ενσωματώνουμε όλα τα στοιχεία (κατά βάση λέξεις) μιας ακολουθίας στο δίκτυό μας (δίχως να αποκλείουμε κάποιες λόγω περιορισμένου μεγέθους παραθύρου) και ανεξαρτήτως αριθμού λέξεων το μοντέλο μας μαθαίνει συγκεκριμένο αριθμό παραμέτρων. Αν αυξηθούν, δηλαδή, οι λέξεις σε μια πρόταση ή το ίδιο το λεξικό μας το μοντέλο δεν θα μεγαλώσει! Επιπλέον, το μοντέλο, δεχόμενο κάθε χρονική στιγμή ως είσοδο ένα διάνυσμα που προέρχεται από την έξοδο του την χρονική στιγμή $t-1$, ενσωματώνει θεωρητικά πληροφορία από όλες τις παρελθοντικές λέξεις.

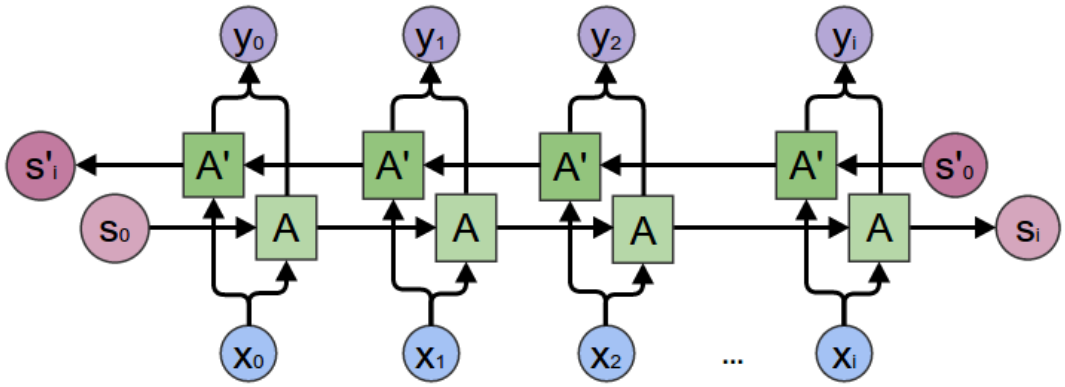
4.3.1 Προβλήματα

Ακριβώς επειδή τα στοιχεία της ακολουθίας εισάγονται σειριακά, ένα πρόβλημα των RNNs είναι οτι δεν παρέχουν την δυνατότητα παραλληλοποίησης των υπολογισμών και αυτό τα καθιστά αργά ως προς την εκπαίδευση. Παρόλα αυτά, η αντικατάστασή τους από πιο σύγχρονα μοντέλα οφείλεται στο πρόβλημα των παραγώγων που μπορούν να πάρουν είτε πολύ μικρές (< 1) είτε πολύ μεγάλες τιμές (> 1). Έτσι, κατά την οπισθοδιάδοση (backpropagation) αν πολλαπλασιάζονται διαρκώς μεγάλες τιμές μεταξύ τους οδηγούν τις παραγώγους στο άπειρο. Αντίστοιχα, αν αν πολλαπλασιάζονται διαρκώς μικρές τιμές μεταξύ τους οδηγούν τις παραγώγους στο μηδέν. Το πρόβλημα αυτό ονομάζεται Vanishing Gradients και Gradient Explosion. Δύο ενδεικτικές λύσεις είναι: (α) το gradient clipping, μια ευριστική που προτάθηκε από τον Mikolov και κόβει ουσιαστικά τις παραγώγους αν ξεπεράσουν κάποιο κατώφλι και (β) η χρήση Rectified Linear Units (ReLUs) αντί για σιγμοϊδείς συναρτήσεις ενεργοποίησης.

Επιπλέον, ενώ θεωρητικά χρησιμοποιούμε μια παρελθοντική πληροφορία, δεν σημαίνει οτι αυτή έτσι όπως προσφέρεται είναι η πλέον ενδεδειγμένη μορφή κωδικοποίησης των παρελθοντικών διαδικασιών του μοντέλου. Οι συσχετίσεις λέξεων που βρίσκονται μακριά μεταξύ τους σε μια ακολουθία είναι δύσκολες και κάποιες φορές ανύπαρκτες [37], [7].

4.3.2 Επεκτάσεις των Αναδρομικών Νευρωνικών Δικτύων

Προτού επεκταθούμε και σε άλλα μοντέλα τα οποία τροποποιούν το εσωτερικό σώμα των υπολογισμών του Αναδρομικού Νευρωνικού Δικτύου, κρίνεται σκόπιμο να αναφερθούμε συνοπτικά σε δύο προεκτάσεις των RNNs οι οποίες είναι χρήσιμες και συμβατές με τις αρχιτεκτονικές που θα αναλύσουμε στη συνέχεια. Η πρώτη επέκταση αφορά την χρήση Αμφίδρομων Αναδρομικών Νευρωνικών Δικτύων (Deep Bidirectional RNNs) και η δεύτερη αφορά την επαύξηση του πλήθους των κρυφών επιπέδων μέσα στο σώμα του RNN.



Σχήμα 4.4: Αμφίδρομο Αναδρομικό Νευρωνικό Δίκτυο [87]

4.3.2.1 Αμφίδρομα Αναδρομικά Νευρωνικά Δίκτυα (Bidirectional RNNs)

Τα Bidirectional RNNs [46] βασίζονται στην ιδέα του να διατρέξουν την ακολουθία από δύο κατευθύνσεις: μια από αριστερά προς τα δεξιά και μια από δεξιά προς τα αριστερά. Επαυξάνουμε, λοιπόν, την εξίσωση 4.6 ως εξής:

$$\vec{h_t} = f(\vec{W_{hh}} h_{t-1}^{\rightarrow} + \vec{W_{xh}} X_t + \vec{b_h}) \tag{4.7}$$

$$\overleftarrow{h_t} = f(\overleftarrow{W_{hh}} h_{t+1}^{\leftarrow} + \overleftarrow{W_{xh}} X_t + \overleftarrow{b_h}) \tag{4.8}$$

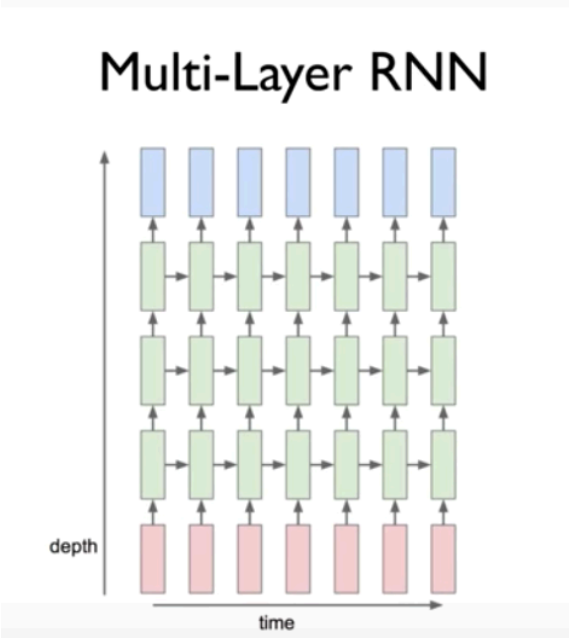
$$h_t = [\vec{h_t}; \overleftarrow{h_t}] \tag{4.9}$$

$$\hat{y} = g(U h_t + c) \tag{4.10}$$

$$\tag{4.11}$$

4.3.2.2 Πολυστρωματικά Αναδρομικά Νευρωνικά Δίκτυα (Multi-layer RNNs)

Η ιδέα είναι να επαυξηθεί το βάθος του δικτύου με επιπρόσθετα στρώματα νευρώνων όπου κάθε στρώμα αντιπροσωπεύει διαφορετικά χαρακτηριστικά της ακολουθίας. Στο σχήμα 4.5, με κόκκινο χρώμα απεικονίζονται τα word embeddings, με πράσινο τα units του RNN και με μπλε οι έξοδοι του μοντέλου. Να παρατηρήσουμε πως στο πρώτο layer ως είσοδο στο μοντέλο έχουμε τα embeddings και την έξοδο του προηγούμενου χρονικά κελιού από το ίδιο στρώμα (δηλαδή του κελιού στα αριστερά). Στα ψηλότερα στρώματα, συνεχίζουμε να λαμβάνουμε ως είσοδο την έξοδο του χρονικά προηγούμενου κελιού, λαμβάνουμε, όμως, και ως είσοδο την έξοδο του κελιού που αντιστοιχεί στην ίδια χρονική στιγμή αλλά στο ακριβώς κατώτερο στρώμα. Για την χρονική στιγμή $t = 0$, ισχύει ό,τι ισχύει και στην απλή περίπτωση RNN, όπου είτε εισάγουμε μια τυχαία τιμή είτε βάζουμε κάποια συγκεκριμένη.



Σχήμα 4.5: Multi-layer RNN

Στη συνέχεια, καταγράφονται και οι εξισώσεις που ικανοποιούν την γενική περίπτωση, αυτή που έχουμε πολλά στρώματα και αμφίδρομο RNN.

$$\vec{h_t}^{(0)} = f(\vec{W_{hh}}^{(0)} \vec{h_{t-1}}^{(0)} + \vec{W_{xh}}^{(0)} X_t + \vec{b_h}^{(0)}) \tag{4.12}$$

$$\overleftarrow{h_t}^{(0)} = f(\overleftarrow{W_{hh}}^{(0)} \overleftarrow{h_{t+1}}^{(0)} + \overleftarrow{W_{xh}}^{(0)} X_t + \overleftarrow{b_h}^{(0)}) \tag{4.13}$$

$$\vec{h_t}^{(i)} = f(\vec{W_{hh}}(i) \vec{h_{t-1}}^{(i)} + \vec{W_{xh}}(i) h_t^{()i-1)} + \vec{b_h}^{(i)}), \quad 1 \leq i \leq L \tag{4.14}$$

$$\overleftarrow{h_t}^{(i)} = f(\overleftarrow{W_{hh}}^{(i)} \overleftarrow{h_{t+1}^{(i)}} + \overleftarrow{W_{xh}}^{(i)} h_t^{(i-1)} + \overleftarrow{b_h}^{(i)}), \quad 1 \leq i \leq L \tag{4.15}$$

$$h_t = [\vec{h_t}^{(L)}; \overleftarrow{h_t}^{(L)}] \tag{4.16}$$

$$\hat{y} = g(U h_t + c) \tag{4.17}$$

Οι παρενθέσεις πάνω από κάθε μεταβλητή δηλώνει το στρώμα της μεταβλητής. Επιπλέον, με L συμβολίζουμε τον αριθμό των στρωμάτων.

4.4 Gated Recurrent Units (GRUs)

Προς βελτίωση των προβλημάτων που δημιουργούνται στα RNNs προτάθηκε το δομικό στοιχείο Gated Recurrent Unit (GRU) όπου επιδιώκει να βελτιώσει την μνήμη του μοντέλου και να καταφέρει να διατηρήσει συσχετίσεις λέξεων που είναι απομακρυσμένες σε μια ακολουθία. Χρησιμοποιούνται, για τον σκοπό αυτό, κάποιες πύλες όπως περιγράφουν οι παρακάτω εξισώσεις.

$$z_t = (W^{(z)} x_t + U^{(z)} h_{t-1}) \quad \text{(Update Gate)} \tag{4.18}$$

$$r_t = (W^{(r)} x_t + U^{(r)} h_{t-1}) \quad \text{(Reset Gate)} \tag{4.19}$$

$$\tilde{h_t} = tanh(r_t \circ U h_{t-1} + W x_t) \quad \text{(New Memory)} \tag{4.20}$$

$$h_t = (1 - z_t) \circ \tilde{h_t} + z_t \circ h_{t-1} \quad \text{(Hidden State)} \tag{4.21}$$

Οι διαφορετικές πύλες εξυπηρετούν και διαφορετικούς σκοπούς. Το z_t όπως βλέπουμε από την εξίσωση 4.21 καθορίζει πόσο πολύ το μοντέλο θα χρησιμοποιήσει την παλιά κρυφή κατάσταση και πόσο πολύ θα εστιάσει στην δημιουργία νέας μνήμης $\tilde{h_t}$. Το reset gate 4.19 είναι υπεύθυνο να διαγράψει την παλιότερη μνήμη αν αυτή δεν χρειάζεται πλέον.

4.5 Αξιολόγηση Γλωσσικών Μοντέλων - Σύγχυση (Perplexity)

Υπάρχει μια ευρεία γκάμα επιλογών ως προς τις μετρικές των Γλωσσικών Μοντέλων. Η πιο συνηθισμένη μετρική προέρχεται από την θεωρία της πληροφορίας, ονομάζεται Σύγχυση (Perplexity) και αποτελεί μια εκθετική συνάρτηση της εντροπίας. Ένα καλό γλωσσικό μοντέλο μπορεί να καταφέρει να προβλέπει με επιτυχία μια ακολουθία που δεν έχει ξαναδεί. Μικρή εντροπία, λοιπόν, είναι προτιμότερη από μεγάλη εντροπία αφού αποτελέσματα τα οποία μπορούμε να προβλέψουμε είναι πιο επιθυμητά από αποτελέσματα που είναι χαοτικά και τυχαία (με μεγαλύτερη εντροπία).

Ορίζουμε, λοιπόν, μαθηματικά την έννοια της Σύγχυσης (Perplexity).

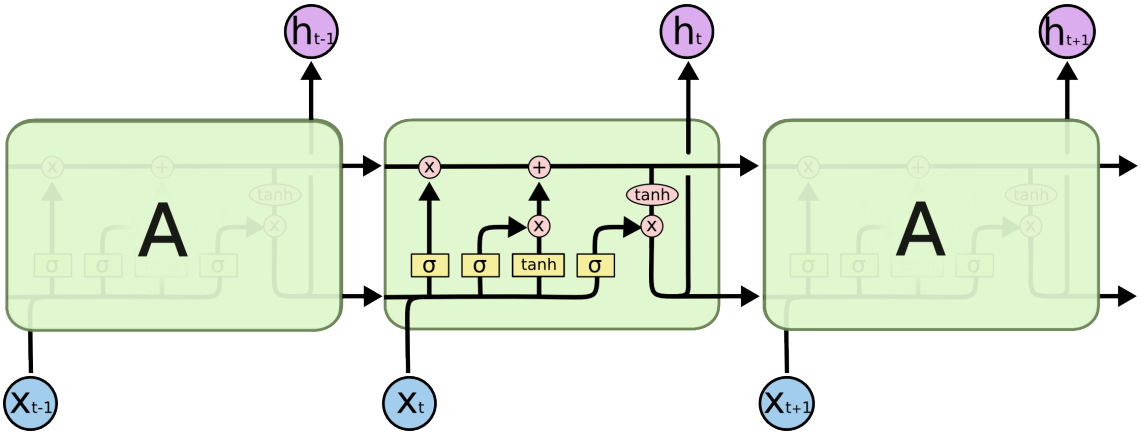
$$\text{Perplexity} = 2^{-\frac{1}{2}\sum_{i=1}^{n} log_2 \text{LM}(w_i|w_{1:-1})} \text{ , όπου} \tag{4.22}$$

n είναι το πλήθος των λέξεων ενός εγγράφου και LM είναι το Γλωσσικό μας Μοντέλο. Εν κατακλείδι, επιθυμούμε μικρές τιμές σύγχυσης έναντι μεγάλων τιμών.

4.6 Long-Short-Term Memories (LSTMs)

Τα LSTMs [38] αποτελούν μια δημοφιλή εκδοχή των RNNs γιατί μετριάζουν πολλά από τα προβλήματα που έχουν προαναφερθεί. Αν και η φιλοσοφία είναι παρόμοια με αυτή των GRUs, η αρχιτεκτονική διαφέρει αρκετά.

$$i_t = (W^{(i)}x_t + U^{(i)}h_{t-1}) \quad \text{(Input Gate)} \tag{4.23}$$
$$f_t = (W^{(f)}x_t + U^{(f)}h_{t-1}) \quad \text{(Forget Gate)} \tag{4.24}$$
$$o_t = (W^{(o)}x_t + U^{(o)}h_{t-1}) \quad \text{(Output/Exposure Gate)} \tag{4.25}$$
$$\tilde{c_t} = tanh(W^{(c)}x_t + U^{(c)}h_{t-1}) \quad \text{(New Memory cell)} \tag{4.26}$$
$$h_t = ft \circ c_{t-1} + i_t \circ \tilde{c_t} \quad \text{(Final Memory cell)} \tag{4.27}$$
$$h_t = o_t \circ tanh(c_t) \tag{4.28}$$

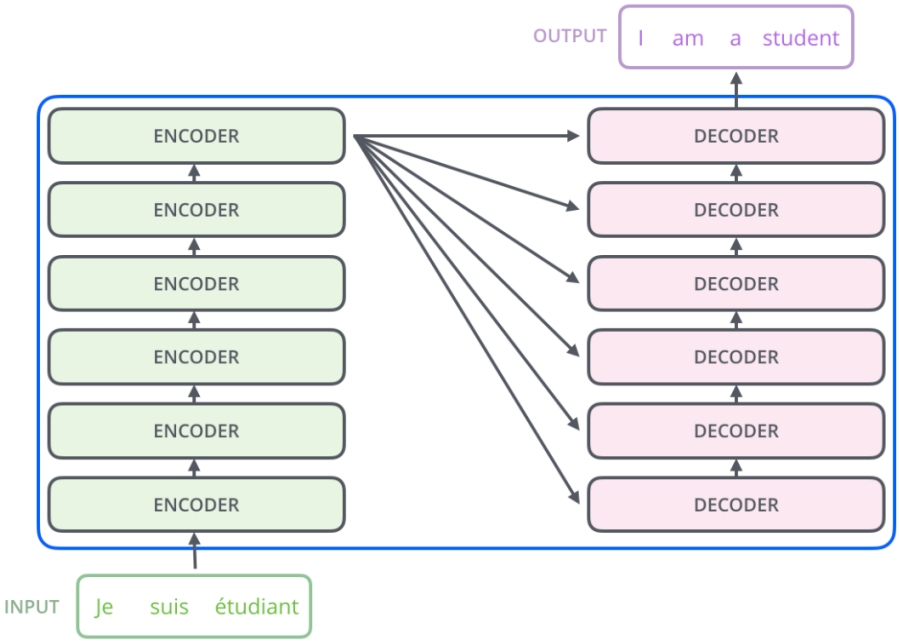


Σχήμα 4.6: Long Short Term Memory (LSTM) [88]

4.7 Transformers

Προκειμένου να περιγραφούν τα μοντέλα που χρησιμοποιήθηκαν (GPT-2 [99] και BERT [23]), κρίνεται αναγκαίο να πραγματοποιηθεί μια γενική εισαγωγή στους Transformers [125] διότι αυτοί αποτελούν την βάση των μοντέλων αυτών. Η μελέτη αυτή είναι βασισμένη στην περιγραφή του Jay Allamar [2]. Τα έως τότε δημοφιλή μοντέλα ήταν βασισμένα στην αναδρομή (recursion) ή στις συνελίξεις (convolutions) και συνέδεαν τον Κωδικοποιητή με τον Αποκωδικοποιητή μέσω ενός Μηχανισμού Προσοχής (Attention Mechanism). Οι Transformers βασίζονται εξ' ολοκλήρου στον Μηχανισμό αυτό και οδηγούν σε αποδοτικότερες υλοποιήσεις και επιτρέπουν την παραλληλοποίηση των υπολογισμών μειώνοντας δραματικά τον χρόνο εκπαίδευσης. Οι Transformers εστίασαν στην επίλυση προβλημάτων αυτόματης μετάφρασης

(Machine Translation) και η αρχιτεκτονική τους εξηγεί το γεγονός αυτό αφού ο κωδικοποιητής, κωδικοποιεί την πληροφορία από μία γλώσσα σε κάποια ενδιάμεση αναπαράσταση, η οποία εισάγεται στον Αποκωδικοποιητή και παράγεται το αποτέλεσμα σε μία άλλη γλώσσα. Η αρχιτεκτονική αυτή, φυσικά, με ορισμένες τροποποιήσεις μπορεί να εφαρμοστεί σε μια ευρεία γκάμα προβλημάτων. Όπως απεικονίζεται και στην Εικόνα 4.7 ένας transformer απο-



Σχήμα 4.7: Η γενική δομή ενός Transformer [2]

τελείται από ένα σύνολο 6 κωδικοποιητών που συνδέεται με ένα άλλο σύνολο 6 αποκωδικοποιητών. Ο αριθμός αυτός αποτελεί αρχιτεκτονική παρέμβαση των ερευνητών. Το σύνολο των κωδικοποιητών όπως και το σύνολο των αποκωδικοποιητών αποτελείται αντίστοιχα από συστήματα τα οποία είναι πανομοιότητα αλλά φυσικά φέρουν διαφορετικές τιμές βαρών. Οι κωδικοποιητές και οι αποκωδικοποιητές δέχονται εισόδους από τα κατώτερα στρώματα και τις μεταφέρουν στα ανώτερα. Κάθε κωδικοποιητής αποτελείται από δύο υποσυστήματα. Το πρώτο είναι ένα στρώμα Αυτο-Προσοχής (Self-Attention Layer) το οποίο επιτρέπει στον κωδικοποιητή να συγκρατεί τις εξαρτήσεις που μπορεί να έχει η υπό μελέτη λέξη με τις υπόλοιπες λέξεις που υπάρχουν στην πρόταση. Το δεύτερο υποσύστημα είναι ένα Νευρωνικό Δίκτυο Εμπρόσθιας Διάδοσης (Feed Forward Neural Network). Παρόμοια είναι και η αρχιτεκτονική των αποκωδικοποιητών με την μόνη διαφορά οτι στα δύο υποσυστήματα που περιγράφηκαν υπάρχει και ένα ενδιάμεσο σε αυτά στρώμα που ονομάζεται Στρώμα Προσοχής Κωδικοποιητή-Αποκωδικοποιητή (Encoder-Decoder Attention). Με παρόμοιο τρόπο, όπως είναι αυτός που εμφανίζεται στα Seq2Seq μοντέλα [117], το υποσύστημα αυτό είναι υπεύθυνο να εντοπίζει και να εστιάζει την προσοχή του σε συγκεκριμένα στοιχεία της εισόδου που έχει ήδη κωδικοποιηθεί από τον κωδικοποιητή.

Συνεχίζοντας ως προς την υλοποίηση του συστήματος, η είσοδος εισάγεται στον πρώτο κωδικοποιητή της σωρού 6 κωδικοποιητών που περιγράφηκε προηγουμένως. Όπως έχει αναλυθεί και σε προηγούμενες ενότητες το σύστημα χρειάζεται να λάβει τις λέξεις μέσω κάποιου διανύσματος λέξεων. Κάθε λέξη, λοιπόν, μετατρέπεται στην κατάλληλη αναπαράσταση και διασχίζει ένα συγκεκριμένο μονοπάτι για τη λέξη αυτή στο δίκτυο. Το Feed Forward Neural Network δεν συγκρατεί συσχετίσεις μεταξύ των μονοπατιών αυτών και συνεπώς των λέξεων.

Αυτό υποδηλώνει οτι οι ροές των μονοπατιών στο σύστημα αυτό μπορούν να παραλληλοποιηθούν. Το σύστημα το οποίο είναι υπεύθυνο για τις συσχετίσεις μεταξύ των λέξεων της πρότασης είναι το σύστημα Self-Attention.

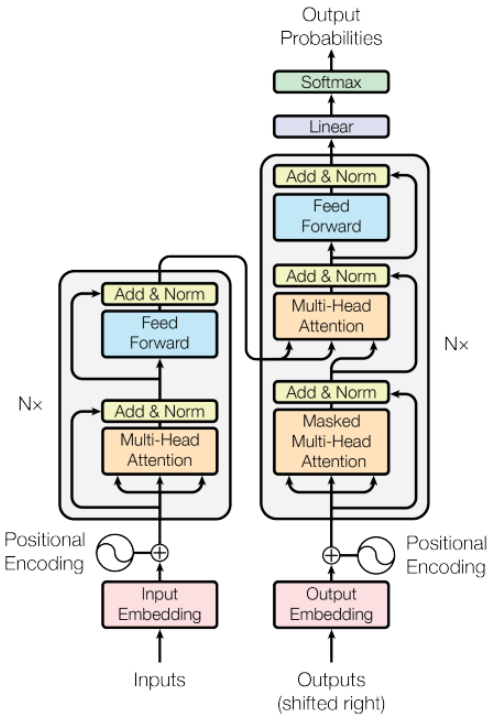
Ο υπολογισμός των εξόδων του Self-Attention συστήματος γίνεται ουσιαστικά σε 2 βασικά βήματα. Το πρώτο βήμα αφορά τη δημιουργία τριών (3) διανυσμάτων από τα διανύσματα λέξεων της πρότασης. Συγκεκριμένα, χρειάζεται να δημιουργηθεί το Query, Key και Value διάνυσμα. Αυτά τα διανύσματα, μεγέθους 64, πολύ μικρότερου δηλαδή από τις διαστάσεις (512) των κρυφών διανυσμάτων, ουσιαστικά προκύπτουν από τον πολλαπλασιασμό των διανυσμάτων εισόδου με συγκεκριμένους πίνακες, οι παράμετροι των οποίων βελτιστοποιούνται κατά τη διαδικασία της εκπαίδευσης. Φυσικά, το μέγεθος αποτελεί αρχιτεκτονική επιλογή αλλά εκτιμάται οτι σε αυτό το μέγεθος είναι δυνατό να υπάρξει μια σταθερή αναπαράσταση ικανή να συγκεντρώσει όλη την πληροφορία μεταξύ των εξαρτήσεων των λέξεων της πρότασης. Ακολουθεί και ένα παράδειγμα, όπως αναλύεται στο [2], προκειμένου να γίνει πλήρως σαφής η λειτουργία του μηχανισμού Self-Attention που αποτελεί ιδιαίτερα σπουδαία ιδέα και στις μετέπειτα εξελίξεις των μοντέλων.

Έστω οτι ως είσοδο εισάγουμε την πρόταση "thinking machines". Αυτή η πρόταση απαρτίζεται από τις λέξεις "thinking" και "machines". Κάθε μία από τις λέξεις αυτές αντιστοιχεί σε ένα διάνυσμα λέξης το οποίο συμβολίζουμε ως x_1 και x_2 αντίστοιχα. Για κάθε λέξη δημιουργούμε τα τρία διανύσματα που προαναφέρθηκαν, q_1, k_1, v_1 και q_2, k_2, v_2, με τη χρήση των πινάκων W^Q, W^K και W^V, όπου τα Q, K και V αντιστοιχούν στις λέξεις Query, Key και Value αντίστοιχα. Για κάθε λέξη, έστω οτι τώρα εστιάζουμε στην λέξη "thinking", πρέπει να υπολογίσουμε μια τιμή score, η οποία υποδηλώνει πόσο πολύ θα πρέπει το σύστημα να δώσει προσοχή και σε άλλες λέξεις της πρότασης καθώς κωδικοποιούμε την είσοδο. Συγκεκριμένα, επειδή έχουμε δύο λέξεις θα προκύψουν και δύο τιμές: $Score_{thinking-thinking} = q_1 * k_1$ και $Score_{thinking-machines} = q_1 * k_2$. Στη συνέχεια διαιρούνται οι τιμές αυτές με $\sqrt{d^k}$, όπου το d^k αντιστοιχεί στις διαστάσεις του διανύσματος k_i. Η διαίρεση αυτή αποσκοπεί σε πιο σταθερές παραγώγους και φυσικά η επιλογή του παρονομαστή αποτελεί επιλογή του ερευνητή. Το αποτέλεσμα της διαίρεσης, διέρχεται από μια Softmax όπου γίνεται κανονικοποίηση και τα αποτελέσματα είναι θετικοί αριθμοί που αθροίζουν στην μονάδα. Έτσι, για παράδειγμα αν το αποτέλεσμα των πράξεων στην λέξη "thinking" είναι 0.88 για τη λέξη "thinking" και 0.12 για τη λέξη "machines", αυτό σημαίνει οτι στη θέση αυτή η λέξη "thinking" παίζει το σπουδαιότερο ρόλο κατά 88% και η λέξη machines συνεισφέρει κατά 12%. Είναι προφανές οτι μετά την εκμάθηση το μεγαλύτερο ποσοστό softmax θα αντιστοιχεί στην ίδια λέξη που μελετάμε αλλά μπορούμε να έχουμε εικόνα και σχετικά με τα ποσοστά των άλλων λέξεων της πρότασης σε σχέση με την υπό μελέτη λέξη. Τέλος, πολλαπλασιάζοντας τα ποσοστά αυτά με τις τιμές v_1 και v_2 αντίστοιχα, προκύπτουν οι έξοδοι z_1 και z_2 που αποτελούν τις εξόδους του συστήματος self-attention μόνο για την λέξη "thinking".

Συνήθης πρακτική είναι και η εισαγωγή multi-headed attention. Συγκεκριμένα, οι Transformers που εισάγονται στην εργασία [125] έχουν 8 attention heads που ουσιαστικά αυτό που σημαίνει είναι οτι προκύπτουν 8 διαφορετικά σετ πινάκων W^Q, W^K και W^V για κάθε Self-Attention υποσύστημα κάθε κωδικοποιητή και κάθε αποκωδικοποιητή. Οι παράμετροι προς βελτιστοποίηση, λοιπόν, αυξάνονται αισθητά αλλά αυτή η τεχνική αποσκοπεί στη βελτίωση της απόδοσης αφού το μοντέλο μπορεί να εστιάσει καλύτερα σε διαφορετικά σημεία της πρότασης και τα διαφορετικά heads μπορεί να προσδίδουν και διαφορετική μορφή προσοχής που θα εστιάζει το μοντέλο. Μπορούν, δηλαδή, να προκύψουν πολλαπλοί υποχώροι αναπαράστασης (representation subspaces). Το Feed Forward Neural Network αναμένει διαφορετικό μέγεθος εισόδου από αυτό που προκύπτει από την εφαρμογή των υπολογισμών που αναφερ-

θήκαμε. Έτσι, αφού συνδέσουμε όλα τα αποτελέσματα σε έναν πίνακα, πολλαπλασιάζουμε τον πίνακα αυτόν με έναν άλλο πίνακα W^O, οι παράμετροι του οποίου βελτιστοποιούνται κατά τη διαδικασία της εκμάθησης. Τέλος, το αποτέλεσμα της πράξης αυτής διέρχεται στο Feed Forward Neural Network.

Παρατηρώντας τις υλοποιήσεις αυτές, είναι εύλογο να συμπεράνει κανείς πως οι Transformers δεν λαμβάνουν υπόψη τους την σειριακή ακολουθία που πραγματικά εμφανίζεται σε μια πρόταση. Για αυτό τον λόγο, χρειάζεται να εισαχθούν και κάποια διανύσματα τα οποία θα προσδίδουν στο σύστημα την αίσθηση της "σειράς" που υπάρχει στην πρόταση. Τα διανύσματα αυτά καλούνται positional embeddings. Η δημιουργία των positional embeddings βασίζεται σε κάποιο συγκεκριμένο πρότυπο (pattern), το οποίο καλείται να μάθει το σύστημα κατά τη διαδικασία της εκπαίδευσης και αυτό το πρότυπο ουσιαστικά υποδηλώνει την "σειρά" τον λέξεων. Αυτά τα embeddings προστίθενται με τα αντίστοιχα διανύσματα λέξεων και ουσιαστικά αυτό που αναμένουμε από την πρόσθεση αυτή είναι εν τέλει να δίνουμε στο σύστημα τα διανύσματα λέξεων με χρονικά χαρακτηριστικά.



Σχήμα 4.8: Η ειδική δομή ενός Transformer [125]

Για λόγους απλότητας παραλείφθηκε και ένα σύστημα το οποίο είναι εξαιρετικά σημαντικό και προτείνεται στη δημοσίευση των transformers [125] και αυτό το σύστημα αφορά το "Add and Norm" σύστημα (True Residuals), όπως παρατηρείται στην Εικόνα 4.8. Επιπλέον, η έξοδος του τελευταίου Κωδικοποιητή μεταφράζεται σε ένα σύνολο από διανύσματα προσοχής (attention vectors): K και V. Η πολύ σημαντική διαφορά του Self-Attention συστήματος στην οποία αξίζει να δοθεί έμφαση για την ακόλουθη μελέτη συγκεκριμένων Transformers είναι οτι το Self-Attention σύστημα του Αποκωδικοποιητή, σε αντίθεση με αυτό του Κωδικοποιητή, δεν μπορεί να λάβει πληροφορία για τις λέξεις όλης της πρότασης και συγκεκριμένα για τις λέξεις που έπονται από την στιγμή που ο αποκωδικοποιητής παράγει τις λέξεις βήμα-βήμα, μία λέξη τη φορά.

4.7.1 Ορισμός Self-Attention

Ιδιαίτερη έμφαση αξίζει να δοθεί στο κομμάτι του Self-Attention, το οποίο απαρτίζεται από τρία διαφορετικά είδη διανυσμάτων Query (Q), Key (K) και Value (V) Vectors και περιγράφεται όπως έχουμε αναφέρει από την εξίσωση

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{4.29}$$

όπου το $\frac{1}{\sqrt{d_k}}$ αποτελεί ένα παράγοντα κλιμάκωσης.

4.7.2 Πλεονεκτήματα Μεταφοράς Μάθησης (Transfer Learning)

Σε αυτό το σημείο κρίνεται σκόπιμο να συμπυκνωθούν τα πλεονεκτήματα που προσφέρουν οι Τεχνικές Μεταφοράς Μάθησης. Το πρώτο σημαντικό πλεονέκτημα σχετίζεται με την ταχύτερη ανάπτυξη και υλοποίηση εφαρμογών που χρησιμοποιούν κάποιο προ-εκπαιδευμένο μοντέλο. Τα προ-εκπαιδευμένα μοντέλα είναι από μόνα τους ήδη σε θέση να ανιχνεύσουν γλωσσικά χαρακτηριστικά (features) λόγω της εκμάθησής τους και ως αποτέλεσμα ο μηχανικός ή προγραμματιστής το μόνο που πρέπει να κάνει είναι να τα συνδυάσει με άλλα μικρά δίκτυα και να προσαρμόσει το σύστημα έτσι ώστε να βελτιστοποιηθεί σε κάποιο συγκεκριμένο πρόβλημα. Λόγω της υπολογιστικά, χρονικά και συνεπώς οικονομικά δαπανηρής διαδικασίας εκμάθησης πάνω σε τεράστιο όγκο κειμένων, το σύστημα που προκύπτει απαιτεί σαφώς λιγότερα δεδομένα και σημαντικά λιγότερο χρόνο και υπολογιστική ισχύ. Το σημαντικό, όμως, πλεονέκτημα που καθιστά δημοφιλείς τις τεχνικές αυτές είναι τα εξαιρετικά καλύτερα αποτελέσματα που προκύπτουν με μικρές τροποποιήσεις ανάλογα με υπό μελέτη πρόβλημα. Επιπλέον, αξίζει να σημειωθεί πως αν για διαφορετικά προβλήματα απαιτείται η αλλαγή ενός μικρού υποδικτύου που ενσωματώνεται πάνω στο προεκπαιδευμένο μοντέλο, αυτό καθιστά αυτές τις τεχνικές αρχιτεκτονικά "διαυγείς" και συνεπώς πιο κατανοητές, υλοποιήσιμες και επεκτάσιμες. Εν κατακλείδι, αυτό που προτάθηκε είναι η ύπαρξη ενός μοντέλου το οποίο αντιλαμβάνεται βασικά χαρακτηριστικά της γλώσσας ανεξαρτήτως προβλήματος και έτσι δεν χρειάζεται κάθε σύστημα που θέλουμε να επιλύσει κάποιο πρόβλημα να δαπανά ώρες εκμάθησης για να μάθει τα βασικά γλωσσικά χαρακτηριστικά των κειμένων προτού βελτιστοποιηθεί σε κάποιο συγκεκριμένο πρόβλημα που έχουμε υποδείξει. Η στροφή αυτή ταυτίζεται και με αυτή που έχει γίνει τα τελευταία χρόνια στο πεδίο της μηχανικής όρασης όπου χρησιμοποιούνται μεγάλα προ-εκπαιδευμένα μοντέλα που έχουν ήδη μάθει να διαχωρίζουν τα βασικά "συστατικά" μιας εικόνας, όπως είναι οι ευθείες ή οι γωνίες [104].

4.7.3 Ιστορική Αναδρομή

Προτού μεταβούμε στον τρόπο υλοποίησης και χρήσης του BERT στο πρόβλημα ταξινόμησης κειμένου ενδείκνυται μια σύντομη ανακεφαλαίωση που αφορά τα υπάρχοντα μοντέλα δίνοντας έμφαση στις διανυσματικές αναπαραστάσεις που χρησιμοποιούνται. Όπως έχει αναλυθεί σε προηγούμενη ενότητα, υπήρχε η τάση να γίνεται ευρεία χρήση του Word2Vec [77] και του Glove [93]. Το αποτέλεσμα αυτών των τεχνικών για κάθε λέξη αποδίδει ένα συγκεκριμένου μήκους διάνυσμα που περιέχει συγκεκριμένες τιμές ανεξαρτήτως περιεχομένου (context) της πρότασης που βρίσκεται η υπό ανάλυση λέξη. Παρόλα αυτά, μια λέξη μπορεί να φέρει διαφορετική σημασία ανάλογα με το περιεχόμενο που βρίσκεται, δηλαδή ανάλογα με τις υπόλοιπες λέξεις με τις οποίες εμφανίζεται σε μια πρόταση. Έχοντας ως γνώμονα αυτή την πεποίθηση,

επιδιώχθηκε να προσδοθεί μια διανυσματική αναπαράσταση σε μια λέξη ανάλογα με το περιβάλλον στο οποίο συναντάται (contextualized word embeddings) [94] [74] [95] . Έτσι, το ELMo αντί να προσδίδει μια σταθερή διανυσματική αναπαράσταση σε κάθε λέξη, βασιζόμενο στο πρόβλημα της Γλωσσικής Μοντελοποίησης και στη χρήση LSTMs, διατρέχει ολόκληρη την πρόταση και από τις δύο κατευθύνσεις, απο δεξιά προς τα αριστερά και αντιστρόφως, και η έξοδος του μοντέλου αποδίδεται ως διανυσματική αναπαράσταση στην οποία εγκολπώνεται και όλη η πληροφορία που προκύπτει από το περιβάλλον, την πρόταση δηλαδή στην οποία αυτή η λέξη εντοπίζεται. Έπειτα, το ULMFiT [40], εκτός από ένα γλωσσικό μοντέλο, εισήγαγε και μια διαδικασία για το πώς θα μπορούσε το μοντέλο αυτό αποδοτικά να βελτιστοποιηθεί μετέπειτα σε κάποιο συγκεκριμένο πρόβλημα.

Στη συνέχεια, δημοσιεύθηκαν οι Transformers που μεταξύ άλλων πλεονεκτημάτων χειρίζονται καλύτερα τις μεγάλες χρονικές συσχετίσεις μεταξύ των λέξεων σε σύγκριση με τα LSTMs προτείνοντας μια ιδανική αρχιτεκτονική για την επίλυση προβλημάτων Αυτόματης Μετάφρασης (Machine Translation) αφού αποτελούνται από ένα δίπολο Κωδικοποιητή (Encoder) και Αποκωδικοποιητή (Decoder). Έπειτα, προέκυψε η ιδέα να μην χρησιμοποιηθεί και ο Κωδικοποιητής και ο Αποκωδικοποιητής στην ίδια αρχιτεκτονική. Σε αυτή την φιλοσοφία, η OpenAI αποφάσισε να χρησιμοποιήσει ένα μοντέλο το οποίο είναι βασισμένο σε πολλά (12) υποσυστήματα Αποκωδικοποιητών (Decoders) τα οποία τοποθετούνται σειριακά το ένα πάνω στο άλλο και επιλύουν προβλήματα γλωσσικής μοντελοποίησης αλλά και άλλα προβλήματα ανάλογα με τον χειρισμό της εξόδου από το μοντέλο.

Το μοντέλο αυτό της OpenAI, είναι ένα forward γλωσσικό μοντέλο που σημαίνει οτι δεν λαμβάνει συσχετίσεις μεταξύ άλλων λέξεων στην πρόταση και από τις δύο κατευθύνσεις (και συγκεκριμένα από δεξιά προς τα αριστερά) όπως επιτυγχανόταν με τα Bi-LSTMs του ELMo. Για τον λόγο αυτό και χρησιμοποιώντας αυτή τη φορά μόνο Κωδικοποιητές (Encoders), η Google εισήγαγε το BERT. Παρόλα αυτά, όπως έχει αναλυθεί σε προηγούμενη ενότητα η γλωσσική μοντελοποίηση δεν λαμβάνει πληροφορία από λέξεις που εμφανίζονται μετά από την λέξη που μελετάμε. Για τον λόγο αυτό, η εκμάθηση πραγματοποιήθηκε με τεχνικές όπου "έκρυβαν" κάποιες λέξεις (masked language models). Επίσης, κατά την εκμάθηση του μοντέλου αυτό σε δεύτερο στάδιο έπρεπε να προβλέψει, δεδομένων δύο προτάσεων, την πιθανότητα η πρόταση B να έπεται της πρότασης A.

4.7.4 BERT

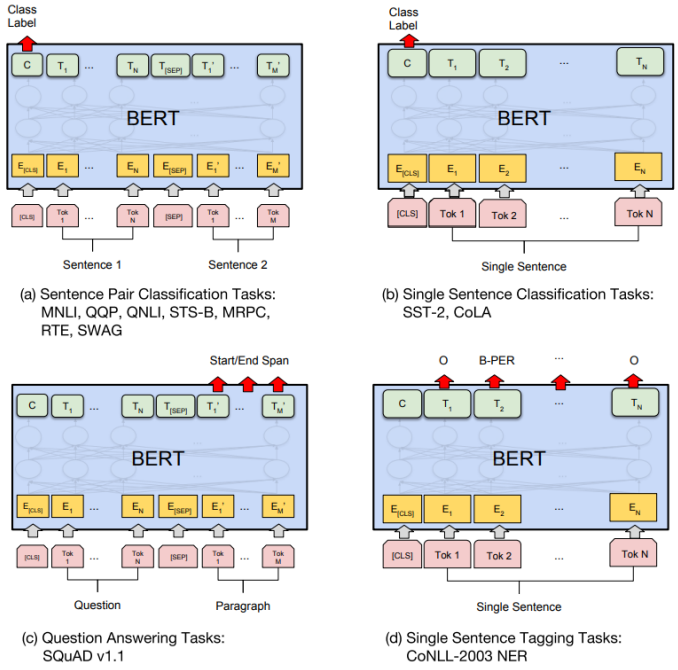
Το 2018, αποτέλεσε μια σημαντική χρονιά για την κοινότητα της Φυσικής Επεξεργασίας Γλώσσας (NLP) από τη στιγμή που δημοσιεύθηκαν τα μοντέλα ELMo της Allen AI [95], Open-GPT της OpenAI [98] και BERT της Google [23]. Έκτοτε άρχισε να δίνεται ιδιαίτερη έμφαση στις Τεχνικές Μεταφοράς Μάθησης και η χρήση τους να αποκτά ευρύ φάσμα εφαρμογών διότι με ελάχιστο χρόνο, κόπο, δεδομένα και υπολογιστική ισχύ, οι ερευνητές του κλάδου μπόρεσαν να επιτύχουν εμφανώς καλύτερα αποτελέσματα από τα υπάρχοντα σε πληθώρα εφαρμογών. Η μόνη διαφοροποίηση έγκειται στο γεγονός οτι πρέπει να χρησιμοποιηθεί ένα προ-εκπαιδευμένο μοντέλο, να προσαρμοστεί και να βελτιστοποιηθεί (finetuning), σε πολλές περιπτώσεις μαζί με ένα μικρό υποδίκτυο, για την επίλυση μιας συγκεκριμένη εργασίας (task).

Το BERT (Bidirectional Encoder Representations from Transformers) [23] είναι βασισμένο σε αρκετές δεσπόζουσες ιδέες της εποχής όπως είναι αυτές της Ημι-επιβλεπόμενης Ακολουθιακής Μάθησης (Semi-Supervided Sequence Learning) [18], του ELMo [95], του ULMFiT [40], του OpenAI Transformer [98] και των Transformers [125]. Συγκεκριμένα, η εκμάθηση

του BERT υλοποιήθηκε με την ιδέα της ημι-επιβλεπόμενης ακολουθιακής μάθησης πάνω σε ένα τεράστιο πλήθος κειμένων που προέρχονται από βιβλία και την Wikipedia, μεταξύ και άλλων πηγών. Το μοντέλο κατά τη διαδικασία της εκμάθησης εστιάζει σε ένα συγκεκριμένο πρόβλημα της γλωσσικής μοντελοποίησης που ονομάζεται masked word prediction. Με αυτό τον τρόπο, μαθαίνει να ανιχνεύει γλωσσικά πρότυπα και είναι ικανό να επεξεργαστεί εν συνεχεία γλωσσικά κείμενα.

Εκτός από την εξαγωγή υψηλής ποιότητας γλωσσικών χαρακτηριστικών από ένα κείμενο, το BERT μαζί με προσθήκες μικρών δικτύων πάνω σε αυτό, μπορεί να χρησιμοποιηθεί για την επίλυση πληθώρας προβλημάτων ταξινόμησης (classification), αναγνώρισης οντοτήτων (entity recognition) και ερωταπαντήσεων (question answering). Στη δική μας μελέτη, το προ-εκπαιδευμένο μοντέλο θα χρησιμοποιηθεί για το πρόβλημα κατηγοριοποίησης κειμένου όπως θα αναλυθεί στη συνέχεια. Αυτό το πρόβλημα θα μπορούσε να λυθεί και με υπάρχοντα μοντέλα όπως είναι για παράδειγμα τα CNNs (Convolutional Neural Networks) ή τα Bi-LSTMs (Bidirectional Long Short Term Memory).

Πέρα από τα προβλήματα που έχουν αναφερθεί οτι μπορεί να επιλύσει το BERT, όπως απεικονίζεται βέβαια και από το παρακάτω σχήμα 4.9, το BERT, όπως και το ELMo, μπορεί να χρησιμοποιηθεί και για την παραγωγή contextualized word embeddings. Σε αυτό το σημείο να σημειωθεί οτι υπάρχουν δύο εκδοχές του BERT. Η πρώτη είναι το κλασικό μοντέλο (12 Encoder - 768 hidden size) που έχει μέγεθος συγκρίσιμο με αυτό του OpenAI Transformer και η δεύτερη είναι ένα πολύ μεγαλύτερο μοντέλο (16 Encoder - 1024 hidden size) το οποίο και οδηγεί σε εμφανώς καλύτερα αποτελέσματα όπως έχει περιγραφεί προηγουμένως.



Σχήμα 4.9: Παραδείγματα Εφαρμογών BERT σε διάφορα Προβλήματα [23]

Όσον αφορά την εφαρμογή του BERT στο πρόβλημα ταξινόμησης κειμένου, χρησιμοποιήθηκε η βιβλιοθήκη που δημόσια παραχωρείται από την HuggingFace σε PyTorch [42]. Συγκεκριμένα, μια πρώτη υλοποίηση κάνει χρήση του μοντέλου BertForSequenceClassification [44], όπου αποτελείται από το κλασικό μοντέλο BERT με την διαφορά οτι στην έξοδό του έχει προστεθεί ένα γραμμικό στρώμα νευρώνων για την κατηγοριοποίηση. Κατά την εκμάθηση όλο

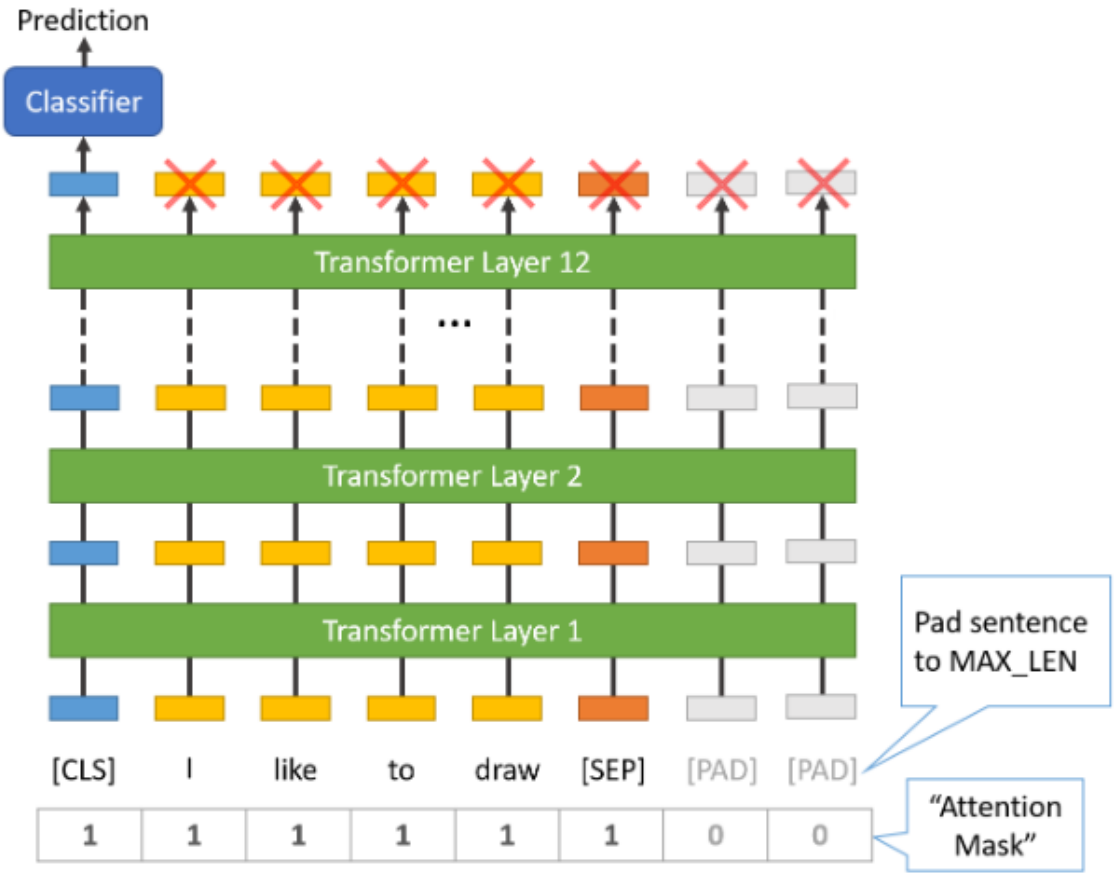
το σύστημα βελτιστοποιείται. Να σημειωθεί οτι επειδή τα μοντέλα αυτά είναι αρκετά μεγάλα, συνήθως μελετώνται τεχνικές παγώματος-ξεπαγώματος (freezing-unfreezing) συγκεκριμένων παραμέτρων, προσαρμογή του learning decay ή ακόμα και βελτιστοποίηση ενός υποσυνόλου των παραμέτρων του συστήματος.

Η πρώτη διαδικασία που πρέπει να πραγματοποιηθεί για την εισαγωγή ενός κειμένου στο BERT είναι η μετατροπή του κειμένου αυτού σε μια μορφή που το μοντέλο αντιλαμβάνεται και ταυτίζεται με αυτή που έγινε κατά τη διαδικασία της εκμάθησής του. Εκτός από τη τμηματοποίηση της πρότασης σε συγκεκριμένη μορφή (WordPieces [139]) θα πρέπει να προστεθούν ειδικά tokens και σε κάθε στοιχείο να δοθούν τα κατάλληλα αναγνωριστικά (ids) που αποτελούν και τους δείκτες του λεξικού. Στο συγκεκριμένο παράδειγμα της κατηγοριοποίησης κειμένου (text classification) απαιτείται η πρόταση να αρχίζει με το token $[CLS]$ που υποδηλώνει την ταξινόμηση (Classification) και να τελειώνει με το token $[SEP]$. Όλες οι προτάσεις θα πρέπει να γίνουν padded ή truncated σε ένα συγκεκριμένο προκαθορισμένο μήκος, το οποίο επηρεάζει και την υπολογιστική και χρονική απόδοση του συστήματος και το μέγιστο μήκος ακολουθίας που μπορεί να δεχθεί ως είσοδο το σύστημα είναι 512 tokens. Επιπλέον, το σύστημα αναμένει και μια μάσκα προσοχής (attention mask) που υποδηλώνει ποια στοιχεία της εισόδου αντιστοιχούν σε ουσιαστική πληροφορία και ποια θα πρέπει να αγνοήσει το σύστημα λόγω του padding. Σε κάθε μία από τις 512 εξόδους του κλασικού συστήματος προκύπτει ένα διάνυσμα 768 τιμών. Το πρώτο διάνυσμα που αντιστοιχεί και στο πρώτο token, δηλαδή στο $[CLS]$ είναι και αυτό που χρησιμοποιείται συνήθως για την ταξινόμηση.

Έστω οτι έχουμε το παραπάνω παράδειγμα και το μέγιστο μήκος ακολουθίας που θα δεχόταν το μοντέλο είναι 8. Η πρόταση "I like to draw" αποτελείται από 4 λέξεις. Στην αρχή και στο τέλος της πρότασης θα πρέπει να προστεθούν τα ειδικά tokens $[CLS]$ και $[SEP]$ αντίστοιχα και συνεπώς θα αποτελείται η πρόταση από 6 στοιχεία. Έστω, οτι ο BertTokenizer [43] κατά την τμηματοποίηση θα σπάσει την πρόταση σε 6 στοιχεία. Επειδή το μέγιστο μήκος ακολουθίας είναι 8 και κάθε είσοδος θα πρέπει αναγκαστικά να φέρει αυτό το μήκος χρειάζεται να προστεθούν $8 - 6 = 2$ επιπλέον $[PAD]$ tokens. Αυτά τα tokens που προστέθηκαν δεν θα πρέπει να ληφθούν υπόψη από το σύστημα. Για τον λόγο αυτό χρησιμοποιείται μια Μάσκα Προσοχής (Attention Mask) η οποία αποτελείται από 1 στις θέσεις των στοιχείων που θα πρέπει να δοθεί προσοχή και σε 0 στις θέσεις των στοιχείων όπου το σύστημα θα πρέπει να αγνοήσει. Συνεπώς, η μάσκα μήκους 8 θα αποτελείται από 6 άσσους στις πρώτες 6 θέσεις και από 2 μηδενικά στις υπολειπόμενες. Αξιοσημείωτο να αναφερθεί είναι οτι οι ερευνητές που δημοσίευσαν το μοντέλο αυτό δίνουν και κάποιες προτάσεις σχετικά με τις παραμέτρους προς βελτιστοποίηση.

4.7.5 GPT-2

Το παρόν υποκεφάλαιο είναι βασισμένο στη δομή του [3]. Το OpenAI GPT-2 [99] δεν είναι καμία καινούρια αρχιτεκτονική. Βασίζεται στους transformers που αποτελούνται μόνο από ένα σύνολο αποκωδικοποιητών. Η διαφορά είναι οτι το μέγεθος του μοντέλου είναι αρκετά μεγάλο και η εκμάθηση έγινε σε εξίσου αρκετά μεγάλο σετ δεδομένων. Συγκεκριμένα, τα δεδομένα αυτά, WebText, έχουν μέγεθος 40GB και το η μικρότερη εκδοχή του μοντέλου απαιτεί χωρητικότητα 500MBs για την αποθήκευση των 117 εκατομμυρίων παραμέτρων του. Να σημειωθεί πως η μεγαλύτερη έκδοση του μοντέλου έχει 1,542 εκατομμύρια παραμέτρους. Η διαφορά του με την αρχιτεκτονική του BERT είναι οτι βασίζεται μόνο σε decoders αντί για encoders. Ανάλογα με το μέγεθος της εκάστοτε έκδοσης του GPT-2 έχουμε και διαφο-



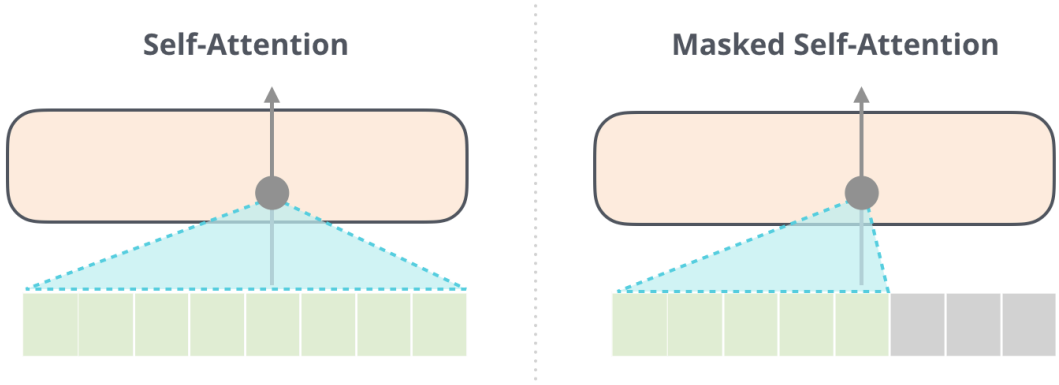
Σχήμα 4.10: Πρακτική Εφαρμογή BERT στο Πρόβλημα της Ταξινόμησης [75]

ρετικό αριθμό από decoders. Συγκεκριμένα, η μικρή, μεσαία, μεγάλη και πολύ μεγάλη έκδοση αποτελείται από 12, 24, 36 και 48 decoders.

Το GPT-2 λειτουργεί ως ένα παραδοσιακό Γλωσσικό Μοντέλο όπου δεδομένης μια ακολουθίας λέξεων προσπαθεί να προβλέψει την επόμενη λέξη. Η έξοδος που παράγεται εισάγεται μαζί με τις προηγούμενες λέξεις στο μοντέλο προκειμένου να παραχθεί μια νέα έξοδος. Η διαδικασία αυτή συνεχίζεται έως ότου παραχθεί ένα ειδικό token που υποδηλώνει τον τερματισμό της πρότασης. Αυτό το χαρακτηριστικό των μοντέλων που χρησιμοποιούν την έξοδο ως είσοδο για να την παραγωγή μιας νέας εξόδου, το έχουμε συναντήσει στα Ακολουθιακά Νευρωνικά Δίκτυα (RNNs). Τα μοντέλα που χρησιμοποιούν αυτή την ιδιότητα συνήθως καλούνται "auto-regressive" μοντέλα. Για παράδειγμα, μοντέλα όπως το BERT δεν ανήκουν σε αυτή την κατηγορία. Γνωστοί transformers που έχουν την ίδια φιλοσοφία με το GPT-2 είναι οι TransformersXL [19] και XLNet [141].

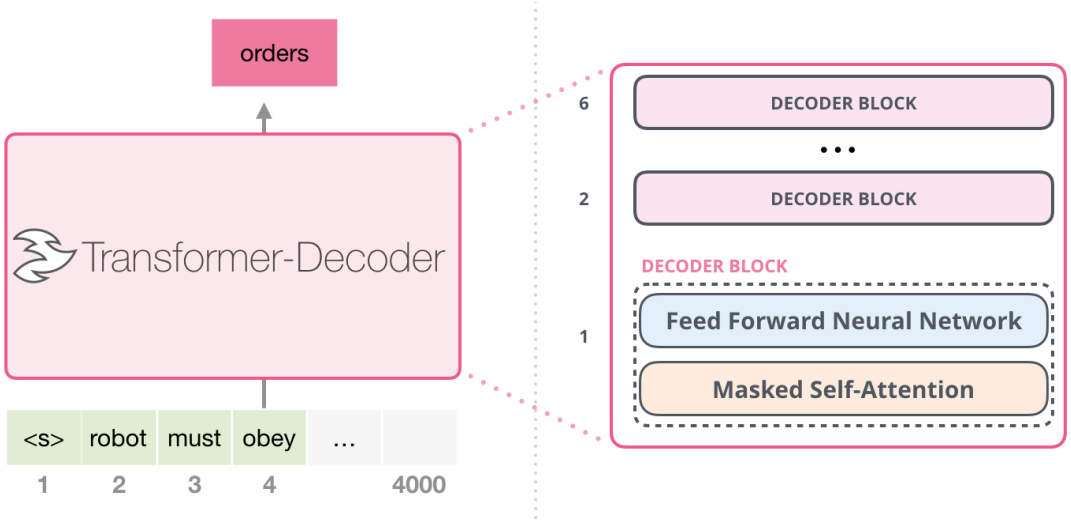
Εκτός από τις ουσιαστικές δομικές διαφορές μεταξύ του BERT και του GPT-2, συναντάμε και άλλες που αφορούν το masking του self-attention. Συγκεκριμένα, το GPT-2 δεν χρησιμοποιεί απλά ένα [MASK] για τις επόμενες λέξεις της ακολουθίας που δεν θα πρέπει να λάβει υπόψη του το μοντέλο τη δεδομένη χρονική στιγμή. Αντιθέτως, αλληλεπιδρά με το self-attention layer με τον τρόπο που απεικονίζεται στην εικόνα 4.11.

Σύμφωνα, λοιπόν, με το [71], οι συγγραφείς αποφάσισαν να εξαλείψουν τα encoder blocks και



Σχήμα 4.11: BEPT Self-Attention (Αριστερά) και GPT-2 Masked Self-Attention (Δεξιά) [3]

να χρησιμοποιήσουν μόνο decoder blocks. Στο πρώιμο στάδιο, η ακολουθία εισέρχεται στο μοντέλο και περνάει μέσα από 6 διαδοχικά decoder blocks. Κάθε decoder block αποτελείται από ένα Masked Self-Attention και ένα Feed Forward Neural Network όπως απεικονίζεται στην εικόνα 4.12. Παρόμοια αρχιτεκτονική προτάθηκε και υιοθετήθηκε και στο [1], με την διαφορά οτι το Γλωσσικό Μοντέλο καλείται να παράγει ένα token τη φορά.

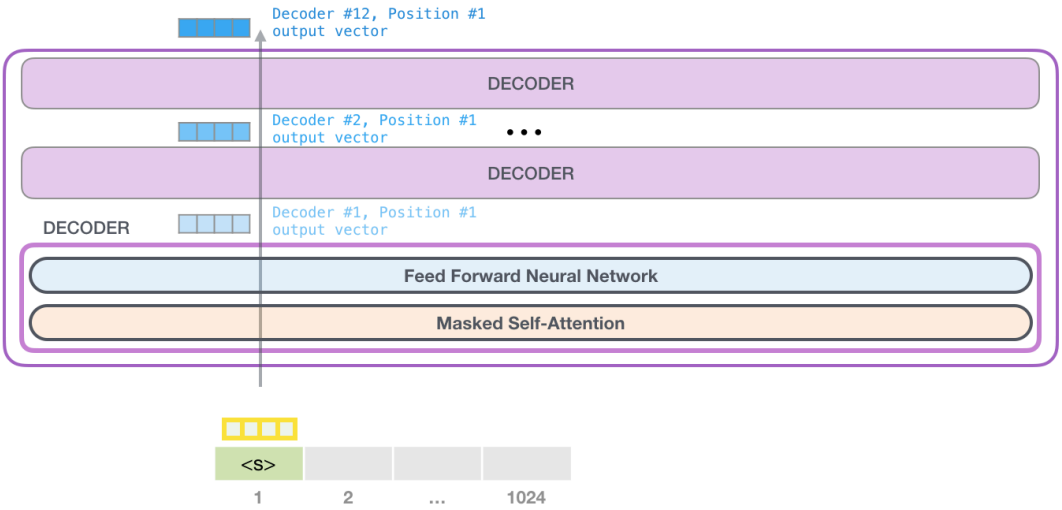


Σχήμα 4.12: Αρχιτεκτονική GPT-2 [3]

Σε αυτό το σημείο να τονίσουμε οτι το GPT-2 μπορεί να δεχτεί ως είσοδο 1024 tokens. Κάθε token διαπερνά το μοντέλο από ένα δικό του μοναδικό μονοπάτι όπως απεικονίζεται στο Σχήμα 4.13. Αυτό είναι και ένα εγγενές χαρακτηριστικό των Transformers που τους έχει κάνει αρκετά δημοφιλείς διότι έτσι μπορούμε να παραλληλοποιήσουμε τους υπολογισμούς. Στην περίπτωση που θελήσει κανείς να χρησιμοποιήσει ένα ήδη εκπαιδευμένο GPT-2 μοντέλο, μπορεί είτε να το αφήσει μόνο του να παράγει προτάσεις (generating unconditional samples) είτε να χρησιμοποιήσει κάποιο prompt και να του επιτρέψει μετέπειτα να συμπληρώσει την ακολουθία (generating interactive unconditional samples). Το αρχικό token που χρησιμοποιεί το μοντέλο πριν παράξει οποιαδήποτε ακολουθία είναι το "$< |endoftext| >$". Όταν ένα token, λοιπόν, διαπερνά το μοντέλο δημιουργεί κάποιο διάνυσμα από το οποίο επιλέγεται η λέξη που θα βγάλει το μοντέλο. Μια λύση είναι να χρησιμοποιηθεί η πιο πιθανή λέξη από το λεξικό του GPT-2. Παρόλα αυτά, δίνεται η δυνατότητα να παραχθούν λέξεις

πιο σπάνιες (χρήση beam search ή τεχνικών sampling όπως περιγράφονται και υλοποιούνται στην ενότητα 7.3.4). Όταν επιλεγεί η κατάλληλη λέξη εισέρχεται ως είσοδος στο μοντέλο. Το δεύτερο, πλέον, μονοπάτι είναι το μόνο ενεργό στον συγκεκριμένο υπολογισμό. Επηρεάζεται από την παλιότερη είσοδο. Όμως, δεν επαναξιολογεί το πρώτο token δεδομένου του δεύτερου! Στη συνέχεια, περιγράφουμε τα υπομέρη του μοντέλου με μεγαλύτερη λεπτομέρεια.

Όπως έχουμε αναλύσει στο Κεφάλαιο 3, κάθε μοντέλο δέχεται ως είσοδο διανύσματα λέξεων. Ανάλογα με το μέγεθος της εκάστοτε έκδοσης του GPT-2 μοντέλου, έχουμε και διαφορετικά μεγέθη διανυσμάτων λέξεων. Συγκεκριμένα, το μικρό, μεσαίο, μεγάλο και πολύ μεγάλο μοντέλο δέχεται διανύσματα λέξεων μήκους 768, 1024, 1280 και 1600 αντίστοιχα. Επιπλέον, από το Κεφάλαιο 4.7 γνωρίζουμε οτι οι transformers δεν αποθηκεύουν κάποια χρονική και χωρική συσχέτιση των λέξεων. Για τον λόγο αυτό, δημιουργούνται και οι κατάλληλες χωρικές αναπαραστάσεις (positional embeddings) τα οποία έχουν μέγεθος όσο και το μέγεθος των διανυσμάτων λέξεων και προσδίδουν την αίσθηση οργάνωσης της χρονικής ακολουθίας μεταξύ των εισόδων. Η είσοδος, λοιπόν, του μοντέλου είναι το αποτέλεσμα του αθροίσματος του διανύσματος λέξεων με το το χωρικό διάνυσμα.



Σχήμα 4.13: Λειτουργία μικρού μοντέλου GPT-2 [3]

Κάθε block, δηλαδή κάθε decoder, είναι ομοιόμορφο ως προς την αρχιτεκτονική του, αποτελείται δηλαδή από ένα Masked Self-Attention block και ένα Feed Forward Neural Network. Η διαφορά στα blocks εμφανίζεται στα βάρη αυτών τα οποία είναι προφανώς διαφορετικά. Παρατηρούμε, λοιπόν, πως από τα δύο υπομέρη ενός αποκωδικοποιητή το Masked Self-Attention προσπαθεί να συμπεριλάβει το περιεχόμενο της ακολουθίας μέχρι την χρονική στιγμή που μελετάμε. Η έξοδος που προκύπτει από το GPT-2 πολλαπλασιάζεται εν τέλει με τον Πίνακα Διανυσμάτων Λέξεων και για κάθε λέξη του λεξικού προκύπτει μια τιμή-βαθμολογία. Η επιλογή της κατάλληλης λέξης μπορεί να γίνει με ποικίλους τρόπους. Το απλούστερο σενάριο είναι η επιλογή της λέξης με τη μεγαλύτερη βαθμολογία (top k = 1). Υπάρχουν, όμως, και άλλες τεχνικές όπως αναλύονται στην ενότητα 7.3.4.

4.8 Σύνοψη

Με το Κεφάλαιο αυτό ολοκληρώνεται η εισαγωγή στο θεωρητικό υπόβαθρο της Επεξεργασίας Φυσικής Γλώσσας. Πραγματοποιήθηκε μια περιγραφή τόσο των παραδοσιακών τεχνικών όσο και των πιο σύγχρονων μοντέλων που συναντώνται στις ερευνητικές και βιομηχανικές εφαρμογές. Στο επόμενο κεφάλαιο, πραγματοποιείται μια εισαγωγή στην Ενισχυτική Μάθηση που είναι αναγκαία για τις υλοποιήσεις που παρουσιάζονται στη συνέχεια της διπλωματικής εργασίας και με αυτόν τον τρόπο θα κλείσει το μέρος των Εισαγωγικών Εννοιών.

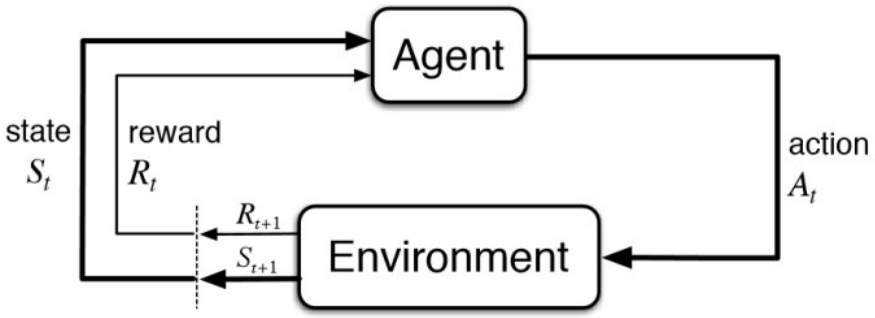
Κεφάλαιο 5

Ενισχυτική Μάθηση (Reinforcement Learning)

Στο Κεφάλαιο αυτό, πραγματοποιείται μια εισαγωγή στην Ενισχυτική Μάθηση (Reinforcement Learning) δίνοντας έμφαση στις ιδέες που είναι χρήσιμες για τις υλοποιήσεις που πλαισιώνουν την διπλωματική εργασία. Η εισαγωγή αυτή είναι βασισμένη στο βιβλίο των Sutton και Barto [118] καθώς επίσης και σε δύο διαδικτυακά μαθήματα από τον David Silver (UCL) [113] και τον Sergey Levine (Berkeley) [62].

Στην Ενότητα 5.1, παρουσιάζεται το πλαίσιο της Ενισχυτικής Μάθησης μαζί με τις Βασικές Ορολογίες. Στη συνέχεια, στην Ενότητα 5.2, μελετώνται οι έννοιες της Επιβράβευσης και της Ανταμοιβής. Στην Ενότητα 5.3, πραγματοποιείται η περιγραφή του προβλήματος μέσω των Μαρκοβιανών Διαδικασιών Λήψης Αποφάσεων και στην Ενότητα 5.4 κατηγοριοποιούνται τα διαφορετικά προβλήματα Ενισχυτικής Μάθησης.

5.1 Περιγραφή Προβλήματος και Ορολογίες



Σχήμα 5.1: Γενικό Πλαίσιο Ενισχυτικής Μάθησης

Το γενικό πλαίσιο της Ενισχυτικής Μάθησης στην απλούστερη μορφή του απεικονίζεται στο Σχήμα 5.1. Αυτό περιγράφεται από έναν **πράκτορα (agent)**, ο οποίος ανάλογα με την **κατάσταση (state)** ή την **παρατήρηση (observation)**, πραγματοποιεί κάποια **ενέργεια (action)** και αλληλεπιδρώντας με το **περιβάλλον (environment)**, τού επιστρέφεται μια

ανταμοιβή (reward) και μια καινούρια κατάσταση ή παρατήρηση. Αν ο πράκτορας λαμβάνει την ολοκληρωμένη κατάσταση του περιβάλλοντος ("είναι") τότε χρησιμοποιείται ο όρος "κατάσταση". Αντιθέτως, αν λαμβάνει μια λειψή ή παραμορφωμένη εικόνα της πραγματικής κατάστασης ("φαίνεσθαι") τότε συνήθως χρησιμοποιείται ο όρος "παρατήρηση". Συνεπώς, ένα περιβάλλον μπορεί να είναι **πλήρως παρατηρήσιμο (fully observable)** ή **μερικώς παρατηρήσιμο (partially observable)**. Στόχος του πράκτορα αλληλεπιδρώντας με το περιβάλλον του είναι η εύρεση μιας κατάλληλης πολιτικής ενεργειών προκειμένου να μεγιστοποιήσει το άθροισμα των ανταμοιβών που θα λάβει.

Σε αυτό το σημείο, αξίζει να σημειωθεί πως με αυτή τη μέθοδο μάθησης γίνεται η υπόθεση οτι όλοι οι στόχοι μπορούν να επιτευχθούν μεγιστοποιώντας κάποια προσδοκώμενη αθροιστική ανταμοιβή. Εξίσου ενδιαφέρον παρουσιάζει το γεγονός οτι στο συγκεκριμένο πλαίσιο, το περιβάλλον αποδίδει κάποια ανταμοιβή στον πράκτορα τη στιγμή που ίσως στην πραγματικότητα το περιβάλλον να μην αποδίδει ανταμοιβές αλλά ανάλογα με τις μεταβολές που προκαλούνται σε αυτό, το αντίτιμο εκτιμάται εμμέσως λόγω της ικανοποίησης του πράκτορα στην νέα πραγματικότητα.

Το σύνολο των παρατηρήσεων, ανταμοιβών και δράσεων του πράκτορα κάθε χρονική στιγμή συνθέτει την έννοια της **ιστορίας (history)**. Ο όρος της κατάστασης επιπλέον χρησιμοποιείται συχνά ως η πληροφορία που συνοψίζει όλη την ιστορία, ικανοποιώντας την Μαρκοβιανή Ιδιότητα, όπως αναλύεται στη συνέχεια. Ο πράκτορας καλείται να μάθει την κατάλληλη και πιο αποδοτική **πολιτική (policy)**, η οποία μπορεί να είναι είτε ντετερμινιστική είτε στοχαστική. Επιπλέον, οι **συναρτήσεις αξιών (value functions)** αποδίδουν κάποια τιμή για την προσδοκώμενη μελλοντική ανταμοιβή που θα λάβει ο πράκτορας. Τέλος, η έννοια του **μοντέλου (model)** σχετίζεται με την πρόβλεψη της επόμενης κατάστασης του περιβάλλοντος. Ακόμα, με τον όρο **επεισόδιο (episode, trial ή trajectory)** αναφερόμαστε στο σύνολο των καταστάσεων, ενεργειών και ανταμοιβών που προκύπτουν μέχρι την τερματική κατάσταση.

5.2 Επιβράβευση (Reward) και Ανταμοιβή (Return)

Η Συνάρτηση Επιβράβευσης (Reward Function) αποτελεί ένα από τα πιο σημαντικά στοιχεία στη μελέτη της Ενισχυτικής Μάθησης. Μολονότι συχνά στη βιβλιογραφία συναντάται με απλούστερες εκφράσεις από τη σχέση 5.1, στην ουσία η Συνάρτηση Επιβράβευσης εκτός από την παροντική ενέργεια (a_t) ενός πράκτορα, εξαρτάται τόσο από την παροντική κατάσταση (s_t) κατά την οποία ενήργησε ο πράκτορας όσο και από την αμέσως επόμενη (s_{t+1}). Η ολοκληρωμένη περιγραφή, λοιπόν, της Συνάρτησης Ανταμοιβής είναι η σχέση 5.1.

$$r_t = R(s_t, a_t, s_{t+1}) \tag{5.1}$$

Παρόλα αυτά, συχνά απλοποιείται στη μορφή $r_t = R(s_t)$ αν εξαρτάται μόνο από την παροντική κατάσταση και στη μορφή $r_t = R(s_t, a_t)$ αν πέρα από την παροντική κατάσταση εξαρτάται και από την ενέργεια του πράκτορα. Με r_t συμβολίζεται η επιβράβευση (reward).

Με τον όρο Ανταμοιβή (Return), γίνεται αναφορά σε μια μακροπρόθεσμη συσσώρευση επιβραβεύσεων. Η Ανταμοιβή αυτή μπορεί είτε να μην περιέχει έναν εκπτωτικό παράγοντα $\gamma \in [0, 1]$ (Εξίσωση 5.2) είτε να περιέχει ένα εκπτωτικό παράγοντα $\gamma \in [0, 1]$ (Εξίσωση 5.3). Στην περίπτωση, που η Ανταμοιβή περιέχει και τον εκπτωτικό παράγοντα γ καλείται Ανταμοιβή με Έκπτωση (Discounted Return). Επιπρόσθετα, το άθροισμα μπορεί να είναι είτε πεπερασμένο

είτε άπειρο. Στην περίπτωση που το άθροισμα είναι πεπερασμένο το άνω όριο αθροίσματος των εξισώσεων 5.2 και 5.3 γίνεται $T \in \mathbb{R}$.

$$R_t = \sum_{k=0}^{\infty} r_{t+k+1} \tag{5.2}$$

$$G_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \tag{5.3}$$

Συνήθως χρησιμοποιείται η Εξίσωση 5.3 για τρεις σημαντικούς λόγους. Ο πρώτος λόγος είναι μαθηματικός και σχετίζεται με την σύγκλιση της σειράς. Δεδομένου οτι οι ακολουθία r_k είναι φραγμένη και ο εκπτωτικός παράγοντας $\gamma \in [0, 1]$, η σειρά θα συγκλίνει. Στην περίπτωση όπου $\gamma = 0$, δεν λαμβάνονται υπόψη οι μελλοντικές επιβραβεύσεις ενώ στην περίπτωση όπου $\gamma = 1$ λαμβάνονται. Στην περίπτωση όπου $\gamma = 1$, δεν αποφεύγεται το πρόβλημα της σύγκλισης. Για αυτό είθισται να χρησιμοποιείται ένα γ πολύ κοντά στο 1. Επιπλέον, ο εκπτωτικός παράγοντας από τη στιγμή που υψώνεται σε κάποια δύναμη, όλο και μεγαλύτερη για πιο μακρινές μελλοντικές καταστάσεις προσδίδει την αίσθηση οτι οι ανταμοιβές του παρόντος έχουν μεγαλύτερη αξία από αυτές του μέλλοντος ("Cash now is better than cash later"). Εν κατακλείδι, με τη χρήση της Εκπτωτικής Ανταμοιβής αποφεύγονται οι άπειρες επαναφορές σε κυκλικές Μαρκοβιανές Διαδικασίες.

5.3 Μαρκοβιανές Διαδικασίες Λήψης Αποφάσεων

Οι Μαρκοβιανές Διαδικασίες Λήψης Αποφάσεων (MDPs) μπορούν να περιγράψουν ένα περιβάλλον της Ενισχυτικής Μάθησης.

5.3.1 Μαρκοβιανή Ιδιότητα

Έστω οτι το περιβάλλον (environment) επιστρέφει τη χρονική στιγμή t στον πράκτορα (agent), μαζί με μια ανταμοιβή (reward), και μια κατάσταση (state) S_t. Μολονότι πράκτορας και περιβάλλον μπορεί να έχουν αλληλεπιδράσει και στο παρελθόν, σε αυτή την κατάσταση S_t θα πρέπει να εγκολπώνονται και όλες οι παρελθοντικές καταστάσεις. Θα πρέπει, δηλαδή, "το μέλλον να είναι ανεξάρτητο του παρελθόντος δεδομένου του παρόντος". Η φράση αυτή συνοψίζει την Μαρκοβιανή ιδιότητα (Markov Property), η οποία ορίζεται και μαθηματικά ως εξής:

$$P[S_{t+1}|S_1, ..., S_t] = P[S_{t+1}|S_t] \tag{5.4}$$

5.3.2 Πίνακας Μετάβασης Καταστάσεων

Ορίζεται, επιπλέον, και ο Πίνακας Μετάβασης Καταστάσεων (State Transition Matrix) ο οποίος σε κάθε στοιχείο του $p_{i,j}$ αποδίδει την πιθανότητα μετάβασης από την κατάσταση i στην κατάσταση j. Η πιθανότητα $p_{i,j}$ υπολογίζεται από τη σχέση 5.5.

$$p_{i,j} = P[S_{t+1} = j|S_t = i] \tag{5.5}$$

Ο Πίνακας Μετάβασης Καταστάσεων (State Transition Matrix), λοιπόν, ορίζεται από τη Σχέση 5.6. Να σημειωθεί πως το άθροισμα των γραμμών ισούται με την μονάδα.

$$P = \begin{bmatrix} p_{11} & \cdots & p_{1n} \\ \vdots & & \vdots \\ p_{n1} & \cdots & p_{nn} \end{bmatrix} \tag{5.6}$$

5.3.3 Μαρκοβιανή Διαδικασία

Κάθε διαδικασία, η οποία ικανοποιεί την Μαρκοβιανή Ιδιότητα 5.3.1 καλείται Μαρκοβιανή Διαδικασία (Markov Process, MP). Μια Μαρκοβιανή Διαδικασία μπορεί να περιγραφεί ώς ένα ζεύγος $\langle S, P \rangle$, όπου S είναι ένα πεπερασμένο σύνολο καταστάσεων πλήθους N και P είναι ένας Πίνακας Μετάβασης Καταστάσεων 5.3.2 $|S| \times |S|$ μεγέθους $N \times N$.

5.3.4 Μαρκοβιανή Διαδικασία με Επιβράβευση

Μια Μαρκοβιανή Διαδικασία με Επιβράβευση (Markov Reward Process, MRP) είναι μια Μαρκοβιανή Διαδικασία 5.3.3 με τη διαφορά οτι χρειάζεται να προστεθεί η έννοια της Συνάρτησης Επιβράβευσης (Reward Function) και του εκπτωτικού παράγοντα (discount factor) $\gamma \in [0, 1]$. Η Μαρκοβιανή Διαδικασία Επιβράβευσης μπορεί να περιγραφεί από μια τετράδα $\langle S, P, R, \gamma \rangle$. Η Συνάρτηση Επιβράβευσης (Reward Function) ορίζεται από την σχέση 5.7.

$$R_S = E[R_{t+1}|S_t = s], \tag{5.7}$$

5.3.5 Συνάρτηση Αξίας Κατάστασης $V(s)$

Η Συνάρτηση Αξίας Κατάστασης (Value Function), δεδομένης μιας συγκεκριμένης πολιτικής (policy) π, αποδίδει για κάποια κατάσταση s την αναμενόμενη ανταμοιβή που θα δοθεί στον πράκτορα αν από την δεδομένη κατάσταση ακολουθήσει την πολιτική π. Η Συνάρτηση Αξίας Κατάστασης (Value Function) ορίζεται από την εξίσωση 5.8.

$$V^{\pi}(s) = E_{\pi}[G_t|S_t = s] \tag{5.8}$$

Η συνάρτηση Αξίας Κατάστασης γίνεται βέλτιστη $V^*(s)$ όταν

$$V^*(s) = max_{\pi} V^{\pi}(s) \tag{5.9}$$

5.3.6 Συνάρτηση Αξίας Δράσης $Q(s, a)$

Η Συνάρτηση Αξίας Δράσης (Action Value Function), δεδομένης μιας συγκεκριμένης πολιτικής (policy) π, αποδίδει για κάποια κατάσταση s την αναμενόμενη ανταμοιβή που θα δοθεί στον πράκτορα αν από την δεδομένη κατάσταση ακολουθήσει την πολιτική π και πραγματοποιήσει τη δράση a. Η Συνάρτηση Αξίας Κατάστασης (Value Function) ορίζεται από την εξίσωση 5.10.

$$Q^{\pi}(s, a) = E_{\pi}[G_t|S_t = s, A_t = a] \tag{5.10}$$

Η συνάρτηση Αξίας Κατάστασης γίνεται βέλτιστη $Q^*(s, a)$ όταν

$$Q^*(s,a) = max_\pi Q^\pi(s,a) \tag{5.11}$$

Ακόμα, η σχέση που συνδέει της Συνάρτηση Αξίας με τη Συνάρτηση Αξίας Δράσης είναι η

$$V^\pi(s) = \sum_{a \in A} \pi(a|s) Q^\pi(s,a) \tag{5.12}$$

Σε αυτό το σημείο, μπορεί επίσης να οριστεί και η Συνάρτηση Πλεονεκτήματος (Advantage Function) $A^\pi(s,a)$ για μια κατάσταση s, μία ενέργεια a και μία πολιτική π ως

$$A^\pi(s,a) = Q^\pi(s,a) - V^\pi(s) \tag{5.13}$$

Επιπλέον, πρέπει να τονιστεί οτι η εύρεση της βέλτιστης πολιτικής π^* επιτυγχάνει και τις βέλτιστες Συναρτήσεις Αξιών. Δηλαδή, ισχύουν οι σχέσεις

$$\pi^* = \arg\max_\pi V^\pi(s) \tag{5.14}$$

$$\pi^* = \arg\max_\pi Q^\pi(s,a) \tag{5.15}$$

Επιπλέον, αν εφαρμοστεί η βέλτιστη πολιτική στις συναρτήσεις αξιών τότε οι συναρτήσεις αυτές θα είναι βέλτιστες, δηλαδή θα ισχύουν οι σχέσεις

$$V^{\pi^*}(s) = V^*(s) \tag{5.16}$$

$$Q^{\pi^*}(s,a) = Q^*(s,a) \tag{5.17}$$

5.3.7 Εξισώσεις Bellman

Αρχικά, αν ορίσουμε τη συνάρτηση μετάβασης $P(s', r|s, a)$ από την κατάσταση s στην κατάσταση s' πραγματοποιώντας την ενέργεια a όπου λαμβάνουμε την ανταμοιβή r ως

$$P(s', r|s, a) = P[S_{t+1} = s', R_{t+1} = r|S_t = s, A_t = a] \tag{5.18}$$

και τώρα ορίσουμε τη Συνάρτηση Μετάβασης Κατάστασης $P^a_{ss'}$ ως τη συνάρτηση που αποδίδει την πιθανότητα μετάβασης από την κατάσταση s στην κατάσταση s' δεδομένης της ενέργειας a, όπως περιγράφεται παρακάτω

$$P^a_{ss'} = P(s'|s,a) = P[S_{t+1} = s'|S_t = s, A_t = a] = \sum_{r \in R} P(s', r|s, a) \tag{5.19}$$

η Συνάρτηση Ανταμοιβής μπορεί να λάβει τη μορφή

$$R(s,a) = E[R_{t+1}|S_t = s, A_t = a] = \sum_{r \in R} r \sum_{s' \in S} P(s', r|s, a) \tag{5.20}$$

Τώρα οι Εξισώσεις των Συναρτήσεων Αξίας Κατάστασης και Αξίας Δράσης μπορούν να γίνουν

$$V^{\pi}(s) = E_{\pi}[G_t|S_t = s] = E(\sum_{k=0}^{\infty} \gamma^k r_{t+k+1}|S_t = s) \tag{5.21}$$

$$= E(r_{t+1} + \sum_{k=1}^{\infty} \gamma^k r_{t+k+1}|S_t = s) \tag{5.22}$$

$$= E(r_{t+1} + \gamma \sum_{k=0}^{\infty} \gamma^k r_{t+k+2}|S_t = s) \tag{5.23}$$

$$= E(r_{t+1} + \gamma G_{t+1}|S_t = s) \tag{5.24}$$

$$= E(r_{t+1} + \gamma V(S_{t+1})|S_t = s) \tag{5.25}$$

$$= R(s,a) + \gamma \sum_{s' \in S} P^a_{ss'} V_{\pi}(s') \tag{5.26}$$

και

$$Q^{\pi}(s,a) = E_{\pi}[G_t|S_t = s, A_t = a] = E(\sum_{k=0}^{\infty} \gamma^k r_{t+k+1}|S_t = s, A_t = a) \tag{5.27}$$

$$= E(r_{t+1} + \sum_{k=1}^{\infty} \gamma^k r_{t+k+1}|S_t = s, A_t = a) \tag{5.28}$$

$$= E(r_{t+1} + \gamma \sum_{k=0}^{\infty} \gamma^k r_{t+k+2}|S_t = s, A_t = a) \tag{5.29}$$

$$= E(r_{t+1} + \gamma G_{t+1}|S_t = s, A_t = a) \tag{5.30}$$

$$= E(r_{t+1} + \gamma V(S_{t+1})|S_t = s, A_t = a) \tag{5.31}$$

$$= E(r_{t+1} + \gamma E_{a \sim \pi} Q(S_{t+1}, a)|S_t = s, A_t = a) \tag{5.32}$$

$$= R(s,a) + \gamma \sum_{s' \in S} P^a_{ss'} \sum_{a' \in A} \pi(a'|s') Q_{\pi}(s', a') \tag{5.33}$$

Η βασική ιδέα των εξισώσεων Bellman είναι οτι η αξία της κατάστασης στην οποία βρίσκεται ένας πράκτορας αντιστοιχεί στο αντίτιμο που αναμένει να λάβει επειδή βρίσκεται στην κατάσταση αυτή συν την αξία που θα προέλθει από την επόμενη κατάσταση στην οποία θα βρεθεί. Η Σχέση 5.26 μπορεί να γραφεί στην μορφή

$$\begin{bmatrix} v(1) \\ \vdots \\ v(n) \end{bmatrix} = \begin{bmatrix} R_1 \\ \vdots \\ R_n \end{bmatrix} + \gamma \begin{bmatrix} p_{11} & \dots & p_{1n} \\ \vdots & & \vdots \\ p_{n1} & \dots & p_{nn} \end{bmatrix} \begin{bmatrix} v(1) \\ \vdots \\ v(n) \end{bmatrix} \tag{5.34}$$

Η εξίσωση 5.34 είναι μια γραμμική εξίσωση, η οποία έχει λύση $v = (1 - \gamma P)^{-1} R$. Είναι γνωστό πως η επίλυση αυτή (στην πράξη) φέρει πολυπλοκότητα $O(n^3)$ [32]. Αυτή η μέθοδος μπορεί να ακολουθηθεί αν το πλήθος των καταστάσεων είναι μικρό. Σε αντίθετη περίπτωση, θα πρέπει να επιστρατευτεί κανείς επαναληπτικές μεθόδους, όπως είναι αυτές του Δυναμικού Προγραμματισμού, του αλγορίθμου Temporal Difference ή των μεθόδων Monte Carlo.

5.3.8 Μαρκοβιανές Διαδικασίες Λήψης Αποφάσεων (MDPs)

Οι Μαρκοβιανές Διαδικασίες Λήψης Αποφάσεων αποτελούν μια επαυξημένη μορφή των Μαρκοβιανών Διαδικασιών με Επιβράβευση 5.3.4. Συγκεκριμένα, οι Μαρκοβιανές Διαδικασίες Λήψης Αποφάσεων (MDPs) μπορούν να περιγραφούν από μια πεντάδα στοιχείων $\langle S, A, P, R, \gamma \rangle$. Το σύνολο αποτελεί ένα πεπερασμένο σύνολο που περιέχει όλες τις πιθανές ενέργειες τις οποίες μπορεί να λάβει ένας πράκτορας. Επιπλέον, διαφοροποιούνται ελάχιστα τόσο ο Πίνακας Μετάβασης Κατάστασης όσο και η Συνάρτηση Ανταμοιβής με τέτοιο τρόπο ώστε να περιλαμβάνουν και την εκάστοτε ενέργεια που λαμβάνει ο πράκτορας. Συγκεκριμένα, ο Πίνακας Μετάβασης Κατάστασης γίνεται $P^a_{ss'} = P[S_{t+1}|S_t = s, A_t = a]$ και η Συνάρτηση Ανταμοιβής γράφεται ως $R^a_s = P[R_{t+1}|S_t = s, A_t = a]$.

5.3.9 Μερικώς Παρατηρήσιμες Μαρκοβιανές Διαδικασίες Λήψης Αποφάσεων (POMDPs)

Οι Μερικώς Παρατηρήσιμες Μαρκοβιανές Διαδικασίες Λήψης Αποφάσεων (POMDPs) αποτελούν μια επαυξημένη μορφή των Μαρκοβιανών Διαδικασιών Λήψης Αποφάσεων. Συγκεκριμένα, οι Μερικώς Παρατηρήσιμες Μαρκοβιανές Διαδικασίες Λήψης Αποφάσεων (MDPs) μπορούν να περιγραφούν από μια επτάδα στοιχείων $\langle S, A, O, P, R, Z, \gamma \rangle$. Με O συμβολίζεται το πεπερασμένο σύνολο των δυνατών παρατηρήσεων ενώ με Z συμβολίζεται η Συνάρτηση Παρατήρησης, η οποία ορίζεται ως $Z^a_{s'o} = P[O_{t+1} = o|S_{t+1} = s', A_t = a]$.

5.3.10 Πολιτική π (Policy)

Με τον όρο πολιτική π (policy) γίνεται λόγος για μια κατανομή πιθανότητας του χώρου των δυνατών ενεργειών που μπορεί να πραγματοποιήσει ένας πράκτορας δεδομένων των καταστάσεων στις οποίες αυτός βρίσκεται. Οι πολιτικές είναι στατικές και ανεξάρτητες του χρόνου όπως επίσης εξαρτώνται από την παροντική κατάσταση.

$$\pi(a|s) = P(A_t = a|S_t = a) \tag{5.35}$$

Μια ενέργεια ενός πράκτορα θα μπορούσε, για παράδειγμα, να προέλθει από μια δειγματοληψία της κατανομής 5.35, $A_t \sim \pi(\cdot|S_t)$, $\forall t > 0$. Επιπλέον, μπορούν να τροποποιηθούν τόσο ο Πίνακας Μετάβασης Κατάστασης

$$P^\pi_{ss'} = \sum_{a \in A} \pi(a|s) P^a_{ss'} \tag{5.36}$$

όσο και η Συνάρτηση Ανταμοιβής

$$R^\pi_s = \sum_{a \in A} \pi(a|s) P^a_s \tag{5.37}$$

5.3.11 Επίλυση σε γνωστές δομές με Δυναμικό Προγραμματισμό

Οι εξισώσεις 5.8 και 5.10 είναι μη γραμμικές και δεν έχουν κλειστή λύση. Για τον λόγο αυτό χρησιμοποιούνται αρκετοί επαναληπτικοί αλγόριθμοι όπως είναι η επαναληπτική μέθοδος πολιτικής (policy iteration), η επαναληπτική μέθοδος αξίας (value iteration), το Q-learning και

Sarsa. Στην περίπτωση που το πρόβλημα ανάγεται στην κατηγορία της Ενισχυτικής Μάθησης με Μοντέλο (model-based) και δεδομένου οτι ένα σύνθετο πρόβλημα μπορεί να διαιρεθεί σε υποπροβλήματα και να αναζητηθεί η λύση σε αυτά μπορεί να χρησιμοποιηθεί Δυναμικός Προγραμματισμός (Dynamic Programming) και οι Επαναληπτικές Μέθοδοι πολιτικής (policy iteration) και αξίας (value iteration). Η εναλλαγή αποτίμησης (policy evaluation) και βελτίωσης πολιτικής (policy improvement) μπορεί να οδηγήσει στην εύρεση βέλτιστων πολιτικών και συναρτήσεων αξιών.

5.3.12 Επίλυση σε άγνωστες δομές

5.3.12.1 Μέθοδοι Monte Carlo

Οι μέθοδοι Monte Carlo έχουν το πλεονέκτημα οτι δεν χρειάζεται να γνωρίζει κανείς την μηχανική του περιβάλλοντος αλλά αντιθέτως βελτιστοποιούνται οι συναρτήσεις αξίας και η πολιτική με βάση την αλληλεπίδραση του πράκτορα με το περιβάλλον του. Να σημειωθεί πως οι βελτιστοποιήσεις γίνονται στο τέλος κάθε επεισοδίου και όχι στα ενδιάμεσα βήματα. Ένα από τα σπουδαιότερα προβλήματα της Ενισχυτικής Μάθησης είναι αυτό του συμβιβασμού μεταξύ εξερεύνησης και αξιοποίησης (exploration vs. exploitation trade-off). Αυτός ο συμβιβασμός εντοπίζεται και στις μεθόδους Monte Carlo. Ένας πράκτορας είτε θα εξερευνήσει το περιβάλλον του (exploration) ακόμα κι αν χρειαστεί να θυσιάσει κάποιες πρόσκαιρες μεγάλες ανταμοιβές είτε θα λάβει τις ανταμοιβές αυτές (exploitation). Ο πράκτορας λαμβάνει αποφάσεις με βάση τις παρελθοντικές του ενέργειες, καταστάσεις και ανταμοιβές. Για να σχηματίσει αποδοτικές πολιτικές θα πρέπει να ενεργήσει με διαφορετικούς τρόπους για να ανακαλύψει ποιοι από αυτούς είναι αποδοτικοί.

5.3.12.2 Temporal-Difference Learning

Σε αντίθεση με τις μεθόδους Monte Carlo, η μέθοδος Temporal-Difference Learning, όντας on-policy μέθοδος και εστιάζοντας σε προβλήματα Ενισχυτικής Μάθησης χωρίς Μοντέλο, μπορεί να ανανεώσει τη συνάρτηση αξίας σε κάθε βήμα και όχι αναγκαστικά στο τέλος ενός επεισοδίου. Η ανανέωση, λοιπόν, της συνάρτησης αξίας κατάστασης χρησιμοποιώντας τη βασική μορφή της μεθόδου TD-Learning, TD(0) γίνεται ως εξής.

$$V(s_t) \leftarrow V(s_t) + a[r_{t+1} + \gamma V(s_{t+1}) - V(s_t)] \tag{5.38}$$

Η τεχνική του TD-Learning όπου οι παράμετροι ανανεώνονται με βάση τις προσεγγίσεις των ανταμοιβών και όχι με τις πραγματικές ανταμοιβές όπως στις Monte-Carlo τεχνικές ονομάζεται bootstrapping.

5.3.12.3 Q-Learning

Η μέθοδος Q-Learning [135] αποτελεί μια μέθοδο που βασίζεται στο TD-Learning. Η διαφορά είναι οτι αποτελεί off-policy μέθοδο (σε αντίθεση με τον αλγόριθμο SARSA) και δεν αποσκοπεί στην εκμάθηση της βέλτιστης πολιτικής αλλά στην εκμάθηση της βέλτιστης συνάρτησης αξίας δράσης Q^*. Αυτή η μέθοδος αποτελεί τον βασικό πυρήνα των μεθόδων που θα χρησιμοποιηθούν στα πειράματα διαπραγματεύσεων που πλαισιώνουν την υλοποίηση της παρούσας διπλωματικής (Κεφάλαιο 8).

Συγκεκριμένα, κάθε χρονική στιγμή t, βρισκόμενοι σε μία κατάσταση S_t επιλέγουμε μία ενέργεια A_t με βάση τις τιμές Q, δηλαδή $A_t = \arg\max_{a \in A} Q(S_t, a)$. Συνήθως, χρησιμοποιείται και η τεχνική ε-greedy, όπου με μία πιθανότητα ε αντί να πραγματοποιήσουμε την ενέργεια όπως αναφέραμε, την επιλέγουμε τυχαία και ομοιόμορφα από το σύνολο των δυνατών ενεργειών. Στη συνέχεια, ανανεώνουμε τις παραμέτρους της Συνάρτησης Αξίας Δράσης $Q(s, a)$ ως εξής

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + a[R_{t+1} + \gamma max_{a \in A} Q(S_{t+1}, a) - Q(S_t, A_t)] \tag{5.39}$$

όπου a είναι μία υπερπαράμετρος. Ανανεώνουμε τον μετρητή χρόνου $t \leftarrow t + 1$ και επαναλαμβάνουμε την ίδια διαδικασία. Το $Q^*(\cdot)$ θα πρέπει να αποθηκεύει όλα τα ζευγάρια κατάστασης-ενέργειας σε έναν τεράστιο πίνακα. Επειδή αυτό είναι αρκετά δύσκολο να γίνει διότι απαιτούνται αρκετοί αποθηκευτικοί πόροι, χρησιμοποιούνται προσεγγιστικές συναρτήσεις (Q-functions). Γνωρίζουμε πως τα Βαθιά Νευρωνικά Δίκτυα μπορούν να προσεγγίσουν μη γραμμικές συναρτήσεις. Επομένως, μια Q-function μπορεί να είναι ένα Βαθύ Νευρωνικό Δίκτυο με παραμέτρους θ.

Το πρόβλημα που προκύπτει σε αυτή την περίπτωση είναι οτι όταν συνδυάζεται ένας off-policy αλγόριθμος με την τεχνική bootstrapping και με μη-γραμμική προσεγγιστική συνάρτηση, η εκμάθηση μπορεί να γίνει ασταθής και να μην συγκλίνει. Το πρόβλημα είναι γνωστό ως "Deadly Triad Issue". Στη δημοσίευση του Mnih [82] εισάγονται τα Deep Q-Networks (DQNs) και αντιμετωπίζεται το εν λόγω ζήτημα με δύο μηχανισμούς: (α) το Experience Replay και (β) την περιοδική ενημέρωση του Target δικτύου.

Initialize replay memory D to capacity N
Initialize action-value function Q with random weights θ
Initialize target action-value function $\hat{Q}$ with weights $\theta^- = \theta$
For episode = 1, M **do**
  Initialize sequence $s_1 = \{x_1\}$ and preprocessed sequence $\phi_1 = \phi(s_1)$
  For t = 1,T **do**
    With probability ε select a random action a_t
    otherwise select $a_t = \text{argmax}_a Q(\phi(s_t), a; \theta)$
    Execute action a_t in emulator and observe reward r_t and image x_{t+1}
    Set $s_{t+1} = s_t, a_t, x_{t+1}$ and preprocess $\phi_{t+1} = \phi(s_{t+1})$
    Store transition $(\phi_t, a_t, r_t, \phi_{t+1})$ in D
    Sample random minibatch of transitions $(\phi_j, a_j, r_j, \phi_{j+1})$ from D
    Set $y_j = \begin{cases} r_j & \text{if episode terminates at step j+1} \\ r_j + \gamma \max_{a'} \hat{Q}(\phi_{j+1}, a'; \theta^-) & \text{otherwise} \end{cases}$
    Perform a gradient descent step on $(y_j - Q(\phi_j, a_j; \theta))^2$ with respect to the network parameters θ
    Every C steps reset $\hat{Q} = Q$
  End For
End For

Σχήμα 5.2: Αλγόριθμος DQN [82]

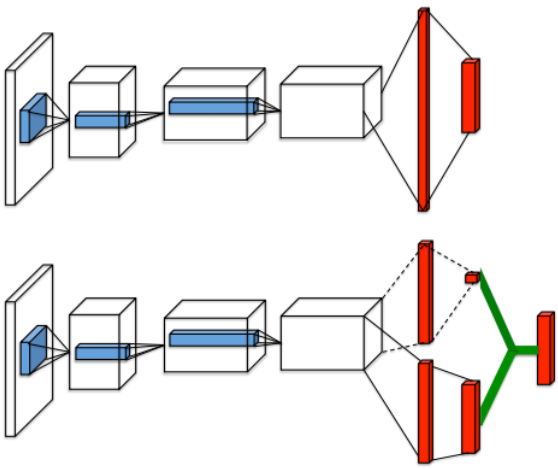
Όσον αφορά το Experience Replay, αποτελεί στην ουσία έναν buffer που αποθηκεύει βήματα των επεισοδίων της μορφής $e_t = (S_t, A_t, R_t, S_{t+1})$ και όταν έρθει η στιγμή της ενημέρωσης των παραμέτρων, επιλέγεται τυχαία από τη μνήμη ένας συγκεκριμένος αριθμός από αυτές τις τετράδες. Με αυτό τον τρόπο αναιρούνται οι συσχετίσεις μεταξύ των βημάτων των επεισοδίων

και εξομαλύνονται οι μεταβολές στην κατανομή των δεδομένων. ταυτόχρονα επιτυγχάνεται μεγαλύτερο data efficiency. Όσον αφορά τη δεύτερη τεχνική, χρησιμοποιείται και ένα δεύτερο δίκτυο πανομοιότυπο με το δίκτυο $\hat{Q}$ (με παραμέτρους θ^-) της Q-συνάρτησης οι παράμετροι του οποίου ενημερώνονται περιοδικά και με αυτόν τον τρόπο σταθεροποιείται η εκμάθηση. Να τονίσουμε πως η συνάρτηση σφάλματος έχει την ακόλουθη μορφή.

$$L(\theta) = E_{(s,a,r,s')\sim U(D)}[(r + \gamma \max_{a'} Q(s', a'; \theta^-) - Q(s, a; \theta))^2] \tag{5.40}$$

όπου με $U(D)$ συμβολίζουμε την ομοιόμορφη κατανομή στη μνήμη D και με θ^- αναφερόμαστε στις παραμέτρους του "παγωμένου" Q-δικτύου. Ο αλγόριθμος DQN, όπως διατυπώθηκε από τον Mnih [82], παρουσιάζεται στην Εικόνα 5.2. Εκτός από το κλασικό DQN μοντέλο [82], στην εργασία μας, χρησιμοποιούνται και μεταγενέστερες εξελίξεις του, όπως είναι τα Double DQNs [124] και τα Dueling DQNs [134]. Το πρόβλημα των απλών DQNs συνήθως είναι το οτι σε θορυβώδη περιβάλλοντα υπερεκτιμούν τις ανταμοιβές και επίσης το οτι χρησιμοποιείται ένα μοντέλο που λαμβάνει αποφάσεις και αξιολογεί τις ενέργειες ταυτόχρονα.

Προς επίλυση των προβλημάτων αυτών, το Double DQN χρησιμοποιεί δύο πανομοιότυπα δίκτυα, όπου έχουν εκπαιδευτεί σε διαφορετικά επεισόδια της "τράπεζας" αποθηκευμένων επεισοδίων εξομαλύνοντας το θόρυβο του περιβάλλοντος και διαχωρίζοντας τις διεργασίες της λήψης και αξιολόγησης ενεργειών. Τα δύο δίκτυα μπορούν να εναλλάσσουν ρόλους με αποτέλεσμα τη συμμετρική ανανέωση των παραμέτρων τους. Μια άλλη ιδέα προτείνεται με το Dueling DQN κατά την οποία υπολογίζεται τόσο η αξία της κατάστασης στην οποία βρίσκεται ένας πράκτορας όσο και η Συνάρτηση Πλεονεκτήματος (Advantage Function). Όπως παρατηρείται στο κάτω μέρος του Σχήματος 5.3, το δίκτυο διαχωρίζεται σε δύο υποδίκτυα. Το πάνω υποδίκτυο υπολογίζει το βαθμωτό μέγεθος $V(s)$ ενώ το κάτω μέρος υπολογίζει τη Συνάρτηση Πλεονεκτήματος για την κατάσταση s και για κάθε δυνατή ενέργεια. Οι έξοδοι των δικτύων συνενώνονται στη συνέχεια.



Σχήμα 5.3: Κλασική Αρχιτεκτονική DQN (Πάνω) και Προτεινόμενη Αρχιτεκτονική Dueling DQN (Κάτω) [134]

5.4 Κατηγοριοποίηση Προβλημάτων Ενισχυτικής Μάθησης

Ένας πρώτος διαχωρισμός μπορεί να γίνει με βάση το αν ο πράκτορας έχει πρόσβαση στο περιβάλλον ή αν μαθαίνει ένα μοντέλο του περιβάλλοντος. Στην περίπτωση που η παραπάνω υπόθεση αληθεύει τότε γίνεται λόγος για Ενισχυτική Μάθηση με Μοντέλο (model-based RL). Σε αντίθετη περίπτωση, γίνεται λόγος για Ενισχυτική Μάθηση χωρίς Μοντέλο (model-free). Στην Ενισχυτική Μάθηση με Μοντέλο, ο πράκτορας μπορεί να χρησιμοποιήσει το μοντέλο αυτό προκειμένου να σχεδιάσει τις στρατηγικές του και να εξετάσει τις πιθανές επιλογές. Το μειονέκτημα έγκειται στο γεγονός οτι ένα τέτοιο μοντέλο δεν προσφέρεται από το περιβάλλον και πρέπει ο πράκτορας να επιδιώξει την εκμάθηση αυτού μέσω της αλληλεπίδρασης με το περιβάλλον. Αν αποβεί επιτυχές το εγχείρημα, ο πράκτορας θα συμπεριφέρεται εξαιρετικά αποδοτικά. Παρόλα αυτά στην πράξη, αυτό αποτελεί μια δύσκολη διαδικασία. Για αυτό το λόγο, έντονη έμφαση δίνεται στην Ενισχυτική Μάθηση χωρίς Μοντέλο.

Μια δεύτερη κατηγοριοποίηση εντοπίζεται στο τι επιλέγει να μάθει ο πράκτορας. Ένας πράκτορας μπορεί να βελτιστοποιείται προκειμένου να μάθει κάποια πολιτική (στοχαστική ή ντετερμινιστική), μια συνάρτηση αξίας κατάστασης, μια συνάρτηση αξίας δράσης ή ένα μοντέλο του περιβάλλοντος. Συγκεκριμένα, στην περίπτωση Ενισχυτικής Μάθησης Χωρίς Μοντέλο (Model-Free RL), συναντώνται συνήθως μέθοδοι Βελτίωσης Πολιτικής (Policy Optimization) και το Q-learning, όπου ο πράκτορας μαθαίνει μια προσέγγιση της Συνάρτησης Αξίας Δράσης.

Ακόμα, μια διαφοροποίηση μεταξύ των διαφορετικών μεθόδων βελτιστοποίησης εντοπίζεται στο αν είναι on-policy ή off-policy. Σε off-policy μεθόδους, οι πράκτορες μπορούν να βελτιστοποιήσουν την πολιτική τους χωρίς να παράγουν νέα δεδομένα από αυτή την πολιτική. Αντιθέτως, σε on-policy μεθόδους, επειδή η πολιτική τους αλλάζει, έστω και ανεπαίσθητα, χρειάζεται να παραχθούν νέα δείγματα.

5.5 Σύνοψη

Στο Κεφάλαιο αυτό, πραγματοποιήθηκε μια εισαγωγή στις βασικές αρχές της Ενισχυτικής Μάθησης και δόθηκε έμφαση στα DQN μοντέλα τα οποία θα χρησιμοποιήσουμε στις υλοποιήσεις του Κεφαλαίου 8. Με αυτό το Κεφάλαιο, ολοκληρώνεται το πρώτο μέρος της εργασίας που αφορά τις Εισαγωγικές Έννοιες. Στα επόμενα Κεφάλαια, μελετάμε τα Διαλογικά Συστήματα και εστιάζουμε στο Πρόβλημα της Διαπραγμάτευσης.

Μέρος ΙΙ

Διαλογικά Συστήματα και Εφαρμογές

Κεφάλαιο 6

Διαλογικά Συστήματα

Με τον όρο διαλογικά συστήματα αναφερόμαστε στα συστήματα που αυτόματα μπορούν να συνδιαλέγονται με κάποιον άνθρωπο ή κάποιο άλλο σύστημα. Ανάλογα με την ύπαρξη ή όχι συγκεκριμένου σκοπού, αυτά κατηγοριοποιούνται σε δύο βασικές κατηγορίες. Η πρώτη αφορά τα συστήματα τα οποία είναι προσανατολισμένα σε ένα συγκεκριμένο σκοπό (task-oriented conversational agents). Τέτοια συστήματα συναντώνται στους αυτόματους τηλεφωνητές τραπεζών, σε εφαρμογές για κλείσιμο κάποιου εισιτηρίου ή τραπεζιού σε εστιατόριο αλλά ακόμα και στις συσκευές οικιακής χρήσης. Αυτοσκοπός τους είναι η υποβοήθηση ενός χρήστη να επιτύχει ένα συγκεκριμένο σκοπό στον ελάχιστο χρόνο και με τον ελάχιστο κόπο. Για αυτό συνήθως, τα συστήματα αυτά επιδιώκουν σύντομες και αρκετά στοχευμένες συνομιλίες. Σε αντίθεση, η δεύτερη κατηγορία διαλογικών συστημάτων, chatbots, επιδιώκει μεγαλύτερες σε χρόνο συζητήσεις και δεν περιορίζεται σε ένα συγκεκριμένο θέμα αλλά η φύση των θεματικών των διαλόγων μπορεί να φέρει μεγάλη ποικιλομορφία. Τέτοια συστήματα χρησιμοποιούνται και για ψυχαγωγικούς λόγους αλλά και για ιατρικούς όπως είναι τα συστήματα που χρησιμοποιούνται ως ψυχαναλυτές. Να σημειωθεί οτι πολλά από αυτά τα συστήματα ήδη από αρκετά νωρίς, συγκριτικά με την εξέλιξη του κλάδου τα τελευταία χρόνια, έχουν καταφέρει να περάσουν το Turing test [121].

Στην Ενότητα 6.1 παρουσιάζονται τα chatbots και στην Ενότητα 6.2 μελετώνται τα διαλογικά συστήματα καθορισμένου (task-oriented) σκοπού. Στη συνέχεια, στην Ενότητα 6.3 περιγράφεται ένας εναλλακτικός τρόπος να μελετάμε τη δημιουργία και εμφάνιση γλώσσας μέσω προσομοιώσεων σε δισδιάστατα ή και τρισδιάστατα περιβάλλοντα. Τέλος, στην Ενότητα 6.4 πραγματοποιείται μια εισαγωγή στο πρόβλημα της Διαπραγμάτευσης.

6.1 Chatbots ή Open-Domain Conversational Agents

Τα chatbots επιδιώκουν συζητήσεις γενικής φύσης που προσομοιάζουν συζητήσεις ανθρώπων πάνω σε ποικίλα θέματα. Πολλά από αυτά, όπως το XiaoIce της Microsoft [144] χρησιμοποιούν κάποια τιμή που προβλέπει τη διασκέδαση του χρήστη. Επίσης, εντοπίζονται δύο κατηγορίες από chatbots, η πρώτη χρησιμοποιεί κανόνες επικοινωνίας (rule-based) ενώ η δεύτερη βασίζεται στην εκμάθηση του συστήματος μέσα από μια ευρεία γκάμα συζητήσεων μεταξύ ανθρώπων (corpus-based). Ιστορικά, το πρώτο chatbot, η ELIZA, δημιουργήθηκε το 1966 από τον Weizenbaum [136] και αποτελεί έναν ροτζεριανό ψυχοθεραπευτή. Η βασική ιδέα αυτού του κλάδου ψυχολογίας είναι να τροποποιούνται και να επαναχρησιμοποιούνται

λεγόμενα του ασθενούς για τη συνέχεια της. Όπως αναφέρεται στο κεφάλαιο 24 του [49], ο Weizenbaum χρησιμοποιεί αυτό το είδος ψυχολογίας διότι οδηγεί σε μια σπάνια μορφή συνομιλίας. Συγκεκριμένα, αν ένας ασθενής αναφέρει "Έκανα μια μεγάλη βόλτα με την βάρκα", ο ψυχίατρος μπορεί να απαντήσει "Μίλησε μου για τις βάρκες". Αυτή η απάντηση του ψυχιάτρου δεν θα έκανε τον ασθενή να απορήσει αν όντως ο ψυχίατρος δεν γνωρίζει τι είναι μια βάρκα. Αυτή η προσέγγιση είναι αρκετά ικανοποιητική υπό το πρίσμα οτι ένα διαλογικό σύστημα δεν γνωρίζει σίγουρα από βάρκες αλλά ο συνομιλητής δεν θα αναρωτηθεί ποτέ για αυτή του την ανικανότητα. Η ELIZA, όπως αναφέρει ο Weizenbaum, δημιουργούσε έντονα συναισθήματα στο προσωπικό-ασθενείς του ως το σημείο που του ζητούσαν να βρίσκονται μόνοι τους με την ELIZA καθόλη τη διάρκεια της συνομιλίας. Παρόλα αυτά, το σύστημα αυτό είναι βασισμένο σε αυστηρούς κανόνες και σε πολύ περιορισμένο πεδίο εφαρμογής και ως επακόλουθο είναι προβλέψιμο, ντετερμινιστικό, μη εξελίξιμο και χαρακτηρίζεται από γλωσσική πενία. Λίγα χρόνια αργότερα, στο ίδιο πεδίο της κλινικής ψυχολογίας, δημιουργείται ο PARRY [14] που χρησιμοποιήθηκε για να μελετήσει τη σχιζοφρένεια. Εκτός από τις κανονικές εκφράσεις στις οποίες άλλωστε είχε βασιστεί και η ELIZA, ο PARRY ανάλογα με τη συναισθηματική του κατάσταση λ.χ. θυμό, φόβο κτλ επέλεγε απαντήσεις από διαφορετικά σύνολα αποκρίσεων αντίστοιχα. Ο PARRY αποτέλεσε το πρώτο γνωστό σύστημα που πέρασε το Turing Test το 1972, αφού οι ψυχίατροι δεν μπορούσαν να διαχωρίσουν αν συνομιλούσαν με τον PARRY ή με κάποιο ψυχασθενή.

Σε αντίθεση με τα συστήματα που παράγουν απαντήσεις μέσω κανόνων, υπάρχουν και αυτά που βασίζονται σε υπάρχουσες συνομιλίες. Τέτοια συστήματα είτε προσπαθούν να ανακτήσουν απαντήσεις από υπάρχοντα κείμενα (information retrieval) είτε παράγουν ακολουθίες λέξεων και κατ' επέκταση αποκρίσεις μέσω μοντέλων που έχουν υποστεί μια μορφή εκμάθησης πάνω σε ένα σύνολο διαλόγων. Συνήθως, τα συστήματα αυτά επειδή δέχονται σχετικά μικρή πληροφορία για το περιεχόμενο του διαλόγου και επιδιώκουν να παράξουν μια απόκριση, καλούνται response generation systems. Τα information retrieval συστήματα βασίζονται στην ιδέα οτι η απάντηση στον χρήστη θα προκύψει από την ανάκτηση κάποια φράσης από ένα σύνολο κειμένων. Έτσι, συχνά, επιλέγεται είτε μια πρόταση η οποία αντιστοιχεί στην ανθρώπινη απάντηση μια παρόμοιας εισόδου που έχει δώσει ο χρήστης [47] [60] είτε σε μια πρόταση η οποία είναι αρκετά όμοια με την πρόταση του χρήστη [101], [133]. Εναλλακτική προσέγγιση, είναι αυτή η χρήση Encoder-Decoder (Seq2Seq) όπου ένα μοντέλο θα πρέπει μέσα από κάποιους διαλόγους να δέχεται μια πρόταση ως είσοδο και να παράγει μια απόκριση ως έξοδο ([111], [108], [115]). Το πρόβλημα που συναντάται σε αυτές τις αρχιτεκτονικές είναι οτι επαναλαμβάνονται πιθανές απαντήσεις όπως 'Δεν ξέρω' ή 'οκ'. Αυτό μπορεί να αντιμετωπιστεί είτε τροποποιώντας τις συναρτήσεις κόστους (χρησιμοποιώντας mutual information objective) [66] είτε να χρησιμοποιηθούν άλλες μέθοδοι παραγωγής λέξεων στον Decoder όπως αυτές αναλύονται στις ενότητες 7.3.5, 7.3.6 και 7.3.7. Πιο φυσικά αποτελέσματα προκύπτουν και από τη χρήση Ενισχυτικής Μάθησης όπως παρουσιάζεται στα [67], [68] . Επιπρόσθετα, τα συστήματα αυτά βασίζονται στην προηγούμενη απόκριση του χρήστη. Προσπάθειες να ενσωματωθεί ένα μεγαλύτερο περιεχόμενο, ιστορικό του διαλόγου εντοπίζονται στο [72]. Επιπρόσθετα, ιδιαίτερο ενδιαφέρον εμφανίζει η δουλειά [30], όπου εκτός από την απόκριση του χρήστη ως είσοδο, το σύστημα δέχεται και διανύσματα που αφορούν το περιεχόμενο (context) του διαλόγου. Τέλος, τα Seq2Seq μοντέλα στον διάλογο είναι εμπνευσμένα από τα προβήματα αυτόματης μετάφρασης και για αυτό φέρουν και τις αντίστοιχες μετρικές όπως BLEU [90], ROUGE [70], METEOR [6]. Παρόλα αυτά, οι μετρικές αυτές δεν αποτελούν ικανοποιητικές μετρικές αξιολόγησης. Για αυτό, κατα κύριο λόγο υπεύθυνοι για την αξιολόγηση είναι οι άνθρωποι (mechanical turkers ή χρήστες). Ιδιαίτερο ενδιαφέρον εμφανίζει η δημιουργία ενός ανταγωνιστικού αξιολογητή (adversarial evaluator) [50] όπου

εμπνευσμένο από το Turing Test, ο αξιολογητής θα πρέπει να μπορεί να ξεχωρίσει ποια πρόταση προέρχεται από άνθρωπο και ποια όχι. Αν, δηλαδή, ένα σύστημα καταφέρει να εξαπατά αυτόν τον ταξινομητή οτι είναι άνθρωπος, αυτό θα είναι ένα αρκετά καλό σύστημα.

6.2 Task-oriented Conversational Agents

Σε αυτό το υποκεφάλαιο, αναλύονται τα διαλογικά συστήματα τα οποία είναι προσανατολισμένα στην υποβοήθηση ενός χρήστη να ολοκληρώσει μια συγκεκριμένη εργασία σε ένα καθορισμένο περιβάλλον, όπως για παράδειγμα είναι η ενοικίαση ενός δωματίου σε ένα ξενοδοχείο ή η αγορά ενός εισιτηρίου σε μια κινηματογραφική αίθουσα. Οι πλέον δημοφιλείς Ψηφιακοί Βοηθοί (Virtual Assistants), όπως είναι οι Siri (Apple), Google Assistant, Alexa (Amazon), Microsoft Cortana και Bixby (Samsung), αποτελούν απτά παραδείγματα της ζήτησης που παρατηρείται στην αγορά για την αυτοματοποίηση ορισμένων διεργασιών η οποία απαιτεί επιτυχημένη επικοινωνία μεταξύ συστήματος και χρήστη.

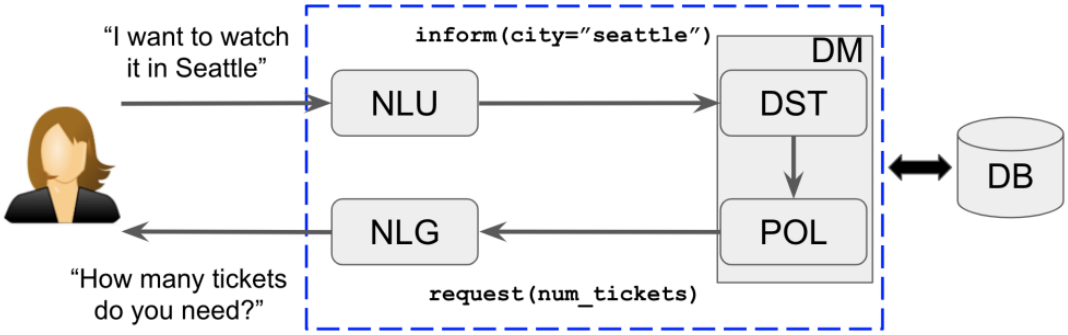
6.2.1 Ιστορική Αναδρομή

Τα διαλογικά συστήματα αυτού του τύπου περιορίζονται σε ένα καθορισμένο περιβάλλον και συνεπώς βασίζονται σε μια οντολογία που προκύπτει από το περιβάλλον αυτό (domain ontology). Η οντολογία αυτή αποτελείται από πλαίσια (frames) όπου κάθε ένα από αυτά αποτελείται από slots τα οποία μπορούν να λάβουν κάποιες τιμές (values). Αυτή η ιδέα εμφανίστηκε για πρώτη φορά το 1977 από τον Bobrow στο σύστημα GUS [9] που χρησιμοποιήθηκε για τον προγραμματισμό ταξιδιών. Στη συγκεκριμένη περίπτωση, για παράδειγμα, ένα slot μπορεί να είναι η πόλη προορισμού του ταξιδιού και μια ενδεικτική τιμή μπορεί να είναι η "Αθήνα". Στην πιο απλή τους μορφή, τα διαλογικά συστήματα αυτού του τύπου προκειμένου να συλλέξουν όλες τις αναγκαίες πληροφορίες για τον προσδιορισμό π.χ. ενός ταξιδιού ήταν βασισμένα σε αρχιτεκτονικές πεπερασμένων αυτομάτων. Επιπλέον, γίνεται ο διαχωρισμός ανάμεσα σε συστήματα που παίρνουν την πρωτοβουλία του διαλόγου και σε αυτά που αφήνουν τον χρήστη ελεύθερο. Το θετικό της περίπτωσης όπου τα συστήματα παίρνουν την πρωτοβουλία, είναι οτι μπορούν να κατευθύνουν την συζήτηση και ως επακόλουθο "γνωρίζουν" ποιες ερωτήσεις απαντάει ο χρήστης ανά πάσα στιγμή και μπορούν να προσαρμόσουν το σύστημα Κατανόησης Φυσικής Γλώσσας (NLU, Natural Language Understanding) να αναμένει συγκεκριμένες απαντήσεις. Το σύστημα GUS αναλαμβάνει πρωτοβουλίες αλλά ταυτόχρονα δίνει και στον χρήστη τη δυνατότητα να επιλέξει το περιεχόμενο της συζήτησης. Εν κατακλείδι, είναι εμφανές πως αυτά τα συστήματα λειτουργούν με βάση συγκεκριμένους κανόνες και αυστηρή ντετερμινιστική δομή, η οποία αφενός απαιτεί μεγάλο κόπο και χρόνο για την κατασκευή και αφετέρου αυτή η κατασκευή είναι δύσκολο να περιλαμβάνει όλες τις πιθανές περιπτώσεις και δεν είναι ελαστική σε μεταβολές που πολύ συχνά συμβαίνουν στον προφορικό λόγο.

6.2.2 Βασικές Έννοιες

Όπως σημειώθηκε στην προηγούμενη υποενότητα, τα διαλογικά συστήματα καλούνται αλληλεπιδρώντας με τον χρήστη να συμπληρώσουν κάποια πεδία (**slots**) με τις επιθυμητές τιμές του χρήστη. Οι διάλογοι αυτοί (**slot-filling ή form-filling dialogues**) εντοπίζονται σε συγκεκριμένα περιβάλλοντα (**domains**) όπως για παράδειγμα είναι αυτό της αγοράς εισιτηρίου για κάποια θεατρική παράσταση και τα πεδία (slots) φέρουν τιμές που αντικατοπτρίζουν

κάποιες αναγκαίες πληροφορίες, όπως για παράδειγμα το όνομα της ταινίας ή ο αριθμός των εισιτηρίων. Τα πεδία είτε ενημερώνουν μια τιμή (**informable**) όπως αριθμός τηλεφώνου είτε ο χρήστης ζητάει να μάθει μια τιμή (**requestable**) όπως τιμή εισιτηρίου είτε χρησιμοποιούνται και με τους δύο τρόπους όπως το όνομα ταινίας.



Σχήμα 6.1: Δομή (modular) Διαλογικού Συστήματος καθορισμένου σκοπού [28]

Το παραπάνω σχεδιάγραμμα περιγράφει την αρχιτεκτονική μεγάλου πλήθους διαλογικών συστημάτων τα οποία είναι προσανατολισμένα για μια συγκεκριμένη εργασία. Ο χρήστης εισάγει στο σύστημα τη φράση "Θέλω να την δω (εννοεί την ταινία) στο Σιάτλ". Το πρώτο υποσύστημα που τίθεται σε λειτουργία είναι το σύστημα Κατανόησης Φυσικής Γλώσσας (**NLU, Natual Language Understanding**). Αυτό μετατρέπει το κείμενο του χρήστη από φυσική γλώσσα σε διαλογική ενέργεια (**dialogue act**). Μια διαλογική ενέργεια μπορεί να εξυπηρετεί στην ενημέρωση κάποιου πεδίου, στο αίτημα για κάποια πληροφορία, σε κάποια πρόταση και σε άλλες ενέργειες που μπορεί να κρίνονται σκόπιμες από τους σχεδιαστές. Στο συγκεκριμένο παράδειγμα, η διαλογική ενέργεια, που έχει τη σημασιολογική μορφή $inform(city = "seattle")$, συμβάλλει στην ενημέρωση του πεδίου "Πόλη" με την τιμή "Σιάτλ". Στη συνέχεια, αυτή η διαλογική ενέργεια εισέρχεται στον **Dialogue Manager**. Πρόκειται για την κεντρική δομή του διαλογικού συστήματος και αποτελείται από δύο υπομέρη. Το πρώτο (**DST, Dialogue State Tracker**) είναι υπεύθυνο για να αναγνωρίζει σε ποια κατάσταση βρίσκεται ο διάλογος με τον χρήστη ανά πάσα χρονική στιγμή και το δεύτερο (**Policy**) δέχεται την κατάσταση του διαλόγου και αποφασίζει ποια διαλογική ενέργεια θα προάγει τον διάλογο με τον χρήστη. Αυτό το ευρύτερο σύστημα, επικοινωνεί συχνά και με κάποια βάση δεδομένων για να αποθηκεύει ή να ανακτά πληροφορίες. Έπειτα, αυτή η διαλογική ενέργεια που επιλέγει ως απάντηση το σύστημα θα πρέπει να παραχθεί σε φυσική γλώσσα που θα είναι κατανοητή από τον χρήστη. Επομένως, το τελευταίο υποσύστημα (**NLG, Natural Language Generation**) εξυπηρετεί αυτή τη λειτουργία και δεχόμενο μια ενέργεια παράγει φυσική γλώσσα και απαντά στον χρήστη.

Αυτή η περιγραφή οδήγησε στη συσχέτιση του προβλήματος με περιβάλλοντα Ενισχυτικής Μάθησης, όπου η πρόταση του χρήστη αντιστοιχεί στην παρατήρηση (observation) του συστήματος και η απάντηση του συστήματος αντιστοιχεί στην ενέργεια (action) [16], [120]. Έτσι, λοιπόν, ο χρήστης μπορεί να αντιμετωπιστεί ως το αντίστοιχο περιβάλλον που χρησιμοποιείται στα προβλήματα της Ενισχυτικής Μάθησης και ο διάλογος να αναχθεί σε ένα πρόβλημα βέλτιστης λήψης απόφασης [61]. Το αντίτιμο μπορεί να προκύψει ανάλογα με την ποιότητα της απάντησης, αλλά κατά κύριο λόγο από την επίτευξη ή όχι του σκοπού ("έκλεισε ή όχι το εισιτήριο") και από χρονικές συνιστώσες, -1 αντίτιμο σε κάθε απάντηση προκειμένου να πετυχαίνει το σύστημα τον επιθυμητό σκοπό στον ελάχιστο δυνατό χρόνο. Να σημειω-

θεί πως η εύρεση κατάλληλων αντιτίμων είναι πολύ σημαντική και επηρεάζει δραματικά τα αποτελέσματα και τη γενικότερη συμπεριφορά του συστήματος.

Οι Walker [131] και Singh [114] ήταν από τους πρώτους που επιχείρησαν να εγκολπώσουν την χρήση ενισχυτικής μάθησης στα διαλογικά συστήματα της κατηγορίας που περιγράφεται στο κεφάλαιο αυτό. Παρόλα αυτά, ανήγαγαν το πρόβλημα σε μια πλήρως παρατηρήσιμη μαρκοβιανή διαδικασία λήψης αποφάσεων υποθέτοντας αφενός οτι το σύστημα NLU λειτουργεί καλά και αφετέρου οτι ο χρήστης δηλώνει ξεκάθαρα τις προθέσεις του. Αυτές οι υποθέσεις είναι αντιφατικές με την πραγματικότητα και για αυτό τον λόγο οι Roy [96] και Williams & Young [138] πρότειναν τη χρήση μερικώς παρατηρήσιμων μαρκοβιανών διαδικασιών λήψης αποφάσεων διαδικασιών (POMDP, Partially Observable Markov Decision Process). Σε αυτή την προσέγγιση, συναντάται συχνά ο όρος "belief state".

Σε αυτό το σημείο αξίζει να αναφερθούν τα προβλήματα των συστημάτων αυτών. Αρχικά, τα συστήματα αυτά απαιτούν λεπτομερή και ακριβή σχεδίαση η οποία εντοπίζεται σε έναν συγκεκριμένο κλάδο (π.χ. εισιτήρια σε κινηματογράφο) και κατ' επέκταση δεν μπορεί να χρησιμοποιηθεί σε άλλους κλάδους (π.χ. αεροπορικά εισιτήρια). Χρειάζεται, δηλαδή, κανείς να σχεδιάσει εκ νέου όλα τα πεδία που θα πρέπει να συμπληρωθούν μολονότι υπάρχει επικάλυψη ορισμένων πεδίων από κλάδο σε κλάδο. Ένα εξίσου σημαντικό πρόβλημα είναι οτι το σύστημα αποτελείται από υποσυστήματα τα οποία βελτιστοποιούνται ξεχωριστά το ένα από το άλλο. Εκτός από το γεγονός οτι μπορεί να μην είναι συμβατή η βέλτιστη λειτουργία του ενός με του άλλου, προκύπτει το πρόβλημα ανάθεσης ευσήμων (credit assignment problem), όπως αναφέρεται στο [28]. Σύμφωνα με το πρόβλημα αυτό, αν το σύστημα παράγει μη ικανοποιητικές απαντήσεις είναι αρκετά δύσκολο να εντοπιστεί πιο σύστημα υπολειτουργεί. Χαρακτηριστικά αναφέρεται από τον McTear [76] οτι "το κλειδί σε ένα επιτυχημένο διαλογικό σύστημα είναι η ενσωμάτωση όλων των εξαρτημάτων (υποσυστημάτων) σε ένα λειτουργικό σύστημα". Για την επίλυση αυτών των ζητημάτων, ερευνώνται πλέον συστήματα με χρήση νευρωνικών δικτύων και ενισχυτικής μάθησης, τα οποία βελτιστοποιούνται ως μία ολότητα απ' άκρου εις άκρον (end-to-end) και θα αναλυθούν στη συνέχεια.

Σημαντικό μέρος της μελέτης των διαλογικών συστημάτων αποτελεί η αξιολόγηση των συστημάτων αυτών. Σε αντίθεση με τα chatbots, λόγω της επίτευξης ενός συγκεκριμένου σκοπού συναντώνται μετρικές ορθότητας (accuracy), ακρίβειας (precision), ανάκλησης (recall), F1 και BLEU [132], [129], [130], [89], [35]. Επιπλέον, αν χρησιμοποιείται ενισχυτική μάθηση, μπορούν να χρησιμοποιηθούν τα αντίτιμα που προκύπτουν από τον διάλογο ως μέσα αξιολόγησης και σύγκρισης των συστημάτων. Εν γένει, μετρικές οι οποίες χρησιμοποιούνται στην περίπτωση των chatbots, μπορούν να χρησιμοποιηθούν και εδώ μαζί με τις μετρικές που αναφέρθηκαν και κατά κύριο λόγο εστιάζουν στην επίτευξη του στόχου που ορίζεται από το περιβάλλον.

Επιπλέον, η χρήση της Ενισχυτικής Μάθησης επιβάλλει την εκμάθηση μέσω διαλόγων με κάποιους χρήστες. Επειδή αυτή η διαδικασία εκτός από χρονοβόρα έχει και μεγάλο κόστος (είτε χρηματικό είτε ως προς την αξιοπιστία της εταιρείας), χρησιμοποιούνται συχνά προσομοιωτές. Στο [115] παρουσιάζεται μια εκτενής βιβλιογραφία σχετικά με την δημιουργία ρεαλιστικών προσομοιωτών [106]. Ιδιαίτερο ενδιαφέρον παρόλα αυτά παρατηρείται στις διαφοροποιήσεις των προσομοιωτών αυτών. Άλλοι προσομοιωτές εστιάζουν στο επίπεδο των διαλογικών ενεργειών ενώ άλλοι σε επίπεδο φυσικής γλώσσας [48]. Επίσης, άλλοι προσομοιωτές χρησιμοποιούν κανόνες προκειμένου να συνδιαλεχθούν (**Agenda-Based Simulation** [105], [69], [122]) ενώ άλλοι βασίζονται σε ένα μοντέλο που έχει υποστεί εκμάθηση από πραγματικές συνομιλίες (**Model-based Simulation**, [25], [61], [12]). Το πεδίο αυτό ερευνάται ακόμα.

Αναγκαία και ουσιαστική μορφή αξιολόγησης είναι αυτή που πραγματοποιείται από ανθρώπους. Παρουσιάζονται δύο διαφορετικά είδη χρηστών-αξιολογητών. Οι πρώτοι εντοπίζονται στο εργαστήριο ή σε πλατφόρμες (crowdsourcing) ενώ οι δεύτεροι είναι οι πραγματικοί χρήστες μιας εφαρμογής. Εναλλακτικές μορφές αξιολόγησης που χρησιμοποιούνται σχετίζονται με τη λογική του self-play ([119], [83]), όπου το σύστημα μπορεί να "παίζει" με τον εαυτό του. Με μικρές τροποποιήσεις ένα σύστημα χρησιμοποιώντας το self-play μπορεί να παράξει νέους διαλόγους οι οποίοι θα χρησιμεύσουν ως δεδομένα από τα οποία μπορεί μετέπειτα να μάθει το σύστημα χρησιμοποιώντας επιβλεπόμενη μάθηση [109]. Εν κατακλείδι, συνίσταται από το [28], να χρησιμοποιούνται υβριδικές μέθοδοι αξιολόγησης.

6.3 Grounded Language Learning: Emergent Communication

Όπως έχει προαναφερθεί, κατά την εκπόνηση της παρούσας διπλωματικής εργασίας σχηματίστηκε η άποψη οτι η γλώσσα δημιουργείται από σαφώς καθορισμένους πράκτορες προκειμένου να επικοινωνήσουν μηνύματα εντός ενός σαφώς καθορισμένου περιβάλλοντος προκειμένου να ικανοποιήσουν κάποια ανάγκη τους. Υπό μια έννοια η γλώσσα "γεννάται" από τη στιγμή που "γεννάται" και η ανάγκη της επίλυσης ενός προβλήματος που απαιτεί κάποια μορφή συνεργασίας. Συγκεκριμένα, σε διαλογικά προβλήματα παρατηρούνται δύο αντίρροπες κατευθύνσεις ως προς τη μελέτη τους. Η πρώτη σχετίζεται με την παραγωγή γλώσσας μέσω της ανάγκης ικανοποίησης του προβλήματος (bottom-up) και η δεύτερη σχετίζεται με τη χρησιμοποίηση υπαρχόντων γλωσσικών μοντέλων τα οποία εστιάζουν στο συγκεκριμένο πρόβλημα (top-down). Η πρώτη περίπτωση αποτελεί μια εξαιρετικά ενδιαφέρουσα επιλογή και ένας κλάδος που μελετά την εμφάνιση γλώσσας όταν αυτή είναι γειωμένη (grounded) σε ένα περιβάλλον είναι αυτός του Grounded Language Learning.

Μια προσπάθεια για εφαρμογή αυτής της θεωρίας επιχειρήθηκε στο [36] όπου ένας πράκτορας εισάγεται σε ένα προσομοιωμένο, τρισδιάστατο περιβάλλον και λαμβάνει ανταμοιβές για την επιτυχή εκτέλεση γραπτών οδηγιών. Κατά την εκπαίδευση με χρήση Μη Επιβλεπόμενης και Ενισχυτικής Μάθησης, ο πράκτορας επιτυγχάνει, χωρίς πρότερη γνώση, να συσχετίσει γλωσσικά σύμβολα σε νοητικές αναπαραστάσεις του φυσικού περιβάλλοντος και σε συναφείς ακολουθίες δράσεων. Η αντίληψή του για την γλώσσα τον βοηθάει να χρησιμοποιήσει γνωστό λεξιλόγιο σε άγνωστες καταστάσεις και να ερμηνεύσει νέες εντολές. Επιπλέον, όσο μεγαλώνει το μέγεθος της σημασιολογικής του γνώσης τόσο αυξάνει η ταχύτητα με την οποία μαθαίνει νέες λέξεις. Με αυτό τον τρόπο, η εκμάθηση και η κατανόηση συσχετίζονται σε ένα συνεχές περιβάλλον όπου ο πράκτορας μαθαίνει end-to-end να "γειώνει" μια γλωσσική έκφραση, να αντιλαμβάνεται δηλαδή, και τη φυσική υπόσταση των λέξεων. Συνδυάζει, λοιπόν, τη χρήση της γλώσσας για ολοκλήρωση ενός προβλήματος με τα οπτικά ερεθίσματα (pixels) που δέχεται κάθε φορά σαν είσοδο.

Για τις προσομοιώσεις, χρησιμοποιήθηκε μια εκτεταμένη έκδοση του DeepMind Lab όπου ο πράκτορας πρέπει να βρει και να συλλέξει αντικείμενα δεχόμενος μία γλωσσική περιγραφή σε κείμενο. Ο πράκτορας δέχεται συνεχή οπτικά ερεθίσματα ως είσοδο μαζί με την εντολή γραμμένη σε κείμενο. Εντοπίζονται 4 δομικά στοιχεία που αφορούν την όραση (vision module), τη γλώσσα (language module), τη μίξη αυτών (mixing module) και την ενέργεια (action module). Επιπλέον, ως προς τις τεχνικές της Ενισχυτικής Μάθησης χρησιμοποιείται ο Αλγόριθμος Advantage Actor-Critic [**?**]. Ενδιαφέρον στη μελέτη αυτή συναντάται και στην πρόβλεψη ενός πράκτορα σχετικά με το πώς αναμένει να μεταβληθεί ο κόσμος ανάλογα με τις δράσεις του (temporal autoencoding).

Ενδιαφέρουσα προσέγγιση είναι και αυτή του Mikolov [78] όπου ερευνά τα βασικά χαρακτηριστικά τα οποία θα πρέπει να έχει ένα ευφυές σύστημα δίνοντας έμφαση στις έννοιες της επικοινωνίας και της μάθησης. Όσον αφορά την επικοινωνία, ενδεικτικά αναφέρεται οτι αυτή θα πρέπει να είναι διαδραστική και πως τα συστήματα που θα παραχθούν θα πρέπει να έχουν την ικανότητα να επικοινωνούν με τους ανθρώπους προκειμένου να μπορούμε να κατανοούμε τα αποτελέσματα που παράγονται ενώ όσον αφορά την ικανότητα μάθησης ένα σύστημα θα πρέπει να προσαρμόζεται, να αυτοδιορθώνεται και να αποκτά κίνητρα. Εξαιρετικό ενδιαφέρον παρουσιάζει η ιδέα χρήσης ενός Δασκάλου που θα οδηγήσει έναν πράκτορα-μαθητή να μαθαίνει τη γλώσσα.

Η επικοινωνία πραγματοποιείται μέσω καναλιών I/O (Input/Output) για λόγους απλότητας. Παρόλα αυτά, όσο πιο περίπλοκο το περιβάλλον, τόσο πιο σύνθετες θα είναι και οι στρατηγικές που θα δημιουργηθούν και έτσι αυτό το περιβάλλον επικοινωνίας θεωρείται πως είναι μια εσφαλμένη αναπαράσταση των πραγματικών περιβάλλοντων [29]. Εξαιρετικό ενδιαφέρον εμφανίζει, επίσης, και η χρήση πολυπρακτορικών συστημάτων όπως συναντάται στα [58] και [85].

6.4 Εισαγωγή στις Διαπραγματεύσεις

Σύμφωνα με τους Pruitt και Carnevale [97], με τον όρο διαπραγμάτευση καλούμε τον διάλογο που στοχεύει στην επίτευξη μιας συμφωνίας μεταξύ πρακτόρων με αντικρουόμενα συμφέροντα. Από την στιγμή, λοιπόν, που τα συμφέροντα δύο ή περισσότερων οντοτήτων είναι αντικρουόμενα και προκύπτει η ανάγκη εύρεσης κάποιας συναίνεσης, προκύπτουν ως φυσικό επακόλουθο διαπραγματεύσεις. Αυτές, λοιπόν, αποτελούν αναπόσπαστο κομμάτι της καθημερινότητας και εμφανίζονται από απλές περιστάσεις όπως είναι η διαπραγμάτευση κάποιας τιμής για ένα προϊόν σε ένα κατάστημα μέχρι σε πιο σύνθετες καταστάσεις με έντονο οικονομικό αντίκτυπο όπως μπορεί να είναι η διαπραγμάτευση μεταξύ δύο ή περισσότερων μονοπωλίων ή οικονομικών κολοσσών. Στην εκπόνηση της παρούσας διπλωματικής εργασίας, δίνεται έμφαση στη διαπραγμάτευση μεταξύ δύο πρακτόρων η οποία αποτελεί την βάση για τη μελέτη προβλημάτων διαπραγμάτευσης μεταξύ περισσότερων των δύο πρακτόρων.

Η διαπραγμάτευση, κατά τον John Nash [86], μπορεί να θεωρηθεί και ως ένα "παιχνίδι μη μηδενικού αθροίσματος" (nonzero-sum game). Αυτή η προτεινόμενη αντιμετώπιση έρχεται σε αντιστοιχία με την προσέγγιση των von Neumann και Morgenstern [128] και σε αντίθεση με την φιλοσοφία οικονομικών προσεγγίσεων ως προς το διμερές μονοπώλιο των Cournot, Bowley, Tintner και Fellner, όπως αναφέρεται στο [86]. Επιπλέον, σε τέτοιου είδους παίγνια τίθεται το ζήτημα της ύπαρξής κάποιας "λογικής" συμπεριφοράς (rational behavior). Παρόλα αυτά, ο προσδιορισμός της είναι αρκετά δύσκολος και επιτυγχάνεται υποθέτοντας τα κίνητρα και τις ανάγκες κάθε πράκτορα. Η μερική ή ολική ικανοποίηση κάποιων από των αναγκών αυτών θα πρέπει να αποδίδει μια ικανοποίηση.

Επιπλέον, ένας πράκτορας σχηματίζει σε κάποιο παίγνιο μια προσμονή (anticipation), δηλαδή μια προσδοκία που απαρτίζεται είτε από μια βεβαιότητα σχετικά με κάποιο ενδεχόμενο είτε από πιθανότητες μεταξύ διαφόρων ενδεχομένων. Ο Nash, λοιπόν, ανέπτυξε τις προϋποθέσεις υπό τις οποίες μπορεί να προκύψει μια ικανοποιητική συνάρτηση ωφελιμότητας (utility function), η οποία μπορεί να αποδώσει μια πραγματική τιμή σε κάθε προσμονή κάθε πράκτορα. Συνήθως, σε προβλήματα διαπραγμάτευσης αντικειμένων μια ικανοποιητική προσέγγιση συνάρτησης ωφελιμότητας σχετίζεται με την ίδια την αξία των αντικειμένων αυτών.

Στο σημείο αυτό να τονιστεί οτι αυτοσκοπός μιας πετυχημένης διαπραγμάτευσης είναι η επίτευξη μιας συναίνεσης, με την έννοια οτι αν δεν υπάρχει δυνατότητα συναίνεσης, δεν υφίσταται η έννοια της διαπραγμάτευσης [63]. Επιπλέον, άλλοτε παρουσιάζονται παίγνια όπου η ήττα ενός πράκτορα αντιστοιχεί σε νίκη του άλλου και άλλοτε παρουσιάζονται παίγνια στα οποία και οι δύο πράκτορες επιδιώκουν την καλύτερη δυνατή "κοινωνική" έκβαση. Αξιοσημείωτο, επίσης, να αναφερθεί είναι το γεγονός οτι τα παίγνια αυτά μπορούν να αποτελέσουν πρόσφορο έδαφος δημιουργίας αυτόματων συστημάτων διαπραγμάτευσης [4], τα οποία μπορούν να ανταγωνιστούν άλλα αυτόματα συστήματα συμπεριλαμβανομένου και του εαυτού τους.

6.5 Σύνοψη

Το πρόβλημα της διαπραγμάτευσης αποτελεί ένα από τα δυσκολότερα προβλήματα καθορισμένου σκοπού (task-oriented) κι ο λόγος έγκειται στο γεγονός οτι απαιτεί το συνδυασμό πλούσιων γλωσσικών χαρακτηριστικών μαζί με συλλογιστική ικανότητα. Μπορεί να αντιμετωπιστεί με τις δύο διαφορετικές προσεγγίσεις που αναφέρθηκαν, την bottom-up και την top-down. Εμείς δίνουμε έμφαση στη δεύτερη προκειμένου να εκμεταλλευτούμε μια πληθώρα σύγχρονων μοντέλων που στιγματίζουν τα τελευταία χρόνια την κοινότητα της Επεξεργασίας Φυσικής Γλώσσας. Δεδομένης αυτής της επιλογής, μπορούμε να χρησιμοποιήσουμε δύο διαφορετικά είδη αρχιτεκτονικών: τις end-to-end και τις modular αρχιτεκτονικές. Στο Κεφάλαιο 7, δίνουμε έμφαση στις end-to-end αρχιτεκτονικές αλλά αποδομώντας το υπό μελέτη πρόβλημα διαπραγμάτευσης σε υποπροβλήματα και προτείνοντας μοντέλα που τα βελτιώνουν επιχειρούμε να κάνουμε και μια νύξη σε modular αρχιτεκτονικές. Όπως θα δούμε στη συνέχεια, ένα πρόβλημα διαπραγμάτευσης απαιτεί να μπορούμε να παράγουμε γλώσσα (γλωσσική μοντελοποίηση), να μπορούμε να κατανοούμε τις προσφορές που μας κάνει ένας πράκτορας (ανάγεται σε πρόβλημα ταξινόμησης) και να λαμβάνουμε τις κατάλληλες αποφάσεις (πρόβλημα βέλτιστης λήψης απόφασης). Τα δύο πρώτα προβλήματα μελετώνται στο Κεφάλαιο 7 ενώ το τελευταίο πρόβλημα συναντάται στο Κεφάλαιο 8.

Κεφάλαιο 7

Εφαρμογή End-to-End Task-oriented Διαλογικών Συστημάτων σε Προβλήματα Διαπραγμάτευσης

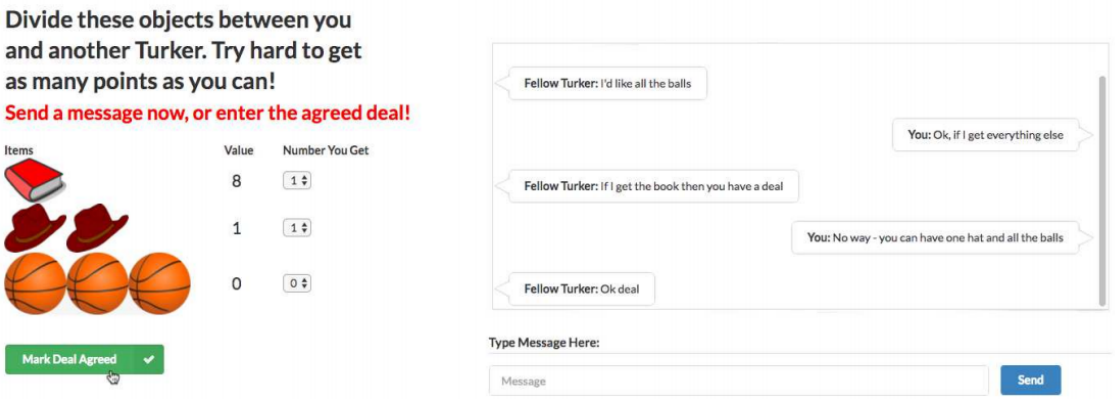
7.1 Εισαγωγή

Το πρόβλημα της διαπραγμάτευσης αποτελεί ένα ενδιαφέρον πρόβλημα τόσο από θεωρητική όσο και από πρακτική σκοπιά. Εστιάζοντας στο πρόβλημα της διαπραγμάτευσης, η δυσκολία του έγκειται στο γεγονός οτι σε ένα διάλογο τέτοιου σκοπού συνυπάρχουν χαρακτηριστικά που απαιτούν από ένα διαλογικό σύστημα γλωσσική επάρκεια και συγχρόνως συλλογιστικές ικανότητες. Στο Κεφάλαιο αυτό, μελετάται η χρήση end-to-end διαλογικών συστημάτων που βασίζονται στα μοντέλα που προτείνονται από τη Facebook [65]. Παρόλα αυτά, γίνεται σε αρκετά σημεία νύξη για χρησιμοποίηση και modular αρχιτεκτονικών που θα μπορούσαν να λειτουργήσουν αποδοτικότερα σε ορισμένες εργασίες (tasks). Οι δύο αυτές προσεγγίσεις έχουν διαφορετικά πλεονεκτήματα και μειονεκτήματα, τα οποία ένας μηχανικός θα πρέπει να γνωρίζει εκ των προτέρων. Ανεξαρτήτως της δυσκολίας των προβλημάτων διαπραγμάτευσης, το αποτέλεσμα τους γίνεται εύκολα αντιληπτό και συνεπώς εξίσου εύκολα μπορεί να αξιολογηθεί και η επιτυχία των διαλογικών συστημάτων.

Αρχικά, μελετάται ένα πρόβλημα διαπραγμάτευσης τριών αντικειμένων μεταξύ δύο πρακτόρων (multi-issue bargaining) που εισήχθηκε από τη Facebook. Αφότου αναπαραχθούν οι αρχιτεκτονικές του [65], προτείνονται στην Ενότητα 7.1.5 κάποιες τροποποιήσεις των μοντέλων. Στην πορεία διαχωρίζεται το υπό μελέτη πρόβλημα σε υποπροβλήματα. Αρχικά, ένα μοντέλο χρειάζεται να παράγει γλώσσα (language modeling και generation, Ενότητα 7.3), να κατανοεί τις προτάσεις του αντιπάλου του (ανάγεται σε πρόβλημα κατηγοριοποίησης, Ενότητα 7.2) και να επιλέγει τις κατάλληλες ενέργειες. Το τελευταίο βήμα, αν και αποτελεί τον ακρογωνιαίο λίθο μιας τέτοιας υλοποίησης κρίνεται οτι χρειάζεται διαφορετικό τρόπο ανάλυσης από αυτόν που προτείνεται στο [65]. Για το λόγο αυτό, μελετάται ξεχωριστά στο επόμενο Κεφάλαιο. Για τα δύο πρώτα υποπροβλήματα, κάνοντας χρήση των Transformers και της Μεταφοράς Μάθησης (Transfer Learning), προτείνονται κάποιες αρχιτεκτονικές που αποδεικνύονται ως βέλτιστες.

7.1.1 Περιγραφή Deal Or No Deal

Το Deal Or No Deal αποτελεί ένα task που σχηματίστηκε από τη Facebook προκειμένου να μελετήσει με end-to-end αρχιτεκτονικές το πρόβλημα της διαπραγμάτευσης σε ημισυνεργατικά περιβάλλοντα δίνοντας έμφαση τόσο στα γλωσσικά όσο και στα στρατηγικά χαρακτηριστικά του προβλήματος [65]. Αρχικά, σε κάθε πράκτορα (agent) εμφανίζονται τα διαθέσιμα αντικείμενα προς διαπραγμάτευση και μία συνάρτηση ανταμοιβής, η οποία καθορίζει και τη βαθμολογία του πράκτορα ανάλογα με το αποτέλεσμα της διαπραγμάτευσης. Μολονότι τα διαθέσιμα αντικείμενα είναι ορατά και από τους δύο πράκτορες, δεν συμβαίνει το ίδιο και με τις συναρτήσεις ανταμοιβών. Κάθε πράκτορας είναι σε θέση να γνωρίζει μόνο τη δική του συνάρτηση ανταμοιβών, η οποία προσδιορίζει την προσωπική του προσέγγιση για την αξία κάθε αντικειμένου. Στόχος, λοιπόν, είναι η επίτευξη συμφωνίας στο διαμοιρασμό των αντικειμένων μέσω ενός καναλιού επικοινωνίας όπου οι πράκτορες συνδιαλέγονται σε φυσική γλώσσα. Η διεπαφή που προτείνεται για τη συλλογή των δεδομένων και περιγράφεται από το Σχήμα 7.1 αποτελεί προσαρμοσμένη έκδοση της διεπαφής που συναντάται στο [20]. Μέσω των μηνυμάτων κάθε πράκτορας επιδιώκει να αναγνωρίσει την συνάρτηση ανταμοιβών του άλλου πράκτορα προκειμένου να οδηγηθεί σε μία συμφέρουσα συμφωνία. Όταν κάποιος πράκτορας επιλέξει οτι έχει πραγματοποιηθεί η συμφωνία, τότε και οι δύο πράκτορες δηλώνουν ποιος είναι ο διαμοιρασμός των αντικειμένων στον οποίο συμφωνούν. Αν είναι διαφορετικός, τότε δεν λαμβάνουν καμία επιβράβευση.



Σχήμα 7.1: Διεπαφή Διαπραγμάτευσης για τη Συλλογή Δεδομένων. Παρουσιάζονται τα διαθέσιμα αντικείμενα με τις αντίστοιχες αξίες (Αριστερά) και το Κανάλι Επικοινωνίας (Δεξιά) [65]

7.1.2 Σχετική Βιβλιογραφία

Το υπό μελέτη πρόβλημα είναι βασισμένο στην υλοποίηση του Devault [22] και ανήκει στην κατηγορία προβλημάτων "multi-issue bargaining", όπου η διαπραγμάτευση πραγματοποιείται πάνω σε ένα πεπερασμένο αριθμό αντικειμένων προκαθορισμένων κατηγοριών. Σε αντίθεση με το Κεφάλαιο 8, η επικοινωνία δεν γίνεται με τη χρήση προκαθορισμένων συμβολικών πράξεων, αλλά αντιθέτως φέρει μεγαλύτερη ποικιλομορφία. Το πρόβλημα που μελετάται από τη Facebook, μεταξύ άλλων, διαφέρει από το πρόβλημα του Devault στο γεγονός οτι περιορίζεται σε γραπτούς διαλόγους. Αντίστοιχες προσεγγίσεις που έχουν συνδυάσει τη φυσική γλώσσα με την επίτευξη συγκεκριμένου σκοπού που ταυτόχρονα απαιτεί τη λήψη κάποια απόφασης είναι αυτές των Rosenfeld [102], Cuayáhuitl [17] και Keizer [52]. Ιδιαίτερο ενδιαφέρον

παρουσιάζει επίσης και η μελέτη του Cao [11], όπου μελετάται η χρήση πολυπρακτορικής ενισχυτικής μάθησης ως η ενδεδειγμένη μέθοδος για την εμφάνιση και δημιουργία γλώσσας μεταξύ των διαφόρων πρακτόρων που αποσκοπούν να λύσουν ένα συγκεκριμένο πρόβλημα.

7.1.3 Δεδομένα

Το σύνολο των δεδομένων που χρησιμοποιούνται για την μελέτη της εφαρμογής task-oriented διαλογικών συστημάτων σε προβλήματα διαπραγματεύσεων βασίζεται στα δεδομένα που συλλέχθηκαν από ανθρώπους που διαπραγματεύονταν σε ήμι-συνεργατικά (semi-cooperative) περιβάλλοντα [65] αποσκοπώντας στον διαμοιρασμό τριών διαφορετικών ειδών αντικειμένων τα οποία μπορεί να συναντώνται και με διαφορετική πληθικότητα.

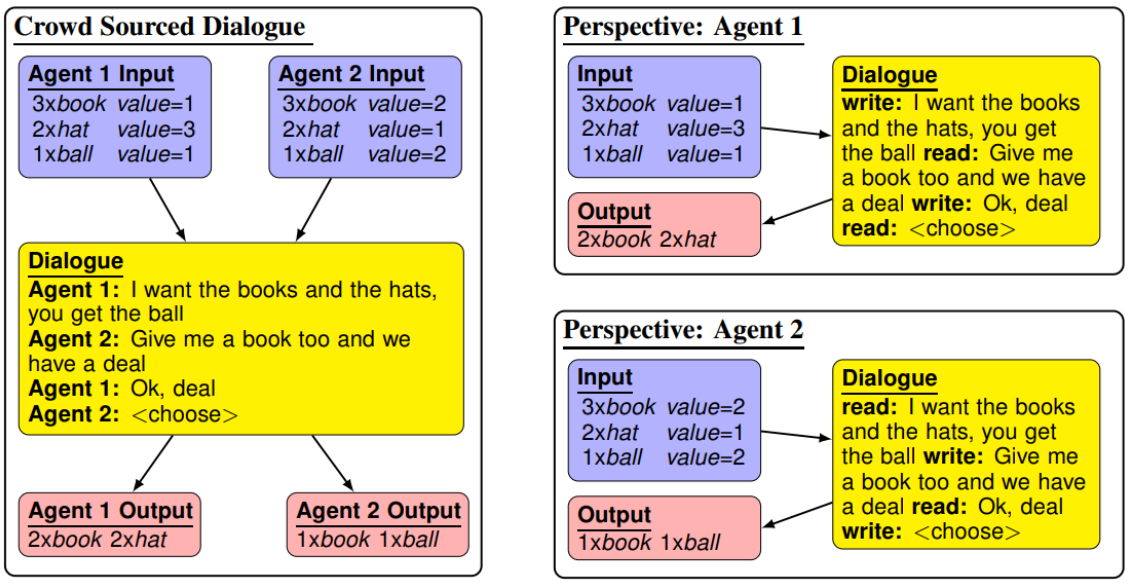
Οι διάλογοι συλλέχθηκαν χρησιμοποιώντας το Amazon Mechanical Turk και αξιοποιώντας εργαζόμενους (workers) από την Αμερική με υψηλή αξιοπιστία όπου με την επίτευξη συμφωνίας κάθε διαλόγου λάμβαναν επιπρόσθετα χρήματα προκειμένου να τους δημιουργηθεί επιπλέον κίνητρο. Συλλέχθηκαν 5808 διάλογοι που περιγράφουν 2236 διαφορετικά σενάρια (λόγω συμμετρίας προκύπτουν 5808*2=11616 διάλογοι). Με τον όρο σενάριο καλείται μια εκδοχή διαπραγμάτευσης με συγκεκριμένα διαθέσιμα αντικείμενα και συγκεκριμένες συναρτήσεις ανταμοιβών για κάθε πράκτορα. Ως σετ αξιολόγησης (test set), χρησιμοποιούνται 526 διάλογοι και συγκεκριμένα 252 σενάρια με σκοπό να ερευνηθεί αν τα μοντέλα διαπραγμάτευσης μπορούν όντως να γενικεύσουν σε νέα σενάρια (λόγω συμμετρίας προκύπτουν 526*2=1052 διάλογοι). Στον Πίνακα 7.1 παρουσιάζονται αναλυτικά όλα τα απαραίτητα στοιχεία που περιγράφουν το σύνολο των δεδομένων που έχει συλλεχθεί από το [65]. Παρόλα αυτά σε αυτό το σημείο πρέπει να τονιστεί οτι στα δεδομένα τα οποία είναι διαθέσιμα μαζί με τον κώδικα της υλοποίησης από τη Facebook [1], το training σετ αποτελείται από 10095 προτάσεις, το validation set αποτελείται από 1087 προτάσεις και το test set αποτελείται από 1052 προτάσεις.

Συγκεκριμένα, παρουσιάζονται το σύνολο των διαθέσιμων διαλόγων (# of Dialogues), το μέσο πλήθος turns ανά διάλογο (Avg. Turns), το μέσο πλήθος λέξεων ανά στιχομυθία (Avg. Words per Turn), το ποσοστό διαλόγων που οδήγησαν σε συμφωνία (Agreed(%)), η μέση βαθμολογία με μέγιστο το 10 (Avg. Score(/10)) και το ποσοστό βέλτιστων κατά Pareto διαλόγων (Pareto Optimal(%)). Ένας διάλογος μπορεί να χαρακτηριστεί ως βέλτιστος κατά Pareto εφόσον καμία αλλαγή στο διαμοιρασμό των αντικειμένων δεν μπορεί να οδηγήσει σε ικανοποίηση για όλους τους πράκτορες (σε αυτή την περίπτωση και για τους δύο πράκτορες που συνδιαλέγονται).

Πίνακας 7.1: Στατιστικά Στοιχεία για την Περιγραφή του Συνόλου των Δεδομένων του Deal Or No Deal [65]

Metric	Dataset
# of Dialogues	5808
Avg. Turns	6.6
Avg. Words per Turn	7.6
Agreed(%)	80.1%
Avg. Score(/10)	6.0
Pareto Optimal(%)	76.9%

[1] https://github.com/facebookresearch/end-to-end-negotiator/tree/690817765705b229e15b55be53a0782f65063ed7

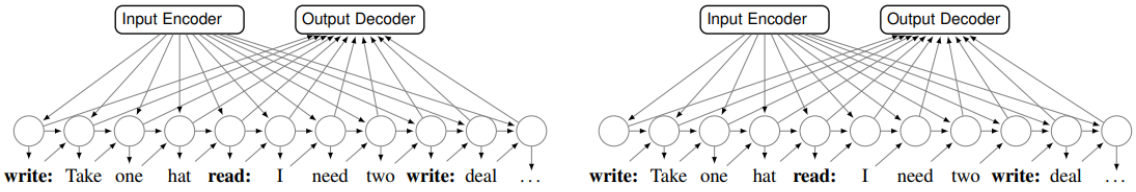


Σχήμα 7.2: Μετατροπή Διαλόγου μεταξύ Ανθρώπων στο Amazon Mechanical Turk (Αριστερά) σε Δεδομένα Εκπαίδευσης (Δεξιά) και από τις δύο οπτικές παικτών [65]

7.1.4 Baseline Αρχιτεκτονική

Το baseline μοντέλο που θα χρησιμοποιηθεί είναι αυτό που περιγράφεται από τη Facebook [65] στην ενότητα 3 της αντίστοιχης δημοσίευσης. Πρόκειται για ένα Seq2Seq μοντέλο που χρησιμοποιεί 4 ακολουθιακά μοντέλα τύπου GRUs [13]: GRU_w, GRU_g, $GRU_{\vec{o}}$ και $GRU_{\overleftarrow{o}}$ όπως θα περιγραφούν στη συνέχεια.

Κάθε παίκτης λαμβάνει ως είσοδο τον στόχο g (goal) που περιέχει τα διαθέσιμα αντικείμενα για τον εκάστοτε γύρο μαζί με τις αντίστοιχες αξίες αυτών που είναι μοναδικές και διαφορετικές από αυτές του αντιπάλου του, τις λέξεις w που απαρτίζουν τον διάλογο, και το αποτέλεσμα o της διαπραγμάτευσης.



(a) Supervised Training (b) Decoding, and Reinforcement Learning

Σχήμα 7.3: Αρχιτεκτονική Baseline Μοντέλου στην Επιβλεπόμενη Μάθηση (Αριστερά) και στην Αποκωδικοποίηση και την Ενισχυτική Μάθηση (Δεξιά) [65]

7.1.4.1 Αναπαράσταση του Στόχου

Αρχικά, κωδικοποιείται η πληροφορία σχετικά με τον αρχικό στόχο g. Υπάρχουν δύο προσεγγίσεις. Η πρώτη χρησιμοποιεί ένα αναδρομικό νευρωνικό δίκτυο GRU και η δεύτερη ένα απλό νευρωνικό δίκτυο. Όσον αφορά την πρώτη προσέγγιση, χρησιμοποιώντας ένα Gated

Recurrent Unit που συμβολίζεται ως GRU_g, λαμβάνεται η έξοδος από το τελευταίο hidden state ως η επιθυμητή αναπαράσταση h_g. Ο στόχος g, όπως προαναφέρθηκε, αποτελείται τόσο από τα διαθέσιμα αντικείμενα όσο και από τις αξίες αυτών. Σε αυτή την κωδικοποίηση δεν υφίσταται διαχωρισμός μεταξύ αντικειμένων και αξιών. Προτού εισαχθεί ο στόχος g στο εκάστοτε δίκτυο, διέρχεται πρώτα από ένα κωδικοποιητή Embedder που μετατρέπει τον στόχο αυτό σε μία αναπαράσταση. Στη συνέχεια, τα embeddings αυτά εισάγονται στο GRU_g και λαμβάνεται εν τέλει η αναπαράσταση h_g. Η αρχικοποίηση των βαρών τόσο του κωδικοποιητή όσο και αυτών του μοντέλου γίνεται με τη χρήση ομοιόμορφης κατανομής.

$$h_t^g = GRU_g(h_{t-1}^g, g_{embedded}) \tag{7.1}$$

Όσον αφορά τη δεύτερη προσέγγιση, επιδιώκεται να διαφοροποιηθούν οι τιμές του g που αφορούν το πλήθος αντικειμένων και την αξία αυτών. Έτσι, αν το διάνυσμα g είναι της παρακάτω μορφής, $g = [n_{books}, v_{books}, n_{hats}, v_{hats}, n_{balls}, v_{balls}]$, όπου με $n_{\text{αντικείμενο}}$ και $v_{\text{αντικείμενο}}$ συμβολίζεται το πλήθος και η αξία του διαθέσιμου αντικείμενου αντίστοιχα, δημιουργούνται δύο νέα διανύσματα $[n_{books}, n_{hats}, n_{balls}]$ και $[v_{books}, v_{hats}, v_{balls}]$. Κάθε ένα από αυτά διέρχεται από ένα διαφορετικό κωδικοποιητή Embedder που μετατρέπει τα διανύσματα αυτά στις κατάλληλες αναπαραστάσεις. Στη συνέχεια, πολλαπλασιάζονται μεταξύ τους στοιχείο προς στοιχείο και δημιουργούν ένα ενιαίο διάνυσμα. Τέλος, η ζητούμενη αναπαράσταση προκύπτει με την εφαρμογή της tanh() και ένα γραμμικό layer που τα μετατρέπει στο επιθυμητό μέγεθος. Ανεξαρτήτως προσέγγισης, η κωδικοποίηση του στόχου g, συμβολίζεται ως h_g σε όλες τις υπόλοιπες διαδικασίες που ακολουθούν. Η υλοποίηση που επιχειρείται από τους ερευνητές της Facebook εστιάζει στην πρώτη προσέγγιση.

7.1.4.2 Γλωσσική Μοντελοποίηση

Δεδομένων των διαλόγων και των παρελθοντικών tokens, πρέπει να δημιουργηθεί ένα μοντέλο το οποίο να έχει την ικανότητα να προβλέπει το επόμενο token x_t. Για τον σκοπό αυτό, οι ερευνητές της δημοσίευσης, σε κάθε χρονική στιγμή t, χρησιμοποιούν ένα μοντέλο GRU_w το οποίο λαμβάνει ως είσοδο το hidden state του μοντέλου h_{t-1}, την embedded αναπαράσταση του προηγούμενου token Ex_{t-1} (με E συμβολίζεται ο πίνακας των embeddings) και την αναπαράσταση h_g. Στην είσοδο, λοιπόν, δεν εισέρχεται μόνο το x_{t-1} όπως σε κάθε Γλωσσικό Μοντέλο που έχουμε αναλύσει ως τώρα στην Ενότητα 4 αλλά και το h_g προκειμένου να κωδικοποιηθεί η πληροφορία που αφορά τόσο το γλωσσικό περιεχόμενο όσο και το context του διαλόγου.

$$h_t = GRU_w(h_{t-1}, [Ex_{t-1}, h^g]) \tag{7.2}$$

Η αναπαράσταση h_t χρησιμοποιείται στη συνέχεια για την πρόβλεψη του token x_t. Συνδέουν, επιπλέον, τον πίνακα E με το h_t όπως στο [73]. Η τεχνική αυτή είναι γνωστή ως weight tying. Αντί να χρησιμοποιηθούν δύο διαφορετικοί πίνακες, ένας που να μετατρέπει την είσοδο σε embeddings και ένας που να μετατρέπει την έξοδο του μοντέλου στην είσοδο της softmax, χρησιμοποιείται ένας πίνακας. Συνήθως η μέθοδος αυτή χρησιμοποιείται για την αποφυγή του overfitting και συνεπώς αντιστοιχεί σε μια μορφή κανονικοποίησης (regularization). Τέλος, η εξίσωση 7.3 χρησιμοποιείται για την παραγωγή του επόμενου token.

$$p(x_t|x_{0...t-1}, g) \propto exp(E^\top h_t) \tag{7.3}$$

7.1.4.3 Ταξινόμηση

Εκτός από τη γλωσσική επάρκεια που θα πρέπει να διακατέχει το μοντέλο, καλούμαστε να προβλέψουμε και το αποτέλεσμα του διαλόγου, τον διαμοιρασμό, δηλαδή, των αντικειμένων στον οποίο θα συμφωνήσουν οι δύο πράκτορες. Με o αναπαρίσταται ένα σετ από tokens τα οποία αντιστοιχούν στο αποτέλεσμα της διαπραγμάτευσης. Το πρόβλημα αυτό, λοιπόν, εμπίπτει στο είδος προβλημάτων ταξινόμησης (classification). Κάθε σετ αποτελείται από 6 αντικείμενα της μορφής $[\text{books}_\text{A} = 1, \quad \text{hats}_\text{A} = 0, \quad \text{balls}_\text{A} = 1, \quad \text{books}_\text{B} = 0, \quad \text{hats}_\text{B} = 4, \quad \text{balls}_\text{B} = 0]$. Τα πρώτα 3 στοιχεία αντιστοιχούν στην επιλογή του ενός πράκτορα και τα υπόλοιπα 3 στην επιλογή του άλλου πράκτορα. Για κάθε ένα από τα 6 στοιχεία, χρησιμοποιείται και ένας διαφορετικός ταξινομητής. Στο σύνολο, λοιπόν, προτείνονται 6 ταξινομητές. Οι έξοδοι, λοιπόν, των ταξινομητών αυτών είναι ανεξάρτητοι μεταξύ τους. Μολονότι αποτελεί μια ενδιαφέρουσα ιδέα η χρήση ενός ταξινομητή για κάθε ένα αντικείμενο από τη σκοπιά και των δύο παικτών, αξίζει να σχολιαστεί πως αν ένας εκ των 6 ταξινομητών σφάλει τότε στην πράξη όλη η πρόβλεψη είναι εσφαλμένη. Με αυτό τον τρόπο ίσως είναι πιθανή η επίτευξη χαμηλού σφάλματος με ταυτόχρονα χαμηλή ακρίβεια.

Οι ταξινομητές αυτοί μοιράζονται αμφίδρομες GRUs [5] και γίνεται χρήση του Μηχανισμού Προσοχής [5]. Συγκεκριμένα, από τις Εξισώσεις 7.4 και 7.5 προκύπτουν τα διανύσματα $h_t^{\vec{o}}$ και $h_t^{\overleftarrow{o}}$ τα οποία στη συνέχεια συνενώνονται και σχηματίζουν το διάνυσμα h_t^o όπως περιγράφεται στην Εξίσωση 7.6. Σημειώνεται πως η φορά των $\vec{o}$ και $\overleftarrow{o}$ προσδιορίζει την φορά εισόδου των δεδομένων στο GRU. Στη συνέχεια, οι Εξισώσεις 7.7 και 7.8 περιγράφουν τον Μηχανισμό Προσοχής (Attention) όπου προκύπτουν τα a_t. Ακόμα, στην Εξίσωση 7.8, τα βάρη εφαρμόζονται στο διάνυσμα h_t, όπως έχει οριστεί στην Εξίσωση 7.2, και συνενώνεται το αποτέλεσμα με το διάνυσμα h^g (Εξίσωση 7.1). Στην Εξίσωση 7.9, το αποτέλεσμα μετασχηματίζεται γραμμικά διέρχεται από τη συνάρτηση ενεργοποίησης $\tanh(\cdot)$, όπου δημιουργείται το διάνυσμα h^s. Τέλος, στην Εξίσωση 7.10, προκύπτουν οι 6 ταξινομητές που συζητήσαμε για $i = \{1, 2, 3, 4, 5, 6\}$ με παραμέτρους W^{o_i} αντίστοιχα.

$$h_t^{\vec{o}} = GRU_{\vec{o}}(h_{t-1}^{\vec{o}}, [Ex_t, h_t]) \tag{7.4}$$

$$h_t^{\overleftarrow{o}} = GRU_{\overleftarrow{o}}(h_{t+1}^{\overleftarrow{o}}, [Ex_t, h_t]) \tag{7.5}$$

$$h_t^o = [h_t^{\vec{o}}, h_t^{\overleftarrow{o}}] \tag{7.6}$$

$$h_t^a = W^a[tanh(W^h h_t^o)] \tag{7.7}$$

$$a_t = \frac{exp(wh_t^a)}{\sum_{t'} exp(wh_t^a)} \tag{7.8}$$

$$h^s = tanh(W^s[h^g, \sum_t a_t h_t]) \tag{7.9}$$

$$p_\theta(_i|x_{0\ldots t-1}, g) \propto exp(W^{o_i} h^s) \tag{7.10}$$

7.1.5 Προτεινόμενη Αρχιτεκτονική

Η προτεινόμενη αρχιτεκτονική αντικαθιστά τα Αναδρομικά Νευρωνικά Δίκτυα τύπου GRU με LSTMs. Επιπλέον, στο μοντέλο εισάγονται και προεκπαιδευμένα διανύσματα λέξεων τύπου Glove αντικαθιστώντας την παραγωγή διανυσμάτων λέξεων που αρχικοποιούνται τυχαία και βελτιστοποιούνται κατά την εκμάθηση του μοντέλου. Συνεπώς, προκύπτουν 3 επιπρόσθετες υλοποιήσεις. Συνολικά, μελετώνται

- GRU και Τυχαία αρχικοποιημένα διανύσματα λέξεων (baseline)

- LSTM και Τυχαία αρχικοποιημένα διανύσματα λέξεων
- GRU και Προεκπαιδευμένα Διανύσματα Λέξεων (Glove)
- LSTM και Προεκπαιδευμένα Διανύσματα Λέξεων (Glove)

7.1.5.1 Συνάρτηση Κόστους

Η συνάρτηση κόστους απαρτίζεται από δύο μέρη. Το πρώτο αφορά το σφάλμα της πρόβλεψης του επόμενου token της ακολουθίας και η δεύτερη αφορά το σφάλμα της πρόβλεψης της έκβασης του διαλόγου. Οι συγγραφείς για να ζυγίσουν την συνεισφορά κάθε σφάλματος στο πρόβλημα χρησιμοποιούν μια υπερπαράμετρο a, όπως φαίνεται στην παρακάτω Εξίσωση. Οπότε,

$$L(\theta) = -\sum_{x,g}\sum_{t} log\ p(x_t|x_{0...t-1}, g) - \sum_{x,g,o}\sum_{j} log\ p(o_j|x_{0...T}, g) \tag{7.11}$$

Σημειώνεται οτι με T συμβολίζεται το μήκος της δοθείσας ακολουθίας από tokens. Ακόμα, σύμφωνα με την δημοσίευση [65], δηλώνεται οτι η προσέγγισή τους διαφοροποιείται από αυτή των Vinyals και Le [127] ως προς το γεγονός οτι οι παράμετροι είναι ίδιες για την εισαγωγή και την παραγωγή (weight tying).

7.1.5.2 Decoding

Κατά την αποκωδικοποίηση, το μοντέλο, δεδομένης της ιστορίας του διαλόγου και του στόχου g, καλείται να παράγει το επόμενο token x_t.

$$x_t \sim p(x_t|x_{0...t-1}, g) \tag{7.12}$$

Το μοντέλο παράγει διαδοχικά tokens έως ότου παράξει ένα ειδικό end-of-turn token που δηλώνει και το πέρας της πρότασης. Εκεί τερματίζει η παραγωγή και η απάντηση στέλνεται στον άλλο πράκτορα. Στην περίπτωση που κάποιος από τους δύο παίκτες παράξει το end-of-dialogue token τότε η διαπραγμάτευση τερματίζει. Στη συνέχεια, το μοντέλο καλείται να αποφασίσει το αποτέλεσμα του διαλόγου. Όπως αναλύθηκε προηγουμένως, το αποτέλεσμα αποτελείται από 6 στοιχεία, κάθε ένα από τα οποία προβλέπεται ανεξάρτητα από τα άλλα. Έπειτα, διερευνάται αν το αποτέλεσμα του διαλόγου είναι εφικτό, δηλαδή αν κάθε αντικείμενο έχει διαμοιραστεί σε έναν ακριβώς παίκτη.

$$o^* = \arg\max_{o \in O} \prod_{i} p(o_i|x_{0...T}, g) \tag{7.13}$$

7.1.5.3 Μετρικές

Η αξιολόγηση των Μοντέλων βασίζεται σε δύο διαφορετικές μετρικές όπου η μία αφορά την αποτίμηση της γλωσσικής επάρκειας και η άλλη την ικανότητα του μοντέλου στο πρόβλημα ταξινόμησης. Συγκεκριμένα, μετρώνται τα σφάλματα των μοντέλων για κάθε μία από τις δύο διαδικασίες. Για τη γλωσσική μοντελοποίηση, χρησιμοποιείται η μετρική της Σύγχυσης (Perplexity). Όσο μικρότερη είναι η τιμή της μετρικής αυτής τόσο καλύτερο είναι το Γλωσσικό

Μοντέλο. Για την ακρίβεια, όσο μικρότερη είναι η μετρική αυτή τόσο καλύτερα προβλέπει το μοντέλο ένα άγνωστο κείμενο που του δίνεται. Να σημειωθεί πως αυτό δεν σημαίνει απαραίτητα οτι ένα μοντέλο με μεγαλύτερο perplexity δεν μπορεί να παράγει γλώσσα, η οποία να είναι γραμματικά, συντακτικά και νοηματικά ορθή. Το μόνο συμπέρασμα που με βεβαιότητα μπορεί να εξαχθεί είναι οτι ένα μοντέλο μας με μεγαλύτερο perplexity αδυνατεί να προβλέψει μια ακολουθία του test set δεδομένης της εκπαίδευσης που έχει κάνει πάνω στο training σετ σε σύγκριση με ένα μοντέλο με χαμηλότερο perplexity. Επιπλέον, επιχειρώντας την αναπαραγωγή των αποτελεσμάτων της δημοσίευσης [65], χρησιμοποιείται και για το πρόβλημα ταξινόμησης, το σφάλμα και το perplexity. Δοκιμάζεται, επιπρόσθετα των μετρικών αυτών και η μετρική της ακρίβειας (accuracy).

7.1.6 Αποτελέσματα και Συζήτηση

Οι Πίνακες 7.2 και 7.3 συγκεντρώνουν τα αποτελέσματα της μελέτης που περιγράφηκε στις προηγούμενες υποενότητες. Σημειώνεται πως στους προαναφερθέντες πίνακες τα αποτελέσματα αναπαράχθηκαν από την παλιότερη έκδοση του κώδικα [2] η οποία εστιάζει στη δημοσίευση [65]. Η νεότερη έκδοση [3] επαυξάνει τον κώδικα και εστιάζει και στις αρχιτεκτονικές που προτείνονται στη δημοσίευση [142]. Ακόμα, η μετρική της σύγχυσης (perplexity) για το πρόβλημα της ταξινόμησης χρησιμποιείται μόνο στον κώδικα που πλαισιώνει την υλοποίηση της δημοσίευσης [65].

Πίνακας 7.2: Αξιολόγηση Baseline Μοντέλου και Προτεινόμενων Μοντέλων με χρήση LSTM και Glove Embeddings ως προς τη Γλωσσική Μοντελοποίηση

	Valid PPL	Test PPL
Baseline Likelihood (B)	5.661	5.476
B + Glove	6.065	5.844
B + LSTMs	5.648	**5.467**
B + Glove + LSTMs	6.003	5.823

Πίνακας 7.3: Αξιολόγηση Baseline Μοντέλου και Προτεινόμενων Μοντέλων με χρήση LSTM και Glove Embeddings ως προς το Πρόβλημα Ταξινόμησης

	Valid Select PPL	Test Select PPL
Baseline Likelihood (B)	1.133	**1.155**
B + Glove	1.165	1.332
B + LSTMs	1.128	1.280
B + Glove + LSTMs	1.229	1.350

Παρατηρούμε πως οι διαφορές των μοντέλων είναι ανεπαίσθητες. Όσον αφορά τη γλωσσική μοντελοποίηση, στο test σετ υπερτερεί το LSTM με Τυχαία Αρχικοποιημένα αλλά Εκπαιδευόμενα Διανύσματα Λέξεων ενώ στο Πρόβλημα της Ταξινόμησης υπερτερεί το Baseline Μοντέλο. Επιπλέον, όσον αφορά το πρόβλημα ταξινόμησης, χρησιμοποιήθηκε και η μετρική της ακρίβειας όπως αναφέρθηκε στο προηγούμενη υποενότητα. Το Baseline μοντέλο με το αρκετά χαμηλό loss και κατ'επέκταση perplexity εμφανίζει μηδενική ακρίβεια (accuracy). Αυτό

[2] https://github.com/facebookresearch/end-to-end-negotiator/tree/690817765705b229e15b55be53a0782f65063ed7
[3] https://github.com/facebookresearch/end-to-end-negotiator

μπορεί να οφείλεται στην αρχιτεκτονική που χρησιμοποιεί 6 ταξινομητές που αντιστοιχούν ένας για κάθε αντικείμενο κάθε παίκτη. Αν το μοντέλο προβλέψει έστω και μία τιμή λάθος, τότε από άποψη σφάλματος, αυτό μπορεί να είναι μειωμένο αλλά από άποψη ακρίβειας (accuracy), η πρόβλεψη θεωρείται λανθασμένη.

Στην πράξη, μια λάθος πρόβλεψη θα οδηγήσει σε λανθασμένα συμπεράσματα ενός πράκτορα για την πορεία του διαλόγου. Ιδιαίτερα αν ο ταξινομητής δεν καταφέρνει να προβλέψει καμία έκβαση σωστά, τότε όλη η λειτουργία του κρίνεται μη ωφέλιμη επί της ουσίας. Για το λόγο αυτό, διαχωρίζονται τα προβλήματα γλωσσικής μοντελοποίησης και ταξινόμησης με τέτοιο τρόπο ώστε να προβεί κανείς σε βελτιστοποιήσεις και στα δύο προβλήματα τα οποία είναι σημαντικά για την επιτυχία της διαπραγμάτευσης. Σε αυτά προστίθεται και αυτό τις λήψης απόφασης και της εξαγωγής στρατηγικών που επιχειρείται σε ένα αποδομημένο περιβάλλον αποκομμένο από γλωσσικές εξαρτήσεις (Κεφάλαιο 8).

7.2 Το Πρόβλημα της Ταξινόμησης (Classification Task)

Σημαντικό ρόλο στα προβλήματα διαπραγματεύσεων έχει η δυνατότητα να μπορεί ένας πράκτορας, δεδομένου ενός διαλόγου, να προβλέπει σε τι διαμοιρασμό αντικειμένων αποσκοπεί ο διάλογος αυτός. Απαιτείται, λοιπόν, η δημιουργία ενός μοντέλου το οποίο θα αντιστοιχεί ένα διάλογο, ο οποίος εισάγεται στο μοντέλο σε μορφή κειμένου, σε μία κατηγορία που ανήκει σε ένα πεπερασμένο σύνολο πιθανών διαμοιρασμών των αντικειμένων. Το ζητούμενο μοντέλο μπορεί να εκπληρώσει δύο σκοπούς. Ο πρώτος αφορά την πρόβλεψη της συμφωνίας μεταξύ δύο πρακτόρων που λαμβάνει χώρα στο τέλος κάθε διαλόγου και ο δεύτερος αφορά την πρόβλεψη της πρότασης διαμοιρασμού αντικειμένου ύστερα από την απάντηση του άλλου πράκτορα. Είναι εμφανές πως ο δεύτερος σκοπός του μοντέλου αυτού είναι αναγκαίος σε κάθε βήμα του διαλόγου, προκειμένου να εκληφθεί επιτυχώς η κατάσταση και η πορεία της διαπραγμάτευσης.

Η υλοποίηση που παρουσιάζεται στην προηγούμενη Ενότητα, μολονότι επιτυγχάνει ένα χαμηλό σφάλμα (loss) και συνεπώς χαμηλή σύγχυση (perplexity) τόσο στο validation όσο και στο test σετ, έχει πολύ χαμηλή ακρίβεια (accuracy) ως προς την πρόβλεψη της κατηγορίας στην οποία αντιστοιχεί ο εκάστοτε διάλογος. Αυτό οφείλεται και στο γεγονός οτι αν ένα εκ των έξι νευρωνικών δικτύων που χρησιμοποιούνται για την πρόβλεψη της πληθικότητας κάθε αντικειμένου από την σκοπιά κάθε παίκτη, σφάλει, τότε όλη η πρόβλεψη θεωρείται εσφαλμένη. Αν και ενδιαφέρουσα η οπτική της χρησιμοποίησης έξι δικτύων, ένα για κάθε αντικείμενο κάθε παίκτη, σκοπός της εργασίας είναι ο πράκτορας να αντιλαμβάνεται με επιτυχία την κατάσταση της διαπραγμάτευσης. Για το λόγο αυτό, η εστίαση θα γίνει στην επίτευξη της υψηλότερης δυνατής ακρίβειας προκειμένου το μοντέλο να είναι λειτουργικό στην πράξη και επί της ουσίας.

Παρουσιάζοντας αρκετές υλοποιήσεις κλασικών Ταξινομητών που συναντώνται σε πληθώρα εφαρμογών, επιδιώκεται να αναδειχθεί η υπεροχή του BERT (Transformer) και της Μεταφοράς Μάθησης (Transfer Learning). Ο Ταξινομητής που χρησιμοποιεί το BERT επιτυγχάνει εξαιρετικά καλύτερα αποτελέσματα σε σύγκριση με κλασικές μεθόδους, με μικρές τροποποιήσεις και σε ελάχιστο χρόνο βελτιστοποίησης στο συγκεκριμένο πρόβλημα. Επιπλέον, ο Ταξινομητής που χρησιμοποιεί το BERT με ελάχιστες τροποποιήσεις μπορεί να εστιάσει και σε διαφορετικά προβλήματα ταξινόμησης σε αντίθεση με τις κλασικές μεθόδους επιδεικνύοντας την σπουδαιότητα της Μεταφοράς Μάθησης (Transfer Learning).

7.2.1 Περιγραφή Προβλήματος Ταξινόμησης

Το πρόβλημα ταξινόμησης, όπως προαναφέρθηκε, αφορά τη δημιουργία ενός μοντέλου το οποίο αντιστοιχεί ένα κείμενο, στην παρούσα εργασία έναν διάλογο, σε μια κατηγορία που ανήκει σε ένα πεπερασμένο σύνολο δυνατών διαμοιρασμών αντικειμένων. Συγκεκριμένα, ο μέγιστος αριθμός πληθικότητας κάθε αντικειμένου που παρατηρείται στα δεδομένα της Facebook [65] είναι 4. Επειδή κανείς μπορεί να μην λάβει και κανένα αντικείμενο, μπορούν να υπάρξουν $(4+1)\times(4+1)\times(4+1) = 125$ πιθανές εκβάσεις (το $+1$ αντιστοιχεί στην περίπτωση 0 αντικειμένων). Σε αυτές τις πιθανές εκβάσεις προστίθενται άλλες τρεις εκβάσεις που αντιστοιχούν στη "διαφωνία", την "αποσύνδεση" και την "ασυμφωνία". Συνολικά, λοιπόν, προκύπτουν 128 πιθανές εκβάσεις και συνεπώς 128 κλάσεις διαμοιρασμών αντικειμένων. Να σημειωθεί οτι ενώ μπορεί να είναι διαθέσιμα λιγότερα αντικείμενα σε ένα παιχνίδι διαπραγμάτευσης (λ.χ. "2 βιβλία, 1 καπέλο και 1 μπάλα") το μοντέλο θα πρέπει να διαλέξει την κατάλληλη κατηγορία διαμοιρασμού αντικειμένων ανάμεσα και στις 128 πιθανές κατηγορίες και όχι ανάμεσα στις $(2+1)\times(1+1)\times(1+1)+3 = 15$ πιθανές εκβάσεις. Το πρόβλημα γίνεται ακόμη πιο δύσκολο καθώς αγνοείται στις υλοποιήσεις το πλήθος των διαθέσιμων αντικειμένων και η αντίστοιχη αξία αυτών για κάθε παίκτη. Η είσοδος του μοντέλου, λοιπόν, είναι ολόκληρος ο διάλογος μεταξύ των δύο παικτών και η ζητούμενη έξοδος είναι ένας δείκτης που αντιστοιχεί σε κάποια από τις 128 πιθανές εκβάσεις του διαλόγου.

7.2.2 Δεδομένα

Τα δεδομένα που χρησιμοποιήθηκαν είναι τα ίδια που χρησιμοποιήθηκαν και στις προηγούμενες υλοποιήσεις. Περαιτέρω λεπτομέρειες αυτών μπορούν να αναζητηθούν στην Ενότητα 7.1.3.

7.2.3 Δημιουργία Λεκτικών Μονάδων (Tokenization)

Τα δεδομένα κειμένων θα πρέπει να έρθουν σε κατάλληλη μορφή προκειμένου να είναι αξιοποιήσιμα από τα μοντέλα που χρησιμοποιούνται στη συνέχεια. Η πρώτη ενέργεια αφορά τη δημιουργία λεκτικών μονάδων (tokenization). Για όλα τα μοντέλα έχει χρησιμοποιηθεί ένας απλός Μετατροπέας κειμένων σε λεκτικές μονάδες (Tokenizer). Ο απλός αυτός Μετατροπέας χωρίζει τις λέξεις ανάλογα με τα κενά (όχι σε επίπεδο χαρακτήρων) και μετατρέπει λέξεις που περιέχουν κεφαλαία γράμματα σε μικρά. Στην περίπτωση, που εμφανιστεί λέξη που δεν ανήκει στο λεξικό χρησιμοποιείται ένα ειδικό σύμβολο που δηλώνει οτι η λέξη είναι εκτος λεξικού. Εξαίρεση αποτελεί η περίπτωση του BERT, στην οποία χρησιμοποιείται συγκεκριμένος Tokenizer, όπως θα αναφερθεί στη συνέχεια. Για τη σύγκριση των αποτελεσμάτων, να σημειωθεί πως ο Tokenizer των μοντέλων πλην του BERT είναι πιο απλός και το λεξικό του αρκετά μικρότερο από αυτό του BERT.

7.2.4 Περιγραφή Μοντέλων

Αρχικά, πραγματοποιείται μια πρώτη σύγκριση μεταξύ των παραδοσιακών μεθόδων ταξινόμησης. Συγκεκριμένα χρησιμοποιήθηκε ένα Απλό Αναδρομικό Νευρωνικό Δίκτυο (Ενότητα 4.3) και στη συνέχεια υλοποιήθηκαν ταξινομητές που χρησιμοποιούσαν Gated Recurrent Units (GRUs, Ενότητα 4.4), Αμφίδρομο Long Short Term Memory (LSTM, Ενότητα 4.6)

και Convolutional Neural Networks (CNNs, Ενότητα 2.4.1). Όλα τα μοντέλα χρησιμοποιούν τον ίδιο Tokenizer που περιγράφτηκε στην Ενότητα 7.2.3.

Ως προς τα διανύσματα λέξεων, όλα τα μοντέλα πραγματοποιούν εκμάθηση των διανυσμάτων λέξεων, τα οποία αρχικοποιούνται με τυχαίες τιμές. Για να επιτευχθεί και η σύγκριση με προ-εκπαιδευμένα διανύσματα λέξεων, χρησιμοποιήθηκαν τα Διανύσματα Λέξεων (Glove Embeddings, 3.5). Τα Glove Embeddings μπορούν να χρησιμοποιηθούν αμετάβλητα κατά τη διαδικασία της εκπαίδευσης (frozen parameters) αλλά μπορούν και να βελτιστοποιούν τις παραμέτρους τους (unfrozen parameters). Τα Glove Embeddings χρησιμοποιούνται στους ταξινομητές που περιέχουν το μοντέλο CNN.

7.2.4.1 Απλό Αναδρομικό Νευρωνικό Δίκτυο (Simple RNN)

Το μοντέλο που χρησιμοποιείται σαν μοντέλο αναφοράς είναι ένα Απλό Νευρωνικό Δίκτυο όπως περιγράφηκε στην Ενότητα 4.3. Η έξοδος που αντιστοιχεί στη τελευταία λέξη της ακολουθίας, που εισάγεται στο μοντέλο, διέρχεται από ένα νευρωνικό δίκτυο, από το οποίο εν τέλει προκύπτει το αποτέλεσμα της Ταξινόμησης. Όσον αφορά τις υπερπαραμέτρους της υλοποίησης, το μέγεθος των διαστάσεων του Αναδρομικού Νευρωνικού Δικτύου ανέρχεται στο 128 και τα διανύσματα λέξεων έχουν μήκος 100. Επιπλέον, χρησιμοποιείται πιθανότητα dropout ίση με 0.1 και το Νευρωνικό Δίκτυο που χρησιμοποιείται στην έξοδο του Απλού Αναδρομικού Νευρωνικού Δικτύου αποτελείται από ένα κρυφό στρώμα 100 νευρώνων. Το μέγεθος του batch που χρησιμοποιήθηκε είναι 128 και επιπλέον εφαρμόστηκε και η τεχνική της Πρόωρης Διακοπής (Early Stopping) με υπομονή (patience) ίση με 3. Ο μέγιστος αριθμός εποχών είναι 30. Ο ρυθμός μάθησης learning rate αρχικά τίθεται στην τιμή 0.001 και σταδιακά μειώνεται με βάση το ReduceLROnPlateau. Ο Βελτιστοποιητής (Optimizer) που χρησιμοποιείται είναι ο Adam Optimizer και για την μέτρηση του σφάλματος χρησιμοποιείται το Cross-Entropy Loss.

7.2.4.2 Αναδρομικό Νευρωνικό Δίκτυο με χρήση Gated Recurrent Units (GRUs)

Στη συνέχεια, χρησιμοποιούνται οι ίδιοι παράμετροι με την μόνη διαφορά οτι αντικαθίσταται το Απλό Αναδρομικό Νευρωνικό Δίκτυο με ένα Gated Recurrent Unit όπως περιγράφηκε στην Ενότητα 4.4. Η έξοδος που αντιστοιχεί στη τελευταία λέξη της ακολουθίας, που εισάγεται στο μοντέλο, διέρχεται από ένα νευρωνικό δίκτυο, από το οποίο εν τέλει προκύπτει το αποτέλεσμα της Ταξινόμησης. Όσον αφορά τις υπερπαραμέτρους της υλοποίησης, το μέγεθος των διαστάσεων του GRU ανέρχεται στο 128 και τα διανύσματα λέξεων έχουν μήκος 100. Επιπλέον, χρησιμοποιείται πιθανότητα dropout ίση με 0.1 και το Νευρωνικό Δίκτυο που χρησιμοποιείται στην έξοδο του GRU αποτελείται από ένα κρυφό στρώμα 100 νευρώνων. Το μέγεθος του batch, η παράμετρος patience της Πρόωρης Διακοπής (Early Stopping), ο μέγιστος αριθμός εποχών, το learning rate, ο βελτιστοποιητής (optimizer) και η Συνάρτηση Κόστους είναι ίδια με την υλοποίηση του προηγούμενου μοντέλου 7.2.4.1.

7.2.4.3 Αμφίδρομο Long Short Term Memory

Στη συνέχεια, χρησιμοποιούνται οι ίδιοι παράμετροι με την μόνη διαφορά οτι αντικαθίσταται το Απλό Αναδρομικό Νευρωνικό Δίκτυο με ένα αμφίδρομο LSTM όπως περιγράφηκε στην Ενότητα 4.6. Η έξοδος που αντιστοιχεί στη τελευταία λέξη της ακολουθίας, που εισάγεται

στο μοντέλο, διέρχεται από ένα νευρωνικό δίκτυο, από το οποίο εν τέλει προκύπτει το αποτέλεσμα της Ταξινόμησης. Όσον αφορά τις υπερπαραμέτρους της υλοποίησης, το μέγεθος των διαστάσεων του LSTM ανέρχεται στο 128 και τα διανύσματα λέξεων έχουν μήκος 100. Επιπλέον, χρησιμοποιείται πιθανότητα dropout ίση με 0.1 και το Νευρωνικό Δίκτυο που χρησιμοποιείται στην έξοδο του LSTM αποτελείται από ένα κρυφό στρώμα 100 νευρώνων. Ομοίως, όπως και στην περίπτωση του 7.2.4.2, το μέγεθος του batch, η παράμετρος patience της Πρόωρης Διακοπής (Early Stopping), ο μέγιστος αριθμός εποχών, το learning rate, ο βελτιστοποιητής (optimizer) και η Συνάρτηση Κόστους είναι ίδια με την υλοποίηση του μοντέλου 7.2.4.1. Ακόμα, για τη συγκεκριμένη υλοποίηση χρησιμοποιήθηκαν δύο στρώματα στο μοντέλο του LSTM αντί για 1.

7.2.4.4 Convolutional Neural Network

Το μοντέλο που χρησιμοποιείται έπειτα δεν ανήκει στην κατηγορία των Αναδρομικών Νευρωνικών Δικτύων, αλλά χρησιμοποιείται σε πληθώρα εφαρμογών Μηχανικής Όρασης. Το Συνελικτικό Νευρωνικό Δίκτυο (Convolutional Neural Network, CNN, 2.4.1) χρησιμοποιεί φίλτρα μεγέθους 2,3 και 4 και stride ίσο με 1. Η έξοδος που αντιστοιχεί στη τελευταία λέξη της ακολουθίας, που εισάγεται στο μοντέλο, διέρχεται από ένα νευρωνικό δίκτυο, από το οποίο εν τέλει προκύπτει το αποτέλεσμα της Ταξινόμησης. Όσον αφορά τις υπερπαραμέτρους της υλοποίησης, τα διανύσματα λέξεων έχουν μήκος 100. Επιπλέον, χρησιμοποιείται πιθανότητα dropout ίση με 0.1 και το Νευρωνικό Δίκτυο που χρησιμοποιείται στην έξοδο του CNN αποτελείται από ένα κρυφό στρώμα 100 νευρώνων. Το μέγεθος του batch που χρησιμοποιήθηκε είναι 64 και ομοίως η παράμετρος patience της Πρόωρης Διακοπής (Early Stopping), ο μέγιστος αριθμός εποχών, το learning rate, ο βελτιστοποιητής (optimizer) και η Συνάρτηση Κόστους είναι ίδια με τις προηγούμενες υλοποιήσεις. Στη συνέχεια, στο μοντέλο αυτό δοκιμάζεται και η χρήση προ-εκπαιδευμένων διανυσμάτων λέξεων Glove 100 διαστάσεων. Επίσης, τα διανύσματα αυτά ανάλογα με το εκάστοτε ζητούμενο μπορεί να είναι παγωμένα (frozen) ή μη-παγωμένα (unfrozen) που σημαίνει οτι μπορεί κατά τη διάρκεια της εκμάθησης να παραμένουν αμετάβλητα ή να βελτιστοποιούνται αντίστοιχα.

7.2.4.5 BERT

Δημιουργείται ένας Ταξινομητής, ο οποίος έχει ως βάση τον Transformer BERT όπως περιγράφηκε στην Ενότητα 4.7.4. Στην παρούσα υλοποίηση έγινε χρήση του μοντέλου "Bert For Sequence Classification" [44]. Το μοντέλο αυτό περιέχει έναν προ-εκπαιδευμένο Transformer BERT και ένα νευρωνικό δίκτυο που λαμβάνει την έξοδο του Transformer και λειτουργεί ως Ταξινομητής. Το μοντέλο αυτό, όπως και πληθώρα μοντέλων, έχουν παραχωρηθεί δημόσια από την HuggingFace τόσο σε PyTorch [42] όσο και σε Tensorflow. Σχετικά με τη δημιουργία λεκτικών μονάδων (tokenization) γίνεται χρήση του BertTokenizer [43]. Ειδικότερα, η πρόταση τμηματοποιείται σε συγκεκριμένη μορφή (WordPieces, [139]).

7.2.5 Μετρικές

Όπως έχει ήδη επισημανθεί, στόχος μας είναι η σωστή πρόβλεψη της έκβασης του διαλόγου. Για το λόγο αυτό χρησιμοποιείται η ακρίβεια (accuracy) ως μετρική για το συγκεκριμένο πρόβλημα.

7.2.6 Αποτελέσματα

Σε αυτή την υποενότητα παρατίθενται και αναλύονται τα αποτελέσματα των υλοποιήσεων που περιγράφηκαν στις προηγούμενες ενότητες. Επιδιώκεται να γίνει μια σύγκριση μεταξύ παραδοσιακών μοντέλων χωρίς προ-εκπαιδευμένα διανύσματα λέξεων. Ακόμα, παρατίθενται τα αποτελέσματα μετά τη χρήση προ-εκπαιδευμένων διανυσμάτων λέξεων Glove, τα οποία είτε παραμένουν αμετάβλητα (frozen), είτε βελτιστοποιούνται περαιτέρω (unfrozen). Τέλος, παρατίθενται τα αποτελέσματα με τη χρήση Μεταφοράς Μάθησης (Transfer Learning) και συγκεκριμένα με τη χρήση του Transformer BERT. Τα αποτελέσματα είναι συγκεντρωμένα στον Πίνακα 7.4. Σημειώνεται οτι το G συμβολίζει τα Glove Embeddings και τα F και U αντιστοιχούν στις λέξεις Frozen και Unfrozen.

Πίνακας 7.4: Αποτελέσματα Σφάλματος και Ακρίβειας για το Πρόβλημα της Ταξινόμησης των τριών Αναδρομικών Νευρωνικών Δικτύων (Απλό RNN, GRU και LSTM) και του CNN χωρίς Προ-εκπαιδευμένα Διανύσματα Λέξεων

	Valid Loss	Valid Accuracy (%)	Test Loss	Test Accuracy (%)
Simple RNN	4.37	10.0	4.15	9.39
GRU	3.64	26.4	3.51	26.91
bi-LSTM	4.03	22.1	3.31	18.34
CNN	3.48	23.7	3.53	22.6
CNN + G (F)	3.47	23.0	3.26	19.08
CNN + G (U)	3.51	23.4	3.18	20.94
BERT	**2.12**	**42**	**2.14**	**42**

Παρατηρείται οτι τα πιο σύνθετα Αναδρομικά Νευρωνικά Δίκτυα και το CNN μοντέλο επιτυγχάνουν καλύτερα αποτελέσματα από το Απλό αναδρομικό Νευρωνικό Δίκτυο. Επιπλέον, στο test σετ το GRU μοντέλο επιτυγχάνει τα υψηλότερα αποτελέσματα ακρίβειας.

Από τον Πίνακα 7.4 συμπεραίνουμε ακόμα πως η χρήση των Unfrozen Glove Embeddings προσφέρει καλύτερα αποτελέσματα από τα Frozen Glove Embeddings. Αυτό, ίσως, οφείλεται στο γεγονός οτι με την βελτιστοποίηση των παραμέτρων των προ-εκπαιδευμένων διανυσμάτων λέξεων, αυτά τα διανύσματα λέξεων εξειδικεύονται για το συγκεκριμένο σετ δεδομένων και πρόβλημα. Παρόλα αυτά η χρήση των Glove Embeddings για το CNN μοντέλο δίνει χειρότερα αποτελέσματα από την περίπτωση που τα διανύσματα λέξεων αρχικοποιούνται τυχαία και βελτιστοποιούνται κατά τη διάρκεια της εκμάθησης. Αυτό μπορεί να οφείλεται στο γεγονός οτι οι παράμετροι των Embeddings του CNN βελτιστοποιούνται σε σχέση με το συγκεκριμένο πρόβλημα σε ικανοποιητικό βαθμό. Όσον αφορά, τα κλασικά μοντέλα μηχανικής μάθησης (αναδρομικά δίκτυα και CNN μοντέλο), το δομικό στοιχείο GRU δίνει τα καλύτερα αποτελέσματα στο συγκεκριμένο πρόβλημα.

Τα καλύτερα αποτελέσματα προκύπτουν με τη χρήση Μεταφοράς Μάθησης και συγκεκριμένα του Ταξινομητή BERT.

7.3 Το Πρόβλημα της Γλωσσικής Μοντελοποίησης (Language Modeling)

Το πρόβλημα της Γλωσσικής Μοντελοποίησης έχει περιγραφεί λεπτομερώς στο Κεφάλαιο 4. Στην ενότητα αυτή, θα αναδειχθεί, όμως, και το ζήτημα της παραγωγής κειμένου (text generation). Συγκεκριμένα, θα παρουσιαστούν οι παραδοσιακές μορφές (Decoding) και οι πιο σύγχρονες εκδοχές που βασίζονται στη δειγματοληψία. Εν τέλει, ο στόχος είναι να αναδειχθεί η ανωτερότητα του GPT-2 έναντι παραδοσιακών μοντέλων γλωσσικής μοντελοποίησης και στη συνέχεια να γίνει χρήση του GPT-2 για την παραγωγή διαλόγων. Αυτό το πρόβλημα είναι αρκετά σημαντικό και η παραγωγή κειμένου αποδεσμεύεται από όλα τα υποπροβλήματα που καλείται κανείς να λύσει.

7.3.1 Περιγραφή Μοντέλων

Τα μοντέλα που θα χρησιμοποιηθούν είναι αυτά που περιγράφηκαν στις Ενότητες 7.1.4 και 7.1.5 και αφορούν την γλωσσική μοντελοποίηση. Πέρα από το baseline μοντέλο που προτείνεται από τη Facebook [65], χρησιμοποιούνται και οι τροποποιήσεις αυτού που κάνουν χρήση των LSTMs και των Glove Διανυσμάτων Λέξεων. Η ενότητα, όμως, αυτή αποσκοπεί να αναδείξει την υπεροχή της Μεταφοράς Μάθησης (Transfer Learning) και συγκεκριμένα του Transformer GPT-2 [99].

7.3.1.1 GPT-2

Δημιουργείται ένα γλωσσικό μοντέλο το οποίο έχει ως βάση τον Transformer GPT-2 όπως περιγράφηκε στην Ενότητα 4.7.5. Στην παρούσα υλοποίηση έγινε χρήση του μοντέλου "GPT2 LMHeadModel" που παραχωρείται από την HuggingFace [42]. Πρόκειται για ένα GPT-2 μοντέλο στο οποίο έχει προστεθεί ένα γραμμικό στρώμα με βάρη τα οποία είναι συσχετισμένα με τις διανυσματικές αναπαραστάσεις της εισόδου. Όσον αφορά τη δημιουργία λεκτικών μονάδων (tokenization) γίνεται χρήση του GPT2Tokenizer που παραχωρείται δημόσια μαζί με το μοντέλο και βασίζεται στο Byte-level Byte-Pair-Encoding [107].

7.3.2 Μετρικές

Η μετρική που χρησιμοποιείται για την αξιολόγηση των υπό μελέτη μοντέλων είναι η σύγχυση (perplexity).

7.3.3 Αποτελέσματα

Στον Πίνακα 7.5 μαζί με τα αποτελέσματα του Πίνακα 7.2 παρουσιάζονται και τα αποτελέσματα που προκύπτουν από την χρήση του GPT-2. Είναι εμφανές πως με μικρές τροποποιήσεις του προεκπαιδευμένου μοντέλου GPT-2 μπορούν να επιτευχθούν εμφανώς καλύτερα αποτελέσματα σε λιγότερο χρόνο και μικρότερο κόστος από ό,τι θα χρειαζόταν να γίνει εκπαίδευση εξ αρχής. Η σπουδαιότητα της Μεταφοράς Μάθησης παρατηρείται και σε αυτό το πρόβλημα. Λόγω των πολύ καλών αποτελεσμάτων του GPT-2 συνιστάται περαιτέρω μελέτη του εν λόγω ζητήματος.

Πίνακας 7.5: Αξιολόγηση Μοντέλων

	Valid PPL	Test PPL
Baseline Likelihood (B)	5.661	5.476
B + Glove	6.065	5.844
B + LSTMs	5.648	5.467
B + Glove + LSTMs	6.003	5.823
GPT-2	**1.417**	**1.556**

7.3.4 Μελλοντική Χρήση - Παραγωγή Γλώσσας (Text Generation)

Εξαιρετική σημασία έχει η Παραγωγή Κειμένου. Από τη στιγμή που αποδεικνύεται η ανωτερότητα του GPT-2 αναλύονται κάποιες τεχνικές παραγωγής κειμένου και παρουσιάζονται συγκεκριμένα παραδείγματα παραγωγής προτάσεων από το μοντέλο GPT-2 για ποιοτική αξιολόγηση και βαθύτερη κατανόηση των τεχνικών αυτών. Συγκεκριμένα, χρησιμοποιήθηκε ο κώδικας που παράχθηκε από την HuggingFace [45] εστιασμένος στα δεδομένα του προβλήματος που καλούμαστε να λύσουμε. Για λόγους πληρότητας, το perplexity πάνω στο σύνολο δεδομένων αξιολόγησης (validation set) ύστερα από την εκμάθηση του μοντέλου ανέρχεται στην τιμή 1.977 που σημαίνει οτι το μοντέλο αυτό έχει μάθει σε εξαιρετικό βαθμό να προβλέπει τα κείμενα του συνόλου δεδομένων και κατ'επέκταση αναμένεται κατά τη διαδικασία του Generation να προκύπτουν και προτάσεις που μοιάζουν σε σημαντικό βαθμό, όπως άλλωστε αποδεικνύεται και στη συνέχεια, με αυτές του συνόλου δεδομένων του προβλήματος. Συγκεκριμένα, χρησιμοποιοποιήθηκε το GPT-2 μοντέλο. Παρόλα αυτά θα μπορούσε να χρησιμοποιηθεί οποιοδήποτε auto-regressive μοντέλο XLNet [141], OpenAi-GPT [98], CTRL [53], Transformer-XL [19], XLM [57], Bart [64] και T5 [100]. Τα μοντέλα αυτά βασίζονται στην υπόθεση οτι η πιθανότητα μιας ακολουθίας λέξεων μπορεί να αποδομηθεί σε γινόμενο δεσμευμένων πιθανοτήτων λέξεων δεδομένων των προηγούμενων.

$$P(w_{1:T}|W_0) = \prod_{t=1}^{T} P(w_t|w_{1:t-1}, W_0) \tag{7.14}$$

όπου το W_0 αντιστοιχεί στο context του διαλόγου και $w_{1:0} = \emptyset$.

Οι μέθοδοι που θα αναλυθούν είναι η Άπληστη Αποκωδικοποίηση (Greedy Decoding), Beam Search και εν συνεχεία δύο μέθοδοι δειγματοληψίας (sampling), η Top-K [26] και η Top-p δειγματοληψία [39]. Να σημειωθεί οτι σε κάθε ένα από τα παραδείγματα χρησιμοποιείται ως context η πρόταση

> YOU: i want the hat and the ball

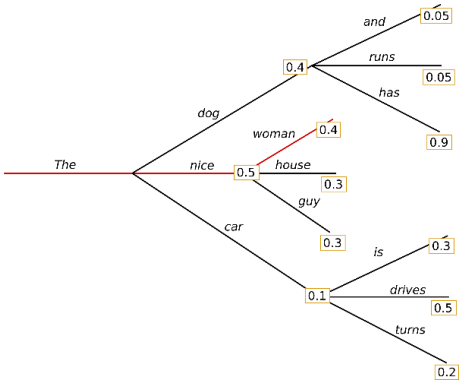
και επιπλέον το μοντέλο, ανεξαρτήτως τεχνικής για Generation, παράγει κείμενο μία λέξη τη φορά έως ότου παράξει κάποια λέξη, η οποία αντιστοιχεί σε ένα token τερματισμού (stop token). Στα παραδείγματα που ακολουθούν χρησιμοποιείται και μία άλλη συνθήκη τερματισμού. Το μοντέλο σταματάει την παραγωγή κειμένου όταν συμπληρώσει τα 50 tokens. Σε αυτά προσμετρώνται και τα tokens της εισόδου.

7.3.5 Greedy Decoding

Η πιο εύκολη μέθοδος παραγωγής κειμένου από auto-regressive μοντέλα είναι η άπληστη αποκωδικοποίηση (greedy decoding). Δεδομένου του context και των προηγούμενων λέξεων που μπορεί να παρήχθησαν, επιλέγεται σε κάθε χρονική στιγμή η λέξη που φέρει τη μεγαλύτερη πιθανότητα να εμφανιστεί. Μαθηματικά, αυτή η επιλογή εκφράζεται ως $w_t = argmax_w P(w|w_{1:t-1})$. Εισάγοντας, λοιπόν, ως context στο μοντέλο τη φράση "YOU: i want the hat and the ball" το μοντέλο παράγει ένα διάλογο έως ότου συναντήσει μια λέξη τερματισμού ή όταν έχει παράξει έναν συγκεκριμένο αριθμό από tokens που έχει εισαχθεί στο μοντέλο ως συνθήκη τερματισμού. Στο συγκεκριμένο παράδειγμα, το μοντέλο μπορεί να παράξει μέχρι 50 tokens, στα οποία προσμετρώνται και τα tokens του context. Για αυτό ενώ παράγεται ένα αποτέλεσμα που θυμίζει κανονικό διάλογο, διακόπτεται απότομα. Το αποτέλεσμα που παράγεται είναι ο διάλογος

> YOU: i want the hat and the ball <eos>
> THEM: i need the hat and the ball <eos>
> YOU: i can not make that deal. i need the hat and the ball <eos>
> THEM: i can

Το πρόβλημα που συνήθως παρατηρείται με το Greedy Decoding όπως και με την Beam Search είναι πως το μοντέλο από ένα σημείο και έπειτα επαναλαμβάνει τις ίδιες εκφράσεις [126] [112]. Αυτό το ζήτημα επιδιώκεται να λυθεί με τις τεχνικές δειγματοληψίας. Το Greedy Decoding είναι μια απλή και εύκολα υλοποιήσιμη μέθοδος αφού επιλέγει την πιο πιθανή λέξη δεδομένης μιας προϋπάρχουσας ακολουθίας. Ελέγχει, δηλαδή, μόνο τις πιθανότητες για τις αμέσως επόμενες λέξεις. Το πρόβλημα έγκειται στο γεγονός οτι πίσω από μια φαινομενικά χαμηλής πιθανότητας λέξη μπορεί να κρύβεται μια λέξη με πολύ μεγάλη πιθανότητα και θεωρητικά το πιο πιθανό μονοπάτι θα παραλειφθεί από το Greedy Decoding. Προς ενίσχυση αυτής της υπόθεσης, παρατίθεται το παρακάτω παράδειγμα, όπως απεικονίζεται στο σχήμα. Το μοντέλο, με τη χρήση Greedy Decoding, θα επιλέξει τη λέξη "nice" και έπειτα τη λέξη "woman". Η πιθανότητα αυτής της ακολουθίας είναι $0.5 \cdot 0.4 = 0.2$. Υπάρχει, όμως, και η ακολουθία "dog"-"has" με πιθανότητα $0.4 \cdot 0.9 = 0.36$ που θα αγνοηθεί πλήρως.



Σχήμα 7.4: Greedy Decoding [41]

7.3.6 Beam Search

Το πρόβλημα αυτό αποσκοπεί να λύσει η Beam Search. Αναλύοντας ταυτόχρονα και άλλα μονοπάτια ελαττώνει την πιθανότητα να αγνοηθούν τα πιθανά μονοπάτια που αγνοεί το Greedy Decoding. Το πλήθος των μονοπατιών αυτών αποτελεί αρχιτεκτονική παρέμβαση του ερευνητή. Η beam search, μέχρι πρότινος αλλά και ακόμα, αποτελούσε και συνεχίζει να αποτελεί μια αξιόπιστη και δημοφιλή μέθοδο για decoding. Χρησιμοποιώντας 5 beams ως παράμετρο πλήθους μονοπατιών, το σύστημά μας επιστρέφει τον ακόλουθο διάλογο

> YOU: i want the hat and the ball <eos>
> THEM: i need the hat and at least one other item <eos>
> YOU: i need the hat and at least one other item <eos>
> THEM: i need the

Παρατηρούμε πως ο διάλογος αυτός είναι διαφορετικός από αυτόν που παρήγαγε η Greedy Decoding, γεγονός που σημαίνει οτι και στην πράξη η Greedy Decoding είχε αγνοήσει πιο πιθανές ακολουθίες από αυτή που παρήγαγε.

Όπως παρατηρούμε και εδώ είναι εμφανές το πρόβλημα της επανάληψης προτάσεων με τη χρήση Greedy Decoding και Beam Search και εν γένει αποτελεί ένα σημαντικό ζήτημα [137]. Ένας τρόπος να αντιμετωπιστεί αυτό το πρόβλημα είναι με τη χρήση "ποινών" σε ακολουθίες που επαναλαμβάνονται (n-gram penalties) [91] [55]. Η ιδέα είναι πως αν κάποια n-γράμματα εμφανιστούν ξανά θα αποδίδεται μηδενική πιθανότητα σε αυτά και ως επακόλουθο δεν θα επαναλαμβάνονται φράσεις. Το αποτέλεσμα της χρήσης n-gram ποινών είναι ο διάλογος

> YOU: i want the hat and the ball. <eos>
> THEM: you can have all the books if i get everything else. i need at least one other item to make it worth it for me. how about you get one book and

Πλέον ως παράμετρο λαμβάνουμε και το πλήθος των n-γραμμάτων. Στο παράδειγμα, που υλοποιήθηκε χρησιμοποιήθηκαν 2-γράμματα. Επιπρόσθετα, εκτός από το πιο πιθανό beam που εν τέλει δείχνουμε στην παρουσίαση των αποτελεσμάτων, μπορεί κανείς να παράγει τα αποτελέσματα όσων beams επιθυμεί. Ο αριθμός αυτός, φυσικά, θα πρέπει να είναι μικρότερος από το μέγιστο πλήθος beams που έχει δοθεί ως παράμετρος στον αλγόριθμο. Ενδεικτικά, με τη χρήση δι-γραμματικών ποινών και 5 beams, προέκυψαν και οι παρακάτω 5 πιθανοί διάλογοι:

> Διάλογος 0:
> YOU: i want the hat and the ball. <eos>
> THEM: you can have all the books if i get everything else. i need at least one other item to make it worth it for me. how about you get one book and
>
> Διάλογος 1:
> YOU: i want the hat and the ball. <eos>
> THEM: you can have all the books if i get everything else. i need at least one other item to make it worthwhile for me. how about you take all of the hats
>
> Διάλογος 2:

YOU: i want the hat and the ball. <eos>
THEM: you can have all the books if i get everything else. i need at least one other item to make it worthwhile for me. how about you get the hats and i

Διάλογος 3:
YOU: i want the hat and the ball. <eos>
THEM: you can have all the books if i get everything else. i need at least one other item to make it worthwhile for me. how about you get the hats and one

Διάλογος 4:
YOU: i want the hat and the ball. <eos>
THEM: you can have all the books if i get everything else. i need at least one other item to make it worth it for me. how about you get the balls and

Παρατηρείται πως οι διάλογοι αυτοί δεν απέχουν αρκετά μεταξύ τους και όπως αναφέρεται [41] αυτό είναι λογικό όταν έχουμε μικρό πλήθος από beams.

Παρόλα αυτά, η Beam Search λειτουργεί σαφώς αποδοτικότερα όταν μπορεί να προβλεφθεί το πλήθος των tokens που θα παραχθούν [140]. Θίγοντας αυτή την ευαισθησία ως προς το πλήθος των tokens προς παραγωγή, η beam search ενώ μπορεί να είναι κατάλληλη σε προβλήματα αυτόματης μετάφρασης (όπου το πλήθος των tokens μπορεί σχετικά να προβλεφθεί) δεν είναι κατάλληλη σε προβλήματα διαλόγων ή παραγωγής ιστοριών (story generation).

7.3.7 Sampling

Σύμφωνα με τη μελέτη του Ari Holtzman [39], διατυπώθηκε πως η η κατανομή των λέξεων που παράγονται από ανθρώπινη ομιλία διαφέρει από αυτή των ακολουθιών που παράγονται μέσω του Greedy Decoding και της Beam Search. Για τον λόγο αυτό εισάγεται κάποια τυχαιότητα στους αλγορίθμους για την παραγωγή κειμένων.

$$w_t \sim P(w|w_{1:t-1}) \tag{7.15}$$

Αν το μοντέλο σε κάθε χρονική στιγμή δειγματοληπτεί την επόμενη λέξη, τότε είναι πολύ πιθανό να παράγει προτάσεις οι οποίες να μην είναι ορθές. Ένας τρόπος να λυθεί το πρόβλημα αυτό είναι μειώνοντας την παράμετρο "Temperature" της softmax. Μειώνοντας το temperature, η πιθανότητα των πιο πιθανών λέξεων αυξάνεται και η πιθανότητα των λιγότερο πιθανών λέξεων μειώνεται. Αν $temperature \to 0$, τότε ο αλγόριθμος δειγματοληψίας μετατρέπεται σε Greedy Decoding.

Χωρίς Temperature:
YOU: i want the hat and the ball then. you can have the books and your balls. <eos>
THEM: ok <eos>
YOU: <selection> YOU: i want the hat and the ball then. you can

Με Temperature = 0.7:
YOU: i want the hat and the ball. <eos>
THEM: i can not make that deal. i want the hat and one book <eos>

YOU: i need the hat, or i can not make a deal. <

7.3.7.1 Top-k sampling

Μια έξυπνη μέθοδος δειγματοληψίας είναι να υφίσταται δειγματοληψία μόνο μεταξύ των k πιο πιθανών λέξεων δεδομένου ενός context και των είδη παραγόμενων λέξεων [26]. Με αυτόν τον τρόπο εισάγεται μια τυχαιότητα στις λέξεις και κατ' επέκταση τις προτάσεις που θα παραχθούν και επιπλέον θέτοντας ένα σχετικά μικρό k αναμένεται η ποιότητα των προτάσεων αυτών να είναι εξαιρετικά ικανοποιητική. Το αποτέλεσμα της μεθόδου αυτής εφαρμοσμένης στο GPT-2 είναι ο διάλογος

YOU: i want the hat and the ball.. i'd be willing to give you the book. <eos>
THEM: okay how about you take the hat and one ball? <eos>
YOU: that's just perfect. deal

Παρόλα αυτά, το γεγονός οτι η δειγματοληψία λαμβάνει χώρα σε έναν περιορισμένο αριθμό k λέξεων κάθε χρονική στιγμή, αυτό σε καμία περίπτωση δεν προϋποθέτει οτι η κατανομή πιθανότητας μεταξύ των λέξεων είναι ομοιόμορφη. Επίσης, μπορεί τη μία χρονική στιγμή να επιλεγεί μια λέξη με μεγάλη πιθανότητα και την αμέσως επόμενη μία με μικρή πιθανότητα. Ακόμα, αν μειώσουμε το k αρκετά, θα αντιμετωπίσουμε τα προβλήματα που είχαμε προηγουμένως όπου δεν υπάρχει ποικιλομορφία στα κείμενα που παράγονται.

7.3.7.2 Top-p (nucleus) sampling

Για αυτό τον σκοπό εισάγεται το Top-p (nucleus) sampling [39]. Αυτή η μέθοδος δεν δεσμεύει ένα συγκεκριμένο αριθμό λέξεων κάθε χρονική στιγμή. Αντιθέτως τίθεται ένα πιθανοτικό κατώφλι p και ο αλγόριθμος δειγματοληπτεί ανάμεσα στο πλήθος των λέξεων, οι αθροιστικές πιθανότητες των οποίων ξεπερνούν το κατώφλι αυτό. Συνεπώς, προτείνεται μια δυναμική προσέγγιση ως προς την δειγματοληψία. Αυτές οι δύο μέθοδοι θα μπορούσαν και να συνδυαστούν μεταξύ τους. Το αποτέλεσμα του διαλόγου με την μέθοδο αυτή είναι το ακόλουθο

YOU: i want the hat and the ball <eos>
THEM: i can not give up the hat <eos>
YOU: no deal. i can give you everything if i can have the hat and the book <eos>
THEM

Αναφέρεται ακόμα πως και οι τεχνικές δειγματοληψίας σε αρκετές περιπτώσεις μπορούν να εμφανίζουν ζητήματα επαναληψιμότητας [137].

7.4 Συμπεράσματα

Σε αυτό το Κεφάλαιο μελετήθηκε η προσέγγιση της Facebook στο Task Deal Or No Deal [65] και προτάθηκαν ορισμένες τροποποιήσεις στις αρχιτεκτονικές αυτές με μικρές διαφορές

στα αποτελέσματα. Εντοπίστηκε πως στο classification task, χρησιμοποιώντας ως μετρική το perplexity οδηγούμαστε σε μη ικανοποιητικά αποτελέσματα σε σχέση με την ακρίβεια και την επιτυχία του συστήματος να κατανοήσει την πρόταση του συνομιλητή. Παρόλα αυτά, το ζήτημα αυτό είναι εξαιρετικά σημαντικό για τη δημιουργία επιτυχημένων αυτόματων διαπραγματευτών. Για το σκοπό αυτό, μελετήθηκε το πρόβλημα της ταξινόμησης απομονωμένα. Παρουσιάστηκαν τόσο παραδοσιακές όσο και πιο σύγχρονες αρχιτεκτονικές και ως συμπέρασμα εξάγεται πως η χρήση του BERT Classifier (Transformer) οδηγεί σε αισθητά καλύτερα αποτελέσματα. Το δεύτερο πρόβλημα που μελετάται από τη Facebook αφορά τη Γλωσσική Μοντελοποίηση με απώτερο σκοπό να παράγεται λόγος. Συγκρίνοντας τις αρχιτεκτονικές που προτείνονται και επαυξάνοντας τα δυνατά μοντέλα με ορισμένες τροποποιήσεις αυτών, δεν παρατηρήθηκε αισθητή διαφορά στις μετρικές του perplexity. Για το λόγο αυτό, χρησιμοποιήθηκε και μία σύγχρονη αρχιτεκτονική Transformer, το GPT-2, που παρουσιάζει εξαιρετικά αποτελέσματα και κρίνεται πως χρειάζεται περαιτέρω μελέτη. Προκειμένου να αποδοθεί μια ολοκληρωμένη οπτική του προβλήματος, παρουσιάστηκαν ορισμένες τεχνικές παραγωγής κειμένου (greedy, beam search, top-k, top-p (nucleus) sampling), η αξιολόγηση των οποίων γίνεται εμπειρικά.

Από τη στιγμή που μπορεί το ευρύτερο πρόβλημα να διαιρεθεί σε υποπροβλήματα, προτείνεται ως επιλογή και η χρήση modular αρχιτεκτονικών. Η δυσκολία, συνήθως, σε αυτές τις περιπτώσεις έγκειται στο ζήτημα του credit assignment, στο πώς θα συνδυαστούν τα υπομέρη μεταξύ τους και στο γεγονός οτι αν αλλάξει το πρόβλημα χρειάζεται να αλλάξουν τα υπομέρη. Μολονότι οι end-to-end προσεγγίσεις λύνουν κάποια από αυτά τα ζητήματα, δεν υπάρχει διαφάνεια και επίσης θεωρείται πως είναι δύσκολη η εύρεση κάποιας κατάλληλης αρχιτεκτονικής που να λύνει πληθώρα προβλημάτων.

Ακόμα, μεγάλη μερίδα ερευνητών [67], [110], [109] πέρα από τα Likelihood Μοντέλα κάνει χρήση και Ενισχυτικής Μάθησης. Η μέθοδος αυτή είναι γνωστή ως two-stage learning [21]. Είναι σύνηθες το φαινόμενο η γλώσσα σε συστήματα που έχουν υποστεί εκμάθηση μέσω της χρήσης ενισχυτικής μάθησης να χειροτερεύει. Αυτό παρατηρείται και στο δικό μας πρόβλημα διαπραγμάτευσης [65]. Για αυτό το λόγο, στο επόμενο Κεφάλαιο, εφαρμόζουμε Ενισχυτική Μάθηση στην ενέργεια που θα πρέπει να λάβει ένας πράκτορας η οποία προϋπάρχει της γλώσσας.

7.5 Μελλοντικές Προεκτάσεις

Συνοψίζουμε κάποιες από τις μελλοντικές προεκτάσεις και τις βελτιστοποιήσεις που θα κάνουμε ή θα προτείναμε να γίνουν. Συγκεκριμένα,

- προτείνεται να δοθεί ακόμα μεγαλύτερη έμφαση στις υλοποιήσεις των Transformers, BERT και GPT-2, διότι τα αποτελέσματα κατά πολύ ξεπερνούν αυτά των παραδοσιακών τεχνικών μηχανικής μάθησης
- θα μπορούσαμε (στο πρόβλημα της ταξινόμησης) διευκολύνουμε το μοντέλο μας, εισάγοντας το πλήθος διαθέσιμων αντικειμένων μαζί με τις αντίστοιχες αξίες
- θα μπορούσαμε (στο πρόβλημα της ταξινόμησης) να αυξήσουμε τα δεδομένα μας και να δημιουργήσουμε υποδιαλόγους από τους υπάρχοντες. Μια σοβαρή υπόθεση που θα μπορούσε να γίνει είναι οτι τα τελευταία turns των διαλόγων καθορίζουν το αποτέλεσμα και συνεπώς θα μπορούσαμε να αγνοήσουμε ένα συγκεκριμένο πλήθος από τα πρώτα turns δημιουργώντας "νέους διαλόγους"

- με τις τεχνικές παραγωγής κειμένων μπορούμε να δημιουργήσουμε rollouts διαλόγων. Αν οι μελλοντικές εκβάσεις διαλόγων που θα παραχθούν μπορούν να συνδυαστούν με ένα αξιόπιστο σύστημα ταξινόμησης, όπως περιγράφηκε, μπορούμε να εφαρμόσουμε κάποιου είδους planning στο παίγνιο της διαπραγμάτευσης. Επίσης, με αυτό τον τρόπο μπορεί να προκύψει και ένας ανορθόδοξος αλλά παραλληλοποιήσιμος τρόπος παραγωγής απαντήσεων κατά τον οποίο παράγονται κείμενα πλούσια σε γλωσσικά χαρακτηριστικά έως ότου ο ταξινομητής ταξινομήσει τα κείμενα αυτά στην κατηγορία που προσδιορίζει την ενέργεια που επιθυμούμε να λάβουμε.

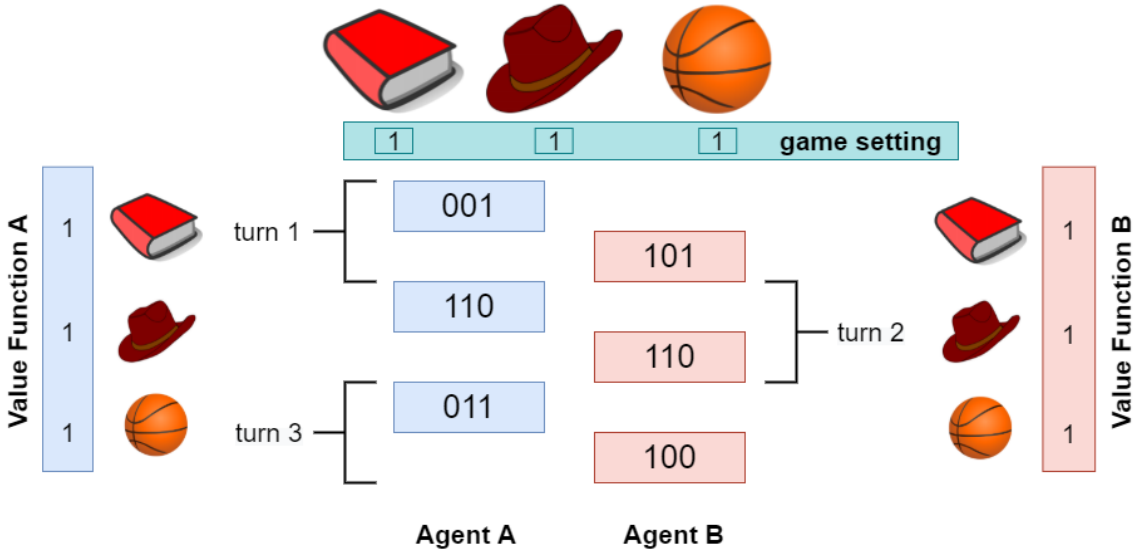
Κεφάλαιο 8

Ανάδειξη Στρατηγικών και Συμπεριφορών Πρακτόρων σε Περιβάλλοντα χωρίς Γλωσσικές Εξαρτήσεις με χρήση Ενισχυτικής Μάθησης

Κατά τη διάρκεια της εκπόνησης της παρούσας διπλωματικής εργασίας, σχηματίστηκε η υπόθεση οτι η γλώσσα ίσως δημιουργείται ανάλογα με τη δομή των πρακτόρων και του περιβάλλοντος, προκειμένου να επικοινωνηθούν μηνύματα μεταξύ των πρακτόρων που αποσκοπούν στην ικανοποίηση κάποιας συγκεκριμένης ανάγκης τους που γεννάται στο περιβάλλον. Σύμφωνα με τη θεωρία αυτή, η ανάγκη προϋπάρχει της γλώσσας ή συγκεκριμένα η στρατηγική, το πλάνο αντιμετώπισης ενός προβλήματος, προϋπάρχει του μηνύματος που σχηματίζεται για την επικοινωνία μέρους του πλάνου. Η γλώσσα καλείται, λοιπόν, να αναπαραστήσει έναν διαπραγματευτικό σκοπό και να τον καταστήσει αντιληπτό από έναν άνθρωπο. Προς επίρρωση αυτού, παρατίθεται ένα παράδειγμα διαπραγμάτευσης όπου είναι διαθέσιμα τρία βιβλία, τέσσερα καπέλα και μια μπάλα ($[3, 4, 1]$) με αντίστοιχες αξίες $[1, 0, 4]$ (σε μονάδες). Τα καπέλα δεν έχουν καμία αξία για τον πράκτορα και ίσως μια πρότασή του να συμπυκνωνόταν με το διάνυσμα $[3, 0, 1]$. Αργότερα, αυτό το διάνυσμα θα πρέπει να το μετατρέψει σε γλώσσα για να επικοινωνήσει την πρότασή του, λ.χ. "Θα ήθελα τα τρία βιβλία και τη μία μπάλα. Μπορείς να κρατήσεις τα καπέλα!". Στο συγκεκριμένο κεφάλαιο, δίνεται έμφαση στο διάνυσμα της πρότασης, στην low-level, δηλαδή, αντιμετώπιση του ζητήματος.

Για να γίνουμε πιο συγκεκριμένοι, στο Σχήμα 8.2, το διάνυσμα του game setting [1,1,1] περιγράφει πόσες μπάλες, καπέλα και βιβλία υπάρχουν διαθέσιμα προς διαπραγμάτευση. Σε αυτή την περίπτωση, υπάρχει ένα αντικείμενο από κάθε είδος. Στη συνέχεια, οι συναρτήσεις αξιών [1,1,1] (Value Functions A και B) δηλώνουν την αξία που έχουν οι μπάλες, τα καπέλα και τα βιβλία αντίστοιχα για κάθε παίκτη. Παρατηρείται, από το Σχήμα 8.2, οτι οι προτάσεις των πρακτόρων A και B στο κανάλι επικοινωνίας δεν γίνονται πλέον με τη μορφή γλώσσας αλλά βασίζονται σε χαμηλού επιπέδου αναπαραστάσεις των προτάσεων. Για παράδειγμα, η πρώτη πρόταση του πράκτορα A αντιστοιχεί στο διάνυσμα [0,0,1]. Αυτό σημαίνει οτι ο πράκτορας A προτείνει να λάβει 0 μπάλες, 0 καπέλα και 1 βιβλίο.

Το πρόβλημα αυτό εμπίπτει στην κατηγορία προβλημάτων δημιουργίας συστημάτων αυτό-



Σχήμα 8.1: Δομή Framework του Περιβάλλοντος χωρίς Γλωσσικές Εξαρτήσεις και οι Βασικές Έννοιές του

ματης διαπραγμάτευσης [4]. Ο κώδικας των προσομοιώσεων που δημιουργήθηκαν και συγκεκριμένα η έννοια του περιβάλλοντος έχει γραφεί σε αντιστοιχία με τα περιβάλλοντα που χρησιμοποιούνται για την υλοποίηση αλγορίθμων ενισχυτικής μάθησης.

8.1 Περιγραφή Προβλήματος

Τα διαθέσιμα αντικείμενα φέρουν περιορισμούς ως προς το πλήθος τους (έως 4 από κάθε είδος) και συνεπώς ο χώρος των δυνατών επιλογών είναι περιορισμένος και μετρήσιμος. Με αυτή την ιδιομορφία του προβλήματος, δεδομένου ενός διανύσματος με τις διαθέσιμες πληθικότητες, ένα σύστημα έχει ένα σχετικά μικρό και πεπερασμένο αριθμό διαθέσιμων επιλογών κάθε χρονική στιγμή. Στο παράδειγμα με game setting = [3,4,1], υπάρχουν $(3+1)*(4+1)*(1+1) = 40$ επιλογές. Το $+1$ προκύπτει διότι ένας παίκτης μπορεί να μην λάβει και κανένα αντικείμενο. Αν και η μη λήψη απόφασης (στασιμότητα) θεωρείται και αυτή μια μορφή απόφασης, τα υπό μελέτη συστήματα σχεδιάζεται να λαμβάνουν πάντα μία. Ανάγεται, λοιπόν, το εν λόγω πρόβλημα, σε ένα πρόβλημα όπου ο κάθε παίκτης έχει ένα πεπερασμένο αριθμό επιλογών όπου κάθε επιλογή τού προσδίδει διαφορετικό αντίτιμο και συνεπώς θα πρέπει να αναγνωρίσει την βέλτιστη και τελεσφόρα πολιτική λήψης αποφάσεων.

Κάθε πράκτορας, λοιπόν, αντιλαμβάνεται ένα observation/state της διαπραγμάτευσης (8.1.1.2) και ενεργεί αποσκοπώντας τη μεγιστοποίηση του κέρδους που ορίζεται από τη συνάρτηση ανταμοιβής (8.1.1.3) έως ότου τερματιστεί το παίγνιο της διαπραγμάτευσης (8.1.1.1). Σε κάθε του ενέργεια, το περιβάλλον (8.1.1) του επιστρέφει κάποια ανταμοιβή και το επόμενο observation/state. Στους πράκτορες δίνεται ένα διάνυσμα που αντιστοιχεί στο πλήθος των διαθέσιμων αντικειμένων από κάθε κατηγορία και η αντίστοιχη συνάρτηση αξιών. Προκειμένου να τηρηθούν οι ευρύτεροι κανόνες που περιγράφονται στο Deal-Or-No-Deal task της Facebook [65] οι δύο πράκτορες δεν μπορούν να παρατηρήσουν τη συνάρτηση αξιών του άλλου πράκτορα με τον οποίο συνδιαλέγονται. Δεδομένου οτι ο μέγιστος αριθμός αντικειμένων της ίδιας κατηγορίας είναι το 4, μπορούν να υπάρξουν $(4+1)*(4+1)*(4+1) = 125$ πιθανές προτάσεις. Ανάλογα με το διάνυσμα που περιγράφει τα διαθέσιμα αντικείμενα, δημιουργείται

και ο αντίστοιχος χώρος πιθανών εκβάσεων.

Το πλαίσιο αυτό ταυτίζεται με το πλαίσιο της Ενισχυτικής Μάθησης, έτσι όπως περιγράφηκε στην Ενότητα 5.3.12.3. Συγκεκριμένα, στις προσομοιώσεις που θα ακολουθήσουν χρησιμοποιούμε το Q-learning.

8.1.1 Περιβάλλον

Ανάλογα με το πώς επιλέγεται να αντιμετωπιστεί ο "αντίπαλος" πράκτορας, το περιβάλλον συναντάται στις προσομοιώσεις σε δύο εκδοχές. Η πρώτη εκδοχή θεωρεί πως ο "αντίπαλος" πράκτορας αποτελεί κομμάτι του περιβάλλοντος διαπραγμάτευσης. Αντιθέτως, η δεύτερη αντιμετωπίζει και τους δύο πράκτορες ως ξεχωριστές οντότητες, οι οποίες αλληλεπιδρούν εντός του περιβάλλοντος το οποίο ουσιαστικά τους παρέχει τα διαθέσιμα αντικείμενα, τις αξίες αυτών, το κανάλι επικοινωνίας και τον μηχανισμό ανταμοιβών σε περίπτωση συμφωνίας ή διαφωνίας. Μολονότι το περιβάλλον περιορίζεται στη χρήση δύο πρακτόρων, μπορεί εύκολα να αναχθεί και σε πολυπρακτορικό περιβάλλον.

8.1.1.1 Συνθήκη Τερματισμού

Το μέγιστο πλήθος προτάσεων από κάθε πράκτορα είναι 3 προτάσεις και συνεπώς ο μέγιστος αριθμός προτάσεων ενός παιγνίου διαπραγμάτευσης ανέρχεται στις 6. Αυτό αποτελεί αρχιτεκτονική επιλογή και θεωρείται ένα κατάλληλο μέγεθος προκειμένου να εξερευνήσει ένας πράκτορας το χώρο των πιθανών εκβάσεων ενός παιγνίου. Το παίγνιο τερματίζει όταν ικανοποιείται μία εκ των δύο ακόλουθων συνθηκών. Η πρώτη σχετίζεται με το μέγεθος της στιχομυθίας και η δεύτερη με το αν επιτεύχθηκε συμφωνία. Η επίτευξη της συμφωνίας δεν αποτελεί κάποια διαφορετική ενέργεια από τις προτάσεις διαμοιρασμού αλλά συγκεκριμένα το περιβάλλον καλείται να την αναγνωρίσει με τον εξής τρόπο. Αν τα διαθέσιμα αντικείμενα είναι $[1, 1, 1]$, ο πράκτορας A έχει προτείνει να λάβει $[1, 0, 1]$ και ο πράκτορας B αποκριθεί πως θέλει να λάβει $[0, 1, 0]$, τότε θεωρείται πως έχουν συμφωνήσει. Με άλλα λόγια, η συμφωνία επιτυγχάνεται όταν η επιθυμία ενός πράκτορα ακολουθείται από μια επιθυμία του άλλου που δεν προκαλεί κάποια σύγκρουση συμφερόντων και κάθε αντικείμενο διαμοιράζεται σε έναν από τους δύο παίκτες.

8.1.1.2 Observation/State

Το περιβάλλον, μετά από την ενέργεια του πράκτορα, επιστρέφει σε αυτόν την επόμενη κατάσταση/παρατήρηση (observation/state). Το διάνυσμα αυτό εισάγεται στην είσοδο του μοντέλου προκειμένου να κάνει την επόμενη ενέργεια. Δημιουργούνται διάφορες εκδοχές του observation/state, η απλούστερη των οποίων συνίσταται από ένα διάνυσμα 9 στοιχείων όπου τα πρώτα 3 αντιστοιχούν στο πλήθος διαθέσιμων αντικειμένων κάθε κατηγορίας, τα επόμενα τρία στη συνάρτηση αξίας του πράκτορα και τα τελευταία 3 στη πρόταση διαμοιρασμού αντικειμένου που έκανε ο "αντίπαλος" πράκτορας, μετασχηματισμένη, βέβαια, στην οπτική μοντέλου μας.

8.1.1.3 Συνάρτηση Ανταμοιβής (Reward Function)

Σπουδαίο ρόλο στα Προβλήματα Ενισχυτικής Μάθησης παίζει η Μηχανική Ανταμοιβών (Reward Engineering). Διαφορετικές Συναρτήσεις Ανταμοιβών (Reward Functions) μπορούν να οδηγήσουν σε διαφορετικές συμπεριφορές των εκπαιδευόμενων πρακτόρων. Για παράδειγμα, μπορεί να τιμωρούν κάθε ενέργεια με μία μικρή ποινή (-1) οδηγώντας τους πράκτορες σε συνοπτικούς διαλόγους ή μπορεί να οδηγούν τους πράκτορες σε συνεργατικές ή ακόμα και ανταγωνιστικές συμπεριφορές.

8.1.2 Περιγραφή Πρακτόρων

Συναντώνται δύο ευρύτερες κατηγορίες πρακτόρων-μοντέλων. Η πρώτη αφορά μοντέλα που βασίζονται στο DQN [82] και σε μεταγενέστερες εξελίξεις του ενώ η δεύτερη αφορά μοντέλα τα οποία δεδομένων των διαθέσιμων επιλογών δειγματοληπτούν τυχαία και ομοιόμορφα την απάντησή τους. Τα τελευταία είναι χρήσιμα στις περιστάσεις όπου επιδιώκεται να χρησιμοποιηθούν ως μέρος του περιβάλλοντος και να παρουσιαστεί η σπουδαιότητα της συνάρτησης ανταμοιβών. Για την εκμάθηση των μοντέλων της πρώτης κατηγορίας, χρησιμοποιείται Ενισχυτική Μάθηση (Reinforcement Learning) και συγκεκριμένα η μέθοδος του Q-learning. Η επιλογή του Q-learning δικαιολογείται από το γεγονός οτι προτιμάται μια model-free προσέγγιση του προβλήματος και η χρήση ενός μοντέλου που μπορεί να αποδώσει μια τιμή προσδοκώμενης ανταμοιβής δεδομένης μιας κατάστασης της διαπραγμάτευσης.

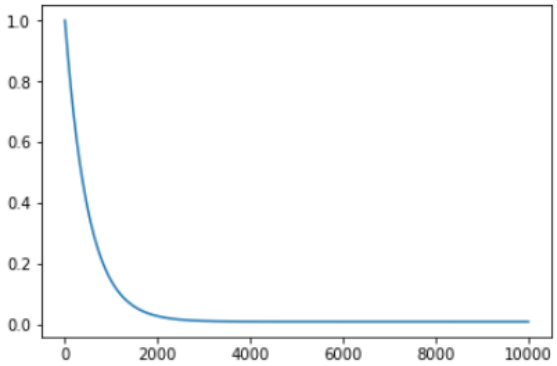
Επιπρόσθετα, μελετάται ένα σπουδαίο και φλέγον ζήτημα της Ενισχυτικής Μάθησης που φέρει την ονομασία "Exploration vs Exploitation Trade-Off". Αυτό σχετίζεται με το κατά πόσο πρέπει ένας πράκτορας να εξερευνήσει τον χώρο προτού αρχίσει να αξιοποιεί τις καταστάσεις στις οποίες βρίσκεται. Συγκεκριμένα, γίνεται χρήση της ε-greedy μεθόδου. Χρησιμοποιείται είτε ένα σταθερό πιθανοτικό κατώφλι είτε μια συνάρτηση που σταδιακά συγκλίνει σε ένα προκαθορισμένο κατώφλι. Αν με x συμβολίζεται ο αριθμός του επεισοδίου τότε η πιθανότητα να κάνει ο πράκτορας μια τυχαία ενέργεια είναι

$$e_{final} + (e_{start} - e_{final}) \exp^{\frac{-x}{e_{decay}}} \tag{8.1}$$

όπου $e_{final} = 0.01$, $e_{start} = 1.0$ και $e_{decay} = 500$. Όσο περνούν τα επεισόδια, η συνάρτηση αυτή τείνει στο 0.01 που σημαίνει οτι υπάρχει 1% πιθανότητα ο πράκτορας να λάβει μια ενέργεια στην τύχη. Αρχικά, η πιθανότητα αυτή είναι αρκετά μεγάλη και αυτό βοηθά τον πράκτορα στην εξερεύνηση του χώρου. Αξιοσημείωτο να αναφερθεί είναι το γεγονός οτι γίνεται χρήση ενός Replay Buffer μεγέθους 1000 θέσεων, στον οποίο αποθηκεύονται πεντάδες της μορφής (state, action, reward, next state, done). Αργότερα, από τον buffer δειγματοληπτείται κάθε φορά ένα συγκεκριμένο πλήθος πεντάδων (4 batches) οι οποίες χρησιμοποιούνται για τον υπολογισμό του TD (Temporal Difference) Σφάλματος και συνεπώς για την ενημέρωση των παραμέτρων του εκάστοτε μοντέλου.

Το DQN μοντέλο [82] στην ουσία αποτελείται από ένα νευρωνικό δίκτυο με ένα κρυφό στρώμα 128 νευρώνων. Ως συνάρτηση ενεργοποίησης, χρησιμοποιείται η ReLU. Επιπρόσθετα, χρησιμοποιούνται και μεταγενέστερες εξελίξεις του DQN, όπως είναι τα Double DQNs [124] και τα Dueling DQNs [134]. Περισσότερες πληροφορίες μπορούν να αναζητηθούν στην Ενότητα 5.3.12.3. Επιπλέον, ο κώδικας που χρησιμοποιείται, βασίζεται στις υλοποιήσεις ενός ανοικτού κώδικα [1] και τροποποιήθηκε περαιτέρω για την ικανοποίηση αναγκών του δικού μας προβλήματος.

[1] https://github.com/higgsfield/RL-Adventure



Σχήμα 8.2: Γραφική Παράσταση Συνάρτησης 8.1 με παραμέτρους $e_{final} = 0.01$, $e_{start} = 1.0$ και $e_{decay} = 500$ για 10000 επεισόδια (οριζόντιος άξονας). Στον κάθετο άξονα αποτυπώνονται οι τιμές πιθανότητας της ε-greedy μεθόδου.

8.1.3 Μετρικές και Οπτικοποίηση Συμπεριφορών

Οι μετρικές αξιολόγησης που χρησιμοποιούνται είναι η μέση ανταμοιβή (reward), το ποσοστό συμφωνίας και ο χρόνος ενός παιγνίου. Στην προσομοίωση που αφορά τη σύγκλιση των πρακτόρων σε σημεία ισορροπίας, μετράται και το πλήθος των σημείων αυτών. Με την έννοια του χρόνου αναφερόμαστε στο άθροισμα του πλήθους των προσφορών που κάνουν και οι δύο πράκτορες. Στην περίπτωση που γίνεται χρήση της Συνάρτησης 8.1, για τον υπολογισμό των μέσων τιμών λαμβάνονται υπόψη μόνο οι μετρήσεις από το σημείο σύγκλισης και έπειτα.

Όσον αφορά την οπτικοποίηση των στρατηγικών που δημιουργούνται, είθισται στα DQN να χρησιμοποιείται είτε t-SNE [123] είτε saliency maps [34]. Παρόλα αυτά, στην παρούσα εργασία δεν χρειάζεται να χρησιμοποιηθούν οι τεχνικές αυτές διότι το observation/state όπως περιγράφηκε έχει σχηματιστεί με τέτοιο τρόπο που καταγράφει όλες τις διαθέσιμες πληροφορίες που χρειάζονται για την ανάδειξη των συμπεριφορών. Συγκεκριμένα, αν εισαχθεί η απλούστερη μορφή του observation/state στην είσοδο του DQN, γνωρίζουμε το πλήθος των διαθέσιμων αντικειμένων, την αξία αυτών για τον πράκτορά μας και την τελευταία πρόταση διαμοιρασμού του "αντιπάλου" μετασχηματισμένη στην οπτική του πράκτορά μας.

Όπως έχει προαναφερθεί το σύνολο των διαθέσιμων επιλογών σε ένα παίγνιο είναι περιορισμένο και μετρήσιμο. Η έξοδος των DQN μοντέλων θα δώσει μία τιμή που αντιστοιχεί στην προβλεπόμενη αξία της κατάστασης και της αντίστοιχης ενέργειας. Για την ανάδειξη, λοιπόν, της συμπεριφοράς ενός πράκτορα, εισάγονται σε αυτόν όλες τα δυνατά observations/states και εξάγονται τα Q-values και οι ενέργειες που λαμβάνονται. Οι ενέργειες αυτές αντιστοιχούν στις προτάσεις που έχουν τη μεγαλύτερη πιθανότητα. Με αυτό τον τρόπο, δημιουργούνται πίνακες με υποθετικά σενάρια και τις αντίστοιχες δράσεις και σε κάθε πείραμα ακολουθεί μια ποιοτική ανάλυση των αποτελεσμάτων. Ιδιαίτερο ενδιαφέρον έχει και η δειγματοληπτική επιλογή δράσεων ανάμεσα σε ένα σετ των καλύτερων ενεργειών με βάση τις τιμές Q.

8.2 Προσομοιώσεις και Αποτελέσματα

Σε αυτήν την ενότητα, παρουσιάζονται οι προσομοιώσεις που διεξήχθησαν και τα αποτελέσματα αυτών. Σε όλα τα πειράματα, η εκμάθηση ολοκληρώνεται μετά από 10000 παίγνια

διαπραγμάτευσης. Στην Ενότητα 8.2.1 συγκρίνονται τα διαφορετικά μοντέλα DQN που αντιμετωπίζουν ένα Μοντέλο Τυχαίας Πολιτικής και στην Ενότητα 8.2.2 μελετάται η επίδραση της Συνάρτησης Ανταμοιβής στη συμπεριφορά των πρακτόρων. Στην Ενότητα 8.2.3 θίγεται το ζήτημα της επαύξησης της παρατήρησης (observation/state) και στην 8.2.4 ερευνάται αν μπορεί το μοντέλο μας να εκμεταλλεύεται μια συγκεκριμένη στρατηγική του αντιπάλου. Έπειτα, στην Ενότητα 8.2.6 δίνεται έμφαση στο ζήτημα του Exploration vs Exploitation Trade-Off και τέλος στην Ενότητα 8.2.7 ερευνάται η επίτευξη κοινωνικής ισορροπίας μεταξύ δύο άπληστων (greedy) πρακτόρων.

8.2.1 Εκπαίδευση Πράκτορα Εναντίον Μοντέλου Τυχαίας Πολιτικής και Σύγκριση DQN Μοντέλων

8.2.1.1 Περιγραφή

Το πρώτο σύστημα προς εκμάθηση αλληλεπιδρά με ένα μοντέλο που λαμβάνει αποφάσεις τυχαία. Να σημειωθεί πως στην είσοδο του μοντέλου εισάγεται μαζί με άλλες παραμέτρους μόνο η τελευταία απάντηση του αντιπάλου, η οποία μετασχηματίζεται στην οπτική του μοντέλου μας. Το παίγνιο διαπραγμάτευσης συνίσταται συνολικά από 3 αντικείμενα: ένα βιβλίο, ένα καπέλο και μια μπάλα. Κάθε αντικείμενο έχει αξία μία μονάδα και για τους δύο πράκτορες και κανείς δεν γνωρίζει την συνάρτηση αξίας του "αντιπάλου" του. Ο πράκτορας σε κάθε του ενέργεια λαμβάνει αρνητικό αντίτιμο της τάξης -1. Σε περίπτωση συμφωνίας, αν η αξία των αντικειμένων του πράκτορά μας είναι μεγαλύτερη από αυτή του συστήματος, το λαμβάνει 10 μονάδες αλλιώς χάνει 10.

8.2.1.2 Αποτελέσματα και Συμπεράσματα

Το επεισόδιο στο οποίο συγκλίνει η συνάρτηση 8.1 στην πιθανότητα 1% που αφορά το exploration-exploitation trade-off είναι το 2645^o επεισόδιο. Οι μετρικές, λοιπόν, υπολογίζονται ως οι μέσες τιμές από εκείνο το επεισόδιο και έπειτα. Τα αποτελέσματα του μοντέλου που περιγράφηκε παρουσιάζονται στον Πίνακα 8.1.

Πίνακας 8.1: Αποτελέσματα Απλών και Σύνθετων DQN Μοντέλων με βραχυπρόθεσμη μνήμη ενός turn εναντίον Random Policy Model σε περιβάλλον με ένα αντικείμενο από κάθε είδος και όμοιες συναρτήσεις τιμών αντικειμένων (1 μονάδα ανά αντικείμενο, -1 σε κάθε κίνηση, +10 Νίκη -10 Ήττα)

	Μέση Ανταμοιβή	Μέσος Χρόνος	Ποσοστό Συμφωνίας (%)
Απλό DQN	6.098	3.542	83.41
Double DQN	6.99	3.756	82.27
Dueling DQN	6.995	3.746	82.4

Παρατηρούμε πως οι διαφορές στα αποτελέσματα των μοντέλων δεν είναι εν γένει τόσο μεγάλες. Παρόλα αυτά, οι πιο σύνθετες εκδοχές του DQN επιτυγχάνουν καλύτερες μέσες ανταμοιβές αλλά και υψηλότερο μέσο χρόνο. Στον Πίνακα 8.2, παρουσιάζονται οι στρατηγικές του Double DQN μοντέλου και τα Q-values που υπολογίζει ανάλογα με την κατάσταση που βρίσκεται. Με τον όρο κατάσταση αναφερόμαστε στην πρόταση που έχει κάνει το "αντίπαλο"

σύστημα. Για παράδειγμα, στην πρώτη γραμμή του Πίνακα 8.2, αν το σύστημα προτείνει στο μοντέλο μας οτι θέλει να λάβει όλα τα αντικείμενα, το μοντέλο μας δεν θα συμφωνήσει, θα αντιπροτείνει να λάβει ένα καπέλο και μία μπάλα. Δεδομένης αυτής της κατάστασης, υπολογίζει πως το αντίστοιχο Q-value είναι ίσο με 0.23. Σημειώνεται πως στην τελευταία γραμμή του πίνακα, η απόφαση που λαμβάνει το μοντέλο αντιστοιχεί στην πρόταση που θα κάνει αν χρειαστεί να ξεκινήσει τη διαδικασία του παιχνιδιού.

Πίνακας 8.2: Στρατηγική Double DQN Μοντέλου με βραχυπρόθεσμη μνήμη ενός στίχου εναντίον Random Policy Model σε περιβάλλον με ένα αντικείμενο από κάθε είδος και όμοιες συναρτήσεις τιμών αντικειμένων (1 μονάδα ανά αντικείμενο, +0 σε κάθε κίνηση, Άθροισμα Αξιών Αντικειμένων ως Αντίτιμο μόνο σε Περίπτωση Συμφωνίας)

Κατάσταση	Απόφαση	Τιμή Q	Αποτέλεσμα
111	011	0.2372	αντιπρόταση
110	011	0.3793	αντιπρόταση
101	110	0.1143	αντιπρόταση
100	011	10.05	συμφωνία
011	110	0.1905	αντιπρόταση
010	101	10.17	συμφωνία
001	110	10.28	συμφωνία
000	111	9.987	συμφωνία
-	011	0.2809	πρόταση

Παρατηρούμε πως το μοντέλο αποδίδει μεγάλες τιμές στα Q-values στις περιπτώσεις που πρόκειται να συμφωνήσει σε συμφέρουσες για αυτό προτάσεις. Αντίθετα, μικρό Q-value αποδίδεται στις περιπτώσεις που έχει μάθει οτι δεν είναι συμφέρουσες και χρειάζεται να οδηγηθεί σε κάποια αντιπροσφορά. Δεδομένου του παιχνιδιού που αποτελείται από 3 υπάρχοντα αντικείμενα, ένα από κάθε είδος και με την ίδια αξία (1 μονάδα), μπορεί κανείς να συμπεράνει οτι το μοντέλο συμφωνεί μόνο στις περιπτώσεις που η συνολική αξία των αντικειμένων που θα λάβει είναι μεγαλύτερη από αυτή του "αντιπάλου".

Ακόμα, οι συμφέρουσες αντιπροτάσεις που επιχειρεί το μοντέλο μας δεν είναι ίδιες κάθε φορά αλλά εμφανίζουν κάποια ποικιλομορφία. Κρίνεται σκόπιμο να τονιστεί οτι το μοντέλο μας μαθαίνει με βάση το περιβάλλον που αλληλεπιδρά αλλά και με βάση τον αντίπαλό του. Οι συμπεριφορές που προκύπτουν σχετίζονται άμεσα με τις ικανότητες του αντιπάλου του. Έτσι, ένα σύστημα που αλληλεπιδρά με έναν κακό αντίπαλο μπορεί να αρχίσει να παίζει και "κακές" κινήσεις που μπορεί να τον οδηγήσουν σε καλά αποτελέσματα με ελάχιστο κόπο και σε ελάχιστο χρόνο. Αντίστοιχα, και οι τιμές Q δεν αποτελούν κάποια αντικειμενική μέτρηση για την αξιολόγηση του παιχνιδιού εν γένει. Αποτελούν, βέβαια, μια αξιόπιστη μέτρηση του μοντέλου μας δεδομένου του περιβάλλοντος και του επιπέδου τού αντιπάλου. Θα μπορούσε κανείς να ισχυριστεί οτι δεν υπάρχει μια καλή στρατηγική αλλά κάθε φορά αναζητείται η πιο κατάλληλη και η πιο αποτελεσματική.

8.2.2 Διαφοροποιήσεις στις Συναρτήσεων Ανταμοιβών

Διαφορετικές συναρτήσεις ανταμοιβών οδηγούν σε διαφορετικές συμπεριφορές των πρακτόρων. Απώτερος σκοπός είναι να θιχτεί η σπουδαιότητα της κατάλληλης επιλογής συνάρτησης ανταμοιβής ανάλογα με το επιθυμητό τελικό αποτέλεσμα. Όπως προαναφέρθηκε, η Μηχανική

Ανταμοιβών (Reward Engineering) αποτελεί ένα από τα πιο κομβικά ζητήματα του κλάδου της Ενισχυτικής Μάθησης.

8.2.2.1 Περιγραφή

Το παίγνιο διαπραγμάτευσης, όπως και στις προηγούμενες υλοποιήσεις, συνίσταται συνολικά από 3 αντικείμενα: ένα βιβλίο, ένα καπέλο και μια μπάλα. Κάθε ένα αντικείμενο έχει αξία μίας μονάδας και για τους δύο πράκτορες και οι συναρτήσεις αξιών είναι ατομικές και κρυφές. Επιπλέον, χρησιμοποιείται βραχυπρόθεσμη μνήμη μίας πρότασης. Πολλές είναι οι φορές που επιδιώκουμε να λύσουμε ένα πρόβλημα με ελάχιστο κόπο και σε ελάχιστο χρόνο. Αυτό στον Σχεδιασμό των Συναρτήσεων Ανταμοιβών αντιστοιχεί στην εισαγωγή ενός αρνητικού reward σε κάθε ενέργεια του πράκτορα. Εμπειρικά, κάθε δράση που λαμβάνει ο οποιοσδήποτε πράκτορας φέρει ένα μικρό κόστος. Ένα άλλο ζήτημα που μελετάται σχετίζεται με το πότε λαμβάνουν αντίτιμα οι πράκτορες και ποια είναι αυτά. Αυτές οι ευριστικές καθορίζουν τους πράκτορες και μπορούν να δημιουργήσουν ανταγωνιστικές, συναγωνιστικές, συναινετικές συμπεριφορές και ούτω καθεξής.

Το πρώτο πείραμα, λοιπόν, εξετάζει αν ένα αρνητικό reward σε κάθε ενέργεια του μοντέλου μας μπορεί να οδηγήσει σε πιο σύντομους διαλόγους. Στη δεύτερη υλοποίηση μελετάται ένα μοντέλο το οποίο μπορεί να λαμβάνει αντίτιμο σε περίπτωση συμφωνίας, χωρίς να συγκρίνει το σύνολο της αξίας των αντικειμένων του με αυτό του αντιπάλου του και στην τρίτη περίπτωση μελετώνται τα αποτελέσματα που προκύπτουν αν εισαχθεί μια αντίθετη συνάρτηση ανταμοιβής.

8.2.2.2 Αποτελέσματα και Συμπεράσματα

Τα αποτελέσματα του πρώτου πειράματος παρατίθενται στον Πίνακα 8.3. Μεταφέρονται τα αποτελέσματα του Απλού DQN, όπως αυτά παρουσιάστηκαν στον Πίνακα 8.1 και σε αυτά προστίθενται τα αποτελέσματα που προκύπτουν από το ίδιο μοντέλο χωρίς τη Χρήση Τιμωρίας (-1) σε κάθε βήμα. Παρατηρείται πως η μέση ανταμοιβή αυξάνεται στην περίπτωση που δεν υφίσταται η τιμωρία στο μοντέλο. Παρόλα αυτά, όταν το μοντέλο τιμωρείται σε κάθε ενέργεια ολοκληρώνει το παίγνιο σε χαμηλότερο χρόνο. Το ποσοστό συμφωνίας όταν δεν χρησιμοποιείται τιμωρία μειώνεται. Μια υπόθεση που θα μπορούσε να γίνει είναι οτι από την στιγμή που δεν κατευθύνεται το μοντέλο στο να πάρει ταχύτερες αποφάσεις μπορεί να οδηγηθεί σε μακρύτερες συνομιλίες όπου εκεί ελοχεύει ο κίνδυνος να τερματιστεί ο διάλογος και να οδηγήσει σε ασυμφωνία.

Πίνακας 8.3: Σύγκριση Επίδρασης Αρνητικού Αντιτίμου σε κάθε δράση του απλού DQN μοντέλου με βραχυπρόθεσμη μνήμη ενός turn εναντίον Random Policy Model σε περιβάλλον με ένα αντικείμενο από κάθε είδος και όμοιες συναρτήσεις τιμών αντικειμένων (1 μονάδα ανά αντικείμενο, +10 Νίκη -10 Ήττα)

	Μέση Ανταμοιβή	Μέσος Χρόνος	Ποσοστό Συμφωνίας (%)
Απλό DQN με Τιμωρία	6.098	3.542	83.41
Απλό DQN χωρίς Τιμωρία	6.522	3.756	76.57

Τα αποτελέσματα της δεύτερης υλοποίησης παρουσιάζονται στον Πίνακα 8.4. Με τον όρο "Επιθετικό" Double DQN αναφερόμαστε στο Double DQN που κατά την εκμάθηση λαμβάνει

αντίτιμα μόνο στην περίπτωση που υπάρχει συμφωνία και η συνολική αξία των αντικειμένων που θα έχει στην κατοχή του θα είναι μεγαλύτερη από αυτή του αντιπάλου του. Η ίδια προσέγγιση είχε πραγματοποιηθεί στην εκμάθηση του Double DQN, η στρατηγική του οποίου παρουσιάστηκε στον Πίνακα 8.2. Σε εκείνη την περίπτωση, όμως, το μοντέλο λάμβανε +10 μονάδες αν είχε καλύτερα αποτελέσματα από τον αντίπαλό του και -10 αντίθετα. Η διαφορά τώρα είναι πως το μοντέλο λαμβάνει ως ανταμοιβή μόνο την αξία των αντικειμένων του αν έχει καλύτερα αποτελέσματα από τον αντίπαλό του και 0 αντίθετα. Αυτό επιχειρείται προκειμένου να μπορεί να συγκριθεί η Μέση ανταμοιβή αυτού του τύπου μοντέλου με την μέση ανταμοιβή του "συνεργάσιμου" DQN. Το "Συνεργάσιμο" DQN Μοντέλο λαμβάνει ως ανταμοιβή το άθροισμα των αξιών των αντικειμένων του σε περίπτωση συμφωνίας ανεξαρτήτως της αξίας των αντικειμένων του "αντιπάλου". Αυτό το σύστημα, λοιπόν, προκειμένου να λάβει ανταμοιβή, χρειάζεται απλά να συμφωνήσει. Όσο καλύτερη, βέβαια, η συμφωνία τόσο μεγαλύτερη και η ανταμοιβή.

Τα αποτελέσματα του πίνακα 8.4 δείχνουν πως το συνεργάσιμο DQN επιτυγχάνει καλύτερα αποτελέσματα σε όλες τις κατηγορίες. Το ποσοστό συμφωνίας είναι κατά πολύ υψηλότερο και σημαντικά υψηλότερη αποδείχθηκε πως είναι και η μέση ανταμοιβή. Επίσης, χαμηλότερος είναι και ο χρόνος. Όσον αφορά τον χρόνο, βέβαια, το συνεργάσιμο μοντέλο προσπαθεί απλά να συμφωνήσει ενώ το επιθετικό θα πρέπει και να συμφωνήσει και να επιτύχει καλύτερα αποτελέσματα σε σχέση με τον αντίπαλό του. Ατό ίσως εξηγεί και τον ελάχιστα μεγαλύτερο χρόνο και τις διαφορές στα ποσοστά συμφωνίας.

Πίνακας 8.4: Σύγκριση Επίδρασης Συνθήκης και Αξίας Επιβράβευσης DQN Μοντέλων με βραχυπρόθεσμη μνήμη μιας στιχομυθίας εναντίον Random Policy Model σε περιβάλλον με ένα αντικείμενο από κάθε είδος και όμοιες συναρτήσεις τιμών αντικειμένων (1 μονάδα ανά αντικείμενο, -1 σε κάθε κίνηση)

	Μέση Ανταμοιβή	Μέσος Χρόνος	Ποσοστό Συμφωνίας (%)
Επιθετικό Double DQN	0.7023	2.617	81.86
Συνεργάσιμο Double DQN	1.061	2.459	99.96

Επιπλέον, αναμένεται πως μια αντίθετη συνάρτηση ανταμοιβής θα οδηγήσει σε αντίθετες συμπεριφορές. Σκοπός είναι να αποδειχθεί οτι το νέο μοντέλο Reverse Double DQN θα εκτελεί αντίθετες ενέργειες σε σχέση με το Double DQN του Πίνακα 8.2. Το μοντέλο, πλέον, σε περίπτωση συμφωνίας, αν έχει αντικείμενα με μεγαλύτερη αξία από αυτή του αντιπάλου του χάνει 10 μονάδες αλλιώς κερδίζει 10. Σε κάθε βήμα λαμβάνει επίσης και ένα αρνητικό reward της τάξης του -1. Σε επίπεδο μετρικών, παρατηρούμε πως οι διαφοροποιήσεις των μοντέλων δεν είναι σημαντικές 8.5.

Πίνακας 8.5: Επίδραση σε Επίπεδο Μετρικών Αντίθετης Συνάρτησης Ανταμοιβής στο Double DQN μοντέλο με βραχυπρόθεσμη μνήμη μιας στιχομυθίας εναντίον Random Policy Model σε περιβάλλον με ένα αντικείμενο από κάθε είδος και όμοιες συναρτήσεις τιμών αντικειμένων (1 μονάδα ανά αντικείμενο, -1 σε κάθε κίνηση)

	Μέση Ανταμοιβή	Μέσος Χρόνος	Ποσοστό Συμφωνίας (%)
Double DQN	6.99	3.756	82.27
Reverse Double DQN	7.013	3.742	82.21

Στον πίνακα 8.6 παρουσιάζονται τα Q-values και οι αντίστοιχες ενέργειες του μοντέλου Reverse Double DQN. Συγκρίνοντας τα αποτελέσματα των στρατηγικών των δύο μοντέλων επιβεβαιώνεται πως οδηγούνται σε αντίθετες ενέργειες κάθε χρονική στιγμή. Με παρόμοιο τρόπο, όπως συνέβαινε στο κλασικό μοντέλο Double DQN, το μοντέλο αποδέχεται συμφέρουσες προτάσεις (να δώσει τα περισσότερα αντικείμενα) και αντιπροτείνει διαμοιρασμούς αντικειμένων σε περίπτωση που η κατάσταση στην οποία βρίσκεται δεν είναι ιδανική. Οι αντιπροτάσεις αυτές είναι συμφέρουσες προς το μοντέλο μας. Εν κατακλείδι, μικρές διαφοροποιήσεις στις Συναρτήσεις Ανταμοιβών μπορούν να οδηγήσουν σε διαφορετικές συμπεριφορές.

Πίνακας 8.6: Στρατηγική Επιθετικού Double DQN Μοντέλου με βραχυπρόθεσμη μνήμη ενός στίχου εναντίον Random Policy Model σε περιβάλλον με ένα αντικείμενο από κάθε είδος και όμοιες συναρτήσεις τιμών αντικειμένων (1 μονάδα ανά αντικείμενο, +0 σε κάθε κίνηση, Άθροισμα Αξιών Αντικειμένων ως Αντίτιμο μόνο σε Περίπτωση Συμφωνίας)

Κατάσταση	Απόφαση	Τιμή Q	Αποτέλεσμα
111	000	9.972	συμφωνία
110	001	9.874	συμφωνία
101	010	9.862	συμφωνία
100	000	-0.0121	αντιπρόταση
011	100	9.8022	συμφωνία
010	000	1.034	αντιπρόταση
001	010	1.344	αντιπρόταση
000	000	0.4776	αντιπρόταση
-	100	0.3176	πρόταση

8.2.3 Επαύξηση του Observation/State με Ιστορία Παιγνίου

8.2.3.1 Περιγραφή

Τα έως τώρα μοντέλα έχουν βραχυπρόθεσμη μνήμη ενός turn. Μια πρόταση που δέχεται το μοντέλο μας θα πρέπει να έχει διαφορετική αντιμετώπιση όταν συμβαίνει στην αρχή της διαπραγμάτευσης και διαφορετική στο τέλος. Μια μέτρια πρόταση που γίνεται στις αρχές θα μπορούσε να οδηγήσει σε μια αντιπρόταση για να επιτευχθούν καλύτερα αποτελέσματα. Όμως, αν αυτή προταθεί στο τέλος, εμφανίζεται ο κίνδυνος της διαφωνίας όπου δεν θα ληφθεί κανένα αντίτιμο. Μελετώνται οι ακόλουθες δύο τακτικές:

- **time sense**: επαυξάνεται το observation/state με άλλη μία τιμή η οποία αντιστοιχεί στον μετρητή των δράσεων που έχει λάβει το μοντέλο μας
- **history augmentation**: αποτελεί πιο γενική περίπτωση και ουσιαστικά επαυξάνει την μνήμη του μοντέλου. Αν η μνήμη σχετίζεται με τα δύο τελευταία turns του παιγνίου, τότε το observation/state θα επαυξηθεί κατά 3 τιμές.

Η αύξηση της ιστορίας μας δίνει και την δυνατότητα να αναγνωρίσουμε τις πιθανές στρατηγικές ενός μοντέλου, όπου για παράδειγμα, αυτό μπορεί να κάνει μια χαμηλότερης αξίας προσφορά προκειμένου αργότερα να λάβει μια μεγαλύτερη.

8.2.3.2 Αποτελέσματα και Συμπεράσματα

Ως baseline μοντέλο χρησιμοποιείται ένα Dueling DQN χωρίς Επαυξημένο Observation/State. Στον πίνακα 8.7 παρουσιάζονται τα αποτελέσματα μοντέλων. Με H συμβολίζεται η επαύξηση ιστορικού και με TS η επαύξηση time sense. Το ιστορικό επαυξάνεται κατά μία ακόμα παρελθοντική κίνηση η οποία αντιστοιχεί στην προηγούμενη ενέργεια του πράκτορά μας.

Πίνακας 8.7: Επίδραση Επαύξησης Observation/State στο Dueling DQN μοντέλο με βραχυπρόθεσμη μνήμη μιας στιχομυθίας εναντίον Random Policy Model σε περιβάλλον με ένα αντικείμενο από κάθε είδος και όμοιες συναρτήσεις τιμών αντικειμένων (1 μονάδα ανά αντικείμενο, -1 σε κάθε κίνηση, +10/-10 σε συμφωνία)

	Μέση Ανταμοιβή	Μέσος Χρόνος	Ποσοστό Συμφωνίας (%)
Dueling DQN	7.025	3.758	82.41
Dueling DQN + TS	7.092	3.723	82.76
Dueling DQN + H	9.958	3.327	99.99
Dueling DQN + TS + H	9.948	3.349	100

Παρατηρούμε πως η χρήση του ιστορικού δίνει αρκετά καλύτερα αποτελέσματα ως προς την μέση ανταμοιβή. Το ποσοστό συμφωνίας είναι αυξημένο επειδή ο πράκτορας μαθαίνει ποιες επιλογές των συμφέρουν να αποδεχτεί. Η επαύξηση του observation/state με μέρος του ιστορικού συμβάλλει σημαντικά στις μεταβολές των αποτελεσμάτων. Αυτό υποδεικνύει οτι η Μαρκοβιανή Ιδιότητα που υποθέσαμε σχετικά με το Observation/State να μην ήταν και η πλέον ενδεδειγμένη. Γενικά, όπως έχει σχολιαστεί αυτή αποτελεί ισχυρή υπόθεση.

8.2.4 Ανίχνευση και Εκμετάλλευση Στρατηγικής Αντιπάλου

Αρχικά, η υπόθεση που γίνεται είναι πως ένα μοντέλο μπορεί να ανιχνεύσει μια συγκεκριμένη στρατηγική του αντιπάλου του αρκεί αυτή να συσχετίζεται με κάποιο δυνατό τρόπο με την προσπάθεια του μοντέλου να επιτύχει κάποια συγκεκριμένα αντίτιμα (rewards). Προς επίρρωση του ισχυρισμού αυτού, θεωρούμε πως το σύστημα ακολουθεί μια συγκεκριμένη πολιτική: στις πρώτες δύο στιχομυθίες διαπραγματεύεται επιλέγοντας τυχαία τις επόμενες κινήσεις του και στην τρίτη και τελευταία στιχομυθία προτείνει την χειρότερη κίνηση για αυτό, να μην λάβει, δηλαδή, κανένα αντικείμενο.

Όσον αφορά την αυστηρή μοντελοποίηση και τις λεπτομέρειες της υλοποίησης, χρησιμοποιήθηκε ένα παίγνιο όπου υπάρχουν τρία βιβλία, ένα καπέλο και δύο μπάλες (game setting=[1,1,1]) με συνάρτηση αντιτίμων μοντέλου [1,1,1] και συστήματος [1,1,1]. Κάθε μία τιμή στις συναρτήσεις αντιτίμων αντιστοιχεί στην αξία των βιβλίων, καπέλων και μπαλών για κάθε πράκτορα που συμμετέχει στο εν λόγω παίγνιο. Επιπλέον, χρησιμοποιείται η μεταβλητή $time\_sense$ προκειμένου να δοθεί έμφαση στον διαχωρισμό των καταστάσεων ανάλογα με το πλήθος των στιχομυθιών που έχουν προηγηθεί. Έτσι, αν για παράδειγμα το μοντέλο μας δεχθεί μια πρόταση από το σύστημα, έστω [1,1,1], αυτή θα αντιστοιχεί σε διαφορετική κατάσταση αν η πρόταση αυτή του συστήματος αντιστοιχεί στην πρώτη του κίνηση ή στην δεύτερή του κίνηση. Σημειώνεται πως για λόγους απλότητας, τους διαλόγους θα τους ξεκινάει πάντα το σύστημα.

8.2.5 Αποτελέσματα και Συμπεράσματα

Πίνακας 8.8: Στρατηγική Απλού DQN Μοντέλου με βραχυπρόθεσμη μνήμη μιας στιχομυθίας εναντίον Random Policy Model σε περιβάλλον με ένα αντικείμενο από κάθε είδος και όμοιες συναρτήσεις τιμών αντικειμένων (1 μονάδα ανά αντικείμενο) (-1 σε κάθε κίνηση, +10 Νίκη -10 Ήττα)

Κατάσταση	Απόφαση	Τιμή Q	Αποτέλεσμα
0111	011	0.2635	αντιπρόταση
1111	111	2.86	αντιπρόταση
2111	111	5.185	αντιπρόταση
0110	001	0.961	συμφωνία
1110	111	2.820	αντιπρόταση
2110	111	4.861	αντιπρόταση
0101	010	0.994	συμφωνία
1101	111	2.92	αντιπρόταση
2101	111	4.843	αντιπρόταση
0100	011	1.993	συμφωνία
1100	111	3.009	αντιπρόταση
2100	111	4.407	αντιπρόταση
0011	100	0.9635	συμφωνία
1011	111	2.842	αντιπρόταση
2011	111	4.773	αντιπρόταση
0010	101	1.929	συμφωνία
1010	111	2.881	αντιπρόταση
2010	111	4.367	αντιπρόταση
0001	110	1.914	συμφωνία
1001	111	2.813	αντιπρόταση
2001	111	4.214	αντιπρόταση
0000	111	2.867	συμφωνία
1000	111	2.902	συμφωνία
2000	111	3.666	συμφωνία
0-1-1-1	111	0.1801	πρόταση
1-1-1-1	111	3.284	πρόταση
2-1-1-1	111	6.275	πρόταση

Σε αυτό το σημείο με βάση τα αποτελέσματα του πίνακα 8.8 μπορούμε να κάνουμε διάφορες επισημάνσεις. Αρχικά, το μοντέλο όταν κληθεί να πάρει την τελευταία του απόφαση, αυτή θα είναι σε κάθε περίπτωση η [1,1,1] διότι γνωρίζει οτι το σύστημα του κάνει την πιο συμφέρουσα για αυτό πρόταση. Επιπλέον, ενδιαφέρον έχει και το οτι το μοντέλο σε κάθε περίπτωση θα πάρει την απόφαση [1,1,1] και στην προτελευταία του ενέργεια! Αυτό οφείλεται διότι έχει μάθει οτι μετά από αυτή την απόφαση το σύστημα θα προτείνει να δώσει όλα τα αντικείμενα στο μοντέλο, οπότε εκείνο προτρέχει να κλείσει την συμφωνία πριν την προτείνει το σύστημα! Σε αυτό το σημείο, πρέπει να τονιστεί πως αυτό είναι ένα ιδιαίτερα ενδιαφέρον ζήτημα που αφορά την Μηχανική Αντιτίμων (Reward Engineering). Ενώ περιμέναμε να προτείνει το μοντέλο μας [1,1,1] στην τελευταία του ενέργεια, αυτό το επιτυγχάνει και από την προτελευταία ενώ αυτή η συμπεριφορά δεν ήταν η προσδοκώμενη εκ πρώτης άποψης. Παρατηρούμε, όμως, πως αν το σύστημα δεν προβεί σε κάποια εξερεύνηση των δυνατών επιλογών (exploration vs

exploitation), συμφωνεί απευθείας λαμβάνοντας λιγότερα συνολικά αντίτιμα από αυτά που θα μπορούσε να λάβει αν συνέχιζε την διαπραγμάτευση. Δίνει έμφαση και εκμεταλλεύεται, λοιπόν, ένα πρόσκαιρο μικρό κέρδος σε σχέση με αυτό που θα μπορούσε να έχει. Εξαίρεση αποτελεί η περίπτωση όπου το σύστημα προτείνει στο μοντέλο μας να πάρει όλα τα αντικείμενα αλλά αυτό το αρνείται και δεν συμφωνεί γιατί θα λάμβανε μηδενικό αντίτιμο. Ιδιαίτερο ενδιαφέρον θα παρουσίαζε η μελέτη αυτή αν τα μοντέλα μας δεν λάμβαναν αποφάσεις με βάση το μέγιστο Q-value.

8.2.6 Exploration vs Exploitation Trade-off

8.2.6.1 Περιγραφή

Ιδιαίτερο ενδιαφέρον στα Προβλήματα Ενισχυτικής Μάθησης παρουσιάζει το Exploration vs Exploitation Trade-Off. Ένας πράκτορας καλείται να ανακαλύψει τις βέλτιστες στρατηγικές σε ένα περιβάλλον προκειμένου να λάβει τα υψηλότερα αντίτιμα. Παρόλα αυτά τίθεται το ερώτημα πότε μπορεί να είναι σίγουρος οτι τα αντίτιμα που λαμβάνει είναι τα υψηλότερα δυνατά. Θα χρειαστεί να οδηγηθεί σε τυχαία μονοπάτια προτού καταλήξει σε κάποιες στρατηγικές που τον ικανοποιούν και επιτακτική κρίνεται η ανάγκη κάποιας εξισορρόπησης μεταξύ της εξερεύνησης και αξιοποίησης καταστάσεων. Η μη εξερεύνηση και η αξιοποίηση κάποιων συγκεκριμένων καταστάσεων μπορεί να οδηγήσουν σε υποβέλτιστες στρατηγικές αλλά και αντίστροφα η μη-αξιοποίηση καταστάσεων και η διαρκής εξερεύνηση μπορεί να αποβεί ανώφελη.

Στη μελέτη μας, χρησιμοποιείται η ε-greedy μέθοδος. Συγκεκριμένα, γίνεται χρήση του Dueling DQN χωρίς επαύξηση Observation/State όπου μελετάται η επίδραση των παραμέτρων της ε-greedy μεθόδου και η εφαρμογή τόσο ενός σταθερού κατωφλίου όσο και της συνάρτησης 8.1.

Αρχικά, παρουσιάζεται ένα πλήρως τυχαίο μοντέλο που εξερευνά διαρκώς το χώρο. Έπειτα, ακολουθούν υλοποιήσεις που εξερευνούν τον χώρο με μια πιθανότητα 10% και 1%, όπως επίσης και μια υλοποίηση που δεν λαμβάνει τυχαία αποφάσεις. Στη συνέχεια, προτείνεται η χρήση της συνάρτησης που αναφέρθηκε προηγουμένως για ομαλή εξερεύνηση του χώρου επιλογών και παρουσιάζονται τα αποτελέσματα για $e_{decay} = 500, 250$ και 1000.

8.2.6.2 Αποτελέσματα και Συμπεράσματα

Στον Πίνακα 8.9 παρουσιάζονται τα αποτελέσματα της μελέτης σχετικά με τη μέθοδο ε-greedy για το Exploration-Exploitation Trade-Off. Ένα μοντέλο το οποίο λαμβάνει τυχαίες αποφάσεις και εξερευνά μονίμως τον χώρο δυνατών επιλογών δεν ανταμείβεται (Fully Random). Σε αντίθεση με αυτό ένα μοντέλο που δεν εξερευνά τον χώρο δεν γνωρίζει ποτέ αν μία επιλογή του είναι βέλτιστη (Not Random). Να σημειωθεί πως στη συγκεκριμένη περίπτωση, οι προσομοιώσεις ήταν ευαίσθητες σχετικά με τα αποτελέσματα του μοντέλου που δεν εξερευνά τον χώρο. Αυτό ίσως οφείλεται στο γεγονός οτι εκμεταλλεύεται μόνο τις καταστάσεις που τυχαία συναντά. Στη συνέχεια, χρησιμοποιείται η μέθοδος ε-greedy. Σε πρώτη περίπτωση χρησιμοποιείται ένα συγκεκριμένο πιθανοτικό κατώφλι 1 % και 10 % (e-greedy 1% και e-greedy 10% αντίστοιχα). Αυτό το κατώφλι εφαρμόζεται από το πρώτο επεισόδιο. Έπειτα, χρησιμοποιείται η συνάρτηση που περιγράφηκε προηγουμένως με διαφορετικές τιμές για το e_{decay}. Μεγαλύτερη τιμή της μεταβλητής αυτής δηλώνει πως η συνάρτηση συγκλίνει πιο αργά στο πιθανοτικό κατώφλι. Αντίστροφες είναι οι συνέπειες για μικρότερη τιμή του e_{decay} που

δεν επιτρέπει σε πολλά επεισόδια να λάβει το μοντέλο μας τυχαίες επιλογές. Όσον αφορά τους υπολογισμούς να τονιστεί οτι μόνο στην περίπτωση που χρησιμοποιείται η συνάρτηση, οι μέσες τιμές υπολογίζονται από το σημείο σύγκλισης και έπειτα.

Πίνακας 8.9: Μελέτη Exploration-Exploitation Trade-off και της Μεθόδου e-greedy στο Dueling DQN μοντέλο με βραχυπρόθεσμη μνήμη μιας στιχομυθίας εναντίον Random Policy Model σε περιβάλλον με ένα αντικείμενο από κάθε είδος και όμοιες συναρτήσεις τιμών αντικειμένων (1 μονάδα ανά αντικείμενο, -1 σε κάθε κίνηση, +10/-10 σε συμφωνία)

	Μέση Ανταμοιβή	Μέσος Χρόνος	Ποσοστό Συμφωνίας %
Fully Random	-1.845	4.897	48.9
Not Random	6.623	3.859	79.25
e-greedy 1%	6.512	3.883	78.09
e-greedy 10%	6.271	3.872	79.48
e-greedy 1% & d=500	7.155	3.728	83.38
e-greedy 1% & d=1000	7.029	3.75	82.53
e-greedy 1% & d=250	6.982	3.746	82.04

Στον Πίνακα, με d συμβολίζουμε το e_{decay}. Παρατηρούμε οτι η χρήση της e-greedy συνάρτησης που περιγράφηκε για $e_{decay} = 500$ δίνει τα καλύτερα αποτελέσματα. Αυτό συμβαίνει γιατί αρχικά επιτρέπει στο μοντέλο μας να εξερευνήσει τον χώρο για 2645 επεισόδια και έπειτα να εκμεταλλευτεί τις καταστάσεις τις οποίες συνάντησε με τον κατάλληλο τρόπο.

8.2.7 Επίτευξη Κοινωνικής Ισορροπίας μεταξύ Άπληστων Πρακτόρων

8.2.7.1 Περιγραφή

Έως τώρα, το "αντίπαλο" σύστημα είναι ένα μοντέλο που επιλέγει τυχαία τις αποκρίσεις του. Σε αυτό το σημείο, εισάγεται η δεύτερη μορφή περιβάλλοντος που περιγράφηκε στην οποία οι δύο πράκτορες διαχωρίζονται από το περιβάλλον και αποτελούν δύο ξεχωριστές οντότητες. Χρησιμοποιούνται δύο πλήρως ανταγωνιστικοί πράκτορες. Χρησιμοποιώντας μια Συνάρτηση Ανταμοιβής που επιβραβεύει τη συμφωνία μεταξύ δύο πρακτόρων, σκοπός είναι οι άπληστοι πράκτορες να οδηγηθούν μέσω trial and error σε μία κοινωνική ισορροπία. Ακόμα, και οι άπληστοι πράκτορες μπορεί να οδηγηθούν σε πράξεις που υποδεικνύουν μια συλλογικότητα αν λόγω της ισορροπίας "τρόμου" θέλουν να αποκομίσουν έστω και ένα μικρό όφελος.

Αρχικά, το περιβάλλον πλέον σχηματίζει ένα non-zero-sum παίγνιο καθώς παρέχει 3 βιβλία, 1 καπέλο και 2 μπάλες ($[3, 1, 2]$) και οι διαθέσιμες συναρτήσεις αξιών είναι οι $f_1 = [0, 8, 1]$ και $f_2 = [1, 3, 2]$. Η δημιουργία των άπληστων πρακτόρων γίνεται ως εξής. Αρχικά, ο πρώτος πράκτορας λαμβάνει ως συνάρτηση αξίας την f_1 και ανταγωνίζεται ένα τυχαίο σύστημα όπως έχει αναλυθεί ως τώρα με συνάρτηση αξίας την f_2. Η μόνη διαφορά είναι πως δεν υπάρχει συμμετρία ως προς τις συναρτήσεις αξιών των αντικειμένων που λαμβάνουν οι κάθε πράκτορες. Αντίστοιχη είναι η εκμάθηση και του δεύτερου πράκτορα με τη διαφορά οτι τώρα χρησιμοποιεί την συνάρτηση f_2 και ανταγωνίζεται τυχαίο σύστημα που έχει ως συνάρτηση αξιών την f_1. Και οι δύο πράκτορες είναι μοντέλα Double DQN. Επίσης, χρησιμοποιήθηκε στην εκμάθηση τιμωρία σε κάθε δράση της τάξης -1 και λαμβάνουν ως επιβράβευση το άθροισμα των αξιών

των αντικειμένων που συμφώνησαν να λάβουν σε περίπτωση που αυτό το άθροισμα είναι μεγαλύτερο από αυτό του αντιπάλου.

8.2.7.2 Αποτελέσματα και Συμπεράσματα

Στον πίνακα 8.10 παρουσιάζονται τα αποτελέσματα των μοντέλων που ανταγωνίζονται τον μοντέλο τυχαίας πολιτικής. Τα μοντέλα αυτά αλληλεπιδρούν μεταξύ τους και λαμβάνουν ανταμοιβές μόνο στην περίπτωση συμφωνίας. Οι ανταμοιβές αυτές αντιστοιχούν στο άθροισμα των αξιών των αντικειμένων που έλαβαν στο τέλος της διαπραγμάτευσης.

Πίνακας 8.10: Μελέτη Επίτευξης Κοινωνικής Ισορροπίας με το Dueling DQN μοντέλο με βραχυπρόθεσμη μνήμη μιας στιχομυθίας εναντίον Random Policy Model σε περιβάλλον με διάνυσμα αντικειμένων $[3, 1, 2]$ (-1 σε κάθε κίνηση)

	Μέση Ανταμοιβή	Μέσος Χρόνος	Ποσοστό Συμφωνίας %
Greedy Model [0,8,1]	4.129	4.897	73.25
Greedy Model [1,3,2]	6.013	4.1	75.32

Στον πίνακα 8.11 παρουσιάζονται τα αποτελέσματα της προσομοίωσης. Στην περίπτωση "Both Not Trainable" οι πράκτορες αλληλεπιδρούν χωρίς να βελτιστοποιούν τις παραμέτρους των μοντέλων τους. Η αντίθετη διαδικασία συμβαίνει στην περίπτωση "Both Trainable". Παρατηρούμε πως ακόμα και οι άπληστοι πράκτορες προκειμένου να συλλέξουν κάποια ανταμοιβή οδηγούνται σε κοινωνική συμπεριφορά. Το ποσοστό συμφωνίας των εκπαιδεύσιμων πρακτόρων είναι αρκετά υψηλότερο από αυτό των μη εκπαιδεύσιμων. Αξιοσημείωτο να αναφερθεί είναι το γεγονός οτι και ο μέσος χρόνος μειώνεται αισθητά που σημαίνει πως σχεδόν αμέσως οι πράκτορες συμφωνούν σε κάποια πρόταση του άλλου πράκτορα.

Πίνακας 8.11: Επίτευξη Κοινωνικής Ισορροπίας μεταξύ Άπληστων Πρακτόρων

	Μέσος Χρόνος	Ποσοστό Συμφωνίας %	Αριθμός Σημείων Ισορροπίας
Both Not Trainable	5.8211	51.99	1
Both Trainable	2.3614	97.06	8

Εξαιρετικό ενδιαφέρον παρουσιάζει και η σημαντική αύξηση των σημείων ισορροπίας. Οι αρχικά άπληστοι πράκτορες ανακαλύπτουν και νέα σημεία ισορροπίας στα οποία μπορούν να συμφωνήσουν. Η μέθοδος δίνει ελπίδες οτι μέσω μιας προσομοίωσης trial-and-error με συλλογικό σκοπό μπορεί να οδηγήσει στην εμφάνιση βέλτιστων σημείων συμφωνίας-ισορροπίας.

8.3 Συμπεράσματα

Στο Κεφάλαιο αυτό μελετήθηκε μια μέθοδος δημιουργίας αυτόματων συστημάτων διαπραγμάτευσης που δίνει έμφαση στη λήψη αποφάσεων, στις συμπεριφορές και τις στρατηγικές που μπορούν να προκύψουν. Αρχικά, σχεδιάσαμε και υλοποιήσαμε ένα περιβάλλον προσομοίωσης διαπραγματεύσεων όπου προσφέρει τη δυνατότητα μελέτης τόσο zero-sum όσο και non-zero-sum παιγνίων. Μελετήσαμε και συγκρίναμε διαφορετικά μοντέλο Q-learning. Συγκεκριμένα, μελετήθηκαν τα μοντέλα DQN, Double DQN και Dueling DQN, με τα δύο τελευταία να

επιτυγχάνουν καλύτερα αποτελέσματα ως προς τη μέση ανταμοιβή. Αυτό ίσως οφείλεται στο γεγονός οτι λύνουν τα προβλήματα του DQN που συνήθως υπερεκτιμά τα Q-values ειδικά σε θορυβώδη περιβάλλοντα.

Στη συνέχεια, μελετήσαμε τη σπουδαιότητα της Συνάρτησης Ανταμοιβής, η οποία καθορίζει πλήρως τις συμπεριφορές και κατ' επέκταση τη στρατηγική των πρακτόρων. Δημιουργήσαμε επιθετικούς και συνεργατικούς διαπραγματευτές, δώσαμε έμφαση στις ποινές που μπορούν να χρησιμοποιηθούν για να συντομεύουμε τα παίγνια και αναστρέψαμε τις συμπεριφορές των πρακτόρων αντιστρέφοντας τη Συνάρτηση Ανταμοιβής. Έπειτα, μελετήθηκε η σπουδαιότητα της επαύξησης του Observation/Space διότι θεωρείται πολύ χρήσιμο όλη η απαραίτητη πληροφορία να είναι ενσωματωμένη στο διάνυσμα που δέχεται το μοντέλο μας. Ακόμα, αποπειραθήκαμε να ανιχνεύσουμε τις στρατηγικές του αντιπάλου που ακολουθούσε μια συγκεκριμένη στρατηγική και αναγνωρίσαμε συμπεριφορές που δεν προβλέψαμε συμπεραίνοντας πως χρειάζεται αρκετή προσοχή στο πώς ορίζουμε τις Συναρτήσεις Ανταμοιβών και τι ακριβώς περιμένουμε από τα μοντέλα μας. Δώσαμε ακόμα έμφαση και στο Δίλημμα Exploration-Exploitation το οποίο αποτελεί ένα αρκετά σημαντικό ζήτημα στον κλάδο της Ενισχυτικής Μάθησης.

Επιπρόσθετα, επεκτείναμε τις υλοποιήσεις μας σε non-zero-sum παίγνια όπου αφότου δημιουργήσαμε δύο άπληστους διαπραγματευτές που αντιμετώπιζαν ως τότε, κατά τα γνωστά, μοντέλα τυχαίας πολιτικής τούς εισάγαμε σε ένα περιβάλλον προσομοίωσης όπου ο ένας αντιμετώπιζε τον άλλο. Στην περίπτωση που δεν βελτιστοποιούσαν τις παραμέτρους τους, τα αποτελέσματα συμφωνίας δεν ήταν καλά. Αντίθετα, στην περίπτωση που βελτιστοποιούσαν τις παραμέτρους τους οδηγήθηκαν σε πολλά νέα σημεία ισορροπίας, μείωσαν τον χρόνο του παιγνίου και εκτόξευσαν τα ποσοστά συμφωνίας οδηγούμενοι σε μία κοινωνική ισορροπία. Υπό μια έννοια πατάσσουν τα εγωκεντρικά τους οφέλη και οδηγούνται σε μία κοινωνική ισορροπία. Εμφανίζει εξαιρετικό ενδιαφέρον η ιδέα της δημιουργίας ισορροπίας μέσω αντίθετων αντικρουόμενων δυνάμεων.

Τέλος, προτείναμε μια μέθοδο οπτικοποίησης των στρατηγικών των μοντέλων μας προκειμένου να είμαστε σε θέση να δημιουργούμε συστήματα τα οποία δεν εξάγουν απλά μία ενέργεια αλλά μας δίνουν αρκετές πληροφορίες σχετικά με το γιατί οδηγήθηκαν σε αυτή τη συμπεριφορά. Πιο συγκεκριμένα, δεδομένης της εισόδου που εισάγεται στο μοντέλο μας και περιέχει όλη την απαραίτητη πληροφορία που χρειαζόμαστε για να προσδιοριστεί πλήρως και σαφώς το παίγνιο, μπορούμε να εξάγουμε όλες τις δυνατές ενέργειες-επιλογές με τα αντίστοιχα Q-value και τα αντίστοιχα διαπραγματευτικά αποτελέσματα. Για λόγους απλότητας, παρουσιάσαμε στους Πίνακες 8.2, 8.6 και 8.8 μόνο την απόφαση με τη βέλτιστη τιμή Q. Τέλος, ας τονίσουμε πως δεν υφίσταται η έννοια της καλύτερης στρατηγικής. Αντιθέτως, υφίσταται η έννοια της καλύτερης στρατηγικής δεδομένου του αντιπάλου, του περιβάλλοντος και των παραμέτρων που τους διέπουν.

8.4 Μελλοντικές Προεκτάσεις

Σε αυτή την Ενότητα θα αναφερθούμε σε ορισμένες μελλοντικές προεκτάσεις και σε κάποιες τροποποιήσεις που θα προτείναμε για μελλοντική μελέτη. Συγκεκριμένα,

- θα μπορούσε κανείς να χρησιμοποιήσει ένα σύστημα που διαπραγματεύεται παράλληλα σε πολλά contexts και value functions. Η χρησιμότητα της οπτικής που μελετήθηκε ωχ τώρα δίνει έμφαση σε ένα context και ένα value function με το σκεπτικό ότι αν

υπάρξουν πολλά contexts θα υπάρξουν πολλά εξειδικευμένα μοντέλα που λειτουργούν παράλληλα

- το συγκεκριμένο framework μπορεί πολύ εύκολα να γενικευτεί και σε πολυπρακτορικά προβλήματα αναμένοντας πολύ πιο σύνθετες επιλογές
- αντιστρέφοντας τη Συνάρτηση Ανταμοιβής, όπως είδαμε, μπορούμε να μετατρέψουμε το διαπραγματευτή σε πωλητή αντικειμένων
- ενδιαφέρον θα είχε και η δυνατότητα πρόβλεψης των χαρακτηριστικών του αντιπάλου μετατρέποντας το RL πρόβλημα από model-free σε model-based
- από τη στιγμή που δεδομένης μιας κατάστασης προκύπτουν οι ενέργειες με τις αντίστοιχες Q-τιμές, θα είχε ενδιαφέρον να αναζητήσουμε νέες μεθόδους παραγωγής απαντήσεων (π.χ., με δειγματοληψία όπως στην περίπτωση παραγωγής κειμένου). Αν εισαχθούν μη-άπληστες επιλογές, θα έχει νόημα και η διαδικασία του self-play, όπου το μοντέλο θα ανταγωνίζεται διαρκώς μία παλιότερη έκδοση του εαυτού του
- από τη στιγμή που δεδομένης μιας κατάστασης προκύπτουν οι ενέργειες με τις αντίστοιχες Q-τιμές, θα είχε ενδιαφέρον η δημιουργία rollouts. Θα μπορούσε, δηλαδή, το μοντέλο μας να συμπληρώνει αυτόματα μελλοντικές πιθανές εκβάσεις των διαλόγων δεδομένης της κατάστασης που βρίσκεται και με κάποια δενδρική αναζήτηση να επιλέγει την εκάστοτε κατάλληλη ενέργεια

Κεφάλαιο 9

Συμπεράσματα - Μελλοντικές Προεκτάσεις

Για τους λόγους που αναφέρθηκαν στην Ενότητα 1.1, στην παρούσα διπλωματική εργασία δόθηκε έμφαση σε προβλήματα καθορισμένου σκοπού (task-oriented). Ένα από τα δυσκολότερα είναι το πρόβλημα της διαπραγμάτευσης διότι απαιτεί ένα συνδυασμό γλωσσικής επάρκειας και συλλογιστικής ικανότητας.

9.1 Σύνοψη και Συμπεράσματα

Όσον αφορά τη Σύνοψη και τα Αποτελέσματα της Διπλωματικής Εργασίας προτείνεται πέρα από τα παρακάτω κείμενα να μελετηθούν και οι Ενότητες 7.4 και 8.3.

9.1.1 Εφαρμογή End-to-End Task-oriented Διαλογικών Συστημάτων σε Προβλήματα Διαπραγμάτευσης

Αρχικά, μελετήθηκε το πρόβλημα Deal Or No Deal που εισάγεται από τη Facebook και εστιάζει στη διαπραγμάτευση δύο πρακτόρων με δεδομένο πλήθος αντικειμένων μέσω ενός καναλιού επικοινωνίας σε γραπτό λόγο. Σε ένα πρόβλημα διαπραγμάτευσης, θα πρέπει το σύστημα να δέχεται ένα κείμενο, να κατανοεί σε τι διαμοιρασμό αντικειμένων αντιστοιχεί η είσοδος, να επεξεργάζεται την κατάλληλη απόφαση και εν τέλει να παράγει γλώσσα προκειμένου να μεταδώσει το μήνυμα της ενέργειας που θα λάβει. Σε μια πρώτη προσέγγιση το πρόβλημα αναλύεται σε δύο υποπροβλήματα: (1) στο Πρόβλημα Ταξινόμησης και (2) στο Πρόβλημα της Γλωσσικής Μοντελοποίησης. Αρχικά, χρησιμοποιείται ο κώδικας της Facebook για το εν λόγω πρόβλημα [142]. Αφότου αναπαρήχθησαν τα αποτελέσματα της δημοσίευσης [65], όπου γίνεται χρήση end-to-end αρχιτεκτονικών, τροποποιήσαμε τα υπάρχοντα μοντέλα, επαυξάνοντας τον κώδικα με άλλα παραδοσιακά μοντέλα μηχανικής μάθησης και με χρήση προ-εκπαιδευμένων διανυσμάτων λέξεων Glove. Τα αποτελέσματα παρουσιάζουν ελάχιστες διαφοροποιήσεις. Παρόλα αυτά, στο πρόβλημα της Ταξινόμησης εντοπίστηκε οτι χρησιμοποιείται μια μετρική αξιολόγησης που δεν συνάδει με τη σπουδαιότητα επίλυσης του προβλήματος. Συγκεκριμένα, στοχεύει, δηλαδή, στην ελαχιστοποίηση του σφάλματος παρά στην ουσία του ζητήματος που είναι η σωστή κατανόηση της πρότασης του άλλου πράκτορα. Για αυτό τον λόγο, αντί για τη μετρική της σύγχυσης στο πρόβλημα ταξινόμησης προτείναμε

να χρησιμοποιηθεί η μετρική της ακρίβειας (accuracy). Έχοντας ένα νέο μέτρο σύγκρισης, υλοποιούμε αρχικά παραδοσιακούς ταξινομητές μηχανικής μάθησης που βασίζονται στα Απλά RNNs, GRUs, LSTMs, CNNs, τα οποία ενισχύονται με τη χρήση Glove (frozen ή unfrozen). Τα καλύτερα αποτελέσματα προκύπτουν από τον Ταξινομητή που χρησιμοποιεί CNN χωρίς προ-εκπαιδευμένες διανυσματικές αναπαραστάσεις λέξεων Glove, επιτυγχάνοντας ποσοστό επιτυχίας 22.6%. Στη συνέχεια, χρησιμοποιούμε έναν Ταξινομητή που βασίζεται στο BERT το οποίο προσφέρει εξαιρετικά αποτελέσματα ακρίβειας της τάξης του 42%. Επιστρέφοντας στο πρόβλημα της γλωσσικής μοντελοποίησης, θελήσαμε να αποδείξουμε οτι και σε αυτή την κατηγορία προβλήματος, η χρήση Μεταφοράς Μάθησης με τη μορφή των Transformers μπορεί να βελτιώσει αισθητά τα αποτελέσματα. Συγκεκριμένα, χρησιμοποιήθηκε ο Transformer GPT-2 και τα αποτελέσματα της σύγχυσης εμφανίζονται αρκετά χαμηλότερα (όσο χαμηλότερη η μετρική της σύγχυσης τόσο καλύτερα τα αποτελέσματα). Στη συνέχεια, κάναμε χρήση του εκπαιδευμένου και βελτιστοποιημένου, πάνω στο σύνολο διαλόγων μας, GPT-2 προκειμένου να παράξουμε γλώσσα με τη μορφή διαλόγων και χρησιμοποιήσαμε ενδεικτικά 4 τακτικές: τη greedy decoding, τη beam search, την top-k δειγματοληψία και τέλος την top-p (nucleus) δειγματοληψία.

Μελετήσαμε, λοιπόν, μια end-to-end αρχιτεκτονική και αποδομήσαμε τα υποπροβλήματα που επιλύει προτείνοντας έμμεσα και μία modular αρχιτεκτονική. Τα πλεονεκτήματα και τα μειονεκτήματα των αρχιτεκτονικών επιλογών πρέπει να ζυγίζονται από τον μηχανικό που τα σχεδιάζει ανάλογα με το εκάστοτε πρόβλημα. Με αυτή τη μελέτη προτείναμε πώς μπορεί αρχικά να δομηθεί ένας σωστός ταξινομητής που να εστιάζει στην ουσία του ζητήματος για το οποίο υλοποιείται και πώς μπορούμε να παράξουμε ένα μοντέλο που δημιουργεί πιθανές συνέχειες διαλόγων (rollouts).

9.1.2 Ανάδειξη Στρατηγικών και Συμπεριφορών Πρακτόρων σε Περιβάλλοντα χωρίς Γλωσσικές Εξαρτήσεις με χρήση Ενισχυτικής Μάθησης

Έως τώρα η πολιτική λήψης αποφάσεων και η σχεδίαση στρατηγικών αγνοούνταν ή λαμβάνονταν υπόψη σε μεταγενέστερο στάδιο. Θεωρήσαμε, βέβαια, πως η απόφαση και η ενέργεια προϋπάρχει του μηνύματος και για αυτό σχεδιάσαμε και υλοποιήσαμε ένα περιβάλλον προσομοίωσης αυτόματης διαπραγμάτευσης το οποίο είναι απεξαρτημένο από τα γλωσσικά χαρακτηριστικά και εστιάζει σε χαμηλού επιπέδου αναπαραστάσεις των ενεργειών. Με αυτόν τον τρόπο επιχειρήσαμε να δώσουμε έμφαση στον πυρήνα του διαπραγματευτικού συστήματος, όπου οι συμπεριφορές των πρακτόρων αναδύθηκαν μέσω προσομοιώσεων trial-and-error με τη χρήση ενισχυτικής μάθησης και συγκεκριμένα της μεθόδου Q-learning. Αρχικά, συγκρίθηκαν 3 είδη DQN μοντέλων: το απλό DQN, το Double DQN και το Duel DQN με τα δύο τελευταία να επιτυγχάνουν καλύτερα αποτελέσματα μέσης ανταμοιβής με λίγο αυξημένο μέσο χρόνο παιγνίου. Τα ποσοστά συμφωνίας στην περίπτωση του απλού DQN ήταν ανεπαίσθητα λίγο καλύτερα. Στη συνέχεια, μελετώντας την επίδραση διαφορετικών συναρτήσεων ανταμοιβών επιτύχαμε να δημιουργήσουμε πράκτορες με συγκεκριμένες συμπεριφορές, οι οποίες είναι ρυθμίσιμες. Εξετάσαμε, ακόμα, το βέλτιστο τρόπο απεικόνισης της διαπραγματευτικής πληροφορίας όπου προτείνεται η επαύξηση της ιστορίας στο διάνυσμα της παρατήρησης του πράκτορα και δημιουργήσαμε ένα σύστημα που επιχειρεί να εκμεταλλευτεί συγκεκριμένες συμπεριφορές του αντιπάλου του. Ακόμα, μελετήθηκε το δίλημμα Exploration vs Exploitation χρησιμοποιώντας τη μέθοδο ε-greedy. Ως τώρα τα συστήματα αυτά προσομοιώνονται αντιμετωπίζοντας ένα μοντέλο τυχαίας πολιτικής. Δημιουργώντας δύο άπληστους πράκτορες με το τρόπο αυτό, τους εισάγουμε να διαπραγματευτούν μεταξύ τους και αναμένουμε, όπως

επιβεβαιώθηκε, πως τα ποσοστά συμφωνίας και τα σημεία ισορροπίας συμφωνίας ήταν αρκετά χαμηλά. Αλληλεπιδρώντας, πλέον, μεταξύ τους πέτυχαν τη σύγκλιση σε πολλά νεότερα σημεία κοινωνικής ισορροπίας και σχεδόν απόλυτα ποσοστά συμφωνίας.

Αναγνωρίζοντας τη σπουδαιότητα της ερμηνευσιμότητας των αποτελεσμάτων και των στρατηγικών των μεθόδων Μηχανικής και Ενισχυτικής Μάθησης, προτάθηκε και υλοποιήθηκε μια μέθοδος οπτικοποίησης των συμπεριφορών που έχει μάθει ένα σύστημα κατά την εκπαίδευσή του. Αυτή η πρόταση δίνει πλήρη έλεγχο και διαφάνεια στις ενέργειες του πράκτορα. Δεδομένης της ιδιομορφίας του προβλήματος οτι ο χώρος των δυνατών εκβάσεων είναι περιορισμένος, για κάθε δυνατή κατάσταση, εξάγουμε από τον πράκτορα την ενέργεια που θα λάβει μαζί με μία τιμή που αξιολογεί την ενέργεια αυτή. Να σημειωθεί πως εμείς λαμβάνουμε ως απόφαση, αυτή που φέρει το μεγαλύτερο Q-value.

9.2 Μελλοντικές Προεκτάσεις

Η μελέτη αυτή φέρει αρκετές και ενδιαφέρουσες μελλοντικές προεκτάσεις. Κάποιες από αυτές παρουσιάζονται παρακάτω. Πέρα από τις προεκτάσεις που μελετώνται στην παρούσα Ενότητα προτείνεται να μελετηθούν και οι Ενότητες 7.5 και 8.4.

- Θα μπορούσε κανείς να χρησιμοποιήσει ένα σύστημα που διαπραγματεύεται παράλληλα σε πολλά contexts και value functions. Η χρησιμότητα της οπτικής που μελετήθηκε δίνει έμφαση σε ένα context και ένα value function με το σκεπτικό ότι αν υπάρξουν πολλά contexts θα υπάρξουν πολλά εξειδικευμένα μοντέλα που λειτουργούν παράλληλα

- Εξαιρετικό ενδιαφέρον θα παρουσίαζε η παραγωγή αποφάσεων με beam search ή με κάποια μορφή δειγματοληψίας. Ακόμα, με αυτό τον τρόπο είναι εύκολη η δημιουργία rollouts, μελλοντικών, δηλαδή, εκβάσεων των διαλόγων και η αναζήτηση της κατάλληλης κίνησης στο δέντρο πιθανών εκβάσεων

- Ακόμα, το περιβάλλον που περιγράφεται μπορεί αρκετά εύκολα να επεκταθεί σε ένα πολυπρακτορικό περιβάλλον, όπου εκεί αναμένονται πιο σύνθετες συμπεριφορές

- Εμείς ως τώρα μελετάμε model-free Ενισχυτική Μάθηση. Ενδιαφέρον θα είχε να εστιάσουμε στη μοντελοποίηση του αντιπάλου και να γίνει πιο προσωποποιημένη η διαπραγμάτευση. Πάντως, ως τώρα, θα μπορούσαμε να δημιουργήσουμε κάποιες πεπερασμένες κλάσεις αυτόματων διαπραγματευτών με ποικιλία συμπεριφορών και να προσπαθούμε τους εκάστοτε αντιπάλους να τους κατατάσσουμε σε μία από τις πιθανές κατηγορίες όπου θα αντιμετωπίσουν κάποιο συγκεκριμένο σύστημα

- Αν θέλουμε να λύσουμε στην ολότητά του το πρόβλημα χρειάζεται να συνδυάσουμε σε ένα ενιαίο modular σύστημα τα προβλήματα που λύσαμε. Το πρώτο σύστημα θα αποτελούταν από τον ταξινομητή BERT ο οποίος μεταφράζει το κείμενο σε μία πρόταση του "αντίπαλου" πράκτορα. Αυτή η πρόταση εισάγεται εντός του αυτόματου διαπραγματευτικού συστήματος το οποίο λαμβάνει μια συγκεκριμένη ενέργεια και αυτή ενέργεια χρειάζεται να αποδοθεί στον αντίπαλο. Αρχικά, εκεί μπορεί να χρησιμοποιηθεί ένα απλό σύστημα που με φτωχή γλώσσα θα δίνει μια απάντηση ή να υλοποιηθεί ένα σύστημα παραγωγής λόγου που θα δέχεται ως είσοδο τον διάλογο και την απόφαση και θα παράγει το αντίστοιχο κείμενο. Αν θέλουμε, οπωσδήποτε να κάνουμε χρήση του γλωσσικού

μοντέλου GPT-2 που έχουμε χρησιμοποιήσει μπορούμε να παράγουμε διαρκώς προτάσεις τις οποίες θα ταξινομούμε ταυτόχρονα έως ότου παράξουμε αυτή την πρόταση που αντιστοιχεί στην ενέργεια που θα λάβει το σύστημά μας.

Βιβλιογραφία

[1] Rami Al-Rfou, Dokook Choe, Noah Constant, Mandy Guo, and Llion Jones. Character-level language modeling with deeper self-attention. *CoRR*, abs/1808.04444, 2018.

[2] Jay Alammar. The illustrated transformer. http://jalammar.github.io/illustrated-transformer/, 2018.

[3] Jay Alammar. The illustrated gpt-2 (visualizing transformer language models). http://jalammar.github.io/illustrated-gpt2/, 2019.

[4] Tim Baarslag, Katsuhide Fujita, Enrico H. Gerding, Koen Hindriks, Takayuki Ito, Nicholas R. Jennings, Catholijn Jonker, Sarit Kraus, Raz Lin, Valentin Robu, and Colin R. Williams. Evaluating practical negotiating agents: Results and analysis of the 2011 international competition. *Artificial Intelligence*, 198:73 – 103, 2013.

[5] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2015.

[6] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. 01 2005.

[7] Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994.

[8] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. A neural probabilistic language model. *J. Mach. Learn. Res.*, 3(null):1137–1155, March 2003.

[9] Daniel G. Bobrow, Ronald M. Kaplan, Martin Kay, Donald A. Norman, Henry Thompson, and Terry Winograd. Gus, a frame-driven dialog system. *Artificial Intelligence*, 8(2):155 – 173, 1977.

[10] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *in COMPSTAT*, 2010.

[11] Kris Cao, Angeliki Lazaridou, Marc Lanctot, Joel Z Leibo, Karl Tuyls, and Stephen Clark. Emergent communication through negotiation, 2018.

[12] Senthilkumar Chandramohan, Matthieu Geist, Fabrice Lefèvre, and Olivier Pietquin. User simulation in dialogue systems using inverse reinforcement learning. In *INTERSPEECH*, 2011.

[13] Junyoung Chung, Çaglar Gülçehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *CoRR*, abs/1412.3555, 2014.

[14] Kenneth Mark Colby, Sylvia Weber, and Franklin Dennis Hilf. Artificial paranoia. *Artificial Intelligence*, 2(1):1 – 25, 1971.

[15] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel P. Kuksa. Natural language processing (almost) from scratch. *CoRR*, abs/1103.0398, 2011.

[16] Mark Core and James Allen. Coding dialogs with the damsl annotation scheme. 01 2001.

[17] Heriberto Cuayáhuitl, Simon Keizer, and Oliver Lemon. Strategic dialogue management via deep reinforcement learning, 2015.

[18] Andrew M. Dai and Quoc V. Le. Semi-supervised sequence learning, 2015.

[19] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G. Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. *CoRR*, abs/1901.02860, 2019.

[20] Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José M. F. Moura, Devi Parikh, and Dhruv Batra. Visual dialog. *CoRR*, abs/1611.08669, 2016.

[21] Abhishek Das, Satwik Kottur, José M. F. Moura, Stefan Lee, and Dhruv Batra. Learning cooperative visual dialog agents with deep reinforcement learning. *CoRR*, abs/1703.06585, 2017.

[22] David Devault, Johnathan Mell, and Jonathan Gratch. Toward natural turn-taking in a virtual human negotiation agent. 03 2015.

[23] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018.

[24] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, 07 2011.

[25] Wieland Eckert, Esther Levin, and R. Pieraccini. User modeling for spoken dialogue system evaluation. pages 80 – 87, 01 1998.

[26] Angela Fan, Mike Lewis, and Yann N. Dauphin. Hierarchical neural story generation. *CoRR*, abs/1805.04833, 2018.

[27] William A. Gale and Kenneth W. Church. What's wrong with adding one. In *Corpus-Based Research into Language. Rodolpi*, 1994.

[28] Jianfeng Gao, Michel Galley, and Lihong Li. Neural approaches to conversational AI. *CoRR*, abs/1809.08267, 2018.

[29] Jon Gauthier and Igor Mordatch. A paradigm for situated and goal-driven language learning. *CoRR*, abs/1610.03585, 2016.

[30] Marjan Ghazvininejad, Chris Brockett, Ming-Wei Chang, Bill Dolan, Jianfeng Gao, Wen-tau Yih, and Michel Galley. A knowledge-grounded neural conversation model. *CoRR*, abs/1702.01932, 2017.

[31] Gene H. Golub and Christian Reinsch. Singular value decomposition and least squares solutions. *Numerische Mathematik*, 14:403–420, 2007.

[32] G.H. Golub and C.F. Van Loan. *Matrix Computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, 2013.

[33] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. The MIT Press, 2016.

[34] Sam Greydanus, Anurag Koul, Jonathan Dodge, and Alan Fern. Visualizing and understanding atari agents. *CoRR*, abs/1711.00138, 2017.

[35] Mikko Hartikainen, Esa-Pekka Salonen, and Markku Turunen. Subjective evaluation of spoken dialogue systems using servqual method. 01 2004.

[36] Karl Moritz Hermann, Felix Hill, Simon Green, Fumin Wang, Ryan Faulkner, Hubert Soyer, David Szepesvari, Wojciech Marian Czarnecki, Max Jaderberg, Denis Teplyashin, Marcus Wainwright, Chris Apps, Demis Hassabis, and Phil Blunsom. Grounded language learning in a simulated 3d world. *CoRR*, abs/1706.06551, 2017.

[37] Sepp Hochreiter. Untersuchungen zu dynamischen neuronalen netzen. 1991.

[38] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9:1735–80, 12 1997.

[39] Ari Holtzman, Jan Buys, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. *CoRR*, abs/1904.09751, 2019.

[40] Jeremy Howard and Sebastian Ruder. Fine-tuned language models for text classification. *CoRR*, abs/1801.06146, 2018.

[41] HuggingFace. Text generation. https://huggingface.co/blog/how-to-generate.

[42] HuggingFace. Transformers documentation. https://huggingface.co/transformers/.

[43] HuggingFace. Transformers documentation: Bert tokenizer. https://huggingface.co/transformers/model_doc/bert.html#berttokenizer.

[44] HuggingFace. Transformers documentation: Bertforsequenceclassification model. https://huggingface.co/transformers/model_doc/bert.html#bertforsequenceclassification.

[45] HuggingFace. Transformers language modeling code. https://github.com/huggingface/transformers/blob/master/examples/language-modeling.

[46] Ozan Irsoy and Claire Cardie. Bidirectional recursive neural networks for token-level labeling with structure. *CoRR*, abs/1312.0493, 2013.

[47] Sina Jafarpour and Chris J.C. Burges. Filter, rank, and transfer the knowledge: Learning to chat. Technical Report MSR-TR-2010-93, July 2010.

[48] Sangkeun Jung, Cheongjae Lee, Kyungduk Kim, Minwoo Jeong, and Gary Lee. Data-driven user simulation for automated evaluation of spoken dialog systems. *Computer Speech and Language*, 23(4):479–509, 10 2009.

[49] Daniel Jurafsky and James H. Martin. *Speech and Language Processing (2nd Edition)*. Prentice-Hall, Inc., USA, 2009.

[50] Anjuli Kannan and Oriol Vinyals. Adversarial evaluation of dialogue models. *CoRR*, abs/1701.08198, 2017.

[51] Andrej Karpathy. Cs231n convolutional neural networks for visual recognition. n. http:// cs231n.github.io/neural-networks-1,.

[52] Simon Keizer, Markus Guhe, Heriberto Cuayáhuitl, Ioannis Efstathiou, Klaus-Peter Engelbrecht, Mihai Dobre, Alex Lascarides, and Oliver Lemon. Evaluating persuasion strategies and deep reinforcement learning methods for negotiation dialogue agents. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 480–484, Valencia, Spain, April 2017. Association for Computational Linguistics.

[53] Nitish Shirish Keskar, Bryan McCann, Lav R. Varshney, Caiming Xiong, and Richard Socher. Ctrl: A conditional transformer language model for controllable generation, 2019.

[54] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.

[55] Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M. Rush. Opennmt: Open-source toolkit for neural machine translation. *CoRR*, abs/1701.02810, 2017.

[56] R. Kneser and H. Ney. Improved backing-off for m-gram language modeling. In *1995 International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 181–184 vol.1, 1995.

[57] Guillaume Lample and Alexis Conneau. Cross-lingual language model pretraining. *CoRR*, abs/1901.07291, 2019.

[58] Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. Multi-agent cooperation and the emergence of (natural) language. 12 2016.

[59] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[60] Anton Leuski and David Traum. Npceditor: Creating virtual human dialogue using information retrieval techniques. *AI Magazine*, 32(2):42–56, Mar. 2011.

[61] E. Levin, R. Pieraccini, and W. Eckert. A stochastic model of human-machine interaction for learning dialog strategies. *IEEE Transactions on Speech and Audio Processing*, 8(1):11–23, 2000.

[62] Sergey Levine. Cs 285 at uc berkeley: Deep reinforcement learning. http://rail.eecs.berkeley.edu/deeprlcourse/, 2019.

[63] R.J. Lewicki, D.M. Saunders, and J.W. Minton. *Essentials of Negotiation*. Management & Organization series. Irwin/McGraw-Hill, 2001.

[64] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, 2019.

[65] Mike Lewis, Denis Yarats, Yann N. Dauphin, Devi Parikh, and Dhruv Batra. Deal or no deal? end-to-end learning for negotiation dialogues. *CoRR*, abs/1706.05125, 2017.

[66] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. A diversity-promoting objective function for neural conversation models. *CoRR*, abs/1510.03055, 2015.

[67] Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, and Dan Jurafsky. Deep reinforcement learning for dialogue generation. *CoRR*, abs/1606.01541, 2016.

[68] Jiwei Li, Will Monroe, Tianlin Shi, Alan Ritter, and Dan Jurafsky. Adversarial learning for neural dialogue generation. *CoRR*, abs/1701.06547, 2017.

[69] Xiujun Li, Zachary C. Lipton, Bhuwan Dhingra, Lihong Li, Jianfeng Gao, and Yun-Nung Chen. A user simulator for task-completion dialogues. *CoRR*, abs/1612.05688, 2016.

[70] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics.

[71] Peter J. Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. Generating wikipedia by summarizing long sequences. *CoRR*, abs/1801.10198, 2018.

[72] Ryan Thomas Lowe, Nissan Pow, Iulian Serban, Laurent Charlin, Chia-Wei Liu, and Joelle Pineau. Training end-to-end dialogue systems with the ubuntu dialogue corpus. *Dialogue Discourse*, 8:31–65, 2017.

[73] Junhua Mao, Wei Xu, Yi Yang, Jiang Wang, Zhiheng Huang, and Alan L. Yuille. Learning like a child: Fast novel visual concept learning from sentence descriptions of images. *CoRR*, abs/1504.06692, 2015.

[74] Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. Learned in translation: Contextualized word vectors, 2017.

[75] Chris McCormick and Nick Ryan. Bert fine-tuning tutorial with pytorch. https://mccormickml.com/2019/07/22/BERT-fine-tuning/, 2019.

[76] Michael F. McTear. Spoken dialogue technology: Enabling the conversational user interface. *ACM Comput. Surv.*, 34(1):90–169, March 2002.

[77] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013.

[78] Tomas Mikolov, Armand Joulin, and Marco Baroni. A roadmap towards machine intelligence, 2015.

[79] Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernocký, and Sanjeev Khudanpur. Recurrent neural network based language model. volume 2, pages 1045–1048, 01 2010.

[80] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. *CoRR*, abs/1310.4546, 2013.

[81] Tom M. Mitchell. *Machine Learning*. McGraw-Hill, New York, 1997.

[82] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin A. Riedmiller. Playing atari with deep reinforcement learning. *CoRR*, abs/1312.5602, 2013.

[83] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei Rusu, Joel Veness, Marc Bellemare, Alex Graves, Martin Riedmiller, Andreas Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518:529–33, 02 2015.

[84] Goku Mohandas. Recurrent neural netwroks (rnn). https://github.com/madewithml/basics.

[85] Igor Mordatch and Pieter Abbeel. Emergence of grounded compositional language in multi-agent populations. *CoRR*, abs/1703.04908, 2017.

[86] John Nash. The bargaining problem. *Econometrica*, 18(2):155–162, 1950.

[87] Christopher Olah. Neural networks, types, and functional programming. https://colah.github.io/posts/2015-09-NN-Types-FP/, 2015.

[88] Christopher Olah. Understanding lstm networks, 2015.

[89] Tim Paek. Empirical methods for evaluating dialog systems. In *Proceedings of the Second SIGdial Workshop on Discourse and Dialogue*, 2001.

[90] Kishore Papineni, Salim Roukos, Todd Ward, and Wei Jing Zhu. Bleu: a method for automatic evaluation of machine translation. 10 2002.

[91] Romain Paulus, Caiming Xiong, and Richard Socher. A deep reinforced model for abstractive summarization. *CoRR*, abs/1705.04304, 2017.

[92] Min Peng, Chongyang Wang, Tong Chen, Guangyuan Liu, and Xiaolan Fu. Dual temporal scale convolutional neural network for micro-expression recognition. *Frontiers in Psychology*, 8, 2017.

[93] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[94] Matthew E. Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power. Semi-supervised sequence tagging with bidirectional language models. *CoRR*, abs/1705.00108, 2017.

[95] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations, 2018.

[96] Joelle Pineau and Sebastian Thrun. Spoken dialog management for robots. 03 2000.

[97] D.G. Pruitt and P.J. Carnevale. *Negotiation in social conflict*. Mapping social psychology. Open University Press, 1993.

[98] Alec Radford. Improving language understanding by generative pre-training. 2018.

[99] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[100] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2019.

[101] Alan Ritter, Colin Cherry, and William B. Dolan. Data-driven response generation in social media. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 583–593, Edinburgh, Scotland, UK., July 2011. Association for Computational Linguistics.

[102] Avi Rosenfeld, Inon Zuckerman, Erel Segal-Halevi, Osnat Drein, and Sarit Kraus. Negochat-a: a chat-based negotiation agent with bounded rationality. *Autonomous Agents and Multi-Agent Systems*, 30, 02 2015.

[103] D. E. Rumelhart and J. L. McClelland. *Learning Internal Representations by Error Propagation*, pages 318–362. 1987.

[104] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Fei-Fei Li. Imagenet large scale visual recognition challenge. *CoRR*, abs/1409.0575, 2014.

[105] J. Schatzmann and S. Young. The hidden agenda user simulation model. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(4):733–747, 2009.

[106] Jost Schatzmann, Kallirroi Georgila, and Steve J. Young. Quantitative evaluation of user simulation techniques for spoken dialogue systems. 2005.

[107] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. *CoRR*, abs/1508.07909, 2015.

[108] Iulian Vlad Serban, Ryan Lowe, Peter Henderson, Laurent Charlin, and Joelle Pineau. A survey of available corpora for building data-driven dialogue systems. *CoRR*, abs/1512.05742, 2015.

[109] Pararth Shah, Dilek Hakkani-Tür, Bing Liu, and Gokhan Tür. Bootstrapping a neural conversational agent with dialogue self-play, crowdsourcing and on-line reinforcement learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 3 (Industry Papers)*, pages 41–51, New Orleans - Louisiana, June 2018. Association for Computational Linguistics.

[110] Pararth Shah, Dilek Z. Hakkani-Tur, and Larry Heck. Interactive reinforcement learning for task-oriented dialogue management. 2016.

[111] Lifeng Shang, Zhengdong Lu, and Hang Li. Neural responding machine for short-text conversation. 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)*, pages 1577–1586, Beijing, China, July 2015. Association for Computational Linguistics.

[112] Louis Shao, Stephan Gouws, Denny Britz, Anna Goldie, Brian Strope, and Ray Kurzweil. Generating long and diverse responses with neural conversation models. *CoRR*, abs/1701.03185, 2017.

[113] David Silver. Advanced topics 2015 (compm050/compgi13): Reinforcement learning. https://www.davidsilver.uk/teaching/, 2015.

[114] S. Singh, D. Litman, M. Kearns, and M. Walker. Optimizing dialogue management with reinforcement learning: Experiments with the njfun system. *Journal of Artificial Intelligence Research*, 16:105–133, Feb 2002.

[115] Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. A neural network approach to context-sensitive generation of conversational responses. *CoRR*, abs/1506.06714, 2015.

[116] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014.

[117] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. *CoRR*, abs/1409.3215, 2014.

[118] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. A Bradford Book, Cambridge, MA, USA, 2018.

[119] Gerald Tesauro. Temporal difference learning and td-gammon. *Commun. ACM*, 38(3):58–68, March 1995.

[120] David Traum. Speech acts for dialogue agents. 06 1999.

[121] A. M. TURING. I.—COMPUTING MACHINERY AND INTELLIGENCE. *Mind*, LIX(236):433–460, 10 1950.

[122] Stefan Ultes, Lina M. Rojas-Barahona, Pei-Hao Su, David Vandyke, Dongho Kim, Iñigo Casanueva, Paweł Budzianowski, Nikola Mrkšić, Tsung-Hsien Wen, Milica Gašić, and Steve Young. PyDial: A multi-domain statistical dialogue system toolkit. In *Proceedings of ACL 2017, System Demonstrations*, pages 73–78, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[123] Laurens van der Maaten and Geoffrey E. Hinton. Visualizing data using t-sne. 2008.

[124] Hado van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. *CoRR*, abs/1509.06461, 2015.

[125] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.

[126] Ashwin K. Vijayakumar, Michael Cogswell, Ramprasaath R. Selvaraju, Qing Sun, Stefan Lee, David J. Crandall, and Dhruv Batra. Diverse beam search: Decoding diverse solutions from neural sequence models. *CoRR*, abs/1610.02424, 2016.

[127] Oriol Vinyals and Quoc V. Le. A neural conversational model. *CoRR*, abs/1506.05869, 2015.

[128] John von Neumann, Oskar Morgenstern, and Ariel Rubinstein. *Theory of Games and Economic Behavior (60th Anniversary Commemorative Edition)*. Princeton University Press, 1944.

[129] MA Walker, DJ Litman, CA Kamm, and A Abella. Evaluating spoken dialogue agents with paradise: Two case studies. *Computer Speech and Language*, 12(4):317 – 347, 1998.

[130] Marilyn Walker, Candace Kamm, and Diane Litman. Towards developing general models of usability with paradise. *Nat. Lang. Eng.*, 6(3–4):363–377, September 2000.

[131] Marilyn A. Walker. An application of reinforcement learning to dialogue strategy selection in a spoken dialogue system for email. *CoRR*, abs/1106.0241, 2011.

[132] Marilyn A. Walker, Diane J. Litman, Candace A. Kamm, and Alicia Abella. PARADISE: A framework for evaluating spoken dialogue agents. In *35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics*, pages 271–280, Madrid, Spain, July 1997. Association for Computational Linguistics.

[133] Hao Wang, Zhengdong Lu, Hang Li, and Enhong Chen. A dataset for research on short-text conversations. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 935–945, Seattle, Washington, USA, October 2013. Association for Computational Linguistics.

[134] Ziyu Wang, Nando de Freitas, and Marc Lanctot. Dueling network architectures for deep reinforcement learning. *CoRR*, abs/1511.06581, 2015.

[135] Christopher J.C.H. Watkins and Peter Dayan. Technical note: Q-learning. *Machine Learning*, 8:279–292, 1992.

[136] Joseph Weizenbaum. Eliza—a computer program for the study of natural language communication between man and machine. *Commun. ACM*, 9(1):36–45, January 1966.

[137] Sean Welleck, Ilia Kulikov, Stephen Roller, Emily Dinan, Kyunghyun Cho, and Jason Weston. Neural text generation with unlikelihood training, 2019.

[138] Jason D. Williams and Steve Young. Partially observable markov decision processes for spoken dialog systems. *Computer Speech and Language*, 21(2):393 – 422, 2007.

[139] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan

Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144, 2016.

[140] Yilin Yang, Liang Huang, and Mingbo Ma. Breaking the beam search curse: A study of (re-)scoring methods and stopping criteria for neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3054–3059, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[141] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. *CoRR*, abs/1906.08237, 2019.

[142] Denis Yarats and Mike Lewis. Hierarchical text generation and planning for strategic dialogue. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, pages 5587–5595, 2018.

[143] Matthew D. Zeiler. ADADELTA: an adaptive learning rate method. *CoRR*, abs/1212.5701, 2012.

[144] Li Zhou, Jianfeng Gao, Di Li, and Heung-Yeung Shum. The design and implementation of xiaoice, an empathetic social chatbot, 2018.