ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΣΗΜΑΤΩΝ, ΕΛΕΓΧΟΥ ΚΑΙ ΡΟΜΠΟΤΙΚΗΣ

Δημιουργία συστάσεων σε διάλογο με χρήση contextual bandits

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Λεωνίδα Αβδελά

Επιβλέπων: -Εισάγετε το όνομα, αρχικό πατρώνυμο και επίθετο του επιβλέποντα-
-Εισάγετε τον τίτλο του επιβλέποντα-

Αθήνα, Ιανουάριος 2023

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ
ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΣΗΜΑΤΩΝ, ΕΛΕΓΧΟΥ ΚΑΙ ΡΟΜΠΟΤΙΚΗΣ

Δημιουργία συστάσεων σε διάλογο με χρήση contextual bandits

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Λεωνίδα Αβδελά

Επιβλέπων: -Εισάγετε το όνομα, αρχικό πατρώνυμο και επίθετο του επιβλέποντα-
-Εισάγετε τον τίτλο του επιβλέποντα-

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την -εισάγετε ημερομηνία-.

.....	.....	.....
-Εσαγετε Ονοματεπωνυμο-	-Εσαγετε Ονοματεπωνυμο-	-Εσαγετε Ονοματεπωνυμο-
-Εσαγετε τίτλο-	-Εσαγετε τίτλο-	-Εσαγετε τίτλο-

Αθήνα, Ιανουάριος 2023.

.....

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

Περίληψη

Η δημιουργία συστάσεων είναι ένα πρόβλημα στο οποίο έχει δείξει ενδιαφέρον τόσο η ακαδημαική κοινότητα, όσο και η βιομηχανία. Πολλές επιτυχημένες εφαρμογές έχουν χτιστεί πάνω στην δημιουργία καλών συστάσεων, όπως ο αλγόριθμος του Netflix, αλλά και οι συστάσεις προϊόντων της Amazon. Η άνθιση προέρχερται κυρίως λόγο του μεγάλου όγκου πληροφοριών που υπάρχουν στο διαδίκτυο, του οποίου η αναζήτηση και επιμέλεια χειροκίνητα από ανθρώπους είναι αδύνατη.

Επιπλέον, η ενισχυτική μάθηση είναι ένας απο τους πλέον διαδεδομένους τρόπους εκπαίδευσης πρακτόρων, κυρίως σε περιβάλλοντα που το τελικό αποτέλεσμα γίνεται γνωστό μετά από πολλά βήματα, και δεν υπάρχει γνωστή βέλτιστη λύση για κάθε βήμα.

Τέλος η χρήση μηχανική μάθησης για την παραγωγή και την κατανόηση κειμένου είναι ένας κλάδος ο οποίος έχει δει μεγάλη άνθιση τα τελευταία λίγα χρόνια

Η τρέχουσα διπλωματική ασχολείται με την δημιουργία ενός συστήματος συστάσεων, το οποίο δουλεύει παράλληλα με ένα διαλογικό σύστημα, το οποίο προτείνει θέματα συζήτησης στον χρήστη. Η επιλογή των πρακτόρων έγινε με βάση την γνώση ότι το περιβάλλον εργασίας περιείχε περιορισμένα δεδομένα, καθώς και με βάση το γεγονός ότι ένας από τους στόχους ήταν η προσαρμογή των συστάσεων ανάλογα με την χρονική περίοδο και τις αλλαγές στις ανάγκες που προκύπτουν με βάση αυτή, οπότε κρίθηκε η χρήση τεχνικών ενισχυτικής μάθησης ως η βέλτιστη λύση.

Λέξεις Κλειδιά

TODO

Abstract

TODO

Keywords

TODO

Ευχαριστίες

Ευχαριστώ την οικογένεια μου και τους καθηγητές μου.

Περιεχόμενα

Κατάλογος Σχημάτων

Κατάλογος Πινάκων

Κεφάλαιο 1

Εισαγωγή

Η εργασία αυτή ασχολείται με την ανάπτυξη ενός συστήματος το οποίο συνδέει τρία βασικά στοιχεία: τις συστάσεις, τα διαλογικά συστήματα και την ενισχυτική μάθηση. Κάθε ένα από αυτά τα στοιχεία θα αναλυθεί σε βάθος στα επόμενα κεφάλαια. Σε αυτή την ενότητα θα γίνει μια εισαγωγή στα κίνητρα, τις βασικές ιδέες που χρησιμοποιήθηκαν στην εργασία, καθώς και τα κύρια χαρακτηριστικά των συστημάτων αυτών και των προϋπάρχουσων υποδομών.

1.1 Κίνητρα

Οι διαλογικοί πράκτορες δημιουργήθηκαν με σκοπό να μειώσουν τον ανθρώπινο κόπο. Αντί να χρειάζεται να υπάρχει ένας άνθρωπος-πράκτορας πάντα διαθέσιμος να εξυπηρετήσει τα ερωτήματα που έρχονται από πελάτες/ενδιαφερόμενους, το όραμα ήταν να υπάρχουν αυτόματα διαλογικά συστήματα που τους εξυπηρετούν με παρόμοια ποιότητα υπηρεσίας. Η έρευνα στα διαλογικά συστήματα ξεκίνησε από τις αρχές του 1970 με το σύστημα βασισμένο σε κανόνες που ονομαζόταν ELIZA[2] και έχει φτάσει στο απόγειο της πλεον, με την δημιουργία του συστήματος ChatGPT, ενός διαλογικού συστήματος γενικού σκοπού βασισμένο στην αρχιτεκτονική των μετασχηματιστών (transformers) και έχει 175 δισεκατομμύρια παραμέτρους[18]. Παρόλα αυτά, η διατήρηση του ενδιαφέροντος του χρήστη κατά την διάρκεια του διαλόγου και η συνέχιση του, είναι ένα πρόβλημα που δεν έχει λυθεί επαρκώς.

Σκοπός της διπλωματικής εργασίας είναι η σύσταση ενδιαφερόντων θεμάτων στον χρήστη του διαλογικού συστήματος Θεανώ, το οποίο δημιουργήθηκε στο ΕΚ Αθηνά. Η Θεανώ είναι ένα διαλογικό σύστημα, το οποίο μπορεί να απαντήσει καίρια ερωτήματα σχετικά με τον Covid-19. Περαιτέρω επεξήγηση της Θεανώς θα γίνει στις επόμενες ενότητες. Τα ενδιαφέροντα θέματα ορίζονται ως τα θέματα τα οποία θα επιτύγχαναν να διατηρήσουν την αλληλεπίδραση με τον χρήστη για μεγαλύτερη διάρκεια. Έτσι, για παράδειγμα, ένας χρήστης θα ρωτήσει κάποια ερώτηση, και αφού το σύστημα απαντήσει, θα του προτείνει και ένα θέμα περαιτέρω συζήτησης, με βάση τα θέματα τα οποία μπορεί να απαντήσει.

Όπως είναι προφανές, κάθε χρήστης έχει διαφορετικό ιστορικό, διαφορετικά ενδιαφέροντα και διαφορετικές πληροφορίες που τον ενδιαφέρουν. Πολλές από αυτές τις πληροφορίες δεν είναι ποτέ διαθέσιμες στο σύστημα μας, ενώ άλλες είναι διαθέσιμες αφού η Θεανώ αρχίσει να επικοινωνεί με τον χρήστη. Έτσι, το σύστημα θα πρέπει να μπορεί να κάνει δύο πράγματα. Το πρώτο είναι να μπορεί να κατανοήσει τα θέματα που σχετίζονται νοηματικά και είναι πιθανό ότι αν ο χρήστης ρωτήσει για το ένα, να ενδιαφέρεται και για το άλλο. Το δεύτερο είναι να μπορεί να κατανοήσει τα ενδιαφέροντα του χρήστη με βάση τις ερωτήσεις που κάνει και τα θέματα που επιθυμεί να μάθει ή όχι. Καθώς οι χρήστες δεν είναι γνωστοί από πριν, και τα ενδιαφέροντα θέματα μπορεί να αλλάζουν ανα περίοδο, η χρήση κλασικής επιβλεπόμενης μάθησης δεν είναι εφικτή. Αυτό συμβαίνει γιατί στην επιβλεπόμενη μάθηση χρειάζονται γνωστά δείγματα και σωστές απαντήσεις σε αυτά δείγματα, οι οποίες δεν υπάρχουν σε αυτή την περίπτωση, καθώς δεν υπάρχει μια ξεκάθαρη ανάγκη για τους χρήστες. Αντίθετα χρειάζεται μια πιο δυναμική προσαρμογή, η οποία επιτυγχάνεται μέσω την σύγχρονης μάθησης (online learning) και της ενισχυτικής μάθησης (reinforcement learning).

1.2 Η Θεανώ

Η Θεανώ είναι ένας διαλογικός πράκτορας που έχει σκοπό την ενημέρωση σχετικά με τον Covid-19.[23] Βασίζεται πάνω στα εργαλεία που προσφέρονται από την εργαλειοθήκη του Rasa, η οποία παρέχει ένα ολοκληρωμένο διαλογικό σύστημα από άκρη σε άκρη. Συγκεκριμένα, παρέχει σύστημα κατανόησης της φυσικής γλώσσας, αναγνώρισης των προθέσεων του συνομιλητή και επιλογής της κατάλληλης απάντησης με βάση αυτό. Τέλος, έχει και ένα συστημα παραγωγής φυσικής γλώσσας για την απάντηση. Επιπλέον, κάθε πρόθεση μπορεί να αντιστοιχιστεί σε μια πράξη και με βάση αυτή να δημιουργηθεί η απάντηση του πράκτορα. Έτσι το σύστημα είναι αρκετά εύρωστο ώστε να μπορεί να ανταποκριθεί τόσο σε αιτήματα του χρήστη που αποσκοπούν στην επίτευξη κάποιου συγκεκριμένου στόχου όσο και σε άλλα που είναι πιο γενικά. Η Θεανώ εκπαιδεύεται τόσο με την χρήση συνθετικών δεδομένων, ειδικά στην αρχή, όσο και με δεδομένα από συνομιλίες που έχει κάνει με τους χρήστες. Η εκπαίδευση γίνεται με την μέθοδο της επιβλεπόμενης μάθησης, και συγκεκριμένα είναι ένα είδος ταξινόμησης. Περισσότερες πληροφορίες για την εκπαίδευση της Θεανώς υπάρχουν στο Κεφάλαιο 4.

Έτσι η Θεανώ μπορεί να απαντήσει ερωτήσεις σχετικά με τα εμβόλια, την κατάσταση των εμβολιασμών τόσο στην Ελλάδα όσο και στο εξωτερικό, την κατάσταση των ΜΕΘ, και άλλα παρόμοια θέματα. Επιπλέον, η Θεανώ, στην βασική της έκδοση, είχε την δυνατότητα να προτείνει τυχαία θέματα για να συνεχίσει την συζήτηση, αφού ο χρήστης ρωτήσει κάτι. Έτσι προσπαθούσε να «κρατήσει» τον χρήστη περισσότερο στην συζήτηση και να τον βοηθήσει να μάθει περισσότερα πράγματα. Η χρήση αυτών των συστάσεων δείχνει να συνεισφέρει στην μεγαλύτερη διάρκεια των διαλόγων μεταξύ χρήστη και Θεανώς. Όμως, η συχνή τους χρήση συνδέεται με χαμηλότερο αίσθημα ότι η Θεανώ καταλαβαίνει τον χρήστη.

Στόχος μας στην εργασία είναι η βελτίωση του συστήματος αυτών των συστάσεων με

στόχο να επιτύχουμε τόσο την μεγαλύτερη διάρκεια των διαλόγων, αλλά ταυτόχρονα να αυξήσουμε και το αίσθημα ότι η Θεανώ συναισθάνεται τον χρήστη.

1.3 Η συνεισφορά μας

Σκοπός της εργασίας είναι η δημιουργία ενός συστήματος συστάσεων το οποίο θα διαλειτουργεί με το προυπάρχον σύστημα της Θεανώς και θα παρέχει διαλογικές συστάσεις στα θέματα που θα κρατήσουν το ενδιαφέρον του χρήστη για παραπάνω διαλογικούς γύρους. Για την εφαρμογή των μεθόδων bandits στο πρόβλημα, χρησιμοποιήσαμε το εργαλείο Vowpal Wabbit, το οποίο προσφέρει έτοιμες πολιτικές bandits, καθώς και εργαλεία για σύγχρονη εκμάθηση. Επιπλέον, για την καλύτερη ροή πληροφορίας μεταξύ του συστήματος συστάσεων και της Θεανώς, τα δύο αυτά συστήματα διαχωρίστηκαν, και δημιουργήθηκε μια νέα μίκρο-υπηρεσία (micro-service) η οποία είναι υπεύθυνη για τις συστάσεις. Αυτό τελικά σημαίνει ότι η υπηρεσία και η λειτουργικότητα των συστάσεων είναι μεταφέρσιμη και σε άλλα περιβάλλοντα. Το σύστημα συστάσεων εκπαιδεύτηκε αρχικά με την χρήση ασύγχρονης εκμάθησης (offline learning) και μετέπειτα εκπαιδεύεται μέσω της αλληλεπίδρασης του με τους χρήστες της Θεανώς. Έτσι μπορεί να ακολουθήσει τα ρεύματα και τα ενδιαφέροντα των χρηστών καθώς αυτά αλλάζουν με τον χρόνο. Τέλος, η υπηρεσία των συστάσεων σχεδιάστηκε με τρόπου που να είναι εύκολα μεταφέρσιμο σε άλλους διαλογικούς πράκτορες σχεδιασμένους με τα εργαλεία του Rasa, με μικρές τροποποιήσεις στον περιβάλλοντα κώδικα.

1.4 Διάρθρωση της εργασίας

Στο Κεφάλαιο 2 παρουσιάζονται οι βασικές γνώσεις Ενισχυτικής Μάθησης. Το κεφάλαιο αυτό παρέχει όλες τις απαραίτητες πληροφορίες για την κατανόηση της λειτουργίας των τεχνικών bandits, οι οποίες παρουσιάζονται αναλυτικότερα στο Κεφάλαιο 3, μαζί με τους βασικότερους αλγορίθμους που χρησιμοποιούνται σήμερα. Έπειτα, στο Κεφάλαιο 4 παρουσιάζονται συνοπτικά οι βασικές ιδέες γύρω από τα διαλογικά συστήματα και την λειτουργία τους. Στο Κεφάλαιο 5 παρουσιάζεται αναλυτικά η δική μας εργασία και συνεισφορές. Κλείνοντας, στο Κεφάλαιο 6, προτείνονται ιδέες για περαιτέρω εξερεύνηση, καθώς και τα σημαντικότερα αποτελέσματα.

Κεφάλαιο 2

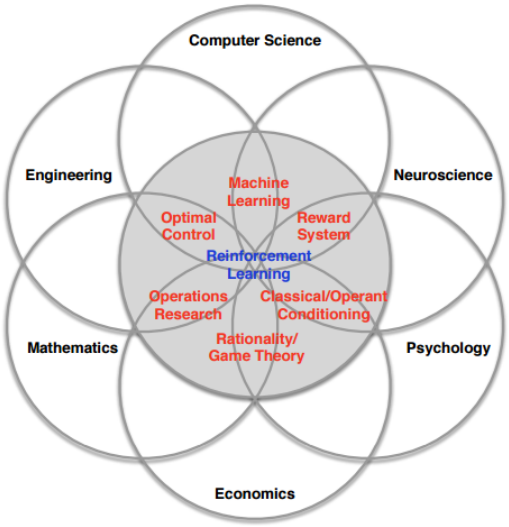
Ενισχυτική Μάθηση

2.1 Γενικά

Η ενισχυτική μάθηση (ΕΜ) (Reinforcement Learning (RL)) είναι είναι ένας γενικός όρος που έχει δοθεί σε μια οικογένεια τεχνικών στις οποίες ένα σύστημα προσπαθεί να μάθει μέσα από την άμεση αλληλεπίδραση με το περιβάλλον [3]. Είναι τομέας της τεχνητής νοημοσύνης και, πιο συγκεκριμένα, της μηχανικής μάθησης.

Η ΕΜ είναι η διαδικασία κατα την οποία ένας πράκτορας (agent) αλληλεπιδρά με το περιβάλλον του, και μαθαίνει τι να κάνει, παρατηρώντας τις συνέπειες των πράξεων του. Δεν δίνουμε πληροφορίες στον πράκτορα σχετικά με το ποίες ενέργειες (actions) να επιλέξει, αλλά πρέπει ο ίδιος να ανακαλύψει ποιες ενέργειες προσφέρουν την μέγιστη ανταμοιβή (reward), δοκιμάζοντας τις [15]. Επιπλέον, σε πολλές περιπτώσεις οι ενέργειες του πράκτορα δεν επηρεάζουν μόνο την άμεση ανταμοιβή που θα πάρει, αλλά και την ανταμοιβή στην επόμενη κατάσταση, και πιθανώς και όλες τις επόμενες ανταμοιβές. Ετσι, μπορεί να υπάρξουν καταστάσεις που ο πρακτορας θα πρέπει να θυσιάσει την άμεση ανταμοιβή για να αποκτήσει καλύτερες ανταμοιβές μακροπρόθεσμα. Σύμφωνα με τα παραπάνω, η ΕΜ στοχεύει να λύσει προβλήματα μέσω τεχνικών *δοκιμής-και-λάθους (trial-and-error)* σε περιβάλλοντα με *καθυστερημένες ανταμοιβές*.

Μια κομβική ιδέα, πάνω στην οποία στηρίζεται η ΕΜ, είναι η υπόθεση της ανταμοιβής (reward hypothesis), η ιδέα ότι κάθε στόχος μπορεί να εκφραστεί ως η μεγιστοποίηση της αναμενόμενης αξίας του σωρευτικού (cummulative) αθροίσματος ενός μονοδιάστατου σήματος. Με απλά λόγια, η υπόθεση θέτει την ιδέα ότι κάθε στόχος μπορεί να εκφραστεί σαν την μεγιστοποίηση μιας ανταμοιβής. Η ανταμοιβή αυτή δεν χρειάζεται απαραίτητα να είναι θετικός αριθμός, αλλά ακόμα και για αρνητικές τιμές της, συνεχίζουμε να καλούμε τον όρο ανταμοιβή. Για παράδειγμα, αν ο στόχος είναι η έξοδος από ένα λαβύρινθο, η ανταμοιβή μπορεί να είναι αρνητική σε κάθε βήμα μέχρι την έξοδο, οπού και γίνεται 0. Τότε ο στόχος τελικά είναι η ελαχιστοποίηση της απόλυτης τιμής της ανταμοιβής, δηλαδή η έξοδος στα λιγότερα βήματα.



Σχήμα 2.1: Τα επιστημονικά πεδία που σχετίζονται με την ενισχυτική μάθηση [8]

Το πεδίο της ΕΜ έχει τις ρίζες του σε δύο περιοχές. Η πρώτη είναι η συμπεριφορική ψυχολογία, από όπου προέρχεται το παράδειγμα της δοκιμής-και-λάθους, και η δεύτερη είναι η περιοχή του βέλτιστου ελέγχου, από όπου η ΕΜ δανείζεται τον μαθηματικό φορμαλισμό (κυρίως τον δυναμικό προγραμματισμό) που υποστηρίζει το πεδίο. Είναι σημαντικό να γνωρίζουμε ότι η ΕΜ βρίσκεται στην τομή πολλών διαφορετικών επιστημονικών πεδίων, οι οποίοι φαίνονται στο Σχήμα 2.1[8]. Όλα αυτά τα πεδία προσεγγίζουν ένα παρόμοιο πρόβλημα, αλλά απο διαφορετική σκοπιά και με διαφορετικές παραμέτρους.

Η ΕΜ πολλές φορές συγχέεται με τις άλλες τεχνικές μηχανικής μάθησης, την επιβλεπόμενη και την μη επιβλεπόμενη μάθηση, παρόλο που έχει αρκετά σημαντικές διαφορές.

Αρχικά, η κύρια διαφορά μεταξύ της επιβλεπόμενης μάθησης (supervised learning) και της ΕΜ είναι ότι στην επιβλεπόμενη μάθηση, το μοντέλο εκπαιδεύεται πάνω σε δείγματα (samples) και ετικέτες (labels), και κάθε πρόβλεψη θεωρείται μοναδικό γεγονός. Στόχος είναι η προσέγγιση μιας άγνωστης συνάρτησης με βάση τα δεδομένα. Αντίθετα, στην ΕΜ, μπορούν να υπάρχουν πολλά βήματα πριν ο πράκτορας μάθει αν η απόφαση που πήρε ήταν σωστή, και είναι πιθανό να μην μάθει ποτέ ποια ήταν η αληθής/βέλτιστη τιμή. Το μόνο που παρατηρεί είναι η επίδραση που είχαν οι πράξεις του στο περιβάλλον.

Όσον αφορά την μη επιβλεπόμενη μάθηση, μπορεί αρχικά να φαίνεται παρόμοια με την ΕΜ. Όμως, στόχος της μη επιβλεπόμενης μάθησης είναι η εύρεση της κρυμμένης δομής δεδομένων τα οποία δεν έχουν ετικέτες. Η ΕΜ έχει διαφορετικό στόχο, ο οποίος είναι η μεγιστοποίηση του σήματος ανταμοιβής. Παρόλο που η εύρεση δομής είναι πολύ σημαντική και στην ΕΜ ώστε να μπορέσει ο πράκτορας να επιλέξει τις κατάλληλες κινήσεις, αυτό από μόνο του δεν

επιτυγχάνει τον στόχο της ΕΜ.

Ενα από τα κύρια προβλήματα της ΕΜ, που δεν συναντάται στις άλλες μορφές μηχανικής μάθησης είναι ο συμβιβασμός μεταξύ εξερεύνησης και εκμετάλλευσης. Για να αποκτήσει ο πράκτορας μεγάλη ανταμοιβή θα προτιμήσει τις ενέργειες που δοκίμασε στο παρελθόν και του προσέφεραν μεγαλύτερη ανταμοιβή. Αλλά για να ανακαλύψει τέτοιες πράξεις, πρέπει να δοκιμάσει πράξεις που δεν έχει δοκιμάσει στο παρελθόν. Έτσι, ο πράκτορας πρέπει να *εκμεταλλευτεί* το τι έχει ήδη βιώσει ώστε να αποκτήσει ανταμοιβές, αλλά πρέπει και να *εξερευνήσει*, ώστε να πάρει καλύτερες αποφάσεις στο μέλλον. Έτσι το δίλημμα είναι ποια από τις δύο στρατηγικές να επιλέξει κάθε φορά, καθώς καμία δεν μπορεί εφαρμοστεί αποκλειστικά, έτσι ώστε να επιτευχθεί ο στόχος του πράκτορα. Ως αποτέλεσμα, ο πράκτορας πρέπει να δοκιμάσει διάφορες κινήσεις και προοδευτικά να προτιμήσει αυτές που θεωρεί καλύτερες. Καθώς πολλές από τα προβλήματα που θέλουμε να λύσουμε με ΕΜ είναι στοχαστικά, ο πράκτορας πρέπει να δοκιμάσει κάθε πράξη πολλές φορές για να πάρει μια αξιόπιστη εκτίμηση της προσδοκώμενης ανταμοιβής.

2.2 Στοιχεία της ΕΜ

Πέρα από τον πράκτορα και το περιβάλλον, υπάρχουν ακόμα τέσσερα κύρια στοιχεία σε ένα σύστημα ΕΜ. Αυτά είναι:

- Η πολιτική (policy), η οποία ορίζει την συμπεριφορά του πράκτορα για μια δοθείσα χρονική στιγμη. Με απλά λόγια, η πολιτική είναι μια αντιστοίχιση των κατατστάσεων που αντιλαμβάνεται ο πράκτορας στις ενέργειες που κάνει σε αυτές τις καταστάσεις (πιο συγκεκριμένα στις πιθανότητες αυτών των ενεργειών). Οι πολιτικές μπορεί να είναι και στοχαστικές, προσδιορίζοντας μια πιθανότητα για κάθε ενέργεια.

- Το σήμα ανταμοιβής, το οποίο αναφέρθηκε και νωρίτερα. Το σήμα αυτό προσδιορίζει τον στόχο ενός προβλήματος ΕΜ. Σε κάθε χρονικό βήμα, το περιβάλλον στέλνει στον πράκτορα έναν αριθμό, την ανταμοιβή. Ο μόνος στόχος του πράκτορα είναι να μεγιστοποιήσει την συνολική ανταμοιβή που λαμβάνει μακροπρόθεσμα. Έτσι το σήμα της ανταμοιβής προσδιορίζει ποιά είναι τα καλά και τα κακά γεγονότα για τον πράκτορα. Είναι σημαντικό η ανταμοιβή να προσδιορίζει ακριβώς το τι θέλουμε να πετύχουμε. Η ανταμοιβή δεν πρέπει να περιέχει πληροφορίες για το πώς θα πετύχουμε τον στόχο. Αυτές οι πληροφορίες μπορούν να τοποθετηθούν μέσα στην πολιτική ή στην συνάρτηση αξίας. Το σήμα είναι η κύρια βάση λόγω της οποίας αλλάζει η πολιτική. Αν ο πράκτορας επιλέξει μια ενέργεια με μικρή ανταμοιβή, τότε η πολιτική του ίσως να αλλάξει για να επιλέξει κάποια άλλη ενέργεια, όταν υπάρξει η ίδια κατάσταση στο μέλλον. Γενικά, οι ανταμοιβές μπορεί να είναι στοχαστικές συναρτήσεις της κατάστασης του περιβάλλοντος και των ενεργειών που επιλέχθηκαν.

- Η συνάρτηση αξίας κάθε κατάστασης (value function). Αντίθετα από το σήμα ανταμοιβής που μας επιστρέφει το τί είναι καλό άμεσα, η συνάρτηση αξίας προσδιορίζει τι

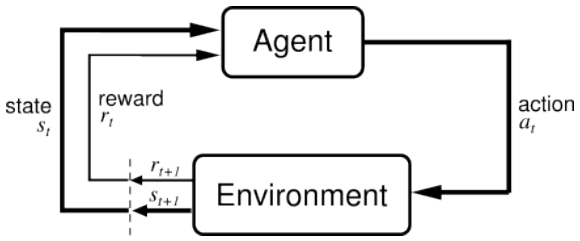
είναι καλό μακροπρόθεσμα. Σε γενικές γραμμές, η αξία μιας κατάστασης είναι η συνολική ανταμοιβή που μπορεί να περιμένει να αποκτήσει ένας πράκτορας στο μέλλον, ξεκινώντας από την συγκεκριμένη κατάσταση. Έτσι, ενώ οι ανταμοιβές προσδιορίζουν την άμεση και εσωτερική ελυστικότητα των καταστάσεων του περιβάλλοντος, η αξία υποδεικνύει την μακροπρόθεσμη ελυστικότητα των καταστάσεων, παίρνοντας υπόψιν τις καταστάσεις που θα ακολουθήσουν και τις διαθέσιμες ανταμοιβές σε αυτές τις καταστάσεις. Έτσι, μια κατάσταση με χαμηλή ανταμοιβή, μπορεί να έχει μεγάλη αξία γιατί οδηγεί σε καταστάσεις με μεγαλύτερες ανταμοιβές. Ετσι η συνάρτηση αξίας προσδιορίζει πόσο καλό είναι για ένα πράκτορα να είναι στην συγκεκριμένη κατάσταση.

- Προαιρετικά, ένα μοντέλο του περιβάλλοντος (model). Το μοντέλο ενός συστήματος ΕΜ μιμείται την συμπεριφορά του συστήματος, ή πιο γενικά, επιτρέπει την δημιουργία συμπερασμάτων για το πώς θα συμπεριφερθεί το περιβάλλον. Τα μοντέλα χρησιμοποιούνται για σχεδιασμό (planning), δηλαδή την επίλογή της σειράς των δράσεων παίρνοντας υπόψιν πιθανές μελλοντικες καταστάσεις, πριν τις βιώσει ο πράκτορας. Μέθοδοι επίλυσης προβλημάτων ΕΜ που χρησιμοποιούν μοντέλα και σχεδιασμό λέγονται μέθοδοι βασισμένοι σε μοντέλα (model-based). Αν οι μέθοδοι δεν έχουν μοντέλο, δηλαδή μέθοδοι που μαθαίνουν ρητά μέσω δοκιμής-και-λάθους, λέγονται μέθοδοι χωρίς μοντέλο (model-free).

Πιο φορμαλιστικά, σε ένα περιβάλλον ΕΜ, ένας αυτόνομος πράκτορας, ελεγχόμενος από ένα αλγόριθμο μηχανικής μάθησης, παρατηρεί μια κατάσταση s_t από το περιβάλλον του σε ένα χρονικό βήμα t. Οι χρονικές στιγμές στην περιγραφή αυτή είναι διακριτές, δηλαδή $t = 0, 1, 2, \ldots$, αλλά θα μπορούσαν να είναι και συνεχείς, χώρις μεγάλες διαφορές. Οι καταστάσεις προέρχονται από τον χώρο καταστάσεων $\mathcal{S}$. Ο πράκτορας αλληλεπιδρά με το περιβάλλον επιλέγοντας μια ενέργεια a_t με βάση την κατάσταση s_t, επιλεγμένη από ένα χώρο ενεργειών $\mathcal{A}(s)$. Όταν ο πράκτορας εκτελέσει την ενέργεια, τότε τόσο το περιβάλλον, μεταβαίνει σε μια νέα κατάσταση s_{t+1}, με βάση την τρέχουσα κατάσταση και την επιλεγμένη ενέργεια [11]. Σε κάθε τριπλέτα (κατάστασης, ενέργειας, νέας κατάστασης), αντιστοιχεί μια πιθανότητα μετάβασης $\Pr\{S_{t+1} = s' | S_t = s, A_t = a\}$. Σε κάθε κατάσταση,ο πράκτορας μπορεί είναι να παρατηρήσει την πλήρη δυναμική του περιβάλλοντος ή μέρος της. Ο πράκτορας επίσης λαμβάνει και μια μονοδιάστατη ανταμοιβή R_t η οποία προέρχεται από την τριπλέτα (κατάστασης, ενέργειας, νέας κατάστασης), και συμβολίζεται ως $R(s_{t-1}, a_{t-1}, s_t)$. Αυτή η ανταμοιβή δεν είναι γνωστή στον πράκτορα στην αρχή και δρα ως μια μορφή ανατροφοδότησης για τις δράσεις του πράκτορα. Αυτή η διαδικασία αναπαριστάται οπτικά στο Σχήμα 2.2.

Συνήθως ο πράκτορας διατηρεί μια εσωτερική κατάσταση, η οποία περιλαμβάνει κομμάτια της κατάστασης του περιβάλλοντος τα οποία θεωρούνται σημαντικά, καθώς και άλλες πληροφορίες. Σε αυτή την εσωτερική κατάσταση, ο πράκτορας διατηρεί μια αντιστοίχηση μεταξύ κατάστασης και ενέργειας, η οποία συμβολίζεται ως $\Pr(a_t|s_t)$.

Η βέλτιστη σειρά ενεργειών προσδιορίζεται από τις ανταμοιβές που παρέχει το περιβάλλον. Ο τελικός στόχος του πράκτορα είναι να μάθει μια πολιτική π, η οποία μεγιστοποιεί την αναμενόμενη απόδοση (σωρευτική, ανταμοιβή με έκπτωση (discounted reward». Δοθείσας



Σχήμα 2.2: Αλληλεπίδραση πράκτορα και περιβάλλοντος

μιας κατάστασης η πολιτική αποφασίζει την επόμενη ενέργεια την οποία θα κάνει ο πράκτορας. Μια βέλτιστη πολιτική είναι η πολιτική η οποία μεγιστοποιεί την αναμενόμενη απόδοση στο συγκεκριμένο περιβάλλον.

Πέρα απο την περιγραφή του συστήματος ΕΜ με βάση την ύπαρξη ή όχι μοντέλου του περιβάλλοντος, ένας άλλος τρόπος να περιγράψουμε τα συστήματα ΕΜ είναι με βάση το περιβάλλον στο οποίο βρίσκονται οι πράκτορες. Η μία περίπτωση είναι να είναι αυτό το περιβάλλον πλήρως παρατηρήσιμο, δηλαδή ο πράκτορας μπορεί να παρατηρήσει κάθε πληροφορία για την δυναμική του περιβάλλοντος. Αυτό φυσικά δεν σημαίνει ότι κάθε παρατήρηση θα είναι χρήσιμη. Έτσι η κατάσταση του πράκτορα θα είναι το υποσύνολο των παρατηρήσεων που είναι χρήσιμες. Αυτά τα περιβάλλοντα ικανοποιούν την Μαρκοβιανή ιδιότητα. Δηλαδή, για κάθε κατάσταση, το μέλλον εξαρτάται μόνο από την τρέχουσα κατάσταση και όχι τις προηγούμενες. Όταν ισχύει αυτή η ιδιότητα τότε μπορούμε να μοντελοποιήσουμε το πρόβλημα ως μια Μαρκοβιανή Διαδικασία Αποφάσεων (Markov Decision Process). Αντίθετα, υπάρχουν περιβάλλοντα που δεν είναι πλήρως παρατηρήσιμα, όπως για παράδειγμα ένα δωμάτιο μέσα στο οποίο κινειται ένα ρομπότ. Σε αυτή την περίπτωση, το ρομπότ δεν γίνεται σε κάθε κίνηση του να γνωρίζει τα πάντα για το περιβάλλον γιατί υπάρχουν πάρα πολλές παράμετροι.

Η Μαρκοβιανή ιδιότητα ορίζεται ως:

$$p(r, s|S_t, A_t) = p(r, s|\mathcal{H}_t, A_t) \tag{2.1}$$

το οποίο σημαίνει ότι η πιθανότητα να βρεθούμε στην κατάσταση s με ανταμοιβή r, αν γνωρίζουμε ολόκληρη την ιστορία της αλληλεπίδρασης του πράκτορα με το περιβάλλον και παίρνοντας και κάνοντας την ενέργεια A_t (αριστερό μέρος) είναι ίδια με την πιθανότητα να βρεθούμε στην κατάσταση s με ανταμοιβή r γνωρίζοντας μόνο την τελευταία κατάσταση S_t στην οποία βρισκόταν ο πράκτορας και την ενέργεια A_t που έκανε (δεξί μέρος).

Σε μια Μαρκοβιανή Διαδικασία Αποφάσεων δημιουργούμε μια εκτίμηση της βέλτιστης $q_*(s, \alpha)$ κάθε ενέργειας α σε κάθε κατάσταση s ή μια εκτίμηση της αξίας $u_*(s)$ κάθε κατάστασης δεδομένης μιας βέλτιστης επιλογής ενεργειών.

2.3 Μαρκοβιανές Διαδικασίες Αποφάσεων

Για την καλύτερη κατανόηση της ΕΜ, είναι χρήσιμο να περιορίσουμε το πρόβλημα στην μορφή του που είναι μια ΜΔΑ. Συγκεκριμένα, σε διακριτό χρόνο, η αρχή της πορείας ενός πράκτορα θα είναι

$$S_0, A_0, R_1, S_1, A_1, R_2, S_2, A_2$$

Σε μια πεπερασμένη ΜΔΑ, η κατάστασεις, οι ενέργειες και οι ανταμοιβές είναι τα σύνολα $(\mathcal{S}, \mathcal{A}, \mathcal{R})$ με πεπερασμένο αριθμό στοιχείων. Σε αυτή την περίπτωση οι τυχαίες μεταβλητές R_t και S_t έχουν μια καλά ορισμένη διακριτή πιθανότητα, η οποία εξαρτάται από την προηγούμενη κατάσταση και ενέργεια. Έτσι η δυναμική της ΜΔΑ ορίζεται από την συνάρτηση, η οποία λόγω της Μαρκοβιανής ιδιότητας προσδιορίζει πλήρως την δυναμική του συστήματος.

$$p(s', r|s, a) = \Pr S_t = s', R_t = r|S_{t-1} = s, A_{t-1} = a \tag{2.2}$$

Όσον αφορά το σήμα ανταμοιβής, ο στόχος είναι η μεγιστοποίηση του σε βάθος χρόνου. Αυτό αναφέρεται στην βιβλιογραφία ως απόδοση (return) οπως αναφέρθηκε και νωρίτερα, η οποία πολύ συχνά αναπαριστάται ως G_t.

Ανάλογα με το αν τερματίζει ένα πρόβλημα, αυτό μπορεί να θεωρηθεί επεισοδικό ή συνεχές. Σε ένα επεισοδικό πρόβλημα, υπάρχει πάντα μια τελική κατάσταση (π.χ. η έξοδος ενός λαβυρίνθου). Σε ένα συνεχές πρόβλημα, δεν υπάρχει αυτός ο διαχωρισμός σε επεισόδια, αλλά η αλληλεπίδραση συνεχίζεται ατέρμονα. Για να μπορέσουμε να υπολογίσουμε την απόδοση ακόμα και σε συνεχή προβλήματα, συνήθως την ορίζουμε ως

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \ldots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{2.3}$$

όπου $0 \leq \gamma \leq 1$ είναι μια παράμετρος η οποία ονομάζεται παράγοντας έκπτωσης. Ο παράγοντας αυτός επηρεάζει το πόσο σημασία έχουν οι μετέπειτα ανταμοιβές. Επίσης, έχει μαθηματική αξία, καθώς εξασφαλίζει ότι η απόδοση είναι πάντα φραγμένη (για $\gamma < 1$).

Για $\gamma = 0$, ο πράκτορας είναι μυοπυκός, δηλαδή ενδιαφέρεται να εξασφαλίσει την καλύτερη ανταμοιβή σε κάθε βήμα, ανεξάρτητα αν αυτό σημαίνει ότι θα χάσει καλύτερες ανταμοιβές αργότερα, ενώ οσο η τιμή πηγαίνει προς το 1, ο πράκτορας βλέπει όλο και πιο μακριά.

Με βάση τα παραπάνω, μπορούμε πλεον να προσδιορίσουμε και πιο φορμαλιστικά την περιγραφή της πολιτικής του πράκτορα και της συνάρτησης αξίας. Οπως αναφέρθηκε ήδη η πολιτική είναι μια αντιστοίχηση μεταξύ καταστάσεων και πιθανοτήτων επιλογής πράξεων. Αν ένας πράκτορας ακολουθεί μια πολιτική π την χρονική στιγμή t, τότε $\pi(a|s)$ είναι η πιθανότητα ότι θα επιλεχθεί $A_t = a$, αν $S_t = s$.

Η συνάρτηση αξίας μιας κατάστασης s όταν ο πράκτορας ακολουθεί μια πολιτική π, με ένδειξη $\upsilon_\pi(s)$ είναι η αναμενόμενη απόδοση όταν ο πράκτορας ξεκινήσει από την κατάσταση s και ακολουθήσει την πολιτική π. Για ΜΔΑ, αυτή ορίζεται ως

$$\upsilon_\pi(s) = \mathbb{E}_\pi[G_t \mid S_t = s] = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \,\middle|\, S_t = s \right] \tag{2.4}$$

για όλες τις καταστάσεις s του συνόλου $\mathcal{S}$. Το $\mathbb{E}_\pi[\cdot]$ είανι η αναμενόμενη τιμή μιας τυχαίας μεταβλητής δεδομένου ότι ο πράκτορας ακολουθεί πολιτική π και t είναι το χρονικό βήμα. Αυτή η συνάρτηση παραπάνω ορίζεται ως συνάρτηση αξίας-κατάστασης για την πολιτική π.

Μπορούμε να ορίσουμε και την συνάρτηση αξίας-ενέργειας για την πολιτική π, η οποία ορίζεται ως

$$q_\pi(s,a) = \mathbb{E}_\pi[G_t | S_t = s, A_t = a] = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \middle| S_t = s, A_t = a \right] \tag{2.5}$$

η οποία προσδιορίζει την αξία του να κανει ο πράκτορας της ενέργεια a στην κατάσταση s υπο την πολιτική π, η οποία είναι η αναμενόμενη απόδοση που θα έχει ο πράκτορας αν στην κατάσταση s κάνει την ενέργεια a και μετά συνεχίσει με την πολιτική π.

Οι δύο συναρτήσεις αξίας, μπορούν να υπολογιστούν με βάση την εμπειρία που αποκτά ο πράκτορας κατά την μετάβαση του μεταξύ των καταστάσεων. Για παράδειγμα αν ακολουθεί την πολιτική π, τότε για κάθε κατάσταση, όσο πιο συχνά την επισκέφτεται, τόσο πιο κοντά θα φτάνει η μέση τιμή στην πραγματική αξία $\upsilon_\pi(s)$ της κατάστασης. Αντίστοια αν κρατάει μέσους όρους για κάθε πράξη σε κάθε κατάταση, θα φτάσει στην πραγματική αξία $q_\pi(s,a)$ της δράσης. Οι τεχνικές αυτές ονομάζονται Μόντε Κάρλο, γιατί βρίσκουν μέσους όρους πολλών τυχαίων δειγμάτων πραγματικών αποδόσεων. Ένας άλλος τρόπος είναι με χρήση συναρτήσεων που προσομοιώνουν τις συναρτήσεις αξιών, και είναι χρήσιμες όταν υπάρχει πολύ μεγάλος αριθμός καταστάσεων και η αποθήκευση όλων των τιμών δεν είναι δυνατή.

Στόχος της ΕΜ είναι η εύρεση βέλτιστων πολιτικών που θα επιφέρουν μεγάλη ανταμοιβή μακροπρόθεσμα. Σε πεπερασμένες ΜΔΑ η βέλτιστη απόφαση είναι πλήρως ορισμένη, και είναι η πολιτική που επιφέρει μεγαλύτερη ή ίση αξία με όλες τις υπόλοιπες πολιτικές. Αυτή η βέλτιστη πολιτική, δεν είναι να βρεθεί σε πραγματικά προβλήματα, γιατί

1. η δυναμική του περιβάλλοντος δεν είναι γνωστή ακριβώς,
2. δεν είναι υπολογιστικά εφικτό να υπολογιστεί η βέλτιστη πολιτική, πχ γιατί ο χώρος καταστάσεων είναι τεράστιος,
3. οι καταστάσεις δεν έχουν την Μαρκοβιανή ιδιότητα.

Σε αυτές τις περιπτώσεις, η μόνη μας επιλογή είναι είτε να δημιουργήσουμε προσεγγίσεις της βέλτιστης πολιτικής ή να χρησιμοποιήσουμε ευριστικές, ή και τα δύο.

2.4 Συμπεράσματα

Το πρόβλημα της Ενισχυτικής Μάθησης, μελετάει την έυρεση βέλτιστων πολιτικών για πράκτορες που κινούνται μέσα σε ένα περιβάλλον, πολιτικών δηλαδή που προσφέρουν την μέγιστη αξία στον πράκτορα. Επειδή πολλές η εύρεση της βέλτιστης πολιτικής δεν είναι πρακτικά δυνατή, χρησιμοποιούμε προσεγγίσεις και ευριστικές.

Στην δική μας εργασία, οι καταστάσεις του περιβάλλοντος είναι γνωστές και πεπερασμένες, και φραγμένες από τον αριθμό των θεμάτων που μπορεί να συζητήσει ο πράκτορας μας, οι οποίες δεν είναι πάρα πολλές. Επιπλέον, μπορούμε να θεωρήσουμε ότι οι καταστάσεις έχουν την Μαρκοβιανή ιδιότητα, καθώς η τρέχουσα κατάσταση μπορεί να περιλαμβάνει χαρακτηριστικά του διαλόγου του πράκτορα με τον χρήστη. Το πρόβλημα είναι ότι οι δυναμικές του συστήματος είναι πλήρως άγνωστες και μεταβλητές, καθώς δεν γνωρίζουμε καμία πληροφορία για τον χρήστη κατα την αρχή της συζήτησης για να μπορέσουμε να επιλέξουμε σωστά προτάσεις, και επιπλέον τα ενδιαφέροντα των χρηστών στο σύνολο μετακινούνται με την πάροδο του χρόνου. Επειδή ο συγκεκριμένος πράκτορας δεν αποκτά πολλές πληροοφορίες για το περιβάλλον, καθώς δεν υπάρχουν πολλοί χρήστες, θα πρέπει να απλοποιήσουμε το πρόβλημα και να μην ασχοληθούμε με το πλήρες πρόβλημα της ενισχυτικής μάθησης, αλλά με ένα πιο περιορισμένο, το πρόβλημα των bandits.

Κεφάλαιο 3

Bandits

3.1 Το πρόβλημα των bandits

Το πρόβλημα των ληστών (bandits), πήρε την ονομασία του από τα μηχανήματα του καζίνο, τους κουλοχέρηδες. Ο όρος one arm bandit προέρχεται από το γεγονός ότι οι κουλοχέρηδες έχουν ένα μοχλό-χέρι και σου ¨κλέβουν' τα χρήματα. Η ορολογία στα Ελληνικά δεν είναι ιδιαίτερα καθιερωμένη, οπότε και θα χρησιμοποιήσουμε την Αγγλική.

Το πρόβλημα των bandits είναι μια απλοποίηση του προβλήματος της Ενισχυτικής Μάθησης, και στην απλούστερη του μορφή το πρόβλημα δεν είναι προσεταιριστικό, δηλαδή κάθε κατάσταση θεωρείται ξεχωριστό γεγονός. Αυτό το γεγονός ότι οι αποφάσεις που κάνει ο πράκτορας στον ένα γύρο δεν επηρεάζουν τις ανταμοιβές και τις επιλογές του πράκτορα στους επόμενους γύρους, είναι και το πιο βασικό χαρακτηριστικό που μας ενδιαφέρει για να απλοποιήσουμε ελαφρώς και το δικό μας πρόβλημα. Ένα ακόμα σύνηθες χαρακτηριστικό των bandits είναι ότι ο πράκτορας μπορεί να παρατηρήσει τις ανταμοιβές του σε κάθε γύρο. Αν δεν μπορεί, τότε το πρόβλημα αυτό λέγεται πρόβλημα *μερικής παρακολούθησης* και δεν είναι κάτι που θα μας απασχολήσει περαιτέρω.

Παρόλο που το πρόβλημα μοιάζει φαινομενικά να είναι πολύ απλούστερο του πλήρους προβλήματος ΕΜ, η επίλυση του δεν είναι τόσο εύκολη. Σαν παράδειγμα [20], μπορούμε να σκεφτούμε ένα κουλοχέρη που έχει δύο μοχλούς, έναν δεξιό και έναν αριστερό, τους οποίους τραβώντας τους για 10 γύρους έχουμε τα παρακάτω αποτελέσματα.

Γύρος	1	2	3	4	5	6	7	8	9	10
Αριστερό	0		10	0		0				10
Δεξί		10			0		0	0	0	

Πίνακας 3.1: Παράδειγμα κουλοχέρη

Υπολογίζοντας την μέση ανταμοιβή που έχουμε από κάθε χέρι, μπορούμε να υπολογίσουμε ότι για το αριστερό χέρι αυτή είναι 4€, ενώ για το δεξί είναι 2€. Άρα το αριστερό χέρι είναι φαινομενικά καλύτερο. Ποια θα ήταν η στρατηγική μας από εδώ και πέρα, αν είχαμε ακόμα 10 ακόμα προσπάθειες· Θα χρησιμοποιούσαμε μόνο το αριστερό χέρι για να εκμεταλλευτούμε αυτό που πιστεύουμε ότι είναι καλύτερο· Θα χρησιμοποιούσαμε το δεξί χέρι, για να εξερευνήσουμε και να μάθουμε αν έχουμε σωστή προσέγγιση της τιμής του· Το δίλημμα μεταξύ εκμετάλλευσης και εξερεύνησης είναι κεντρικό στα προβλήματα αυτά.

Η ορολογία που χρησιμοποιούμε στα προβλήματα bandits είναι η ίδια που χρησιμοποιήσαμε και στα προβλήματα ΕΜ, με την διαφορά ότι το πλήθος των γύρων που θα παίξει ο πράκτορας ονομάζονται **ορίζοντας**. Τα προβλήματα των bandits είναι προβλήματα που η δυναμική του περιβάλλοντος είναι άγνωστη, και έτσι ο πράκτορας πρέπει να την ανακαλύψει παίζοντας. Το μόνο που γνωρίζει ο πράκτορας για το περιβάλλον είναι ότι βρίσκεται σε μια οικογένεια περιβαλλόντων $\mathcal{E}$.

Κύρια μετρική της ποιότητας μιας πολιτικής είναι η μετάνοια (regret), η οποία εκφράζει την διαφορά μεταξύ των αναμενόμενων ανταμοιβών μιας πολιτικής π σε n γύρους, η οποία δεν είναι απαραίτητα αυτή που ακολούθησε ο πράκτορας, και των ανταμοιβών που πραγματικά πήρε ο πράκτορας σε n γύρους, σύμφωνα με την πολιτική που ακολούθησε. Είναι χρήσιμο να υπολογίζουμε την μετάνοια και σε σχέση με μια οικογένεια πολιτικών Π. Τότε η μετάνοια είναι η διαφορά της πολιτικής που ακολούθησε ο πράκτορας σε σχέση με την πολιτική η οποία έχει τις μεγαλύτερες ανταμοιβές μεταξύ των πολιτικών της οικογένειας Π (με άλλα λόγια την καλύτερη πολιτική). Συνήθως επιλέγουμε την οικογένεια Π, ώστε η βέλτιστη πολιτική για όλη την οικογένεια $\mathcal{E}$ να βρίσκεται μέσα στην οικογένεια πολιτικών Π. Ήδη διαισθητικά καταλαβαίνουμε ότι μεγάλη μετάνοια σημαίνει ότι ο πράκτορας δεν τα πάει καλά, ενώ μικρή ότι η πολιτική που ακολουθεί είναι κοντά στην βέλτιστη.

Για να μπορέσουμε να λύσουμε το πρόβλημα, συνήθως μειώνουμε τόσο την οικογένεια πρακτόρων, όσο και την οικογένεια των περιβαλλόντων, ώστε να περιέχει στοιχεία που έχουν συγκεκριμένες επιθυμητές ιδιότητες. Στόχος κάθε φορά είναι να δημιουργήσουμε αλγορίθμους που να πετυχαίνουν όσο καλύτερη μετάνοια είναι δυνατό.

Για παράδειγμα, μια εύκολη οικογένεια προβλημάτων είναι τα στοχαστικά, χρονικά αμετάβλητα προβλήματα bandits. Σε αυτή την οικογένεια προβλημάτων, το περιβάλλον παράγει ανταμοιβές με βάση μια πράξη, οι οποίες προέρχονται από μια κατανομή η οποία είναι σχετική στην πράξη αυτή και ανεξάρτητες από τις προηγούμενες πράξεις. Επίσης οι ανταμοιβές είναι χρονικά αμετάβλητες, είναι συναρτήσεις δηλαδή οι οποίες δεν έχουν ως παράμετρο τους τον χρόνο.

Από την άλλη, αν δεν θέλουμε να κάνουμε καμία υπόθεση για το περιβάλλον, θα μπορούσαμε να υποθέσουμε ότι το μόνο που ξέρουμε είναι ότι οι ανταμοιβές επιλέγονται χωρίς να υπάρχει γνώση των πράξεων του πράκτορα και απλά είναι στοιχεία σε ένα πεπερασμένο σύνολο. Ουσιαστικά αυτό είναι το πρόβλημα των ανταγωνιστικών (adversarial) bandits, όπου ουσιαστικά το περιβάλλον θεωρείται αντίπαλος. Ο αντίπαλος μπορεί να έχει πολύ μεγάλη

ισχύ, ακόμα και την ικανότητα να δει τον κώδικα των αλγορίθμων και να διαλέξει ανταμοιβές αντίστοιχα. Παρ'όλα αυτά το πρόβλημα αυτό δεν είναι πολύ δυσκολότερο από το στοχαστικό πρόβλημα.

Ανάμεσα στα δύο αυτά άκρα, υπάρχουν πολλές επιλογές και υποθέσεις που μπορούμε να κάνουμε σχετικά με το περιβάλλον και τις πολιτικές.

3.2 Στοχαστικοί bandits

Πιο φορμαλιστικά ένα στοχαστικό σύστημα bandits είναι μια συλλογή απο κατανομές $v = (P_a : a \in \mathcal{A})$, όπου $\mathcal{A}$ είναι, όπως πάντα το σύνολο των δυνατών δράσεων. Το περιβάλλον και ο πράκτορας αλληλεπιδρούν διαδοχικά για n γύρους. Συνήθως το πλήθος των γύρων (ο ορίζοντας) είναι πεπερασμένος, αλλά πολλές η αλληλεπίδραση είναι αέναη. Σε κάθε γύρο $t \in \{1, \dots, n\}$ ο πράκτορας επιλέγει μια δράση $A_t \in \mathcal{A}$, η οποία τροφοδοτεί το περιβάλλον. Το περιβάλλον τότε επιλέγει μια ανταμοιβή $X_t \in \mathbb{R}$ από μια κατανομή P_{A_t} και επιστρέφει το X_t στον πράκτορα. Η αλληλεπίδραση μεταξύ πράκτορα και περιβάλλοντος παράγει ένα μέτρο πιθανότητας στην αλληλουχία των αποτελεσμάτων $A_1, X_1, A_2, X_2, \dots, A_n, X_n$. Αυτή η αλληλουχία ικανοποιεί τις παρακάτω υποθέσεις:

1. Η δεσμευμένη πιθανότητα της ανταμοιβής X_t δεδομένου $A_1, X_1, \dots, A_{t-1}, X_{t-1}, A_t$ είναι P_{A_t} που διαισθητικά σημαίνει ότι το περιβάλλον παίρνει ένα δείγμα X_t από την κατανομή P_{A_t} στον γύρο t.
2. Ο δεσμευμένος κανόνας της δράσης A_t δεδομένων $A_1, X_1, \dots A_{t-1}, X_{t-1}$ είναι $\pi(\cdot|A_1, X_1, \dots, A_{t-1}, X_{t-1})$ όπου $\pi_1, \pi_2, \dots$ είναι η αλληλουχία από Μαρκοβιανούς πυρήνες που περιγράφουν τον πράκτορα. Διαισθητικά αυτό σημαίνει ότι ο πράκτορας δεν μπορεί να χρησιμοποιήσει παρατηρήσεις από το μέλλον σε τρέχουσες αποφάσεις.

Ο στόχος του πράκτορα είναι να μεγιστοποιήσει την συνολική ανταμοιβή $S_n = \sum_{t=1} nX_t$ (η οποία διαφέρει ελαφρά από την ανταμοιβή του προβλήματος EM, καθώς εδώ $\gamma = 1$). Η συνολική ανταμοιβή είναι μια τυχαία ποσότητα η οποία εξαρτάται από τις πράξεις του πράκτορα και τις ανταμοιβές που πήρε από το περιβάλλον.

Αν έχουμε την πολιτική $v = (P_a : a \in \mathcal{A})$, τότε μπορούμε να ορίσουμε το μέσο όρο κάθε χεριού

$$\mu_\alpha(v) = \int_{-\infty}^{\infty} x dP_a(x)$$

Τότε μπορούμε να ορίσουμε το $\mu^*(v) = \max_{a \in \mathcal{A}} \mu_a(v)$ ο μέγιστος μέσος όρος των χεριών.

Τότε η μετάνοια της πολιτικής π σε ένα πρόβλημα bandit είναι

$$R_n(\pi, v) = n\mu^*(v) - \mathbb{E}\left[\sum_{t=1}^{n} X_t\right] \tag{3.1}$$

όπου η αναμενόμενη τιμή υπολογίζεται με βάση την πιθανότητα των αποτελεσμάτων που δημιουργούνται από την αλληλεπίδραση του π και του v.

Όλες οι πολιτικές βασίζονται στην παρατήρηση ότι για να μειώσουμε την μετάνοια, ο αλγόριθμος πρέπει να ανακαλύψει την δράση/χέρι με τον μεγαλύτερο μέσο όρο. Συνήθως αυτό σημαίνει ότι ο πράκτορας πρέπει να παίξει κάθε χέρι κάποιον αριθμό φορών, ώστε να δημιουργήσει μια εκτίμηση της μέσης τιμής του χεριού, και στην συνέχεια να παίξει το χέρι με την μεγαλύτερη τιμή. Έτσι το πρόβλημα μπορεί να συνοψιστεί ως την προσπάθεια να ανακαλύψει ο πράκτορας πόσο συχνά πρέπει να παίξει κάθε χέρι, ώστε να μπορεί με στατιστική βεβαιότητα να πει ότι έχει βρει το βέλτιστο χέρι.

3.3 Ανταγωνιστικοί Bandits

Το πλαίσιο των ανταγωνιστικών bandits έχει τις ρίζες του στην θεωρία παιγνίων. Ένα παράδειγμα ενός τέτοιου προβλήματος είναι το εξής παιχνίδι. Παίζουμε μια ένα φίλο μας ένα απλό παιχνίδι με bandits, όπου ο ορίζοντας είναι $n = 1$ και έχουμε 2 δράσεις. Το παιχνίδι έχει την ακόλουθη μορφή:

- Λέμε στον φίλο μας την στρατηγική με βάση την οποία θα επιλέξουμε την δράση.
- Ο φίλος μας διαλέγει κρυφά ανταμοιβές $x_1 \in \{0, 1\}$ και $x_2 \in \{0, 1\}$.
- Εφαρμόζουμε την στρατηγική που επιλέξαμε $A \in \{1, 2\}$ και παίρνουμε ανταμοιβή x_A.
- Η μετάνοια είναι $R = \max x_1, x_2 - x_A$

Προφανώς αν ο φίλος μας επιλέξει και τις δύο ανταμοιβές να είναι 0, τότε η μετάνοια θα είναι πάντα 0. Ο τρόπος για να είναι η στρατηγική επιτυχημένη είναι η τυχαιότητα στις επιλογές μας. Έτσι παρόλο που ο αντίπαλος γνωρίζει την στρατηγική μας, δεν γνωρίζει ακριβώς τις επιλογές που θα κάνουμε. Για παράδειγμα, μπορούμε να πούμε στον φίλο μας, ¨Θα επιλέξω την κίνηση $A = 1$ με πιθανότητα $1/2$' και η αναμενόμενη μετάνοια γίνεται $R = 1/2$. Οσο μεγαλώνει ο ορίζοντας, το πλεονέκτημα του αντιπάλου όλο και μειώνεται.

Πιο φορμαλιστικά, αν $k > 1$ ο αριθμός των χεριών, τότε ένα πρόβλημα ανταγωνιστικού bandit k-χεριών είναι μια αυθαίρετη σειρά από διανύσματα ανταμοιβών $(x_t)_{t=1}^n$, όπου $x_t \in [0, 1]^k$. Σε κάθε γύρο ο πράκτορας διαλέγει μια κατανομή πράξεων $P_t \in P_{k-1}$. Τότε η δράση $A_t \in [k]$ είναι ένα δείγμα από την κατανομή P_t, και ο πράκτορας λαμβάνει ανταμοιβή x_{tA_t}.

Η πολιτική σε αυτή την περίπτωση είναι μια συνάρτηση $\pi : ([k] \times [0, 1])^* \to \mathcal{P}_{k-1}$, η οποία χαρτογραφεί ακολουθίες της ιστορίας σε κατανομές πάνω σε πράξεις. Η επίδοσης της πολιτικής π σε ένα περιβάλλον x μετριέται από την αναμενόμενη μετάνοια, η οποία είναι η αναμενόμενη απώλεια σε κέρδος της πολιτικής π σε σχέση με την καλύτερη πολιτική που επιλέγει ένα χέρι κάθε φορά.

$$R_n(\pi, x) = \max_{i \in [k]} \sum_{t=1}^{n} x_{t_i} - \mathbb{E}\left[\sum_{t=1}^{n} x_{tA_t}\right] \tag{3.2}$$

όπου η αναμενόμενη τιμή είναι πάνω στην τυχαιοτητα των πράξεων του πράκτορα.

Η μετάνοια χειρότερης περίπτωσης σε όλα τα περιβάλλοντα είναι

$$R_n^*(\pi) = \sup_{x \in [0,1]^{n \times k}} R_n(\pi, x)$$

Για να φτιάξουμε πολιτικές που είναι υπο-γραμμικές στο n στην χειρότερη περίπτωση, δηλαδή πολιτικές π που ισχύει

$$\lim_{n \to \infty} \frac{R_n^*(\pi)}{n} = 0$$

θα πρέπει να χρησιμοποιήσουμε πολιτικές με τυχαιότητα.

3.4 Γνωστοί αλγόριθμοι

Παρακάτω παρουσιάζονται κάποιοι από τους πιο γνωστούς αλγορίθμους bandits. Οι ε-άπληστοι (ε-greedy), Ανώτατο Όριο Εμπιστοσύνης (Upper Confidence Bound - UCB) και η δειγματοληψία Thompson αποτελούν λύσεις στο πρόβλημα των στοχαστικών bandits, ενώ ο EXP3 αποτελεί λύση στο πρόβλημα των ανταγωνιστικών bandits.

3.4.1 Αλγόριθμος ε-greedy

Ο αλγόριθμος ε-greedy είναι ο απλούστερος αλγόριθμος και ίσως η πιο προφανής λύση του διλήμματος μεταξύ εξερεύνησης και εκμετάλλευσης. Η πολιτική εξερευνά ένα τυχαίο χέρι με πιθανότητα ε, ενώ με πιθανότητα $1-\varepsilon$, η πολιτική εκμεταλλεύεται την λύση με την μεγαλύτερη ανταμοιβή κατά μέση τιμή. Στην κλασική έκδοση του αλγορίθμου το ε είναι σταθερά, αλλά αυτό δεν είναι απαραίτητο. Αντίθετα βγάζει νόημα το ε να εξαρτάται από τις επαναλήψεις (γραμμική μείωση, εκθετική μείωση, εξερεύνηση με πιθανότητα ε για κάποιες επαναλήψεις και εκμετάλλευση με πιθανότητα $1-\varepsilon$ και καμία εξερεύνηση αργότερα. Έτσι η επόμενη κίνηση A_t επιλέγεται απο τον τύπο.

$$A_t = \begin{cases} randint(1, k), & \text{if } n \leq \varepsilon \\ \underset{a}{argmax}\, Q_{t-1}(a), & \text{otherwise} \end{cases}$$

όπου το n είναι μια τυχαία μεταβλητή η οποία προέρχεται από μια ομοιόμορφη κατανομή ανάμεσα στο 0 και το 1. Η $randint$ είναι μια συνάρτηση που επιστρέφει ένα συγκεκριμένο ακέραιο μέσα στο δοθέν εύρος, k είναι το πλήθος των χεριών, $Q_{t-1}(a)$ είναι η αναμενόμενη μέση τιμή του χεριού του $a-$οστού χεριού την χρονική στιγμή $t-1$.

Αυτή η πολιτική είναι απλή και υπάρχουν πολιτικές που επιφέρουν καλύτερη μετάνοια, αφού υπάρχουν πιο έξυπνοι τρόποι εξερεύνησης σε σχέση με την τυχαία επιλογή. Αφού ο αλγόριθμος έχει τρέξει για κάποιους γύρους, μπορούμε να δούμε ήδη ότι κάποια από τα μπράτσα έχουν κακή απόδοση, και δεν υπάρχει ανάγκη περαιτέρω εξερεύνησης τους, οπότε αυτή η εξερεύνηση θα μπορούσε να χρησιμεύσει για χέρια που έχουν καλύτερες πιθανότητες να είναι το βέλτιστο χέρι.

3.4.2 Upper Confidence Bound (UCB)

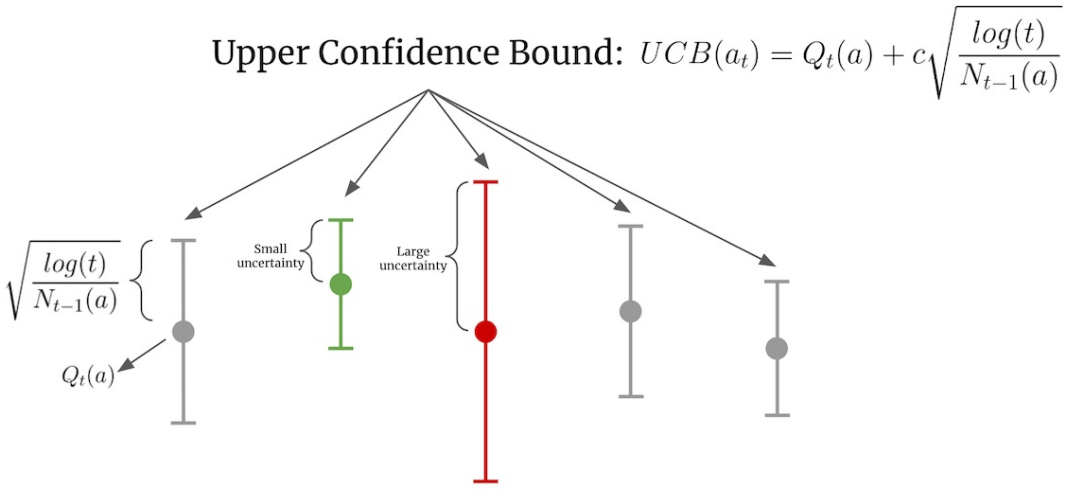
Η βασική ιδέα του αλγορίθμου Μέγιστου Ορίου Εμπιστοσύνης (Upper Confidence Bound - UCB) είναι η επιλογή πάντα του χεριού με το υψηλότερο μέγιστο όριο. Μπορεί να περιγραφή σαν αισιοδοξία στην αντιμετώπιση αβεβαιότητας. Το προβλεπόμενο μέγιστο όριο αποτελείται από δύο στοιχεία: την προβλεπόμενη μέγιστη ανταμοιβή και την αβεβαιότητα, όπως φαίνονται στην εξίσωση.

$$A_t = \underset{a}{argmax} \left[Q_{t-1}(a) + \sqrt{\frac{\log(t-1)}{N_{t-1}(a)}} \right] \tag{3.3}$$

όπου $Q_{t-1}(a)$ είναι η προβλεπόμενη μέση ανταμοιβή του a-οστού χεριού την χρονική στιγμή $t-1$, $t-1$ είναι το πλήθος των χεριών που έχουν τραβηχθεί μέχρι τώρα (ή ο αριθμός των βημάτων γενικότερα) και N_{t-1} είναι το πλήθος των φορών που το a-οστό χέρι έχει τραβηχθεί. Έτσι χέρια με μεγαλύτερη μέση ανταμοιβή έχουν μεγαλύτερη τιμή μεγίστου ορίου. Χέρια που δεν έχουν εξερευνηθεί, τείνουν να έχουν καλύτερα σκορ λόγω εκτιμήσεων αβεβαιότητας. Αυτό θα επιφέρει μικρότερη μετάνοια σε σχέση με τον ε-άπληστο αλγόριθμο, καθώς μικρότερο ποσό εξερευνησης θα καταναλωθεί σε εμφανώς μη-βέλτιστα χέρια. Οπτικά τα παραπάνω φαίνονται στο Σχήμα 3.1, όπου έχουμε ένα πράσινο χέρι που έχει επιλεχθεί πολλές φορές και ένα κόκκινο που έχει επιλεχθεί λίγες. Όπως βλέπουμε το άνω όριο εμπιστοσυνης του κόκκινου είναι το ψηλότερο σε αυτό το βήμα, οπότε αυτό είναι το χέρι που θα επιλεχθεί, το διάστημα εμπιστοσύνης του θα μικρύνει και το κέντρο θα μετακινηθεί ανάλογα με την ανταμοιβή.

3.4.3 Δειγματοληψία Thompson

Η δειγματοληψία Thompson χτίζει μια κατανομή πιθανότητας βασισμένη σε ιστορικές ανταμοιβές και μετά δειγματοληπτεί από την κατανομή καθε δράσης για την επιλογή αυτής που επιφέρει την μέγιστη αναμενόμενη ανταμοιβή. Στην απλή περίπτωση που η ανταμοιβή είναι δυαδική (0 ή 1) και άρα θέλουμε να υπολογίσουμε την πιθανότητα να υπάρχει ανταμοιβή, χρησιμοποιειται η κατανομή Βήτα για την μοντελοποίηση των κατανομών των ανταμοιβών του κάθε χεριού. Η κατανομή Βήτα παίρνει δύο παραμέτρους, τα α και β, όπου α είναι οι φορές που η ανταμοιβή είναι 1 και β είναι η φορές που η ανταμοιβή ήταν 0. Η μέση τιμή της κατανομής είναι $\frac{\alpha}{\alpha+\beta}$, το οποίο αντιστοιχεί στο κλάσμα των επιτυχιών προς το σύνολο των προσπαθειών. Για την επιλογή μιας δράσης, δειγματοληπτούμε από την κατανομή Βήτα κάθε χεριού και επιλέγουμε τον χέρι με την υψηλότερη δειγματολημμένη τιμή [13].



Σχήμα 3.1: Οπτικοποίηση UCB [25] διάφορων χεριών

Η κατανομή Βήτα για διάφορα α και β φαίνεται στο Σχήμα 3.2. Όσο αυξάνεται το πλήθων των α και β, η κατανομή στενεύει και άρα βρίσκεται πιο κοντά στην πιθανότητα που έχει το κάθε χέρι να επιφέρει ανταμοιβή. Έτσι χέρια που έχουν δοκιμαστεί λιγότερο και έχουν πιο διευρυμένη κατανομή, έχουν πιθανότητες να δοκιμαστούν ως εξερεύνηση για να εξεταστεί αν επιφέρουν καλύτερες ανταμοιβές.

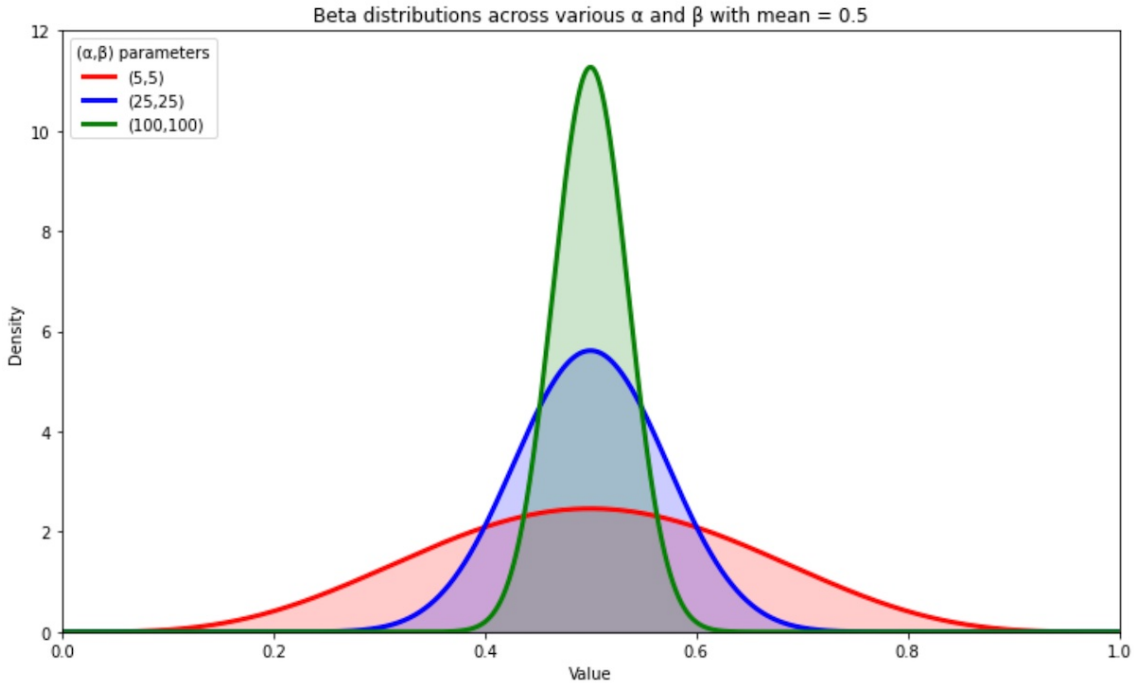
Η δειγματοληψία Thompson δουλεύει και σε περιπτώσεις που η ανταμοιβή δεν είναι δυαδική, απλά η κατανομή που χρησιμοποιειται δεν είναι η Βήτα, αλλά κάποια άλλη.

3.4.4 Ο αλγόριθμος EXP3

Το EXP3 σημαίνει Exponential-weight algorithm for Exploration and Exploitation, δηλαδή Αλγόριθμος εκθετικών βαρών για αναζήτηση και εκμετάλλευση. Ο τρόπος που δουλεύει είναι διατηρώντας μια λίστα με τα βάρη κάθε δράσης και χρησιμοποιώντας τα για να επιλέξει τυχαία ποια δράση να κάνει μετά. Τέλος, τα αντίστοιχα βάρη αυξάνονται/μειώνονται όταν η ανταμοιβή είναι καλή/κακή. Επίσης υπάρχει ο παράγοντας γ, ο οποίος ορίζει την θέληση να επιλέξουμε μια δράση με ομοιόμορφη τυχαιότητα. Έτσι, αν $\gamma = 1$, τα βάρη δεν έχουν καμία επίδραση στις επιλογές σε κάθε βήμα.

Ο αλγόριθμος μπορεί να περιγραφεί ως:

1. Δοθέντος $\gamma \in [0, 1]$, αρχικοποιούμε τα βάρη $w_i(1) = 1$ για $\iota = 1, \ldots, K$, όπου K είναι τα χέρια.
2. Για κάθε γύρο t:



Σχήμα 3.2: Η κατανομή Βήτα στενεύει όσο τα α και β μεγαλώνουν

(α′) Θέτουμε $p_i(t) = (1-\gamma)\frac{w_i(t)}{\sum_{j=1}^{K} w_j(t)} + \frac{\gamma}{K}$ για κάθε i.

(β′) Επιλέγουμε την επόμενη δράση i_t τυχαία με βάση την κατανομή του $p_i(t)$.

(γ′) Παρατηρούμε την ανταμοιβή $x_{i_t}(t)$.

(δ′) Ορίζουμε την αναμενόμενη ανταμοιβή $\hat{x}_{i_t}(t)$ να είναι $x_{i_t}(t)/p_{i_t}(t)$. Αυτό το βήμα εξασφαλίζει ότι η δεσμευμένη προσδοκία της αναμενόμενης ανταμοιβής είναι η πραγματική ανταμοιβή.

(ε′) Θέτουμε $w_{i_t}(t+1) = w_{i_t}(t)e^{\gamma \hat{x}_{i_t}(t)/K}$

(ϛ′) Θέτουμε όλα τα άλλα $w_j(t+1) = w_j(t)$.

3.5 Contextual Bandits

Η περίπτωση bandits που μας ενδιαφέρει είναι αυτή στην οποία ο αλγόριθμος έχει πρόσβαση σε πληροφορίες σχετικά με το συγκείμενο του περιβάλλοντος, τις οποίες θα μπορούσε να χρησιμοποιήσει για να πάρει καλύτερες αποφάσεις. Το πρόβλημα και η μετρική (η μετάνοια) που μελετήσαμε νωρίτερα δεν χρησιμοποιούσαν τέτοια δεδομένα και προσπαθούσαν να επιλέξουν την καλύτερη κίνηση. Το πρόβλημα αυτό στην μορφή που περιγράφουμε εδώ, μελετήθηκε από τον John Langford και τον Tong Zhang στο [4], καθώς σε προβλήματα στον πραγματικό κόσμο πάντα υπάρχουν επιπλέον πληροφορίες που μπορεί να χρησιμοποιήσει ο πράκτορας για να πάρει καλύτερες αποφάσεις. Συγκεκριμένα το πρόβλημα που προσπαθούσαν να λύσουν είναι η αντιστοίχηση διαφημισεων σε περιεχόμενο ιστοσελίδων στο ίντερνετ.

Για να μελετήσουμε αυτή την νέα περίπτωση θα χρειαστεί να επεκτείνουμε το πλαίσιο στο οποίο εργαζόμαστε, καθώς για τον ορισμό της μετάνοιας, ώστε μπορέσουμε να μοντελοποιήσουμε αυτα τα προβλήματα, τα οποία περιέχουν πληροφορίες συγκειμένου. Είναι σημαντικό να έχουμε υπ΄όψιν ότι όταν σχεδιάζουμε μια καινούρια μετρική έχουμε να συμβιβαστούμε μεταξύ της μεροληψίας και της διακύμανσης bias-variance trade-off. Μεροληψίας από την άποψη ότι δεν θέλουμε να βρούμε μια κακή μετρική με την οποία να συγκριθούμε, γιατί τότε κάθε αλγόριθμος που θα έχει παρόμοια απόδοση με την μετρική θα έχει κακή απόδοση. Από την άλλη, ο συναγωνισμός με μια καλύτερη μετρική μπορεί να είναι πολύ δύσκολος από την προοπτική της μάθησης και αυτή η τιμωρία μπορεί να υπερτερεί των πλεονεκτημάτων.

Αν προσεγγίσουμε τους contextual bandits με χρήση των ιδεών από τους ανταγωνιστικούς bandits, τότε το πρόβλημα παίρνει την παρακάτω μορφή:

1. Ο αντίπαλος κρυφά διαλέγει ανταμοιβές $(x_t)_{t=1}^n$ με $x_t \in [0,1]^k$
2. Ο αντίπαλος κρυφά διαλέγει συγκείμενο $(c_t)_{t=1}^n$ με $c_t \in \mathcal{C}$, όπου $\mathcal{C}$ είναι το σταθερό σύνολο των πιθανών συγκειμένων.
3. Για τους γύρους $t = 1, 2, \ldots, n$:

(α′) Ο πράκτορας παρατηρεί συγκείμενο $c_t \in \mathcal{C}$

(β′) Ο πράκτορας διαλέγει κατανομή $P_t \in \mathcal{P}_{k-1}$ και παίρνει δείγμα A_t από το P_t

(γ′) Ο πράκτορας παρατηρεί ανταμοιβή $X_t = x_{tA_t}$

Το πλαίσιο των contextual bandits μας προσφέρει το εργαλείο για να περιγράψουμε το πρόβλημα των συστάσεων που θέλουμε να λύσουμε. Μετά την εισαγωγή του προβλήματος σαν μέθοδο για την επίλυση τέτοιου είδους προβλημάτων το 2007, έχει υπάρξει σημαντική έρευνα, αλλά και χρήση στην βιομηχανία τέτοιων μεθόδων, ειδικά σε περιπτώσεις που το πλήθος των αντικειμένων αλλάζει δυναμικά, πχ νέα άρθρα προστίθενται κάθε μέρα, ενώ τα παλιά άρθρα είναι πλεον λιγότερο σημαντικά. Έτσι εταιρίες όπως η Microsoft, το LinkedIn, το Netflix και άλλες χρησιμοποιούν τέτοιες μεθόδους για να προτείνουν άρθρα, διαφημίσεις ή ταινίες αντίστοιχα. Το Netflix χρησιμοποιεί επίσης contextual bandits για να επιλέξει την εικόνα που δείχνει για κάθε ταινία [12]. Στα επόμενα μέρη της διπλωματικής, δεν αναλύσουμε περισσότερο τις ιδιότητες των αλγορίθμων αυτών, ουτε θα προχωρήσουμε σε αναλύσεις της μετάνοιας τους, αλλά θα τους χρησιμοποιήσουμε μόνο βάση της ανάλυσης στις εργασίες που εισήχθηκαν.

Κεφάλαιο 4

Διάλογος και Συστάσεις

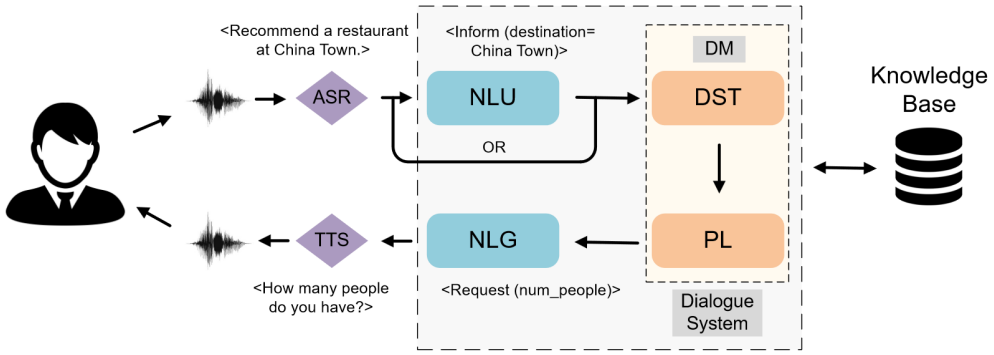
Σε αυτό το κεφάλαιο θα περιγράψουμε πολύ περιληπτικά την χρήση της ενισχυτικής μάθησης σε διαλογικά συστήματα, ώστε να μπορέσουμε να το συνδέσουμε με το πρόβλημα μας, και μετά θα περιγράψουμε την χρήση ενισχυτικής μάθησης σε προβλήματα συστάσεων, που είναι και πιο κοντά στο πεδίο του προβλήματος μας.

4.1 Διαλογικά Συστήματα

Η ιδέα της δημιουργίας ενός πράκτορα που θα μπορεί να απαντήσει σε ανθρώπινες ερωτήσεις, ξεκίνησε από το σύγγραμα του Alan Turing, Computing Machinery and Intelligence [1]. Τα πρώτα μοντέλα που δημιουργήθηκαν βασίζονταν σε κανόνες, όπου αναγνώριζαν κάποιες λέξεις-κλειδιά στο κείμενο και ανάλογα με αυτές απαντούσαν στον χρήστη. Το πρόβλημα αυτών των συστημάτων ήταν ότι ήταν πολύ δύσκολο να επεκταθούν, καθώς και να γενικεύσουν, αφού η δημιουργοί πρέπει να προσθέσουν χειροκίνητα τους επιπλέον κανόνες. Τα τελευταία χρόνια, η δημιουργία διαλογικών συστημάτων γίνεται ολοένα και περισσότερο με χρήση βαθιάς μηχανικής μάθησης, παρόλο που η χρήση κανόνων είναι ακόμα βολική σε κάποιες περιπτώσεις.

Τα διαλογικά συστήματα συνήθως χωρίζονται σε δύο κατηγορίες ανάλογα με τον σκοπό τους:

1. Συγκεκριμένου σκοπού (task-oriented systems). Τα διαλογικά αυτά συστήματα έχουν ως στόχο να βοηθήσουν τον χρήστη να πετύχει συγκεκριμένους στόχους, ιδανικά σε όσο λιγότερους γύρους διαλόγου γίνεται. Για παράδειγμα τέτοια συστήματα είναι συστήματα μέσω των οποίων ο χρήστης μπορεί να κλείσει εισιτήρια, ή να λάβει υποστήριξη σχετικά με ένα πρόβλημα του.
2. Ανοιχτού σκοπού (open-domain systems). Τα διαλογικά συστήματα αυτά δεν έχουν κάποιο σκοπό, αλλά εστιάζουν στο να δώσουν ρεαλιστικές απαντήσεις σε συζητήσεις με τον χρήστη.



Σχήμα 4.1: Σύστημα συγκεκριμένου σκοπού [22]

Συχνά τα συστήματα συγκεκριμένου σκοπού οργανώνονται σε μια αλληλουχία μερών όπως αυτή που φαίνεται στο Σχήμα 4.1:

- **Το κομμάτι της κατανόησης της εισόδου του χρήστη.** Αυτό το κομμάτι είναι υπευθυνο για την ταξινόμηση των διάφορων λέξεων σε μέρη του λόγου, την αναγνώριση ονομάτων, αλλά και την αναγνώριση της πρόθεσης του χρήστη με βάση το τι είπε. Κάποια συστήματα δεν χρησιμοποιούν αυτό το κομμάτι και χρησιμοποιούν το ίδιο το μήνυμα του χρήστη ως είσοδο στο επόμενο κομμάτι, όπως στο Σχήμα 4.1. Αυτό συμβαίνει για να μειώσουν την επίδραση του λάθους του πρώτου αυτού του κομματιού και την μεταφορά λαθών στα μετέπειτα κομμάτια.
- **Το κομμάτι της διαχείρισης της κατάστασης του διαλόγου,** το οποίο ρυθμίζει τις καταστάσεις του διαλόγου με βάση την τρέχουσα είσοδο και την ιστορία του διαλόγου. Η κατάσταση του διαλόγου περιέχει σχετικές δράσεις του χρήστη και ζευγάρια θέσης-τιμής.
- **Το κομμάτι της εκμάθησης της πολιτικής του διαλόγου**, το οποίο με βάση τις καταστάσεις του διαλόγου που παίρνει από το προηγούμενο κομμάτι, επιλέγει την επόμενη δράση του διαλογικού πράκτορα.
- **Το κομμάτι της παραγωγής της απάντησης του συστήματος**, το οποίο μετατρέπει τις δράσεις που επιλέχθηκαν από το προηγούμενο κομμάτι σε φυσική γλώσσα, η οποία θα επιστραφεί στον χρήστη.

Άλλες φορές προτιμάται ένα σύστημα που υλοποιεί όλες τις παραπάνω λειτουργίες από άκρη σε άκρη, το οποίο μπορεί να πετύχει καλύτερη βελτιστοποίηση, καθώς δεν υπάρχει μεταφορά σφάλματος μεταξύ των διάφορων κομματιών.

4.1.1 Κατανόηση γλώσσας

Η κατανόηση γλώσσας είναι ένας τομέας της Μηχανικής Μάθησης, ο οποίος έχει αναπτυχθεί σε μεγάλο βαθμό τα τελευταία χρόνια.

Όσον αφορά την ταξινόμηση προθέσεων, μια από τις πιο διαδεδομένες μεθόδους είναι η χρήση διανυσματικής αναπαράστασης λέξεων (word embeddings) για την εξαγωγή των χαρακτηριστικών και μετά ενός μοντέλου, όπως για παράδειγμα ενός SVM για την ταξινόμηση με βάση αυτές τις αναπαραστάσεις. Σε αυτή την αναπαράσταση, οι λέξεις αναπαρίστανται σαν πυκνά αριθμητικά διανύσματα, διατηρώντας όμως σημασιολογικά και συντακτικά χαρακτηριστικά των λέξεων. Έτσι παρόμοιες λέξεις αναπαρίστανται από παρόμοια διανύσματα. Κάθε αναπαράσταση, είναι μοναδική για την γλώσσα στην οποία εκπαιδεύτηκε το μοντέλο αυτό. Το πρώτο τέτοιο μοντέλο προτάθηκε από τον Mikolov στο [7]. Μετά μετατροπή του κειμένου σε διανύσματα, εκπαιδεύεται ένα SVM να κάνει την ταξινόμηση.

Όσον αφορά την εξαγωγή οντοτήτων, η τεχνική που χρησιμοποιείται είναι αρκετά παρόμοια.

Τα τελευταία χρόνια, οι μετασχηματιστές transformers, οι οποίοι προτάθηκαν αρχικά από ερευνητές στην Google[14], μπορούν να επιλύσουν προβλήματα κατανοησης φυσικής γλώσσας πολύ καλύτερα από οποιαδήποτε προηγούμενη τεχνολογία. Χάρη στην ιδέα της αυτό-προσοχής (self-attention), οι μετασχηματιστές είναι πολύ καλοί στο να εντοπίζουν τις συσχετίσεις μεταξύ των λέξεων και να έχουν καλύτερη κατανόηση του νοήματος μιας πρότασης. Έτσι, μετά την εκπαίδευση τους, με την χρήση fine-tuning, μπορούν να χρησιμοποιηθούν για τα παραπάνω προβλήματα, ξεπερνώντας τα προηγούμενα καλύτερα μοντέλα.

4.1.2 Διαχείριση κατάστασης και πολιτικής διαλόγου

Όσον αφορά τους διαλογικούς πράκτορες, υπάρχουν τρία επίπεδα για την διαχείριση της κατάστασης του διαλόγου και την επιλογή της επόμενης κατάστασης, τα οποία ιστορικά αναπτύχθηκαν με αυτή την σειρά.

Το πρώτο επίπεδο, είναι η μη διατήρηση της κατάστασης του διαλόγου. Αυτό σημαίνει ότι ο διαλογικός πράκτορας είναι ικανός να απαντήσει μόνο την τρέχουσα ερώτηση. Αυτό μπορεί να περιγραφή σαν ένα ερώτημα σε μηχανή αναζήτησης, η οποία όμως μπορεί να εξάγει και να καταλάβει προθέσεις και άλλα δομημένα δεδομένα. Ένα παράδειγμα τέτοιου ερωτήματος θα μπορούσε να είναι ¨βρες μου μια φτηνή πτήση από το Άμστερνταμ στην Ρώμη¨.

Το δεύτερο επίπεδο, είναι η υπαρξη κάποιας έννοιας κατάστασης με χρήση μηχανών πεπερασμένης κατάστασης. Ουσιαστικά ο δημιουργός του συστήματος σχεδιάζει κανόνες με βάση τους οποίους γίνονται οι μεταβάσεις από μια κατάσταση σε μια άλλη. Το πρόβλημα είναι ότι γρήγορα αυτές οι συνθήκες γίνονται δύσκολες στην διαχείριση και την αποσφαλμάτωση. Επιπλέον, συνήθως οι χρήστες δεν ακολουθούν απόλυτα τα ¨χαρούμενα μονοπάτια¨, δηλαδή τις αλληλουχίες διαλόγου που ο χρήστης συμπεριφέρεται γίνονται ακριβώς όπως θα έπρεπε, χωρίς παρεκτροπές. Πολλές φορές οι χρήστες θα ρωτήσουν ερωτήσεις για τις οποίες δεν υπάρχει

προϋπάρχον κανόνας, οπότε και αυτές οι επιλογές θα πρέπει να προστεθούν αργότερα. Έτσι τελικα το σύστημα καταλήγει να είναι μια σειρά από ακραίες περιπτώσεις, οι οποίες πολλές φορές είναι και αντιφατικές μεταξύ τους.

Το τρίτο επίπεδο, είναι η χρήση τεχνικών μηχανικής μάθησης για την εκπαίδευση της πολιτικής του διαλόγου. Δυστυχώς υπάρχουν πολλά προβλήματα στην ιδέα αυτή. Η δημιουργία ενός συστήματος με χρήση επιβλεπόμενης μάθησης θα απαιτούσε την ύπαρξη ενός μεγάλου συνόλου δεδομένων, το οποίο περιέχει σχόλια σχετικά με την πρόθεση του χρήστη. Κάτι τέτοιο είναι πολύ ακριβό και δύσκολο να δημιουργηθεί, καθώς πρέπει να γίνει χειροκίνητα. Μια άλλη ιδέα θα ήταν η χρήση ενισχυτικής μάθησης, η οποία περιγράφεται σε παρακάτω ενότητα. Επιπλέον, το Rasa, το οποίο χρησιμοποιούμε για την εργασία μας, εφαρμόζει μια τεχνική παρόμοια με την EM.

4.1.3 Παραγωγή γλώσσας

Ο πιό απλός τρόπος παραγωγής γλώσσας είναι η χρήση προτύπων (templates) απαντήσεων, στις οποίες μπορούν να προστίθενται συγκεκριμένες οντότητες σε προκαθορισμένες θέσεις (slots) μέσα στο κείμενο.

Οπως και στην κατανόηση κειμένου, και εδώ τα τελευταία χρόνια η σημαντικότερη πρόοδος είναι η τεχνολογία των μετασχηματιστών,οι οποίοι έχουν αλλάξει πλήρως το τοπίο. Τρανό παράδειγμα σε αυτό είναι το ChatGPT που αναφέρθηκε και παραπάνω, το οποίο είναι ικανό να κατανοήσει και να παράγει φυσική γλώσσα η οποία είναι πολύ εύγλωττη και καθαρή.

4.1.4 Ενισχυτική Μάθηση και Διαλογικά Συστήματα

Όσον αφορά την χρήση ενισχυτικής μάθησης, σε συστήματα συγκεκριμένου σκοπού είναι στην διαχείριση του διαλόγου. Πιο συγκεκριμένα δύο συνήθεις χρήσεις είναι για την παρακολούθηση της κατάστασης του διαλόγου και την εκμάθηση της πολιτικής. Η δεύτερη χρήση είναι ιδιαίτερα συνήθης και αρκετά επιτυχημένη, καθώς περιγράφεται ακριβώς από ένα πρόβλημα EM (πχ [17]).

Σε συστήματα ανοιχτού σκοπού, η χρήση της ενισχυτικής μάθησης είναι κυρίως η επιλογή απαντήσεων παρά η παραγωγή τους, καθώς τα παραγωγικά generative συστήματα είναι πολύ καλύτερα στην παραγωγή λόγου.

Για παράδειγμα, σε μια από τις πρώτες δουλείες οι οποίες χρησιμοποιησαν EM σε διάλογο [10], οι συγγραφείς προσπάθησαν να ενώσουν τις ιδέες από τα seq2seq μοντέλα και την EM, ώστε να δημιουργήσουν συστήματα τα οποία επιστρέφουν καλύτερες απαντήσεις. Έτσι δημιούργησαν μια μετρική ανταμοιβής η οποία αξιολογούσε την ποιότητα των απαντήσεων. Αρχικά εκπαίδευσαν ένα seq2seq μοντέλο με επιβλεπόμενη μάθηση, και μετά με βάση αυτό προσομοίασαν διαλόγους μεταξύ δυο πρακτόρων, ξεκινώντας από μια πρόταση από το σύνολο εκπαίδευσης.

Ενα από τα πιο διάσημα σύγχρονα παραδείγματα είναι η χρήση του στην εκπαίδευση του

InstructGPT. Στο [24], οι ερευνητές έκαναν fine-tune το ΓΠΤ-3 με χρήση ενισχυτικής μάθησης και ανατροφοδότησης από ανθρώπους. Συγκεκριμένα, κατα το fine-tuning, το μοντέλο ¨ρώταγε' τους χρήστες ποια από τις απαντήσεις θεωρούσαν καλύτερη σε σχέση με ένα ερώτημα, οι χρήστες ταξινομούσαν τις απαντήσεις από καλύτερη προς χειρότερη, και με βάση αυτό εκπαιδεύτηκε ένα μοντέλο ανταμοιβών. Έπειτα, για την εκπαίδευση της πολιτικής των απαντήσεων του συστήματος, ενα ερώτημα επιλεγόταν από το σύνολο των δεδομένων, η πολιτική του συστήματος παράγει μια έξοδο και το μοντέλο ανταμοιβών επιλέγει την ανταμοιβή για αυτή την έξοδο. Με βάση αυτό ανανεώνεται η ανταμοιβή της πολιτικής.

4.2 Συστήματα Συστάσεων

Στην πραγματική ζωή, υπάρχουν πολλές περιπτώσεις που πρέπει να πάρουμε μια απόφαση, χωρίς να έχουμε κάποια προηγούμενη πληροφορία σχετικά με την κάθε επιλογή. Έτσι πολλές φορές βασιζόμαστε σε συστάσεις από άλλους, οι οποίοι έχουν παραπάνω εμπειρία στο θέμα, να μας βοηθήσουν.Έτσι δημιουργήθηκε το πρώτο αυτοματοποιημένο σύστημα με αυτό το σκοπό, το οποίο πήρε το όνομα *συνεργατικό φιλτράρισμα* (collaborative filtering). Αργότερα δημιουργήθηκε ο γενικότερος όρος *σύστημα συστάσεων* για να αναδείξει δύο γεγονότα: 1) Η μέθοδος δεν χρειάζεται να βασίζεται σε αφανή συνεργασία μεταξύ των χρηστών, 2) η μέθοδος μπορεί να προτείνει ενδιαφέροντα αντικείμενα, και όχι να τα φιλτράρει.

Ένα σύστημα συστάσεων αποτελείται από εργαλεία και αλγορίθμους οι οποίοι αναπτύχθηκαν με την ιδέα να βοηθήσουν τους χρήστες να βρουν αντικείμενα που τους ενδιαφέρουν. Σε μια γενική μορφή, ο στόχος είναι η δημιουργία του προφίλ των χρηστών βασισμένη στην ανατροφοδότηση μεταξύ συστήματος και χρήστη και η σύσταση αντικειμένων που να ταιριάζουν στο προφίλ αυτό. Το πρόβλημα απαντάται σε πολλούς κλάδους όπως της υγείας, της διασκέδασης, της ενημέρωσης, κλπ.

Παραδοσιακά το πρόβλημα των συστάσεων θεωρούνταν ένα πρόβλημα ταξινόμησης ή πρόβλεψης, αλλά πλεον η ακαδημαϊκή κοινότητα συμφωνεί ότι η αλληλεπίδραση μεταξύ χρήστη και συστήματος μοντελοποιειται καλύτερα ως ενα πρόβλημα αποφάσεων με διαδοχικά βήματα [21]. Έτσι μπορεί να περιγραφεί από μια Μαρκοβιανή Διαδικασία Αποφάσεων και να λυθεί με χρήση ενισχυτικής μάθησης.

Πριν περάσουμε σε τεχνικές με χρήση ενισχυτικής μάθησης, είναι σημαντικό να γνωρίσουμε περιληπτικά τους κλασικούς αλγορίθμους. Αυτοί είναι:

- *Συνεργατικό φιλτράρισμα*: Η ιδέα της μεθόδου είναι η ομαδοποίηση του χρήστη (ή των αντικειμένων) σε ομάδες με παρόμοια χαρακτηριστικά. Όταν το φιλτράρισμα γίνεται με βάση τον χρήστη, οι προτάσεις γίνονται με βάση τις παρόμοιες προτιμήσεις διάφορων χρηστών. Από την άλλη, όταν αναφερόμαστε σε φιλτράρισμα με βάση τα αντικείμενα, οι προτάσεις γίνονται με βάση τα αντικείμενα τα οποία σχετίζονται με αυτά που ο χρήστης έχει ήδη αλληλεπιδράσει. Η μέθοδος αυτή μπορεί να χωριστεί σε δύο προσεγγίσεις: με βάση την μνήμη, όπου ουσιαστικά για κάθε χρήστη/αντικείμενο γίνεται μια

σύγκριση ομοιότητας (πχ ομοιότητα συνημιτόνου) με τους υπόλοιπους και μετά με χρήση k-κοντινότερων γειτόνων, ή με βάση κάποιο μοντέλο, όπου ουσιαστικά δημιουργειται ένα μοντέλο που εκπαιδεύεται με τεχνικής μηχανικής μάθησης να βρίσκει ομοιότητες μεταξύ χρήστες [9].

- *Παραγοντοποίηση πινάκων*: Αυτή η μέθοδος χρησιμοποιείται και ξεχωριστά και ως μέρος του φιλτραρίσματος. Η ιδέα είναι η αναπαράσταση του χρήστη και των αντικειμένων ως αντικείμενα σε ένα χώρο λίγων διαστάσεων. Έπειτα η συμβατότητα χρήστη και προϊόντος υπολογίζεται είτε με χρήση εσωτερικού γινομένου, ή με χρήση κάποιου νευρωνικού δικτύου αν οι σχέσει είναι μη γραμμικές.

Οι κλασικές μέθοδοι, έχουν διάφορα προβλήματα, όπως το ότι το σύστημα δεν μπορεί να προσφέρει χρήσιμες συστάσεις σε ένα νέο χρήστη, ή οταν προστίθεται ένα νέο αντικείμενο (cold-start), δεν μπορεί να προσφέρει προτάσεις σε χρήστες που δεν ανήκουν σε κάποια κατηγορία ξεκάθαρα (gray sheep), ενώ δεν μπορεί να κλιμακώσει, να ανταπεξέλθει σε ποικιλία, έχει χαμηλής ποιότητας προτάσεις, και είναι υπολογιστικά ακριβό [5].

Μια άλλη προσέγγιση στις συστάσεις είναι μέσω χρήσης βαθιάς μάθησης, όμως είναι δύσκολο να καταλάβουμε πως δουλεύουν αυτά τα μοντέλα και χρειάζονται πάρα πολλά δεδομένα και υπολογισμούς για να κάνουν καλές προβλέψεις.

4.2.1 Συστάσεις και Ενισχυτική Μάθηση

Σε αντίθεση με τις κλασικές μεθόδους, η ΕΜ μπορεί να διαχειριστεί ακολουθιακές και δυναμικές αλληλεπιδράσεις μεταξύ συστήματος και χρήστη και να λάβει υπ΄όψιν την μακροχρόνια αφοσίωση των χρηστών. Τρία βασικά χαρακτηριστικά που κάνουν την ΕΜ κατάλληλη για το πρόβλημα των συστάσεων είναι:

1. Η ΕΜ μπορεί να διαχειριστεί τις δυναμικές της ακολουθιακής αλληλεπίδρασης μεταξύ χρήστη και συστήματος προσαρμόζοντας τις πράξεις με βάσει την συνεχή ανατροφοδότηση που λαμβάνει από το περιβάλλον.

2. Η ΕΜ μπορεί να λάβει υπόψιν την μακροχρόνια διατήρηση του ενδιαφέροντος του χρήστη στο σύστημα

3. Παρόλο που η ύπαρξη βαθμολογιών από τους χρήστες είναι χρήσιμες, η ΕΜ δεν τις χρειάζεται. Βελτιστοποιεί την πολιτική της αλληλεπιδρώντας ακολουθιακά με το περιβάλλον.

Σύμφωνα με το [21], υπάρχουν 4 στοιχεία που πρέπει να σχεδιαστούν σωστά σε ένα σύστημα συστάσεων με ΕΜ. Αυτά είναι:

1. **Η αναπαράσταση των καταστάσεων.** Στην διεπαφή πράκτορα-περιβάλλοντος, η κατάσταση μπορεί να είναι οποιαδήποτε πληροφορία διαθέσιμη στον πράκτορα. Η αναπαράσταση θα μπορούσε να είναι τόσο υψηλού επιπέδου όσο συμβολικές περιγραφές αντικειμένων σε ένα δωμάτιο ή τόσο χαμηλού επιπέδου όσο μετρήσεις από αισθητήρες.

Αυτό που είναι σημαντικό είναι να ισχύει η Μαρκοβιανή ιδιότητα, το οποίο σημαίνει ότι το σήμα της κατάστασης δεν χρειάζεται να μεταφέρει όλες τις πληροφορίες σχετικά με το περιβάλλον στον πράκτορα, αλλά να συνοψίζει τις προηγούμενες πληροφορίες, ώστε να μην χαθούν αυτές που είναι σημαντικές. Ένα σήμα κατάστασης με αυτή την ιδιότητα ονομάζεται Μαρκοβιανό. Γενικά, η επιλογή της αναπαράστασης της κατάστασης είναι περισσότερο τέχνη παρά επιστήμη. Σε ένα σύστημα συστάσεων με ΕΜ, η αναπαράστασης της κατάστασης θα πρέπει να συνοψίζει πληροφορίες σχετικά με χρήστες, αντικείμενα και συγκείμενο. Πιο συγκεκριμένα η αναπαράσταση θα μπορούσε να χωριστεί σε τρεις κατηγορίες:

(α′) **Αντικείμενα ως καταστάσεις**. Όταν ο χώρος των αντικειμένων είναι μικρός, είναι δυνατό να θεωρήσουμε κάθε αντικείμενο σαν κατάσταση. Όμως αυτή η προσέγγιση δεν κλιμακώνει όταν το χώρος των αντικειμένων είναι μεγάλος. Για να αντιμετωπίσουμε το πρόβλημα κλιμάκωσης σε μεγαλύτερους χώρους αντικειμένων, οι ερευνητές βρήκαν ότι οι καταστάσεις μπορούν να υποδεικνύουν ένα σύνολο από αντικείμενα που καταναλώθηκαν/αξιολογήθηκαν νωρίτερα από τον χρήστη.

(β′) **Χαρακτηριστικά από χρήστες, αντικείμενα και συγκείμενο**. Μια αρκετά διαδεδομένη μέθοδος αναπαράστασης κατάστασης είναι η εξαγωγή χαρακτηριστικών για τους χρήστες, τα αντικείμενα και το συγκείμενο. χαρακτηριστικά των χρηστών μπορεί να είναι δημογραφικές πληροφορίες, όπως ηλικία και φύλο. Χαρακτηριστικά των αντικειμένων μπορεί να είναι η τιμή, η κατηγορία και η δημοτικότητα. Τέλος χαρακτηριστικά του συγκειμένου μπορεί να είναι ο χρόνος, η πλατφόρμα, και η τοποθεσία.

(γ′) **Κωδικοποιημένα embeddings**. Για αποτελεσματική μάθηση, τα βαθειά μοντέλα σε συστήματα συστάσεων βαθιάς ενισχυτικής μάθησης, χρειάζονται καταστάσεις οι οποίες είναι πυκνά, και χαμηλών-διαστάσεων διανύσματα. Συνήθως, αρχικά τα χαρακτηριστικά των χρηστών, των αντικείμενων, και του συγκειμένου, μεταφράζονται σε συνεχή διανύσματα, τα οποία είναι πυκνά και χαμηλών διαστάσεων, και ονομάζονται embeddings. Έπειτα για καλύτερη εκπαίδευση, αυτά τα embeddings, μπορούν να κωδικοποιηθούν χρησιμοποιώντας ένα μοντέλο RNN, το οποίο μπορεί να βοηθήσει το μοντέλο να μάθει τις ακολουθιακές προτιμήσεις του χρήστη [16]. Συνήθως προτιμούνται τα GRU σε σχέση με τα LSTM, καθώς έχουν λιγότερες παραμέτρους και μπορούν να επιτύχουν καλύτερη ή ίση απόδοση. Για να εστιάσουν στα σημαντικά κομμάτια της εισόδου, κάποιοι χρησιμοποιούν και ένα επίπεδο attention. Τέλος τα Κωδικοποιημένα διανύσματα συνενώνονται για να δημιουργήσουν την κατάσταση.

2. **Η βελτιστοποίηση της πολιτικής**. Αφού οριστούν οι καταστάσεις, η πολιτική επιλέγει ποια δράση να επιλέξει, σε κάθε κατάσταση. Για την βελτιστοποίηση της πολιτικής,διάφοροι αλγόριθμοι ΕΜ έχουν χρησιμοποιηθεί από συστήματα συστάσεων με ΕΜ. Πριν την ανάπτυξη της βαθιάς ΕΜ, οι μέθοδοι ΕΜ που χρησιμοποιούνταν από συστήματα συστάσεων με ΕΜ μπορούσαν να ταξινομηθούν σε πινακοειδής και προσεγγιστικές

μεθόδους. Οι πινακοειδής περιλαμβάνουν τεχνικές όπως επανάληψη πολιτικής(policy iteration), q-learning, Sarsa, Sarsa(λ), R-learning και MCTS.Οι προσεγγιστικές τεχνικές είναι οι fitted Q και οι επανάληψης της κλίσης της αξίας (gradient value iteration). Από την άλλη, οι μέθοδοι βαθιάς ΕΜ, χωρίζονται σε 3 κατηγορίες: βασισμένες στην αξία (DQN), κλίσης της πολιτικής (REINFORCE, REINFORCE-wb) και μεθόδους ηθοποιού-κριτικού ((actor-critic, DDPG, PPO)).

3. **Ο ορισμός των ανταμοιβών**.Το σήμα της ανταμοιβής ορίζει το πόσο καλά ή κακά τα πάει ο πράκτορας με βάση τις δράσεις που επιλέγει. Έτσι η σχεδίαση ενός σήματος ανταμοιβών που δίνει επαρκείς πληροφορίες, είναι απαραίτητο για την επιτυχία/εκπαίδευση του πράκτορα. Πράγματι, στην ΕΜ, το σήμα της ανταμοιβής λέει στον πράκτορα τί να κάνει, αλλά όχι πώς. Γενικά, η σχεδίαση ενός σωστού συστήματος ανταμοιβών είναι ένα δύσκολο πρόβλημα, που επιλύεται με την διαδικασία της δοκιμής-και-λάθους. Δεν υπάρχει κάποιος ξεκάθαρος κανόνας για το πώς να σχεδιαστεί μια καλή συνάρτηση ανταμοιβής για το συγκεκριμένο πρόβλημα. Στην βιβλιογραφία για συστήματα συστάσεων με ΕΜ, οι επιλογές είναι συνήθως δύο. Είτε η συνάρτηση ανταμοιβής είναι μια απλή αριθμητική τιμή, ή είναι μια ή πολλαπλές παρατηρήσεις από το περιβάλλον.

4. **Η κατασκευή του περιβάλλοντος**. Γενικά η αξιολόγηση συστημάτων συστάσεων είναι δύσκολη. Ως αποτέλεσμα, η δημιουργία ενός κατάλληλου περιβάλλοντος για να εκπαιδευτεί και να αξιολογηθεί ο πράκτορας στο σύστημα συστάσεων με ΕΜ είναι δύσκολη. Για να ξεχωρίσουμε καλύτερα μεταξύ διάφορων μεθόδων κατασκευής περιβάλλοντος, τις χωρίζουμε σε τρεις κατηγορίες: offline, προσομοιωση και online. Στην offline μέθοδο, το περιβάλλον είναι ένα στατικό σύνολο δεδομένων το οποίο περιέχει την αξιολόγηση κάποιων χρηστών σε κάποια αντικείμενα. Μια συνήθης πρακτική των offline μεθόδων είναι η εκπαίδευση του πράκτορα σε δεδομένα εκπαίδευσης (συνήθως το 70-80% των δεδομένων) και μετά η αξιολόγηση στα εναπομείναντα. Όσον αφορά την προσομοίωση, συνήθως χτίζεται κάποιο μοντέλο του χρήστη και ο αλγόριθμος αξιολογείται καθώς αλληλεπιδρά με το μοντέλο του χρήστη. Αυτός ο χρήστης μπορεί να έχει απλά μια προκαθορισμένη συμπεριφορά ή να είναι πιο περίπλοκος και να έχει δημιουργηθεί από τα δοθέντα δεδομένα. Στην online μέθοδο, ο αλγόριθμος αξιολογειται καθώς αλληλεπιδρά με πραγματικούς χρήστες σε πραγματικό χρόνο.

Η χρήση contextual bandits για συστήματα συστάσεων, απλοποιεί κάποια από τα παραπάνω προβλήματα.

Κεφάλαιο 5

Η συνεισφορά μας

Στο κεφάλαιο αυτό αναλύεται η δική μας συνεισφορά και τα πειράματα μας. Αρχικά, περιγράφονται αναλυτικά το διαλογικό σύστημα Rasa, καθώς και το σύστημα ενισχυτικής μάθησης Vowpal Wabbit. Έπειτα, περιγράφεται η διαδικασία που ακολουθήσαμε, καθώς και τα προβλήματα που ανέκυψαν κατά την διαδικασία. Τέλος περιγράφεται η πλήρης αρχιτεκτονική του συστήματος και η λειτουργία του.

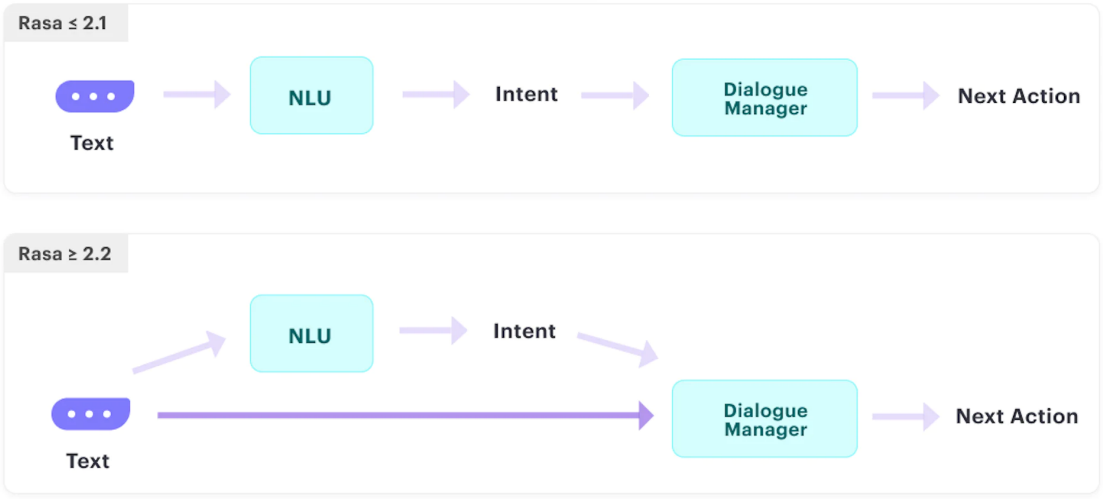
5.1 Το διαλογικό σύστημα Rasa

Το Rasa είναι ένα πλαίσιο (framework) ανοιχτού κώδικα για την ανάπτυξη και την έκδοση εφαρμογών συνομιλίας τεχνητής νοημοσύνης. Παρέχει ένα σύνολο εργαλείων και βιβλιοθηκών που επιτρέπουν στους προγραμματιστές να δημιουργούν διαλογικούς πράκτορες, εικονικούς βοηθούς και άλλους συνομιλητές. Το Rasa έχει σχεδιαστεί για να προσφέρει ευελιξία και έλεγχο της συμπεριφοράς και των δυνατοτήτων αυτών των συστημάτων τεχνητής νοημοσύνης.

Το πλαίσιο Rasa μπορεί νοητικά να χωριστεί σε δύο κομματια. Τα κομμάτια αυτά στο παρελθόν ήταν και ξεχωριστές υπηρεσίες, αλλά πλέον όλα βρίσκονται σε ένα πακέτο: το Rasa NLU (Natural Language Understanding) υπεύθυνο για την κατανόηση φυσικής γλώσσας και το Rasa Core.

Το Rasa NLU είναι υπεύθυνο για την κατανόηση των εισροών των χρηστών και την εξαγωγή σχετικών πληροφοριών από αυτές. Χρησιμοποιεί τεχνικές μηχανικής μάθησης για την επεξεργασία και την ταξινόμηση των μηνυμάτων των χρηστών, εξάγοντας οντότητες και προθέσεις.

Το Rasa Core χειρίζεται τη διαχείριση του διαλόγου και τη διαδικασία λήψης αποφάσεων. Χρησιμοποιεί ενισχυτική μάθηση για να εκπαιδεύσει μοντέλα που μπορούν να προβλέψουν την επόμενη καλύτερη ενέργεια με βάση την τρέχουσα κατάσταση της συνομιλίας. Το Rasa



Σχήμα 5.1: Η αρχιτεκτονική του συστήματος με και χωρίς προθέσεις

Core επιτρέπει στους προγραμματιστές να ορίζουν τις ροές συνομιλιών, να χειρίζονται τις απαντήσεις των χρηστών και να διαχειρίζονται το περιβάλλον και την κατάσταση.

Ένα από τα βασικά πλεονεκτήματα του Rasa είναι η ευελιξία και η δυνατότητα προσαρμογής του. Οι προγραμματιστές μπορούν να ορίσουν τα δικά τους μοντέλα γλώσσας για συγκεκριμένο τομέα, να τους εκπαιδεύσουν χρησιμοποιώντας τα δικά τους δεδομένα και να τα βελτιστοποιήσουν για να επιτύχουν καλύτερη απόδοση. Το Rasa υποστηρίζει επίσης την ενοποίηση με άλλες υπηρεσίες και πλατφόρμες, επιτρέποντας στους προγραμματιστές να συνδέουν τους διαλογικούς τους πράκτορες τους με διάφορα κανάλια, όπως ιστότοπους, εφαρμογές ανταλλαγής μηνυμάτων και φωνητικές διεπαφές.

Το Rasa βασίζεται στην ιδέα της ανάπτυξης βασισμένης στις συνομιλίες (Conversation Driver Development). Αυτό σημαίνει ότι η καλύτερη πηγή πληροφορίας για την βελτίωση του διαλογικού πράκτορα προέρχεται από τις πραγματικές συνομιλίες που δημιουργούνται από την επικοινωνία με χρήστες.

Αν και αρχικά το Rasa χρησιμοποιούσε σε μεγάλο βαθμό την ιδέα των προθέσεων για την δημιουργία των πρακτόρων, τον τελευταίο καιρό έχουν αρχίσει να δοκιμάζουν να χτίσουν εργαλεία, τα οποία θα επιτρέψουν στους πράκτορες να λειτουργήσουν χωρίς προθέσεις, δημιουργώντας ένα σύστημα άκρη-σε-άκρη (end-to-end), στο οποίο το σύστημα διαχείρισης του διαλόγου θα χρησιμοποιεί το κείμενο της απάντησης για να επιλέξει την επόμενη κατάσταση του διαλόγου, αντί να βασίζεται στην πρόθεση, όπως φαίνεται στο Σχήμα 5.1. Φυσικά, αυτό δεν σημαίνει ότι οι προθέσεις δεν έχουν χρήση, αλλά σε περιπτώσεις που το σύστημα δεν μπορεί με μεγάλη βεβαιότητα να προβλέψει την πρόθεση, η χρήση της πολιτικής χωρίς προθέσεις μπορεί να φέρει καλύτερα αποτελέσματα.

5.1.1 Rasa NLU

Το Rasa NLU είναι το κομμάτι της εφαρμογής υπευθυνο για την κατανόηση της φυσικής γλώσσας. Το Rasa χρησιμοποιεί μια τεχνική η οποία ονομάζεται DIET (Dual Intent and Entity Transformer), το οποίο είχε καλύτερα αποτελέσματα από fine-tuning ενός μοντέλου BERT, μετασχηματιστή που το 2020 θεωρούνταν τελευταίας τεχνολογίας. Το DIET έχει διττό ρόλο, καθώς διαχειρίζεται τόσο την αναγνώριση των προθέσεων, όσο και την εξαγωγή προθέσεων. Η αρχιτεκτονική του DIET φαίνεται στο Σχήμα 5.2.

Η δομή του μοντέλου είναι η εξής. Έστω ότι, για παράδειγμα, ο χρήστης λέει "play ping pong", οταν ο πράκτορας τον ρωτήσει τι θα ήθελε να κάνει. Η πρόταση θα σπάσει σε λέξεις (tokens), και κάθε μια θα περάσει μέσα από ένα κομμάτι του δικτύου με δύο διαδρομές. Η πρώτη είναι η προ-εκπαιδευμένη διαδρομή, κάτα την οποία η λέξη περνάει μέσα από ένα προ-εκπαιδευμένο δίκτυο και επιστρέφει μια διανυσματική αναπαράσταση της λέξης. Αυτό το δίκτυο μπορεί να είναι κάποιος μετασχηματιστής για παράδειγμα. Η άλλη διαδρομή μετατρέπει πρώτα την λέξη σε μια αραιή αναπαράσταση της με βάση τις λέξεις και τα n-grams που δημιουργούνται και μετά περνάει μέσα από ένα νευρωνικό δίκτυο, κάνοντας την πράξη $g(Wx + b)$, όπου x είναι η αραιή αναπαράσταση, W τα βάρη του δικτύου και b η κλίση (bias). Έπειτα τα δύο διανύσματα από τις δύο διαδρομές ενώνονται και περνάνε μέσα από ένα ακόμα νευρωνικό δίκτυο, το οποίο έχει ως έξοδο ένα διανυσμα 256 διαστάσεων. Τα νευρωνικά δίκτυα αυτά δεν είναι πλήρως συνδεδεμένα, αλλά έχουν αφαιρεθεί κάποιες ακμές με χρήση drop-out. Επιπλέον όλα τα νευρωνικά δίκτυα που βρίσκονται στο ίδιο επίπεδο, έχουν τα ίδια βάρη.

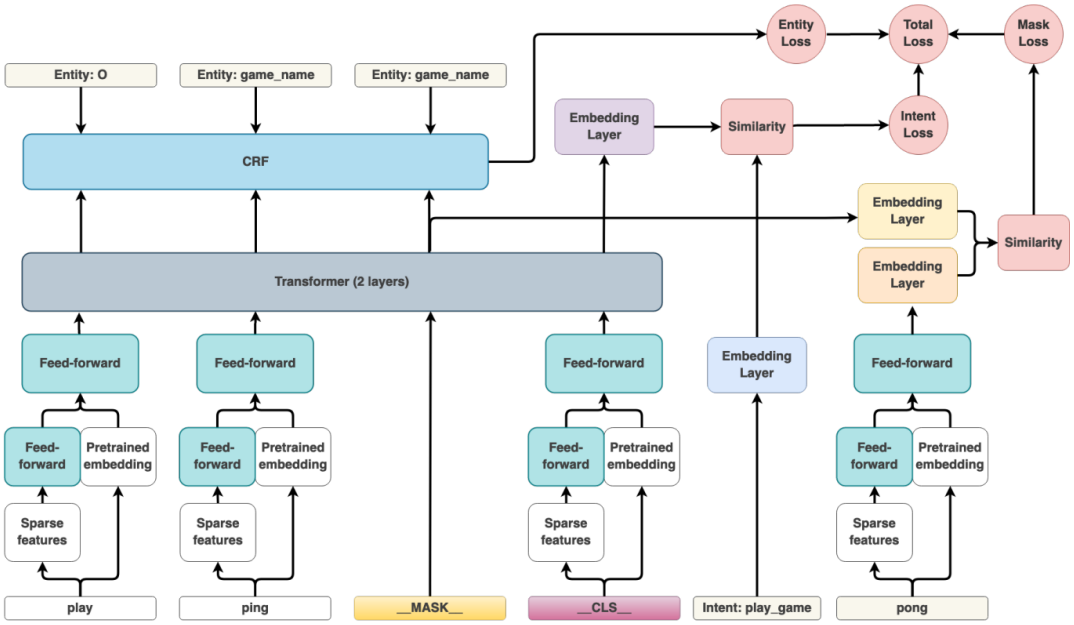
Πέρα από τις λέξεις, στο σύστημα εισέρχεται και το `__CLS__` token, το οποίο ουσιαστικά είναι η αναπαράσταση ολόκληρης της πρότασης. Αυτή η αναπαράσταση ουσιαστικά δημιουργείται είτε με την ένωση των διανυσμάτων των επιμέρους λέξεων, στο κομμάτι της δημιουργίας της αραιής αναπαράστασης είτε είναι η αναπαράσταση της πρότασης μέσα από το προ-εκπαιδευμένο δίκτυο.

Ακόμα, κατα την εκπαίδευση, μια από τις λέξεις συγκαλύπτεται με μια μάσκα, το `__MASK__`. Στο παράδειγμα του Σχήματος 5.2, η λέξη αυτή είναι το pong. Αυτή η λέξη μετά θα πρέπει να προβλεφθεί από τον μετασχηματιστή. Αυτη η πρόβλεψη θα περάσει μετά από ενα δίκτυο διανυσματικής αναπαράστασης και θα συγκριθεί με την πραγματική λέξη. Αυτό μας δείχνει πόσο καλά το μοντέλο μας μπορεί να γενικεύσει την γλώσσα.

Ακόμα, κατά την εκπαίδευση, μια παρόμοια διαδικασία συμβαίνει μεταξύ του `__CLS__` token και της (γνωστής) πρόθεσης του χρήστη, καθώς στηριζόμαστε στην ιδέα ότι το token, αφού αναπαριστά ολόκληρη την πρόταση, φέρει πληροφορία σχετικά με την πρόθεση.

Τέλος, οι λέξεις αφού περάσουν μέσα από τον μετασχηματιστή, συγκρίνονται με τις πραγματικές οντότητες που αναπαριστούν μέσα σε ένα υπό συνθήκη τυχαίο πεδίο (Conditional Random Field) και υπολογίζεται μια απώλεια.

Έτσι το σύστημα ολόκληρο εκπαιδεύεται με βάση την συνολική απώλεια που προκύπτει από το σφάλμα στην αναγνώριση οντοτήτων, στην πρόβλεψη της λέξης μέσα στην πρόταση



Σχήμα 5.2: Η αρχιτεκτονική του συστήματος DIET[19]

και την κατανοηση της πρόθεσης του χρήστη.

5.1.2 Rasa Core

Στο κομμάτι αυτό γίνεται η διαχείριση του διαλόγου και η επιλογή της επόμενης κίνησης. Ακόμα γίνεται η παραγωγή του διαλόγου. Θα εξηγήσουμε περιληπτικά το πώς λειτουργούν τα μέρη αυτά.

Το Rasa χρησιμοποιεί ένα σύνολο από πολιτικές για την διαχείριση του διαλόγου, και ανάλογα με την προτεραιότητα τους και την εμπιστοσύνη που έχουν στην πρόβλεψη τους, επιλέγει ποια θα χρησιμοποιήσει.

Μια από τις πιο σημαντικές είναι η πολιτική TED (Transformer Embedding Dialogue).Η πολιτική TED χρησιμοποιείται για την επιλογή της επόμενης ενέργειας και την αναγνώριση οντοτήτων. Για την επιλογή της ενέργειας, η πολιτική ενώνει σε ένα διάνυσμα την αναπαράσταση της προηγούμενης πράξης, της πρόθεσης και πληροφοριών που είναι απαραίτητο να είναι διαθέσιμες μακροχρόνια (όπως πληροφορίες από κάποια φόρμα). Έπειτα, αυτές περνάνε μέσα από ένα μετασχηματιστή, παρέα με τις αναπαραστάσεις των προηγούμενων καταστάσεων. Η έξοδος του μετασχηματιστή περνάει μέσα από ένα νευρωνικό δίκτυο όπου γίνεται η πρόβλεψη της πρότασης που θα κάνει το σύστημα, μαζί με μια βεβαιότητα στην πρόταση αυτή.

Με αυτό τον τρόπο, το σύστημα μπορεί να διαχειριστεί την επικοινωνία με τον χρήστη όταν αυτός ξεφεύγει από το χαρούμενο μονοπάτι και μπορεί να επιστρέψει στην αρχική συζήτηση.

Για την εκπαίδευση αυτού του συστήματος, χρησιμοποιούνται δεδομένα, τα οποία λέγονται ιστορίες (stories). Οι ιστορίες αυτές περιγράφουν τα διαλογικά μονοπάτια των χρηστών, και είναι η βάση της ανάπτυξης με βάση την συνομιλία. Καθώς οι χρήστες χρησιμοποιούν τον διαλογικό πράκτορα και κάνουν συζητήσεις μαζί του, οι προγραμματιστές μπορούν να αναγνωρίσουν ποια ειναι τα πιο συχνά καλά ή κακά μονοπάτια που παίρνουν και να τα καταγράψουν σαν ιστορίες για να εκπαιδευτεί ο πράκτορας να τα διαχειρίζεται.

Οσον αφορά το κομμάτι της παραγωγής διαλόγου, αυτό βασίζεται πάνω στην ιδέα των προτύπων (templates). Ο προγραμματιστής για κάθε δράση προσθέτει πρότυπα απαντήσεων μαζί με θέσεις όπου μπορούν να μπουν συγκεκριμένες πληροφορίες οι οποίες προέρχονται από τον χρήστη.

5.1.3 Rasa Action Server

Πέρα από τα παραπάνω, ένα σημαντικό κομμάτι για την εργασία μας είναι οι προσαρμοσμένες ενέργειες (custom actions), οι οποίες υλοποιούνται στον Rasa Action Server. Η υπηρεσία αυτή επιτρέπει στους προγραμματιστές να γράψουν ενέργειες οι οποίες είναι προσαρμοσμένες στο τί θέλουμε να κάνουμε. Το Rasa στέλνει στον Action server ενα αίτημα με πληροφορίες όπως την δράση που προβλέφθηκε, το αναγνωριστικό της συζήτησης, την καταγραφή ολόκληρης της συζήτησης και τα περιεχόμενα του περιβάλλοντος του πράκτορα. Με βάση αυτά, ο action server επιστρέφει απαντήσεις και γεγονότα τα οποία θα χρησιμοποιήσει μετά το Rasa για αν απαντήσει στον χρήστη.

5.2 Vowpal Wabbit

Το Vowpal Wabbit[6] είναι μια βιβλιοθήκη μηχανικής μάθησης ανοιχτού κώδικα, η οποία αναπτύχθηκε αρχικά στα εργαστήρια της Yahoo, και πλέον αναπτύσσεται από την ερευνητική ομάδα της Microsoft. Το όνομα 'Vowpal Wabbit' είναι παιχνιδισμός στην φράση 'vorpal wabbit' από το ποίημα του Λιούις Κάρολ 'Jabberwocky¨.

Το Vowpal Wabbit είναι διάσημο για την αποδοτικότητα του και την επεκτασιμότητα του, αφού είναι ικανό να διαχειριστεί δεδομένα πολλών terrabyte και να εκτελέσει σύγχρονη εκπαίδευση πολύ γρήγορα. Χρησιμοποιεί έναν σύγχρονο αλγόριθμο κατάβασης κλίσης για να ανανεώσει τα μοντέλα σταδιακά, καθώς έρχονται νέα δεδομένα. Αυτό του επιτρέπει την αποδοτική επεξεργασία ροών δεδομένων ή δεδομένων με μεγάλο αριθμό παραδειγμάτων.

Επιπλέον, το Vowpal Wabbit είναι διάσημο για τα εργαλεία διαδραστικής μάθησης που προσφέρει, όπως είναι το Contextual Bandits αλλά και άλλες μορφές ενισχυτικής μάθησης. Τέλος, προσφέρει εργαλεία τόσο για σύγχρονη όσο και για ασύγχρονη εκπαίδευση των μοντέλων.

Τα κύρια χαρακτηριστικά του Vowpal Wabbit που το κάνουν ένα εξαιρετικά δυνατό εργαλείο είναι:

1. **Μορφή εισαγωγής των δεδομένων**. Η μορφή εισόδου για τον αλγόριθμο εκμάθησης είναι ουσιαστικά πιο ευέλικτη από ό,τι θα περίμενε κανείς. Τα παραδείγματα μπορεί να έχουν χαρακτηριστικά που αποτελούνται από κείμενο ελεύθερης μορφής, το οποίο ερμηνεύεται με έναν τρόπο bag-of-words. Μπορεί ακόμη να υπάρχουν πολλά σύνολα κειμένου ελεύθερης μορφής σε διαφορετικούς χώρους ονομάτων.

2. **Ταχύτητα** Ο αλγόριθμος εκμάθησης είναι αρκετά γρήγορος — παρόμοιος με τις λίγες άλλες βιβλιοθήκες σύγχρονων αλγορίθμων μάθησης εκεί έξω. Για παράδειγμα, μπορεί να εφαρμοστεί αποτελεσματικά σε μαθησιακά προβλήματα με αραιά σύνολα δεδομένων μεγέθους terrabyte (για παράδειγμα 1012 αραιά χαρακτηριστικά). Ως άλλο παράδειγμα, είναι περίπου 3 φορές πιο γρήγορο από το svmsgd του Leon Bottou στο παράδειγμα RCV1 στο συνολικό χρόνο εκτέλεσης.

3. **Επεκτασιμότητα** Η επεκτασιμότητα είναι διαφορετική από την ταχύτητα. Το σημαντικό χαρακτηριστικό εδώ είναι ότι το αποτύπωμα μνήμης του προγράμματος είναι ανεξάρτητο από το πλήθος των δεδομένων. Αυτό σημαίνει ότι το σετ εκπαίδευσης δεν φορτώνεται στην κύρια μνήμη πριν ξεκινήσει η εκπαίδευση. Επιπλέον, το μέγεθος του συνόλου των χαρακτηριστικών περιορίζεται ανεξάρτητα από τον όγκο των δεδομένων εκπαίδευσης χρησιμοποιώντας τεχνικές κατακερματισμού.

4. **Σύζευξη χαρακτηριστικών (feature pairing)** Τα υποσύνολα χαρακτηριστικών μπορούν να αντιστοιχιστούν εσωτερικά έτσι ώστε ο αλγόριθμος να είναι γραμμικός στο εξωτερικό γινόμενο των υποσυνόλων. Αυτό είναι χρήσιμο για προβλήματα κατάταξης. Ο David Grangier φαίνεται να έχει ένα παρόμοιο κόλπο στον κώδικα PAMIR. Η εναλλακτική λύση της ρητής επέκτασης των χαρακτηριστικών πριν από την τροφοδοσία τους στον αλγόριθμο εκπαίδευσης μπορεί να είναι τόσο εντατική σε υπολογισμούς όσο και σε χώρο, ανάλογα με τον τρόπο χειρισμού της.

5.3 Προϋπάρχουσα αρχιτεκτονική - Θεανώ

Το σύστημα πάνω στο οποίο η εργασία βασίστηκε ονομάζεται Θεανώ. Η Θεανώ είναι η διαλογική βοηθός του Ερευνητικού Κέντρου "Αθηνά", με σκοπό να δώσει έγκυρες πληροφορίες για τον κορωνοϊό. Η υλοποίηση έγινε από τους ερευνητές του Ινστιτούτου Επεξεργασίας του Λόγου.

Η Θεανώ έχει στόχο να κρατάει ενήμερους τους πολίτες για την εξέλιξη της πανδημίας, ώστε να μπορούν να παίρνουν αποφάσεις με βάση τις οδηγίες των ειδικών, καθώς και να μειώνει τον πανικό, προσφέροντας τη δυνατότητα αυτοαξιολόγησης των συμπτωμάτων.

Παρέχει πληροφορίες για τα νέα κρούσματα και τους θανάτους στην Ελλάδα, σε χώρες του εξωτερικού και συνολικά στον κόσμο. Γνωρίζει για την πληρότητα σε ΜΕΘ και για τον αριθμό των ατόμων που έχουν εμβολιαστεί σε μια συγκεκριμένη ημερομηνία ή συνολικά στην Ελλάδα. Καλύπτει πληθώρα συχνών ερωτήσεων, όπως από που ξεκίνησε ο κορωνοϊός, πώς μεταδίδεται, ποια είναι η σωστή χρήση της μάσκας κ.λπ.

Επιπλέον, η Θεανώ βρίσκει φαρμακεία σχεδόν σε όλες τις πόλεις-περιοχές της Ελλάδας, ενώ με 6 απλές ερωτήσεις, σαν ένα μικρό διαγνωστικό τεστ, βοηθά τους χρήστες που έχουν συμπτώματα να αναγνωρίσουν αν υπάρχει πιθανότητα να είναι φορείς ή όχι.

Η Θεανώ είναι χτισμένη πάνω στο Rasa. Συγκεκριμένα, χρησιμοποιεί το DIET για την αναγνώριση προθέσεων, και την πολιτική TED για την διαχείριση του διαλόγου. Επιπλέον, χρησιμοποιεί το Duckling, γραμμένο από την Facebook για την εξαγωγή οντοτήτων. Ακόμα,για να να μπορέσει η Θεανώ να διαχειριστεί τα greeklish, στην αρχή της επεξεργασίας της πρότασης του χρήστη, χρησιμοποιεί ένα μεταφραστή από greeklish σε Ελληνικά, γραμμένο στο Αθηνά.

Όσον αφορά τις δράσεις, για κάθε πρόθεση του χρήστη, υπάρχει η αντίστοιχη λειτουργικότητα στον Action Server. Ακόμα, λόγω του γεγονότος ότι η Θεανώ λειτουργούσε κάποιο καιρό πριν την εργασία μας, υπάρχουν ήδη ιστορίες που διαχειρίζονται τις περισσότερες καταστάσεις.

Όσον αφορά το κομμάτι που εργαστήκαμε εμείς, πριν την έναρξη της εργασίας, υπήρχε ήδη μια δράση, η οποία εμφανιζόταν μετά από τις απαντήσεις της Θεανώς η οποία καλούνταν με κάποια μη-μηδενική πιθανότητα και τυχαία πρότεινε κάποια από τις υπόλοιπες λειτουργίες της που δεν είχαν συζητηθεί με τον χρήστη ακόμα. Ο στόχος ήταν η διατήρηση του ενδιαφέροντος του χρήστη για μεγαλύτερο χρονικό διάστημα δίνοντας του περισσότερα θέματα για συζήτηση.

5.4 Στόχοι, περιορισμοί και παραδοχές

Στόχος της εργασίας ήταν η δημιουργία και ενσωμάτωση ενός συστήματος ενισχυτικής μάθησης στην Θεανώ.Το σύστημα αυτό θα βοηθήσει την Θεανώ να κάνει συστάσεις στον χρήστη σχετικά με θέματα που είναι σχετικά με τα ως τώρα ενδιαφέροντα του και την συνομιλία που έχει κάνει με την Θεανώ. Κάποιοι βασικοί περιορισμοί που προέρχονται από την εως τώρα λειτουργία του συστήματος μας και οι τρόποι που μας επηρεάζουν είναι οι παρακάτω:

- **Περιορισμένες πληροφορίες χρηστών**: Καθώς το σύστημα δεν κρατάει κάποιο ιστορικό συνομιλιών, και δεν έχει προηγούμενη γνώση των χρηστών που αλληλεπιδρούν με αυτό, οι πληροφορίες που έχουμε για αυτούς προέρχονται μόνο μέσα από τις ερωτήσεις που κάνουν οι ίδιοι κατά την διάρκεια των συνομιλιών. Αυτό σημαίνει πώς σε κάθε συνομιλία, ο πράκτορας ξεκινάει με μηδενική γνώση του συνομιλητή και θα πρέπει μέσα από τις περιορισμένες συναναστροφές του να εξάγει πληροφορίες σχετικά με τα θέματα που μπορεί να είναι ενδιαφέροντα σε αυτόν.
- **Περιορισμένα ιστορικά δεδομένα**: Για να μπορέσει το σύστημα να κάνει προτάσεις, θα πρέπει να έχει εκπαιδευτεί στο τί οι χρήστες συσχετίζουν κατα την διάρκεια μιας συνομιλίας. Για να το κάνουμε αυτό εκπαιδεύσαμε το μοντέλο βασιζόμενοι στην τυχαία πολιτική την οποία είχε εως τώρα η Θεανώ. Η τεχνική αυτή ασύγχρονης εκπαίδευσης έχει τον περιορισμό ότι βασίζεται σε περιορισμένα δεδομένα. Τα ιστορικά

Συνολικές συνομιλίες	468
Κενές συνομιλίες	151
Συνομιλίες με σύσταση	261
Συνολικές συστάσεις	1043
Μέσο πλήθος μηνυμάτων χρήστη	5.68
Μέσο πλήθος μηνυμάτων Θεανώς	9.25
Μέσο πλήθος συστάσεων ανα συνομιλία	2.23
Επιτυχία τυχαίας πολιτικής	45.93%

Πίνακας 5.1: Στατιστικά από την ανάλυση των ιστορικών δεδομένων

δεδομένα στην κατοχή μας από την χρήση της Θεανώς είναι 468 συνομιλίες, με συνολικά 1043 τυχαίες συστάσεις, τις οποίες μπορούμε να χρησιμοποιήσουμε.

Με βάση τα παραπάνω, πήραμε τις επόμενες σχεδιαστικές αποφάσεις:

- Αν θεωρήσουμε κάθε συνομιλία ως μοναδική και προσπαθήσουμε να λύσουμε το πλήρες πρόβλημα της ενισχυτικής μάθησης, θα είχαμε μόλις 468 δείγματα, από τα οποία μόλις 261 περιέχουν οποιαδήποτε σύσταση από την Θεανώ. Για να αυξήσουμε την ποσότητα των δεδομένων που έχουμε, καθώς και για να απλοποιήσουμε το πρόβλημα, κάνουμε την παραδοχή ότι η σειρά που γίνονται οι προτάσεις, καθώς και τα ερωτήματα του χρήστη σχετικά με πληροφορίες δεν επηρεάζουν το τι θα ενδιαφέρει τον χρήστη. Έτσι κάθε πρόταση μπορεί να γίνει μόνο με βάση τα θέματα που γνωρίζει η Θεανώ ότι έχουν συζητηθεί και ανεξάρτητα από την σειρά που συζητήθηκαν. Έτσι μπορούμε πρακτικά να αντιμετωπίσουμε το πρόβλημα σαν ένα πρόβλημα contextual bandits.
- Η προσαρμοσμένη ενέργεια του Action Server, η οποία είναι υπεύθυνη για την πρόταση επόμενου θέματος καλείται σε συγκεκριμένες περιπτώσεις και όχι μετά από κάθε απάντηση του χρήστη. Αυτό κάνει δύσκολη την εύρεση της ανταμοιβής μέσα στο ιστορικό της συνομιλίας. Οπότε ένα πρόβλημα που έπρεπε να λύσουμε είναι το πώς θα γίνει η καταγραφή της ανταμοιβής με τρόπο εύρωστο, ώστε να μπορεί να καταγράφεται ανεξάρτητα της απάντησης και της αντίδρασης του χρήστη.

5.5 Ανάλυση δεδομένων & αναπαράσταση τους

Το πρώτο βήμα για την δημιουργία του μοντέλου μας ήταν η κατανόηση των ιστορικών δεδομένων στην κατοχή μας και μετέπειτα η εκπαίδευση του μοντέλου με βάση αυτό.

Τα ιστορικά δεδομένα που έχουμε στην κατοχή μας προέρχονται από την αλληλεπίδραση της Θεανώς με το ευρύ κοινό την περίοδο 4 Οκτωβρίου 2021 μέχρι 11 Ιανουαρίου 2022. Τα δεδομένα αυτά είναι σε μορφή JSON, όπως προέρχονται από την καταγραφή του διαλόγου του Rasa. Από αυτές τις συζητήσεις, 151 δεν έχουν κανένα γύρο μεταξύ χρήστη και της Θεανώς. Κάποια στατιστικά από τα δεδομένα φαίνονται στο Πίνακα 5.1.

Όπως φαίνεται, από τα αρχικά δεδομένα θα πρέπει να γίνει κάποιος διαχωρισμός ώστε να διατηρηθούν μόνο αυτά που είναι πράγματι χρήσιμα σε εμας, δηλαδή οι 1043 συστάσεις. Για να φτιάξουμε τα δεδομένα σε μορφή που να είναι κατάλληλη για το Vowpal Wabbit, θα πρέπει να εξάγουμε πληροφορίες για την κάθε σύσταση. Για αυτές συλλέξαμε τις προηγούμενες προθέσεις του χρήστη και τις προηγούμενες συστάσεις της Θεανώς και τις ενώσαμε για να δημιουργήσουμε το συγκείμενο (context) για κάθε σύσταση.

Οι πληροφορίες που περιέχει το συγκείμενο παίζουν αρκετά μεγάλο ρόλο στις συστάσεις στην περίπτωση μας, καθώς το συγκείμενο περιγράφει ουσιαστικά τα χαρακτηριστικά που λαμβάνει υπόψιν του το Vowpal Wabbit για να επιλέξει την σύσταση. Μια επιλογή που δοκιμάσαμε ήταν η ύπαρξη ως συγκείμενο μόνο των θεμάτων που μπορεί να προτείνει σαν σύσταση η Θεανώ. Αυτή όμως η προσέγγιση δημιουργεί ένα πολύ περιορισμένο συγκείμενο, και πολλές φορές δεν υπήρχε συγκείμενο στα παραδείγματα, ειδικά στην αρχή του διαλόγου. Μια άλλη προσέγγιση θα ήταν η χρήση όλων των προθέσεων σαν συγκείμενο. Αυτό όμως δημιουργεί ένα πολύ πιο σύνθετο συγκείμενο, για το οποίο τελικά υπάρχουν πολύ λίγα παραδείγματα, ενώ αρκετές προθέσεις δεν έχουν πραγματική αξία, όταν σκεφτόμαστε τον διάλογο (πχ η πρόθεση 'affirmative¨). Οπότε τελικά καταλήξαμε σε ένα συμβιβασμό μεταξύ των δύο αυτών προσεγγίσεων, κατά τον οποίο επιλέξαμε ένα υποσύνολο των προθέσεων, τα οποία δεν θεωρούμε σημαντικά και δεν έχουμε λόγο να χρησιμοποιήσουμε ως συγκείμενο, ενώ όλες οι υπόλοιπες προθέσεις προστιθενται σε αυτό.

Για να δημιουργηθεί το συγκείμενο, αφαιρούμε κάποιες από τις προθέσεις του χρήστη οι οποίες δεν προσφέρουν πληροφορία για τα ενδιαφέροντα του. Κάποιες από αυτές τις προθέσεις που αφαιρούνται είναι η θετική και η αρνητική απάντηση, οι ερωτήσεις που είναι ουσιαστικά ψιλοκουβέντα (chit-chat) - ερωτήσεις όπως τί είναι το Αθηνά, ποιο είναι το όνομα σου κλπ -, καθώς και ερωτήσεις που έχουν σχέση με την συμπλήρωση κάποιας μακροχρόνιας πληροφορίας. Αυτό θεωρήσαμε ότι θα μας βοηθήσει, καθώς το συγκείμενο είναι αρκετά μικρότερο έτσι και μας επιτρέπει να εστιάσουμε μόνο στις πράξεις που έχουν αξία για τον χρήστη.

Επιπλέον, για κάθε σύσταση θα πρέπει να βρούμε την ανταμοιβή και να την προσδιορίσουμε αριθμητικά. Για την περίπτωση μας, επιλέξαμε μια δυαδική αξιολόγηση. Ο τρόπος που προσδιορίσαμε αν μια σύσταση ήταν επιτυχημένη ή όχι ήταν ο εξής. Θεωρούμε ότι η απάντηση του χρήστη είναι αρνητική, αν:

- Ο χρήστης κλείσει την συνομιλία, αρα δεν υπάρχουν περαιτέρω απαντήσεις του χρήστη.
- Δεν υπάρχει απάντηση από τον χρήστη για κάποιο λόγο.
- Αν η απάντηση του χρήστη στην σύσταση είναι αρνητική ή δεν είναι σχετική.
- Αν ο χρήστης ζητήσει να συζητηθεί κάποιο άλλο θέμα.

Στις υπόλοιπες περιπτώσεις θεωρούμε ότι η απάντηση του χρήστη ήταν θετική. Ενδεικτικά, τα δεδομένα από μια συζήτηση μπορούν να βρεθούν στον Πίνακα 5.2. Το 'no_action' ουσιαστικά υποδηλώνει την γενική ερώτηση ¨Θα ήθελες να μάθεις για κάποιο άλλο θέμα;¨.

Α/Α	Προηγούμενες προθέσεις χρήστη	Προηγούμενες συστάσεις	Συγκείμενο	Τρέχουσα σύσταση	Αποτέλεσμα
1	affirmative, covid_stats		covid_stats	no_action	ΔΕΚΤΗ
2	affirmative, covid_stats, vaccines	no_action	covid_stats, vaccines	no_action	ΔΕΚΤΗ
3	affirmative, covid_stats, vaccines, covid_stats, greetings, ask_chitchat	no_action	covid_stats, vaccines	no_action	ΜΗ ΔΕΚΤΗ

Πίνακας 5.2: Παράδειγμα συζήτησης

5.6 Εκπαίδευση του μοντέλου

Για την εκπαίδευση του μοντέλου χρησιμοποιήσαμε την ασύγχρονη αξιολόγηση πολιτικής offline policy evaluation (OPE) του Vowpal Wabbit.

Στόχος της τεχνικής αυτής, σε ένα περιβάλλον contextual bandits είναι η χρήση δεδομένων, τα οποία έχουν δημιουργηθεί από μια άλλη πολιτική (ας την ονομάσουμε την πολιτική που χρησιμοποιείται στην παραγωγή, στην περίπτωση μας είναι η τυχαία πολιτική) με στόχο την εκτίμηση την αξίας μιας νέας υποψήφιας πολιτικής ασύγχρονα, χωρίς την χρήση της στην παραγωγή. Η χρησιμότητα είναι ότι έτσι μπορούμε να εκτιμήσουμε την απόδοση μιας πολιτικής, χωρίς να την εκθέσουμε στο περιβάλλον παραγωγής, και να την συγκρίνουμε με την υπάρχουσα.

Όσον αφορά την αναπαράσταση των δεδομένων, το Vowpal Wabbit, έχει το δικό του τρόπο να τα αναπαριστά τα παραδείγματα καταστάσεων. Συγκεκριμένα ο τρόπος αυτός είναι ο παρακάτω:

```
shared |test_types find_new_icus covid_stats
2:1.0:0.1 |ill_questionnaire_form
|vaccine_stats_form
|pharmacy_form
|ways_of_protection
|symptoms
|test_cost
|what_to_do_if_positive
|underlying_conditions
```

```
|no_action
```

Listing 5.1: Μορφή εισόδου Vowpal Wabbit

οπου, με την γραμμή shared δηλώνουμε ποιο είναι το συγκείμενο για την συγκεκριμένη κατάσταση, ενώ όλες οι επόμενες γραμμές δηλώνουν τις δράσεις που μπορεί να πάρει η πολιτική. Η πρώτη γραμμή σε αυτή την περίπτωση μας δείχνει την επιλογή που έκανε η τυχαία πολιτική. Συγκεκριμένα υπάρχουν 3 πεδία πριν την δράση, τα οποία χωρίζονται με άνω κάτω τελεία. Το 2 δηλώνει τον αριθμό της δράσης, και δεν επηρεάζει τον αλγόριθμο που χρησιμοποιούμε. Το δεύτερο πεδίο (1.0) δηλώνει την ανταμοιβή της δράσης που επιλέχθηκε. Αρα στο παράδειγμα, ο χρήστης αποδέχτηκε την σύσταση. Το τελευταίο πεδίο υποδηλώνει την πιθανότητα να επιλεχθεί η συγκεκριμένη δράση από την τυχαία πολιτική. Κάθε παράδειγμα τελειώνει με μια νέα γραμμή.

5.6.1 Ασύγχρονη εκπαίδευση

Σε περιβάλλοντα επιβλεπουσης μάθησης, η κλασική προσέγγιση στην ασύγχρονη εκπαίδευση είναι η εκπαίδευση στο σύνολο δεδομένων για εκπαίδευση και η εκτίμηση της γενικευμένης απόδοσης σε ένα σύνολο αξιολόγησης. Σε περιβάλλοντα σύγχρονης εκπαίδευσης, συνήθως χρησιμοποιείται η προοδευτική επαλήθευση. Σε περιβάλλοντα contextual bandits κανένα από τα δύο δεν είναι δυνατα, καθώς όπως σε όλα τα προβλήματα ενισχυτικής μάθησης, το πρόβλημα είναι ένα προβλημα μερικών πληροφοριών: δεν μαθαίνουμε ποτέ τις ανταμοιβές για τις επιλογές που δεν έγιναν. Η μόνη πηγή πληροφορίας είναι τα δεδομένα που δημιουργούνται από την πολιτική στην παραγωγή, που μπορεί να κάνει πολύ διαφορετικές επιλογές από την υποψήφια πολιτική.

Για να γίνει εκτίμηση πολιτικών contextual bandits ασύγχρονα, είναι η χρήση εκτιμητών, οι οποίοι δημιουργούν ψεύτικες ανταμοιβές για τις πράξεις που δεν επιλέχθηκαν, δημιουργώντας έτσι ένα ΄ψεύτικο΄ σύνολο δεδομένων επιβλέπουσας μάθησης, με βάση το οποίο μπορεί να εκτιμηθεί η απόδοση, χρησιμοποιώντας είτε προοδευτική επαλήθευση ή ένα σύνολο αξιολόγησης.

Το Vowpal Wabbit έχει υλοποιημένους αρκετούς εκτιμητές για να μετατρέψει την αξιολόγηση πολιτικής σε αξιολόγηση επιβλεπόμενης μάθησης. Η πιο απλή μέθοδος, είναι η απευθείας μέθοδος (direct method), η οποία εκπαιδεύσει ένα μοντέλο παλινδρόμησης το οποίο εκτιμάει το κόστος (αρνητική ανταμοιβή) ενός ζεύγους (δράση, συγκείμενο). Όπως είναι εμφανές, αυτή η μέθοδος είναι γενικά προκατειλημμένη, καθώς λόγω του προβλήματος μερικών πληροφοριών, βλέπουμε συνήθως περισσότερες ανταμοιβές για καλές δράσεις, παρά για κακές (αν η πολιτική στην παραγωγή είναι καλή). Στην περίπτωση μας η τυχαία πολιτική έχει 45.9% επιτυχία, οπότε έχουμε μια καλή ισορροπία μεταξύ καλών και κακών αποφάσεων. Οι προκατειλημμένοι εκτιμητές δεν μπορούν να χρησιμοποιηθούν για ασύγχρονη αξιολόγηση πολιτικής, αλλά το Vowpal Wabbit προσφέρει επαληθεύσιμα μη-προκατειλημμένους εκτιμητές, όπως είναι η στάθμιση αντίστροφης τάσης (inverse propensity weighting) και διπλά εύρωστος

εκτιμητής (doubly robust estimator), οι οποίοι μπορούν να χρησιμοποιηθούν για αυτό τον λόγο.

Για την πολιτική που θα ακολουθεί το σύστημα, επιλέξαμε την `cb_explore_adf`, η οποία είναι η πολιτική στην οποία το σύνολο των διαθέσιμων δράσεων αλλάζει ανάλογα με τον χρόνο ή όταν υπάρχουν επιπλέον πληροφορίες για κάθε δράση. Η πολιτική αυτή, εφαρμόζει κάποια τεχνική εξερεύνησης, την οποία μπορούμε να ρυθμίσουμε εμεις.

Για την ασύγχρονη εκπαίδευση της πολιτικής, έγινε σταδιακή εκπαίδευση της πάνω στα ιστορικά δεδομένα, ώστε να προσομοιώνει την λειτουργία της πολιτικής σε πραγματικά σενάρια. Σε αυτή την περίπτωση κάθε παράδειγμα εκπαίδευσης χρησιμοποιείται μόνο μια φορά.

Το ερώτημα που θέλουμε να απαντήσουμε εδώ για την υποψήφια πολιτική είναι το πόσο καλά θα γενικεύσει σε νέα παραδείγματα που θα έρχονται κατά την λειτουργία της στην παραγωγή, δεδομένου ότι η πολιτική αυτή εξελίσσεται συνεχώς. Μπορούμε λοιπόν να σκεφτούμε την πολιτική σαν ένα σύνολο υπερπαραμέτρων, στο οποίο ο στόχος είναι να βρούμε το σύνολο το οποίο μαθαίνει καλύτερα σε ένα σύγχρονο περιβάλλον εκπαίδευσης.

Για να απαντήσουμε αυτό το ερώτημα χρησιμοποιούμε προοδευτική επαλήθευση (progressive validation), η οποία είναι υλοποιημένη στο Vowpal Wabbit.

Έτσι, εκπαιδεύσαμε την πολιτική σε όλα τα ιστορικά δεδομένα που είχαμε, και η απώλεια (loss) που έχει η πολιτική μας δίνει μια εικόνα του πόσο καλά γενικεύει.

Επιπλέον, δοκιμάσαμε και τους δύο μη-προκατειλημμένους εκτιμητές. Τα αποτελέσματα που πήραμε όσον αφορά την απώλεια που έχει η πολιτική μας, είναι ίδια και για τους δύο. Ακόμα, δοκιμάζοντας διάφορες τεχνικές εξερεύνησης παρατηρούμε ότι σχεδόν όλες προσφέρουν την ελάχιστη απώλεια που μπορούμε να πετύχουμε στο σύνολο αξιολόγησης.

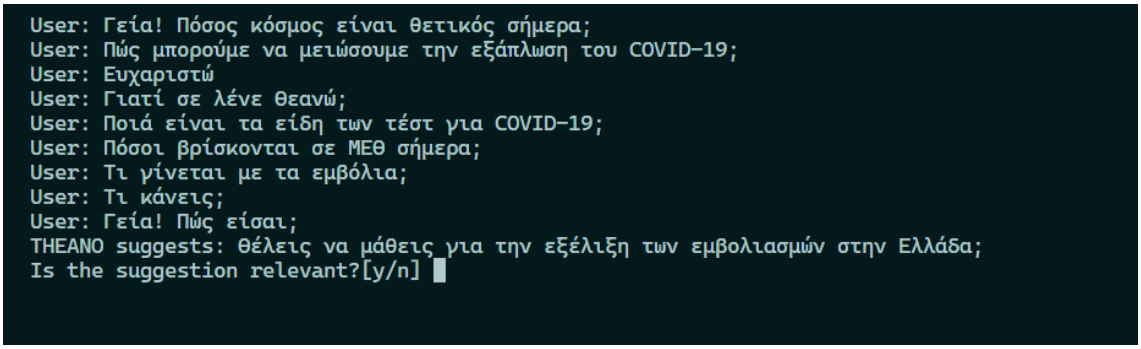
Συγκεκριμένα, τα αποτελέσματα που πετύχαμε στην ασύγχρονη εκπαίδευση φαίνονται στον Πίνακα 5.3. Όπως βλέπουμε η πολιτική που εκπαιδεύουμε έχει μεγαλύτερη απώλεια από την τυχαία. Δηλαδή με βάση τις εκτιμήσεις των προτιμήσεων των χρηστών που κάνει ο εκτιμητής, η πολιτική μας κάνει προτάσεις που απορρίπτονται περισσότερες φορές. Μπορούμε να δουμε διάφορους λόγους που μπορεί να συμβαίνει αυτό:

- Το πλήθος των χαρακτηριστικών είναι πολύ μεγάλο, ενώ το πλήθος των δεδομένων μικρό, και έτσι το σύστημα δεν μπορεί να πετύχει καλές αποδόσεις, συγκεκριμένα στο σύνολο εκπαίδευσης υπάρχουν πάνω από 22000 χαρακτηριστικά, ενώ στα δεδομένα αξιολόγησης 4328.
- Υπάρχει μια προκατάληψη προς την δράση 'no_action¨, καθώς είναι η πιο χρησιμοποιημένη δράση με 594 χρήσεις στα 1043 παραδείγματα, και είναι πολύ πιο συχνό να έχει θετική απάντηση, καθώς οποιαδήποτε απάντηση του χρήστη πέρα από την αρνητική και τον τερματισμό της συνομιλίας θεωρείται θετική απάντηση.

Τα παραπάνω, επαληθεύονται και πειραματικά, καθώς δοκιμάζοντας διάφορα συγκείμενα για να δούμε τις δράσεις που θα προτείνει η πολιτική, βλέπουμε ότι δίνει ίδια πιθανότητα σε

Απώλεια τυχαίας πολιτικής	Απώλεια πολιτικής μας
0.459	0.530

Πίνακας 5.3: Ασύγχρονη εκπαίδευση



Σχήμα 5.3: Δείγμα διαλόγου ΕΜ με ανθρώπινη ανατροφοδότηση

όλες τις επιλογές, προσομοιώνει δηλαδή την τυχαία.

5.6.2 Ενισχυτική μάθηση με ανθρώπινη ανατροφοδότηση

Με σκοπό να βελτιώσουμε την απόδοση της πολιτικής μας, χρησιμοποιήσαμε τεχνικές ενισχυτικής μάθησης με ανθρώπινη ανατροφοδότηση. Η τεχνική αυτή αναπτύσσεται τα τελευταία χρόνια, και έγινε γνωστή χάρη στην διάδοση του ChatGPT, στο οποίο χρησιμοποιήθηκαν τέτοιες τεχνικές για fine-tuning[24].

Στην δική μας περίπτωση, χρησιμοποιήσαμε την τεχνική για να δημιουργήσουμε ένα baseline από παραδείγματα, τα οποία μπορούν μετά να χρησιμοποιηθούν για παραπάνω ασύγχρονη εκπαίδευση της πολιτικής. συγκεκριμένα, φτιάξαμε ένα σύστημα το οποίο χρησιμοποιώντας όλες τις προθέσεις του χρήστη που είναι διαθέσιμες στην Θεανώ. Το σύστημα επιλέγει ένα τυχαίο πλήθος από αυτές και φτιάχνει ένα πλασματικό διάλογο με αιτήματα του χρήστη προς την Θεανώ. Έπειτα, δημιουργείται μια σύσταση, και ο πραγματικός χρήστης του συστήματος, μπορεί να κρίνει αν η σύσταση είναι σχετική ή όχι. Έτσι μαζεύουμε παραδείγματα τα οποία πιο ξεκάθαρα διακρίνουν την ποιότητα των συστάσεων. Στο Σχήμα 5.3 μπορεί να φανεί ένα παράδειγμα ενός τέτοιου διαλόγου. Οι διάλογοι αυτοί μετατρέπονται μετά στην μορφή εισόδου του Vowpal Wabbit και αποθηκεύονται για μετέπειτα χρήση.

5.7 Ενσωμάτωση με το Rasa

Εφόσον δημιουργήσαμε μια πολιτική, θα πρέπει να την ενσωματώσουμε στο Rasa για να μπορέσουμε να την εκπαιδεύσουμε περαιτέρω σύγχρονα.

5.7.1 Αρχική προσέγγιση

Η αρχική προσέγγιση που δοκιμάσαμε ήταν η προσθήκη της πολιτικής/του μοντέλου μέσα στην προϋπαρχουσα συνάρτηση που υπήρχε στον action server. Για να το κάνουμε αυτό, χρειάζεται αρχικά να εξάγουμε πληροφορίες από την καταγραφή της κατάστασης του Rasa. Έτσι εξάγουμε τις τιμές των μακροχρόνιων πληροφοριών, τις προηγούμενες προθέσεις του χρήστη και δημιουργούμε τις εναπομείνασες προθέσεις. Αν δεν έχει μείνει καμία πρόθεση, το οποίο σημαίνει ότι όλα τα θέματα που είναι διαθέσιμα από την Θεανώ έχουν συζητηθεί, τότε χρησιμοποιούμε την γενική πρόθεση `no_action` που αντιστοιχεί στην ερώτηση ¨Θα ήθελες να μάθεις κάτι άλλο;¨.

Αν υπάρχουν επιπλέον προτάσεις, τότε καλείται η συνάρτηση `smart_suggest`, η οποία επικαλείται η πολιτική για να κάνει πρόταση. Για χρησιμοποιηθούν τα δεδομένα ως είσοδος στην πολιτική του Vowpal Wabbit, χρειάζεται να μετατραπούν στην μορφή εισόδου που αποδέχεται αυτό.

Η έξοδος της πολιτικής είναι μια συνάρτηση πυκνότητας πιθανότητας, δηλαδή μια λίστα από πιθανότητες, η καθεμία από τις οποίες αντιστοιχεί σε μια δράση. Δηλαδή η πιθανότητα στην αντίστοιχη θέση στην λίστα αντιστοιχεί στην πιθανότητα επιλογής της συγκεκριμένης δράσης. Έτσι για να καταλήξουμε σε κάποια απόφαση/δράση θα πρέπει να δειγματοληπτήσουμε αυτή την λίστα.

Για παράδειγμα, δοθεισας μιας λίστας $[0.7, 0.1, 0.1, 0.1]$, θα επιλέγαμε το πρώτο αντικείμενο με πιθανότητα 70%.

Με αυτή την δειγματοληψία, καταλήγουμε τελικά σε μια δράση/σύσταση από την πολιτική μας για το τι να προτείνει η Θεανώ ως επόμενο θέμα συζήτησης.

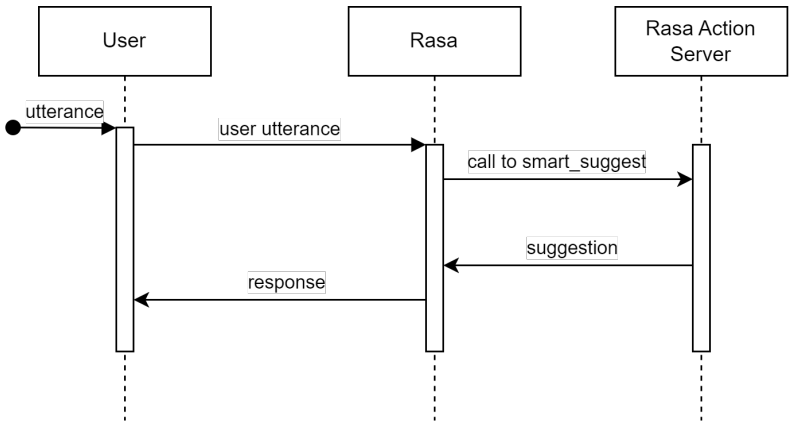
Όσον αφορά την σύγχρονη εκπαίδευση με βάση την αντίδραση του χρήστη στην πρόταση της Θεανώς, η προσέγγιση μας ήταν την επόμενη φορά που καλείται η λειτουργικότητα της πρότασης, να γίνεται πρώτα μια αναζήτηση του ιστορικού για τον εντοπισμό της αντίδρασης του χρήστη. Αυτή η προσέγγιση αποδείχτηκε προβληματική, καθώς η αναζήτηση στην πορεία του διαλόγου είναι περίπλοκη, μιας και μπορούν να υπάρχουν πολλά βήματα διαλόγου μεταξύ της πρώτης κλήσης της σύστασης και της δεύτερης.

Μια περιγραφής της επικοινωνίας μεταξύ Rasa, Rasa Action server και χρήστη φαίνεται στο Σχήμα 5.4. Αυτό που παραλείπεται είναι η διαδικασία μέσα στον Rasa Action server, η οποία περιγράφηκε παραπάνω.

5.7.2 Εξωτερική υπηρεσία

Για να μπορέσουμε να έχουμε καλύτερη γνώση της απάντησης του χρήστη, και άρα να μπορέσουμε να υπολογίσουμε την ανταμοιβή χρειαζόμαστε πιο άμεση πρόσβαση στην πληροφορία της κατάστασης του διαλόγου του Rasa.

Για το σκοπό αυτό, καταλήξαμε ότι ο πιο άμεσος τρόπο να πάρουμε πληροφορίες για τον



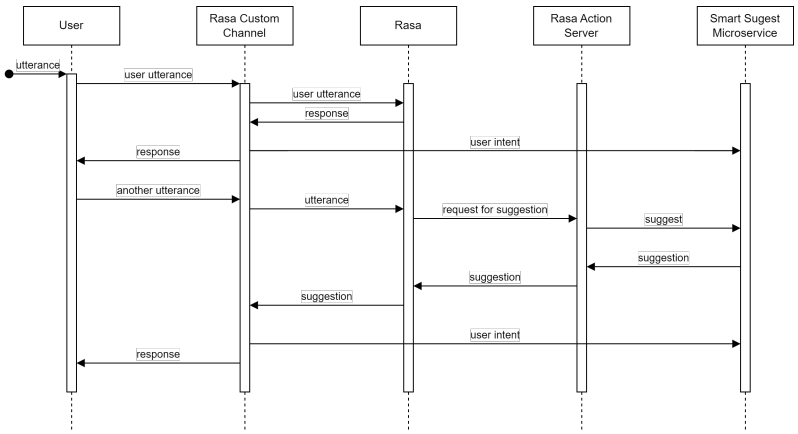
Σχήμα 5.4: Ακολουθιακό διάγραμμα επικοινωνίας μεταξύ Rasa και χρήστη

διάλογο ήταν με την ενσωμάτωση ενός προσαρμοσμένου καναλιού (custom channel), μιας επέκτασης που προσφέρει το Rasa. Ο σκοπός των προσαρμοσμένων καναλιών είναι η εύκολη επέκταση του Rasa σε περισσότερες πλατφόρμες φωνής και κειμένου. Μέσα σε αυτό το κανάλι όμως, μπορούμε να προσθέσουμε και επιπλέον λογική, η οποία θα μας επιτρέψει να καλούμε μια εξωτερική υπηρεσία, ένα τελικό σημείο μιας διεπαφή προγραμματισμού εφαρμογών (API endpoint), στην οποία θα στέλνεται η κατάσταση του διαλόγου. Ο κώδικας του προσαρμοσμένου καναλιού καλείται κάθε φορά που ο χρήστης στέλνει κάτι προς το Rasa.

Αυτή η αλλαγή απαιτεί κάποιες σημαντικές αλλαγές στην αρχιτεκτονική του συνολικού συστήματος. Συγκεκριμένα, πέρα από το προσαρμοσμένο κανάλι, θα πρέπει να δημιουργήσουμε την εξωτερική υπηρεσία, η οποία θα είναι πλεον υπεύθυνη για την διαχείριση της πολιτικής, τόσο την πρόβλεψη όσο και την σύγχρονη εκπαίδευση. Επιπλέον, ο action server θα πρέπει να καλεί την εξωτερική αυτή υπηρεσια για να κάνει την πρόβλεψη.

Η υπηρεσία αυτή δημιουργήθηκε έχοντας υπόψιν την έννοια των μίκρο-υπηρεσιών (microservices), υπηρεσιών δηλαδή που υπηρετούν ένα και μοναδικό σκοπό. Στην περίπτωση μας αυτός ο σκοπός είναι η περίκλειση της πολιτικής. Έτσι αυτή η υπηρεσία έχει δύο άκρα που επικοινωνεί με το εξωτερικό περιβάλλον της. Το πρώτο είναι το `/suggest`, το οποίο λαμβάνει δεδομένα για την κατάσταση του διαλόγου του χρήστη και με βάση αυτά παράγει και επιστρέφει μια σύσταση για την συνέχεια της συζήτησης. Το δεύτερο είναι το `/intent`, το οποίο καλείται όταν ο χρήστης στέλνει κάποιο μήνυμα στην Θεανώ. Η λειτουργικότητα του είναι να ελέγχει αν έχει γίνει σύσταση προς τον χρήστη, και αν έχει γίνει να αξιολογεί την ανταπόκριση του χρήστη και με βάση αυτή να μοντελοποιεί την ανταμοιβή. Τα δεδομένα σχετικά με την κατάσταση κάθε διαλόγου αποθηκεύονται σε μια προσωρινή βάση δεδομένων και κάθε μια ώρα καθαρίζονται. Με αυτό τον τρόπο μπορούμε να ελέγξουμε και αν ο χρήστης έφυγε από την συνομιλία μετά απο μια σύσταση και να εκπαιδεύσουμε την πολιτική με βάση αυτό.

Για να δημιουργήσει το Rasa την κατάσταση του διαλόγου, θα πρέπει πρώτα το αίτημα του



Σχήμα 5.5: Ακολουθιακό διάγραμμα επικοινωνίας μεταξύ Rasa και χρήστη, όταν χρησιμοποιούμε την εξωτερική υπηρεσία.

χρήστη να περάσει μέσα από το κανάλι και το ίδιο το Rasa για να δημιουργηθεί η κατάσταση αυτή πρώτα. Έτσι το κανάλι καλεί την υπηρεσία μας και της μεταφέρει την πρόθεση του μόνο αφού το αίτημα του χρήστη έχει περάσει μέσα από Rasa. Όλα τα παραπάνω, γίνονται αρκετά εμφανή στο Σχήμα 5.5.

Για την δημιουργία της υπηρεσίας, χρησιμοποιήσαμε την γλώσσα Python, και συγκεκριμένα την βιβλιοθήκη FastAPI. Το FastAPI είναι ένα πλαίσιο ανάπτυξης REST εφαρμογών, το οποίο είναι γνωστό για την ευκολία του, την ταχύτητα του και την ευρωστότητα του. Χρησιμοποιείται από αρκετές μεγάλες εταιρίες, όπως η Microsoft και το Netflix για την ανάπτυξη κάποιον λειτουργιών τους. Για την αποθήκευση δεδομένων χρησιμοποιήθηκε μια βάση sqlite, ενώ για την διαχείριση της βάσης χρησιμοποιήθηκε η βιβλιοθήκη sqlAlchemy. Ολη η υπηρεσία τοποθετείται μέσα σε ένα Docker container με σκοπό την απομονωμένη εκτέλεση της.

Κεφάλαιο 6

Αποτελέσματα και περαιτέρω εργασία

6.1 Τελευταίες Τεχνολογίες (State of the Art)

Οι τελευταίες τεχνολογίες οσον αφορά τον διάλογο εμφανίστηκαν αφού είχαμε ξεκινήσει την εργασία μας. Αυτή είναι το ChatGPT, το οποίο δημιουργήθηκε από την OpenAI, το οποίο έφερε μια επανάσταση στο πως αντιλαμβανόμαστε τους διαλογικούς πράκτορες και το τι μπορούν να κάνουν. Συγκεκριμένα το ChatGPT μπορεί να παρέχει αναλυτικές και καλά δομημένες απαντήσεις σε πολλές κατηγορίες θεμάτων, μπορεί να κρατήσει υπόψιν του τι έχει συζητηθεί νωρίτερα στην συζήτηση και παρέχει απαντήσεις σε πολλές γλώσσες. Παρ΄όλα αυτά μπορεί να παρέχει με βεβαιότητα απαντήσεις οι οποίες είναι ανακριβείς ή και τελείως λάθος. Το ChatGPT βασίζεται στα γλωσσικά μοντέλα GPT3.5 και GPT4, τα οποία είναι κάποια από τα μεγαλύτερα που υπάρχουν αυτή την στιγμή. Το ChatGPT οδήγησε στην διάδοση της προόδου της Τεχνητής Νοημοσύνης και του τι μπορεί να καταφέρει και έφερε στο προσκήνιο πολλές συζητήσεις σχετικά με την ταχύτητα εξέλιξης της και τις κοινωνικές επιπτώσεις που μπορεί να έχει η τεχνολογία. Μετά την εμφάνιση του ChatGPT, εμφανίστηκαν επίσης και άλλα παρόμοια μοντέλα όπως το Bard από την Google, το οποίο βασίζεται στο γλωσσικό μοντέλο PaLM, το οποίο έχει 540 δισεκατομμύρια παραμέτρους, καθώς και το γλωσσικό μοντέλο LLaMA από την Meta, το οποίο έχει στην μεγαλύτερη του έκδοση 65 δισεκατομμύρια παραμέτρους. Οσον αφορά τις τελευταίες τεχνολογίες στο κομμάτι των συστάσεων, ένα μεγάλο ρεύμα έρευνας έχει επικεντρωθεί στην χρήση τεχνικών Multi-Armed Bandits για την παραγωγή των συστάσεων. Οι τεχνικές αυτές ουσιαστικά είναι ύπο-κλάδος της Ενισχυτικής Μάθησης, ο οποίος προσπαθεί να λύσει ένα πιο απλοποιημένο πρόβλημα. Στη βιομηχανία, μεγάλες εταιρίες τεχνολογίας χρησιμοποιούν τεχνικές bandits για να λύσουν προβλήματα όπως η σύσταση νέων να δείξουν στην αρχική σελίδα τους, ή των εξωφυλλων στις ταινίες που είναι προτεινόμενες για τον χρήστη. Αυτή είναι η προσέγγιση που ακολουθήσαμε και εμείς για να λύσουμε το πρόβλημα της σύστασης. Συγκεκριμένα δοκιμάσαμε διάφορους αλγορίθμους και

προσπαθήσαμε να εντοπίσουμε ποιοί ανταπεξέρχονται καλύτερα, με βάση το γεγονός ότι ο όγκος προηγούμενων δεδομένων που έχουμε είναι περιορισμένος.

6.2 Αποτελέσματα

6.3 Επεκτάσεις

Βιβλιογραφία

[1] A. M. Turing, "Computing machinery and intelligence", English, *Mind*, New Series, vol. 59, no. 236, pp. 433–460, 1950, ISSN: 00264423. [Online]. Available: http://www.jstor.org/stable/2251299.

[2] Θ. Ωειζενβαυμ, «ΕΛΙΖΑ—α ϲομπυτερ προγραμ φορ τηε στυδψ οφ νατυραλ λανγυαγε ϲομμυνιϲατιον βετωεεν μαν ανδ μαϲηινε», *ϑμμυν. ACM*, τόμ. 9, αρϑμ. 1, σσ. 36–45, Ιαν. 1966, ISSN: 0001-0782. DOI: 10.1145/365153.365168. διεύϑν.: https://doi.org/10.1145/365153.365168.

[3] Ι. Βλαχάβας, Π. Κεφαλάς, Ν. Βασιλειάδης, Φ. Κόκκορας και Η. Σακελλαρίου, *Τεχνητή Νοημοσύνη*. Πανεπιστήμιο Μακεδονίας, 2006.

[4] J. Langford and T. Zhang, "The Epoch-Greedy Algorithm for Multi-armed Bandits with Side Information", in *Advances in Neural Information Processing Systems*, J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds., vol. 20, Curran Associates, Inc., 2007.

[5] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, *Recommender Systems: An Introduction*. Cambridge University Press, 2010. DOI: 10.1017/CBO9780511763113.

[6] A. Agarwal, O. Chapelle, M. Dudik, and J. Langford, *A reliable effective terascale linear learning system*, 2013. arXiv: 1110.4198 [cs.LG].

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

[8] D. Silver, *Lectures on reinforcement learning*, URL: https://www.davidsilver.uk/teaching/, 2015.

[9] P. H. Aditya, I. Budi, and Q. Munajat, "A comparative analysis of memory-based and model-based collaborative filtering on the implementation of recommender system for e-commerce in indonesia: A case study pt x", in *2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 2016, pp. 303–308. DOI: 10.1109/ICACSIS.2016.7872755.

[10] J. Li, W. Monroe, A. Ritter, M. Galley, J. Gao, and D. Jurafsky, *Deep reinforcement learning for dialogue generation*, 2016. DOI: 10.48550/ARXIV.1606.01541. [Online]. Available: https://arxiv.org/abs/1606.01541.

[11] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey", *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 26–38, Nov. 2017, ISSN: 1558-0792. DOI: 10.1109/MSP.2017.2743240.

[12] N. T. Blog, *Artwork Personalization at Netflix*, en, Dec. 2017. [Online]. Available: https://netflixtechblog.com/artwork-personalization-c589f074ad76 (visited on 01/30/2023).

[13] D. Russo, B. Van Roy, A. Kazerouni, I. Osband, and Z. Wen, "A tutorial on thompson sampling", 2017. DOI: 10.48550/ARXIV.1707.02038. [Online]. Available: https://arxiv.org/abs/1707.02038.

[14] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need", in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17, Long Beach, California, USA: Curran Associates Inc., 2017, pp. 6000–6010, ISBN: 9781510860964.

[15] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: A Bradford Book, 2018, ISBN: 0262039249.

[16] X. Zhao, L. Zhang, Z. Ding, L. Xia, J. Tang, and D. Yin, "Recommendations with negative feedback via pairwise deep reinforcement learning", in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ACM, Jul. 2018. DOI: 10.1145/3219819.3219886. [Online]. Available: https://doi.org/10.1145%2F3219819.3219886.

[17] Z. Zhang, X. Li, J. Gao, and E. Chen, *Budgeted policy learning for task-oriented dialogue systems*, 2019. DOI: 10.48550/ARXIV.1906.00499. [Online]. Available: https://arxiv.org/abs/1906.00499.

[18] T. Βροων, B. Μαννν, N. Pψδερ κ.ά., «Λανγυαγε Μοδελς αρε Φεω-Σηοτ Λεαρνερς», στο *Αδανσες ιν Νευραλ Ινφορματιον Προσεσσινγ Σψστεμς*, H. Λαροχηελλε, M. Ρανζατο, P. Ηαδσελλ, M. Βαλςαν και H. Λιν, επιμελητές, τόμ. 33, Ὗρραν Ασσοχιατες, Ινς., 2020, σσ. 1877–1901. διεύθν.: https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

[19] T. Bunk, D. Varshneya, V. Vlasov, and A. Nichol, *Diet: Lightweight language understanding for dialogue systems*, 2020. arXiv: 2004.09936 [cs.CL].

[20] T. Lattimore and C. Szepesvári, *Bandit Algorithms*. Cambridge University Press, 2020, ISBN: 9781108486828.

[21] M. M. Afsar, T. Crump, and B. Far, *Reinforcement learning based recommender systems: A survey*, 2021. DOI: 10.48550/ARXIV.2101.06286. [Online]. Available: https://arxiv.org/abs/2101.06286.

[22] J. Ni, T. Young, V. Pandelea, F. Xue, V. Adiga, and E. Cambria, "Recent advances in deep learning based dialogue systems: A systematic survey", *CoRR*, vol. abs/2105.04387, 2021. arXiv: 2105.04387. [Online]. Available: https://arxiv.org/abs/2105.04387.

[23] N. ἔντουρα, K. Παλιος, Ψ. ἄσιλακις, Γ. Παρασκεοπουλος, N. Κατσαμανις και ". Κατσουρος, «Τηεανο: Α Γρεεκ-σπεακινγ ςονερσατιοναλ αγεντ φορ "ΟΎΙΔ-19», στο *Προςεεδινγς οφ τηε 1στ Ωορκσηοπ ον ΝΛΠ φορ Ποσιτιε Ιμπαςτ*, A. Φιελδ, Σ. Πραβηυμοψε, M. Σαπ, Z. Θιν, Θ. Ζηαο και ". Βροςκεττ, επιμελητές, Ονλινε: Ασσοχιατιον φορ ὅμπυτατιοναλ

Λινγυιστιχς, Αύγ. 2021, σσ. 36–46. DOI: 10.18653/1/2021.νλπ4ποσιμπαχτ-1.5. διεύθν.: https://aclanthology.org/2021.nlp4posimpact-1.5.

[24] L. Ouyang, J. Wu, X. Jiang, *et al.*, *Training language models to follow instructions with human feedback*, 2022. DOI: 10.48550/ARXIV.2203.02155. [Online]. Available: https://arxiv.org/abs/2203.02155.

[25] Z. Yan, "Bandits for recommender systems", *eugeneyan.com*, May 2022. [Online]. Available: https://eugeneyan.com/writing/bandits/.