ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΕΡΓΑΣΤΗΡΙΟ ΣΥΣΤΗΜΑΤΩΝ ΑΠΟΦΑΣΕΩΝ ΚΑΙ ΔΙΟΙΚΗΣΗΣ
ΤΟΜΕΑΣ ΗΛΕΚΤΡΙΚΩΝ ΒΙΟΜΗΧΑΝΙΚΩΝ ΔΙΑΤΑΞΕΩΝ ΚΑΙ
ΣΥΣΤΗΜΑΤΩΝ ΑΠΟΦΑΣΕΩΝ

ΕΦΑΡΜΟΓΕΣ ΕΝΙΣΧΥΤΙΚΗΣ ΜΑΘΗΣΗΣ ΣΕ ΣΥΣΤΗΜΑΤΑ ΣΥΣΤΑΣΕΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Κατσικερού Θεόδωρου

Επιβλέπων : Ασκούνης Δημήτριος
Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2021

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΕΡΓΑΣΤΗΡΙΟ ΣΥΣΤΗΜΑΤΩΝ ΑΠΟΦΑΣΕΩΝ ΚΑΙ ΔΙΟΙΚΗΣΗΣ
ΤΟΜΕΑΣ ΗΛΕΚΤΡΙΚΩΝ ΒΙΟΜΗΧΑΝΙΚΩΝ ΔΙΑΤΑΞΕΩΝ ΚΑΙ
ΣΥΣΤΗΜΑΤΩΝ ΑΠΟΦΑΣΕΩΝ

ΕΦΑΡΜΟΓΕΣ ΕΝΙΣΧΥΤΙΚΗΣ ΜΑΘΗΣΗΣ ΣΕ ΣΥΣΤΗΜΑΤΑ ΣΥΣΤΑΣΕΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Κατσικερού Θεόδωρου

Επιβλέπων: Ασκούνης Δημήτριος
Καθηγητής Ε.Μ.Π

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 5η Νοεμβρίου 2021.

.....	.....	.....
Δημήτριος Ασκούνης	Ιωάννης Ψαρράς	Χρυσόστομος Δούκας
Καθηγητής Ε.Μ.Π	Καθηγητής Ε.Μ.Π	Αναπ. Καθηγητής Ε.Μ.Π

Αθήνα, Οκτώβριος 2021

.....

Κατσικερός Θεόδωρος

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

ΠΕΡΙΛΗΨΗ

Τα Συστήματα Συστάσεων είναι ένας τύπος συστημάτων που χρησιμοποιείται σχεδόν σε κάθε online πλατφόρμα, ώστε να προτείνονται στους χρήστες αντικείμενα που μπορεί να τους ενδιαφέρουν, κυρίως για διαφημιστικούς/εμπορικούς λόγους. Τα νέα δεδομένα που φέρνει η εποχή της τεχνολογίας όμως με την αύξηση της ροής δεδομένων στο διαδίκτυο και οι δυνατότητες που προσφέρει η επιστήμη των Big Data, επιβάλλουν τη χρήση Μηχανικής Μάθησης και στον τομέα των Συστημάτων Συστάσεων, προκειμένου να γίνονται πιο αποτελεσματικές και εύστοχες συστάσεις. Έχει υπάρξει αρκετή έρευνα ως προς την εφαρμογή της Μηχανικής Μάθησης σε Συστήματα Συστάσεων, κυρίως με χρήση της Επιβλεπόμενης Μάθησης που μέχρι πρότινως θεωρούταν η ιδανική για εφαρμογή στο συγκεκριμένο κλάδο. Παρόλα αυτά, η προσπάθεια της έρευνας τα τελευταία χρόνια συγκεντρώνεται στην εφαρμογή πιο σύνθετων μεθόδων, όπως η Ενισχυτική Μάθηση, αφού είναι πολλά υποσχόμενη όσον αφορά τη διαχείριση μεγάλου όγκου δεδομένων.

Στην παρούσα διπλωματική εργασία, μελετήθηκαν και παρουσιάστηκαν παραδείγματα εφαρμογής της Ενισχητικής Μάθησης σε Συστήματα Συστάσεων σε διάφορους τομείς, όπως το e-commerce (Feed Streaming Recommendation), η Μουσική (Music Recommendation), οι Ειδήσεις (News Recommendation) και οι Διαφημίσεις (Ad Recommendation). Αναλύθηκε σε μεγάλο βαθμό η τεχνική υλοποίησης των αλγορίθμων, εξηγήθηκε γιατί η Ενισχυτική Μάθηση ήταν το κατάλληλο εργαλείο για την κάθε περίπτωση ξεχωριστά, παρουσιάστηκαν τα ωφέλη μέσω παρουσίασης πραγματικών πειραμάτων που έχουν διεξαχθεί, και τέλος επισημάνθηκαν οι δυσκολίες που προκύπτουν λόγω της εφαρμογής της Ενισχυτικής Μάθησης. Προφανώς, η ενσωμάτωση τεχνικών Ενισχυτικής Μάθησης στα Συστήματα Συστάσεων, επιφέρει επιπλέον πολυπλοκότητα και δυσκολία υλοποίησης των εκάστοτε αλγορίθμων. Στη συγκεκριμένη περίπτωση ωστόσο, τα πειραματικά αποτελέσματα δηλώνουν με σαφή τρόπο πως το τίμημα αυτό αξίζει να το πληρώσει κανείς, καθώς τα Συστήματα Συστάσεων που χρησιμοποιούν ως εργαλείο την Ενισχυτική Μάθηση είναι πολύ καλύτερα από τα παραδοσιακά Συστήματα Συστάσεων σε κάποιους τομείς εφαρμογών.

Λέξεις κλειδια: Μηχανική μάθηση, Ενισχυτική μάθηση, Συστήματα Συστάσεων, Τεχνητή Νοημοσύνη, Επιβλεπόμενη Μάθηση

ABSTRACT

Recommendation systems are used on almost every online platform, to promote items that may be of interest to the user, mainly for advertising / commercial purposes. However, with the increase of data flow on the internet and the possibilities offered by the Big Data science and technologies, the use of Machine Learning in the field of Recommendation Systems has become mandatory, in order to achieve more effective and accurate recommendations. There has been a lot of research on the application of Machine Learning in Recommendation Systems, but mainly with the use of Supervised Learning which until recently was considered the most ideal form of Machine Learning for assisting applications in the recommendation system industry. However, research effort in recent years has focused on the application of more complex methods, such as Reinforcement Learning, as it is more appropriate than other techniques in terms of managing large volumes of data.

In this thesis, the purpose was to examine and present examples of applications of Reinforcement Learning in Recommendation Systems in various fields, such as e-commerce (Feed Streaming), Music, News and Ads. The technique of implementing the Reinforcement Learning algorithms has been analyzed, the facts why Reinforcement Learning was suitable for each case individually has been explained, the benefits were presented through the presentation of real experiments that have been conducted, and finally the difficulties arising from the implementation of Reinforcement Learning were pointed out. Obviously, when choosing Reinforcement Learning instead of simpler traditional techniques a cost in complexity and difficulty has to be paid. In this case, however, the experimental results clearly indicate that it is worth the cost, as the Recommendation Systems that use Reinforcement Learning as a tool are much better than the more traditional referral systems in several sectors.

Key words: Machine Learning, Reinforcement Learning, Recommendation Systems, Artificial Intelligence, Supervised Learning

ΕΥΧΑΡΙΣΤΙΕΣ

Με την ολοκλήρωση της παρούσας διπλωματικής εργασίας νιώθω την ανάγκη να εκφράσω τις ιδιαίτερες ευχαριστίες μου στον Καθηγητή του Ε.Μ.Π. κ. Δημήτριο Ασκούνη για την ευκαιρία που μου έδωσε με την εκπόνηση της εργασίας αυτής.

Επίσης, θα ήθελα να ευχαριστήσω θερμά τον Καρακόλη Ευάγγελο για την πολύ καλή συνεργασία που είχαμε καθ' όλη τη διάρκεια εκπόνησης της παρούσας διπλωματικής εργασίας, καθώς και για την πολύτιμη βοήθεια και καθοδήγηση που μου προσέφερε.

Ευχαριστώ ακόμα τον καθηγητή κ. Ιωάννη Ψαρρά και τον Αναπληρωτή Καθηγητή Ε.Μ.Π κ. Χρυσόστομο Δούκα για την συμμετοχή τους στην επιτροπή εξέτασης της διπλωματικής εργασίας μου.

Τέλος θα ήθελα να ευχαριστήσω την οικογένεια μου για όλη τη στήριξη και βοήθεια που μου έδωσαν.

Contents

Table of Figures

ΚΕΦΑΛΑΙΟ 1
ΕΙΣΑΓΩΓΗ

1.1 ΣΥΣΤΗΜΑΤΑ ΣΥΣΤΑΣΕΩΝ

Στις μέρες μας, η ποσότητα της πληροφορίας που είναι διαθέσιμη στο διαδίκτυο αυξάνεται με εκρηκτικούς ρυθμούς παγκοσμίως. Παρά το γεγονός πως αυτό παρέχει στους χρήστες περισσότερες επιλογές και περισσότερες ευκαιρίες για γνώση, ταυτόχρονα κάνει πιο δύσκολη την επιλογή της χρήσιμης/ενδιαφέρουσας πληροφορίας, ανάμεσα στις πολλές άχρηστες. Αυτό το πρόβλημα είναι ευρέως γνωστό ως «information overload» [1].

Προκειμένου να αντιμετωπιστεί αυτό το πρόβλημα, στις online πλατφόρμες έχουν εισαχθεί τα Συστήματα Συστάσεων. Αυτά θα μπορούσαν εν συντομία να χαρακτηριστούν ως εξατομικευμένα συστήματα πληροφορίας, που χρησιμοποιούνται για να προβλέψουν το πόσο χρήσιμη ειναι μια πληροφορία για τον χρήστη, ή γενικότερα ως ένα σύστημα που καθοδηγεί τον χρήστη προς ενδιαφέροντα αντικείμενα ή πληροφορίες μέσα από μία εκτενή γκάμα διαθέσιμων επιλογών.

Φυσικά, τα Συστήματα Συστάσεων χρησιμοποιούνται σε πάρα πολλές εφαρμογές, όπως την πρόβλεψη προϊόντων που ο χρήστης είναι πιθανό να αγοράσει, ταινίες που θα μπορούσε να δει, τραγούδια που θα μπορούσε να ακούσει, ειδήσεις που θα του φαίνονταν ενδιαφέρουσες και πολλά άλλα.

Θα ήταν χρήσιμο να εξηγηθεί λίγο η λειτουργία των Συστημάτων Συστάσεων. Αρχικά, προκειμένου τα συστήματα αυτά να κάνουν κατά το δυνατόν επιτυχημένες συστάσεις, χρειάζονται κάποια «εμπειρία». Αυτήν την αποκτούν μέσω διαφόρων μορφών Μηχανικής Μάθησης. Δημιουργούν κάποιες συσχετίσεις ανάμεσα στους χρήστες και τα προϊόντα (user-product), τα προϊόντα με άλλα παρεμφερή προϊόντα (product-product), ή χρήστες με πανομοιότυπα ενδιαφέροντα (user-user) [2].

<u>USER-PRODUCT</u>

Η συσχέτιση αυτή υπάρχει όταν κάποιοι χρήστες έχουν κάποια προτίμηση σε συγκεκριμένα θέματα που τους αφορούν. Για παράδειγμα, ένας παίκτης ποδοσφαίρου μπορεί να έχει προτίμηση σε αντικείμενα που αφορούν το ποδόσφαιρο. Έτσι, μια ιστοσελίδα ηλεκτρονικών πωλήσεων (e-commerce) θα χτίσει μια user-product σχέση ανάμεσα σε παίκτες ποδοσφαίρου-ποδοσφαιρικά αντικείμενα.

PRODUCT-PRODUCT

Αυτού του τύπου οι συσχετίσεις υπάρχουν όταν κάποια προϊόντα είναι φύσει παραμφερή είτε βάσει εμφάνισης, είτε βάσει περιγραφής. Παραδείγματα αποτελούν ταινίες ή βιβλία ίδιου είδους (π.χ. thriller, comedy, drama, …), φαγητά ίδιας κουζίνας, άρθρα που αφορούν την ίδια είδηση.

USER-USER

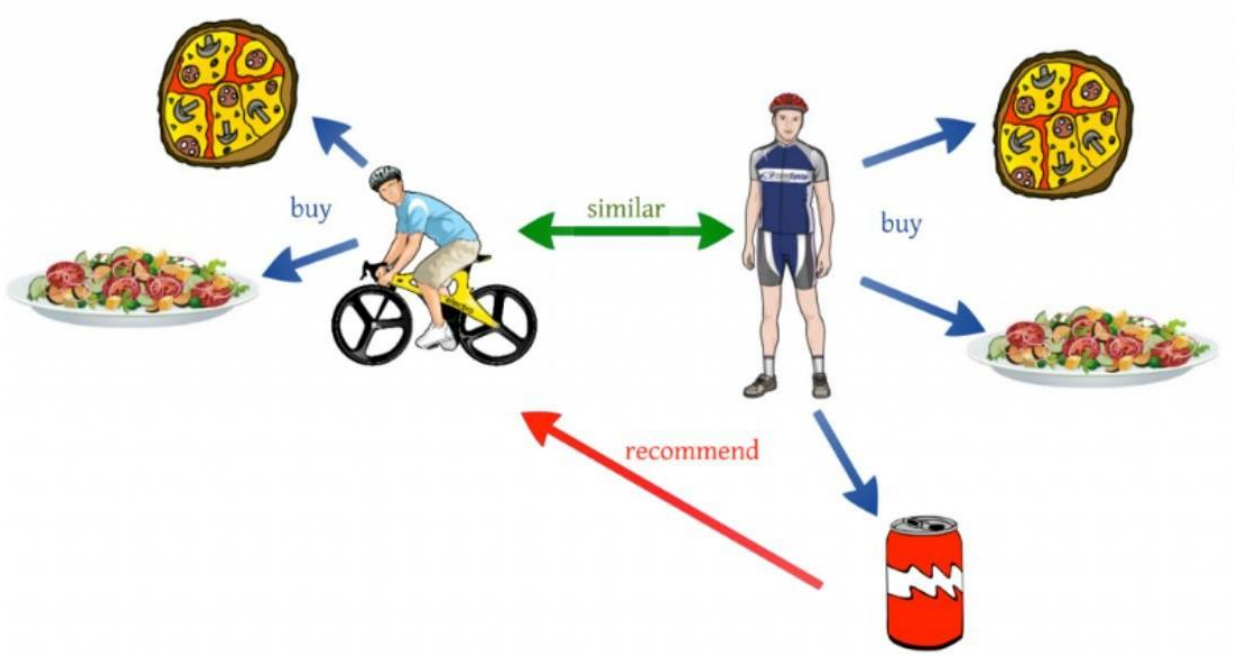
Η συσχέτιση αυτή υπάρχει όταν οι χρήστες έχουν κοινό ενδιαφέρον σε κάποια αντικείμενα. Παραδείγματα αποτελούν οι κοινοί φίλοι, παρεμφερές ιστορικό, ίδια ηλικία, ίδιο φύλο, ίδιος τόπος κατοικίας κ.α.

Για να είναι εφικτές όλες οι παραπάνω συσχετίσεις, τα Συστήματα Συστάσεων χρησιμοποιούν δεδομένα που αφορούν:

1. τη συμπεριφορά του χρήστη, η οποία συλλέγεται από διάφορες αξιολογήσεις που κάνει, clicks σε σελίδες ή προϊόντα, ιστορικό αγορών.

2. τα δημογραφικά χαρακτηριστικά, δηλαδή ηλικία, τόπο κατοικίας, εκπαίδευση, εισόδημα, τοποθεσία.

3. τα χαρακτηριστικά των αντικειμένων, δηλαδή τα είδος ενός βιβλίου, το cast σε μια ταινία, την κουζίνα στην περίπτωση του φαγητού.



Εικόνα 1.1: Λειτουργία Συστήματος Συστάσεων

1.2 ΕΝΙΣΧΥΤΙΚΗ ΜΑΘΗΣΗ (REINFORCEMENT LEARNING)

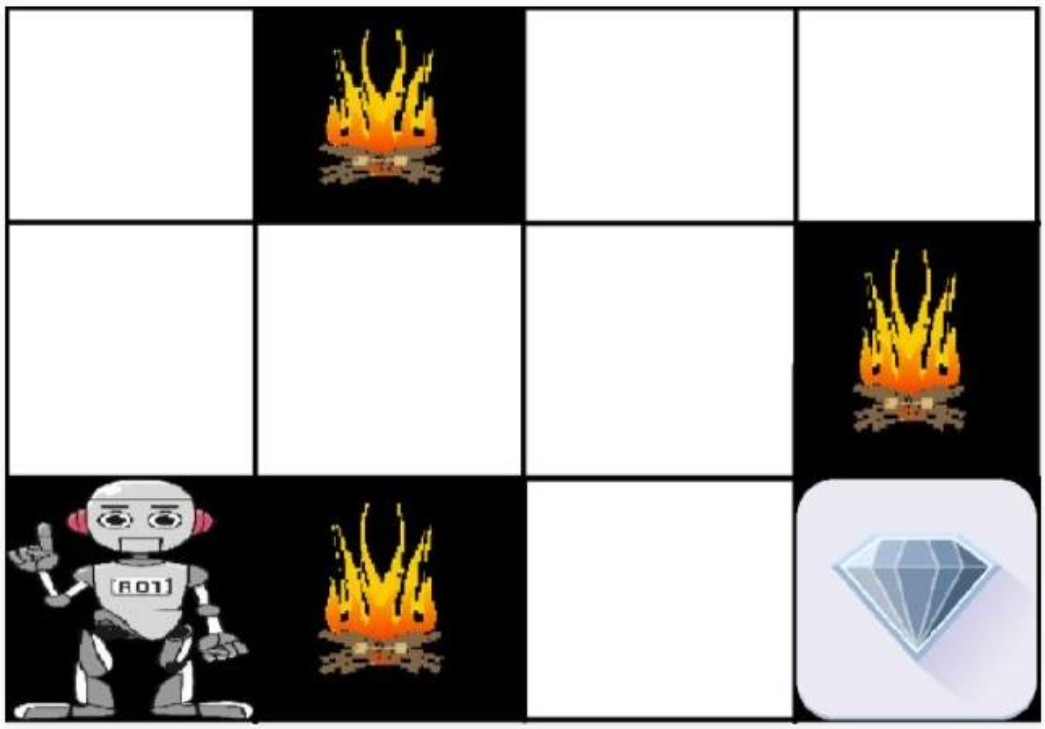
Η Ενισχυτική Μάθηση είναι ένας από τους τρεις κύριους κλάδους της Μηχανικής Μάθησης, μαζί με την Επιβλεπόμενη (Supervised) και τη Μη Επιβλεπόμενη Μάθηση (Unsupervised Learning).

Η Επιβλεπόμενη Μάθηση [3] εκπαιδεύεται χρησιμοποιώντας επισημειωμένα (labeled) σύνολα δεδομένων για εκπαίδευση που δίνονται από τον προγραμματιστή και με βάση αυτά δημιουργεί μια συνάρτηση που επιτρέπει αργότερα να κάνει όσο το δυνατόν καλύτερη κατηγοριοποίηση στις εισόδους που δίνονται.

Η Μη Επιβλεπόμενη Μάθηση [4] δεν χρειάζεται την ανθρώπινη εποπτεία και ψάχνει για μοτίβα σε μη-επισημειωμένα δεδομένα. Με βάση αυτά τα μοτίβα φτιάχνει μια συνάρτηση κατηγοριοποίησης.

Ένα σύστημα Ενισχυτικής Μάθησης [5] δεν κάνει τίποτα από τα παραπάνω. Ουσιαστικά, εκπαιδεύεται μόνο του στην πράξη, ξεκινώντας συνήθως από τυχαίες ανταποκρίσεις σε εισόδους. Δυνειτικά βελτιώνεται μέσω μιας συνάρτησης ανταμοιβής που διορθώνει την ανταπόκριση και βελτιώνει τις εξόδους. Πρακτικά, λοιπόν, ο ίδιος ο χρήστης είναι αυτός που εν αγνοία του ανταμοίβει ή τιμωρεί το σύστημα για κάθε επιθυμητή ή ανεπιθύμητη, αντίστοιχα, ανταπόκριση.

Ένα παράδειγμα που θα βοηθήσει στην κατανόηση της έννοιας είναι το εξής (Εικόνα 1.2):



Εικόνα 1.2: Παράδειγμα εφαρμογής Ενισχυτικής Μάθησης

Η Εικόνα 1.2 δείχνει έναν χάρτη, όπου υπάρχει ένα ρομπότ, ένα διαμάντι και τρεις διαφορετικές εστίες φωτιάς. Προφανώς το ρομπότ θέλει να φτάσει στο διαμάντι δίχως να καεί από τη φωτιά. Το ρομπότ μαθαίνει δοκιμάζοντας όλα τα μονοπάτια και εν τέλει διαλέγει το μονοπάτι που του δίνει τη μεγαλύτερη ανταμοιβή σε συνδυασμό με τα λιγότερα βήματα. Η συνάρτηση ανταμοιβής δίνει στο ρομπότ κάποια ανταμοιβή για κάθε σωστή κίνηση, ενώ το τιμωρεί για κάθε λανθασμένη. Η συνολική ανταμοιβή υπολογίζεται όταν φτάσει στον τελικό στόχο, που είναι το διαμάντι. Η ανταμοιβή ή τιμωρία σε κάθε βήμα δίνεται μέσω του δυναμικού προγραμματισμού, ώστε να μη χρειάζεται η διαδικασία να ξεκινάει από την αρχή κάθε φορά που το ρομπότ μετακινείται σε κάποιο πεδίο με φωτιά.

Τα βασικά στοιχεία της Ενισχυτικής Μάθησης είναι:

1. Η είσοδος, η οποία ορίζει την αρχική κατάσταση του συστήματος.

2. Η έξοδος. Κάθε πρόβλημα έχει πολλές πιθανές, αλλά θέλουμε να επιλέγεται η βέλτιστη.

3. Η εκπαίδευση του συστήματος. Αυτή εξαρτάται από την είσοδο. Το σύστημα θα επιστρέψει κάποια έξοδο και ουσιαστικά ο χρήστης επιλέγει αν θα το ανταμείψει ή θα το τιμωρήσει βάσει της εξόδου αυτής.

4. Το γεγονός ότι το σύστημα συνεχίζει να εκπαιδεύεται όσο χρησιμοποιείται.

5. Η καλύτερη λύση αποφασίζεται με μοναδικό κριτήριο τη μέγιστη ανταμοιβή.

1.3 ΕΝΙΣΧΥΤΙΚΗ ΜΑΘΗΣΗ ΣΕ ΣΥΣΤΗΜΑΤΑ ΣΥΣΤΑΣΕΩΝ

Γενικά, τα περισσότερα Συστήματα Συστάσεων είναι βασισμένα σε Επιβλεπόμενη Μάθηση. Η Επιβλεπόμενη Μάθηση είναι δεδομένα μια πολύ αποτελεσματική μέθοδος και τα συστήματα που τη χρησιμοποιούν έχουν πολύ αξιόλογα αποτελέσματα. Παρόλα αυτά, σε ορισμένες εφαρμογές, θέλουμε οι συστάσεις που θα γίνονται να είναι εξατομικευμένες για κάθε χρήστη. Η μέθοδος της Επιβλεπόμενης Μάθησης δεν διαχωρίζει τους «επισκέπτες» από τις «επισκέψεις». Αυτό σημαίνει πως η κάθε επίσκεψη εκλαμβάνεται από το σύστημα ως ένας νέος επισκέπτης, η εκμάθηση του Συστήματος Συστάσεων γίνεται για όλους τους χρήστες συνολικά και, συνεπώς, οι συστάσεις του δεν είναι προσαρμοσμένες στον κάθε χρήστη και στις ανάγκες του, αλλά κοινές.

Η μετρική που συνήθως χρησιμοποιείται σε τέτοια συστήματα είναι το Click Through Rate (CTR) [6], η οποία είναι η πλέον κατάλληλη για την εκτίμηση της απόδοσης τέτοιων άπληστων αλγορίθμων. Όμως, καθώς οι χρήστες τείνουν να επιστρέφουν στην εκάστοτε ιστοσελίδα, αυτές οι μέθοδοι κρίνονται ανεπαρκείς όταν το σύστημα καλείται να λάβει υπόψιν την μακροπρόθεσμη επιρροή που ασκείται στον χρήστη και στα ενδιαφέροντά του μέσω της συνεχούς τριβής του με τα δεδομένα της ιστοσελίδας. Έτσι, δημιουργείται η ανάγκη να διαχωρίσουμε τις έννοιες «επίσκεψη» και «επισκέπτης».

Οι αλγόριθμοι Ενισχυτικής Μάθησης προσπαθούν να βελτιστοποιήσουν την μακροπρόθεσμη απόδοση του συστήματος. Η φύση των αλγορίθμων αυτών τους επιτρέπει να λάβουν υπόψιν όλη τη διαθέσιμη γνώση για τον χρήστη προκειμένου να διαλέξουν τις συστάσεις που μεγιστοποιούν τον συνολικό αριθμό των ανταποκρίσεων του χρήστη. Αυτή η μετρική ονομάζεται user’s life-time value (LTV) [6].

Σε αντίθεση με τις άλλες προσεγγίσεις, οι αλγόριθμοι Ενισχυτικής Μάθησης διαφοροποιούν τον «επισκέπτη» από την «επίσκεψη» και ομαδοποιούν όλες τις επισκέψεις ενός χρήστη (σε χρονολογική σειρά). Δηλαδή, μοντελοποιούν τους επισκέπτες, και όχι τις επισκέψεις. Αυτό σημαίνει ότι παρόλο που εμείς κρίνουμε την απόδοση του αλγορίθμου με βάση το CTR, αυτός

δεν προσπαθει να βελτιώσει τη μετρική αυτή, και θα ήταν καταλληλότερο να τον κρίνουμε με βάση το LTV, δηλαδή τον απαιτούμενο συνολικό αριθμό ανταποκρίσεων ανά χρήστη.

Παρόλες αυτές τις ιδανικές ιδιότητες, υπάρχουν κάποια βασικά εμπόδια στη χρήση της Ενισχυτικής Μάθησης στα Συτήματα Συστάσεων:

1. Το «cold start» [7]. Όπως εξηγήσαμε παραπάνω, η Ενισχυτική Μάθηση δεν εκπαιδεύεται μέσω κάποιου συνόλου δεδομένων για εκπαίδευση, αλλά καθαρά μέσω της εμπειρίας που αποκτά από τις ανταμοιβές και τις τιμωρίες που δέχεται. Γίνεται, λοιπόν, κατανοητό πως στην αρχή οι συστάσεις του θα είναι εντελώς τυχαίες.

2. Ο υπολογισμός μιας LTV πολιτικής.

3. Η αξιολόγηση της LTV πολιτικής μας, από τη στιγμή που μπορούμε να χρησιμοποιήσουμε μόνο ιστορικά δεδομένα, τα οποία έχουν παραχθεί από άλλες πολιτικές.

ΚΕΦΑΛΑΙΟ 2

ΣΥΣΤΗΜΑΤΑ ΣΥΣΤΑΣΕΩΝ ΗΛΕΚΤΡΟΝΙΚΟΥ ΕΜΠΟΡΙΟΥ

2.1 ΠΑΡΟΥΣΙΑΣΗ ΠΡΟΒΛΗΜΑΤΟΣ

Στο συγκεκριμένο Κεφάλαιο θα παρουσιαστεί η έρευνα που έγινε για τη δημιουργία ενός αλγορίθμου [8] , που ενσωματώνει τεχνικές Ενισχυτικής Μάθησης σε ένα Σύστημα Συστάσεων στον τομέα του ηλεκτρονικού εμπορίου (e-commerce).

Τα συστήματα συστάσεων βοηθούν τον χρήστη στην εύρεση πληροφοριών προτείνοντας αγαθά (όπως προϊόντα, ειδήσεις, υπηρεσίες κλπ.) που ταιριάζουν κατά το δυνατόν καλύτερα στις ανάγκες και τις προτιμήσεις του. Οι χρήστες είναι σε θέση να βλέπουν συνεχώς ατελείωτα αντικείμενα που εμφανίζονται στην οθόνη τους από σελίδες ροής πληροφοριών, όπως το Yahoo News, το Facebook, το Amazon. Ειδικότερα, σε σελίδες όπως το Amazon, που αφορά ροή προϊόντων, οι χρήστες μπορεί να επιλέξουν τα αντικείμενα και να δουν λεπτομέρειες για αυτά. Μπορούν, επίσης, να απορρίψουν (skip) αντικείμενα που δεν τους φαίνονται ελκυστικά και να συνεχίσουν να ψάχνουν, ή ακόμη και να απομακρυνθούν από τη σελίδα/ σύστημα συστάσεων λόγω της εμφάνισης πολλών αδιάφορων αντικειμένων. Υπό αυτό το πρίσμα, η αύξηση του αριθμού των clicks δεν θα είναι ο αυτοσκοπός πλέον. Είναι κομβικής σημασίας η αύξηση της ευχαρίστησης του χρήστη μέσω της αλληλεπίδρασής του με τις e-commerce σελίδες, η οποία μεταφράζεται σε δύο μετρικές: άμεση αντίδραση (π.χ. click, αγορά) και μακροπρόθεσμη αντίδραση, που σημαίνει ότι ο χρήστης θέλει να μείνει για πολλή ώρα στη σελίδα ροής πληροφοριών, ή αρχίζει να την επισκέφτεται συχνά και επαναλαμβανόμενα.

Ωστόσο, τα πιο παραδοσιακά συστήματα συστάσεων επικεντρώνουν την προσπάθεια στην αύξηση άμεσων μετρικών όπως το Click Through Rate (CTR). Όσο περισσότερο αυξάνεται η αλληλεπίδραση του χρήστη με την ιστοσελίδα, ένα καλό σύστημα συστάσεων πρέπει να προσέξει όχι μόνο να βελτιώσει τις άμεσες μετρικές και να επιφέρει μεγαλύτερο αριθμό ή πυκνότητα σε clicks, αλλά να μπορέσει να κάνει και τους χρήστες να παραμένουν ενεργοί στην ιστοσελίδα και να μη φεύγουν. Αυτό μετράται με μακροπρόθεσμες μετρικές. Οι μακροπρόθεσμες μετρικές συνήθως είναι πιο πολύπλοκες από τις άμεσες ή βραχυπρόθεσμες. Μερικά παραδείγματα μακροπρόθεσμων μετρικών είναι το dwell time [9] (το οποίο ουσιαστικά δείχνει πόση ώρα ο χρήστης παρέμεινε σε μια συγκεκριμένη σελίδα της εφαρμογής και όχι συνολικά στην εφαρμογή, δηλαδή το κατά πόσο εξάντλησε το περιεχόμενο της συγκεριμένης σελίδας), το depth of the page-viewing (δηλαδή το πόσες σελίδες «βάθος» έφτασε η αναζήτηση του χρήστη προτού αποχωρήσει), ο ενδιάμεσος χρόνος μεταξύ δύο διαδοχικών επισκέψεων,

κ.α.. Δυστυχώς, επειδή το να μοντελοποιηθούν τέτοιου τύπου μετρικές είναι από μόνο του δύσκολο, η βελτιστοποίησή τους αποτελεί μεγάλη πρόκληση.

Η Ενισχυτική Μάθηση που είναι μια μέθοδος η οποία εξ ορισμού φτιάχτηκε για να μεγιστοποιεί τις μακροπρόθεσμες ανταμοιβές, φαίνεται να είναι μια πολύ καλή λύση για την βελτιστοποίηση των άμεσων και μακροπρόθεσμων μετρικών που κάνουν την εμπειρία του χρήστη πιο ευχάριστη στην εκάστοτε πλατφόρμα. Η εφαρμογή της Ενισχυτικής Μάθησης για το σκοπό αυτό είναι από μόνη της ένα απαιτητικό πρόβλημα. Όπως προαναφέρθηκε, η μοντελοποίηση των μακροπρόθεσμων μετρικών είναι δύσκολη γιατί χρειάζονται πολλές μεταβλητές, ώστε ο συνδυασμός τους να χτίσει ένα αξιόπιστο σύστημα συστάσεων. Συνεπώς, η εκ νέου δημιουργία ενός συστήματος συστάσεων με αυτές τις προδιαγραφές θα ήταν απαγορευτική, αφού προκειμένου να υπολογισθούν μακροπρόθεσμα όλες οι απαραίτητες για την ορθή λειτουργία του συστήματος μεταβλητές, θα υπήρχε ένα χρονικό διάστημα κατά το οποίο η λειτουργία του θα ήταν άστοχη, θα δημιουργούσε κακή εμπειρία στον χρήστη, ή στη χειρότερη περίπτωση θα τον εκνεύριζε και θα τον απομάκρυνε από την εφαρμογή για πάντα. Μια άλλη εναλλακτική, είναι η εκπαίδευση ενός συστήματος συστάσεων offline, χρησιμοποιώντας τα καταγεγραμμένα δεδομένα, μετριάζοντας έτσι το κόστος (σε «χαμένο» χρόνο) που θα προέκυπτε στην προηγούμενη περίπτωση, αν δηλαδή χρησιμοποιούταν μια μέθοδος εκπαίδευσης τύπου trial and error κάνοντας στην αρχή σχεδόν τυχαίες συστάσεις. Δυστυχώς, οι υπάρχουσες μέθοδοι συμπεριλαμβανομένων και της Monte Carlo (MC) [10] [11] και Temporal Difference (TD) [12] έχουν περιορισμούς στην offline εκμάθηση πολιτικής σε πραγματικά συστήματα συστάσεων. Οι μέθοδοι τύπου MC δεν μπορούν να διαχειριστούν μεγάλο όγκο δεδομένων, όπως για παράδειγμα έναν κατάλογο δισεκατομμυρίων διαθέσιμων στοιχείων προς πρόταση, τα οποία δεδομένα υπάρχουν σε ένα πραγματικό σύστημα συστάσεων. Οι μέθοδοι τύπου TD βελτιώνουν την αποδοτικότητα χρησιμοποιώντας τεχνικές bootstrapping [13] στην εκτίμηση, το οποίο ωστόσο συνοδεύεται από ένα άλλο πρόβλημα, που ονομάζεται Deadly Triad [14]. Το πρόβλημα αυτό σχετίζεται με την αστάθεια και την απόκλιση που προκύπτει οποτεδήποτε συνδυάζονται συναρτησεις προσέγγισης, bootstrapping και offline εκπαίδευσης. Δυστυχώς, οι υπερσύγχρονες μέθοδοι στα συστήματα συστάσεων, που έχουν σχεδιαστεί με αρχιτεκτονικές νευρώνων, είναι δεδομένο πως θα αντιμετωπίσουν το Deadly Triad πρόβλημα κατά τη διάρκεια της offline εκπαίδευσης.

Προκειμένου να αποφευχθούν τα ανωτέρω προβλήματα, στο συγκεκριμένο paper προτείνεται το FeedRec, ένα framework βασισμένο στο Reinforcement Learning, το οποίο θα βοηθήσει στη μακροπρόθεσμη σχέση του χρήστη με την πλατφόρμα. Πιο συγκεκριμένα, η ροή αντικειμένων προς σύσταση παρουσιάζεται ως μια Markov Decision Process (MDP) [15] [16] και σχεδιάζεται ένα Q-Network [17] ώστε να βελτιστοποιεί απευθείας τις μετρικές που προσδιορίζουν το «δέσιμο» του χρήστη με την εφαρμογή. Επίσης, για να αποφευχθεί το πρόβλημα της αστάθειας

σύγκλισης στο offline Q-Learning [18], χρησιμοποιείται ένα S-Network, το οποίο προσομοιώνει τα περιβάλλοντα, ώστε να βοηθήσει την εκμάθηση πολιτικής. Στο Q-Network, για να δειχθεί η πληροφορία των πολύπλευρων μακροπρόθεσμων συμπεριφορών του χρήστη, μοντελοποιείται από το LSTM [19] μια αλυσίδα συμπεριφορών χρήστη, η οποία αποτελείται από όλες τις πιθανές συμπεριφορές, όπως το click, το skip, η αναζήτηση, η ταξινόμηση, το dwell, η επιστροφή του χρήστη στην εφαρμογή, κλπ.. Όταν μοντελοποιούνται τέτοιου είδος συμπεριφορές, προκύπτουν δυο προβλήματα: οι αριθμοί για συγκεκριμένες ενέργειες χρηστών είναι αρκετά ανισόρροποι (π.χ. τα clicks είναι πολύ λιγότερα από τα skips) και μακροπρόθεσμη συμπεριφορά του χρήστη είναι πιο πολύπλοκη στην αναπαράσταση. Εφαρμόζεται έτσι ολοκληρωμένο ιεραρχικό LSTM με χρονικό ιστορικό στο Q-Network ώστε να χαρακτηριστούν οι συμπεριφορές χρήστη που εξαρτώνται από την πάροδο του χρόνου.

Από την άλλη μεριά, προκειμένου να χρησιμοποιηθούν αποτελεσματικά τα ιστορικά δεδομένα και να αποδευχθεί το Deadly Triad πρόβλημα στο offline Q-Learning, εισάγεται ένα μοντέλο περιβάλλοντος, που ονομάζεται S-Network, ώστε να προσομοιώσει το περιβάλλον και να παράξει προσομοιωτικά εμπειρείες χρηστών, βοηθώντας την offline εκμάθηση πολιτικής. Στο πλαίσιο αυτού του paper έγιναν εκτενή πειράματα τόσο σε εικονικό σύνολο δεδομένων, όσο και σε πραγματικό e-commerce σύνολο. Τα αποτελέσματα έδειξαν πως ο προτεινόμενος αλγόριθμος έχει εξαιρετικά αποτελέσματα σε σύγκριση με τα υπερσύγχρονα αλλά καθιερωμένα συστήματα που χρησιμοποιούνται για την βελτιστοποίηση της μακροπρόθεσμης σχέσης του χρήστη με την εκάστοτε πλατφόρμα.

Συνοψίζοντας τα όσα θα παρουσιαστούν στη συνέχεια:

1) Προτείνεται ένα Reinforcement Learning framework, το FeedRec, το οποίο στόχο έχει να βελτιστοποιεί απευθείας τη σχέση του χρήστη με την εφαρμογή ροής συστάσεων, τόσο βραχυπρόθεσμα όσο και μακροπρόθεσμα.
2) Γίνεται προσπάθεια μοντελοποίησης των συμπεριφορών χρήστη, συμπεριλαμβανομένης και της άμεσης σχέσης (click,skip κλπ.) αλλά και της μακροπρόθεσμης (dwell time, επαναλαμβανόμενες επισκέψεις, κλπ.), μέσω ενός Q-Network με ιεραρχική αρχιτεκτονική LSTM.
3) Σχεδιάζεται ένα αποτελεσματικό και ασφαλές framework εκπαίδευσης προκειμένου να αποφευχθούν προβλήματα σύγκλισης και αστάθειας.

2.2 ΠΑΡΑΔΟΣΙΑΚΑ ΣΥΣΤΗΜΑΤΑ ΣΥΣΤΑΣΕΩΝ

Τα περισσότερα από τα υπάρχοντα συστήματα συστάσεων προσπαθούν να ισορροπήσουν τις άμεσες μετρικές και παράγοντες όπως η ποικιλία και η καινοτομία στις συστάσεις. Από πλευράς άμεσων μετρικών, έχει γίνει αρκετή δουλειά που συγκεντρώνεται στην βελτίωση του feedback που μπορεί να παρθεί μέσω των clicks του χρήστη, των βαθμολογιών που δίνει και του dwell time που ξοδεύει στα προτεινόμενα αντικείμενα. Στην πραγματικότητα, οι άμεσες μετρικές έχουν «κατηγορηθεί» ως ανεπαρκείς για ασφαλή αποτελέσματα και ότι δεν αναπαριστούν την πραγματική σχέση του χρήστη με την εκάστοτε πλατφόρμα. Σαν συμπλήρωμα, έχουν προταθεί πολλοί αλγόριθμοι που προσπαθούν να διεγείρουν την ευχαρίστηση στην εμπειρεία του χρήστη μέσω της πρότασης διαφορετικών αντικειμένων από αυτά που ήδη εχουν προταθεί. Ωστόσο, όλη αυτη η δουλειά δεν μπορεί να μοντελοποιήσει την επαναληπτική αλληλεπίδραση του χρήστη με την πλατφόρμα. Επιπλέον, κανένας από τους υπάρχοντες αλγορίθμους δεν μπορεί να βελτιστοποιήσει άμεσα τις μακροπρόθεσμες μετρικές που δείχνουν τη σχέση του χρήστη με την πλατφόρμα.

2.3 ΣΥΣΤΗΜΑΤΑ ΣΥΣΤΑΣΕΩΝ ΒΑΣΙΣΜΕΝΑ ΣΤΗΝ ΕΝΙΣΧΥΤΙΚΗ ΜΑΘΗΣΗ

Έχουν προταθεί λύσεις βασισμένες στο contextual bandit [20] για να μοντελοποιήσουν την αλληλεπίδραση με τους χρήστες και να χειριστούν το γνωστό δίλημμα εξερεύνησης νέων αντικειμένων/εκμετάλλευσης των ήδη γνωστών, που υπάρχει πάντα στα online συστήματα συστάσεων. Από τη μία πλευρά, αυτές οι πρακτικές εφαρμογές θεωρούν ως δεδομένο πως τα ενδιαφέροντα των χρηστών παραμένουν τα ίδια ή αλλάζουν με ομαλό τρόπο, πράγμα το οποίο δεν μπορεί να ισχύσει σε ένα πραγματικό e-commerce σύστημα με συνεχή ροή συστάσεων. Από την άλλη πλευρά, ενώ έχει προταθεί να γίνει προσπάθεια να βελτιστοποιηθεί η μακροπρόθεσμη μετρική του χρόνου μεταξύ δύο διαδοχικών επισκέψεων του χρήστη, δεν υπάρχει κάποια συστηματική λύση για τη βελτιστοποίηση μακροπρόθεσμων μετρικών που αφορούν τη σχέση του χρήστη με την πλατφόρμα.

Εκτός από τις βασισμένες σε contextual bandit λύσεις, έχουν προταθεί και πολλές βασισμένες στο Markov Decision Process (MDP) όσον αφορά τη διαδικασία πρότασης αντικειμένων. Σε όλες τις προτεινόμενες λύσεις, δίνεται βάση κυρίως στη βελτίωση της εκτίμησης των άμεσων/βραχυπρόθεσμων μετρικών. Στο συγκεκριμένο paper παρουσιάζεται μια λύση βασισμένη στο MDP που προσπαθεί να βρει την αλλαγή στα ενδιαφέροντα του χρήστη και να βελτιστοποιήσει άμεσα τόσο τις βραχυπρόθεσμες όσο και τις μακροπρόθεσμες μετρικές που προσδιορίζουν τη σχέση χρήστη-πλατφόρμας.

2.4 ΔΙΑΤΥΠΩΣΗ ΠΡΟΒΛΗΜΑΤΟΣ

2.4.1 ΣΥΣΤΑΣΗ ΣΕ ΠΛΑΤΦΟΡΜΑ ΡΟΗΣ ΠΡΟΤΑΣΕΩΝ

Στη σύσταση σε πλατφόρμα ροής πληροφοριών, το σύστημα συστάσεων αλληλεπιδρά με έναν χρήστη u σε διακεκριμένες χρονικές στιγμές. Σε κάθε χρονική στιγμή t, το σύστημα προτείνει ένα αντικείμενο i_t και λαμβάνει ένα feedback f_t από τον χρήστη. Το i_t ανήκει στο σύνολο των αντικειμένων προς σύσταση και το f_t είναι η συμπεριφορά του χρήστη u ως προς το αντικείμενο i_t, όπως το click, η αγορά, το skip, η αποχώρηση από την πλατφόρμα κλπ. Η διαδικασία αλληλεπίδρασης δημιουργεί μια αλληλουχία $X_t = \{u, (i_1, f_1, d_1), \dots, (i_t, f_t, d_t)\}$, όπου το d_t δηλώνει το διάστημα παραμονής του χρήστη στην σύσταση και προσδιορίζει σε έναν βαθμό την προτίμηση του χρήστη ως προς τις συστάσεις. Με δεδομένο το X_t, το σύστημα πρέπει να επιλέξει το αντικείμενο i_{t+1} προς σύσταση, με στόχο να μεγιστοποιήσει το μακροπρόθεσμο «δέσιμο» του χρήστη με την πλατφόρμα (π.χ. τα συνολικά click ή το browsing depth). Στο συγκεκριμένο paper, η μελέτη επικεντρώνεται στη βελτίωση της ποιότητας των αντικειμένων που εμφανίζονται σε μια ροή από αντικείμενα προς σύσταση.

2.4.2 ΤΟ ΠΡΟΒΛΗΜΑ ΩΣ MARKOV DECISION PROCESS (MDP)

Ένα MDP πρόβλημα ορίζεται από το M = [S,A,P,R,γ], όπου S είναι το σύνολο καταστάσεων, A το σύνολο των ενεργειών, P : S x A x S → |R είναι η συνάρτηση μετάβασης, R: S x A → |R είναι η συνάρτηση ανταμοιβής με r(s,a) την άμεση ανταμοιβή του (s,a) και γ ∈ [0, 1] τον συντελεστή έκπτωσης. Μια πολιτική π : S x A → [0,1] αναθέτει σε κάθε κατάσταση s μια κατανομή πιθανών ενεργειών α, όπου η κάθε ενέργεια α, έχει πιθανότητα π(α/s). Στην πλατφόρμα ροής συστάσεων, τα <S, A, P> ορίζονται ως εξής:

- Το S είναι ένα σύνολο από καταστάσεις. Σε κάθε χρονική στιγμή t, το σύνολο καταστάσεων ορίζεται ως $S_t = X_{t-1}$. Αρχικά, ισχύει ότι $S_1 = \{u\}$, δηλαδή περιέχει μόνο τις πληροφορίες του χρήστη. Τη χρονική στιγμή t, $S_t = S_{t-1} + \{(i_{t-1}, f_{t-1}, d_{t-1})\}$. Δηλαδή, σε κάθε βήμα, το σύνολο καταστάσεων εμπλουτίζεται από μια τούπλα του προηγούμενου βήματος, που περίεχει το προτεινόμενο αντικείμενο, το feedback και το dwell time.
- Το A είναι ένα πεπερασμένο σύνολο από ενέργειες. Οι διαθέσιμες πιθανές ενέργειες εξαρτώνται από την κατάσταση s και συμβολίζονται με A(s). Το $A(s_1)$ αρχικοποιείται με όλα τα προς πρόταση αντικείμενα και σε κάθε επόμενο βήμα το $A(s_t)$ ενημερώνεται, αφαιρώντας τα αντικείμενα που προτάθηκαν από το $A(s_{t-1})$ και η ενέργεια α_t είναι το αντικείμενο i_t που προτάθηκε.
- Το P είναι η συνάρτηση μετάβασης με πιθανότητα $p(s_{t+1} \backslash s_t, i_t)$ να υπάρξει η κατάσταση s_{t+1} αν στην κατάσταση s_t επιλεχθεί η πρόταση του αντικειμένου i_t. Εδώ, υπάρχει μια αβεβαιότητα, που όπως είναι φυσικό, προκύπτει από το feedback f_t του χρήστη.

2.4.3 ΑΦΟΣΙΩΣΗ ΧΡΗΣΤΗ ΚΑΙ ΣΥΝΑΡΤΗΣΗ ΑΝΤΑΜΟΙΒΗΣ

Όπως ήδη αναφέρθηκε, σε αντίθεση με τα παραδοσιακά συστήματα συστάσεων, οι άμεσες μετρικές (clicks, αγορές, κλπ.) δεν είναι οι μοναδικές που λαμβάνονται υπόψιν όταν υπολογίζεται η αφοσίωση και η ευχαρίστηση του χρήστη. Μάλιστα, οι μακροπρόθεσμες μετρικές είναι σημαντικότερες. Τέτοιες μετρικές είναι το βάθος αναζήτησης (browsing depth), οι φορές που ο χρήστης επισκέπτεται πάλι την πλατφόρμα, και το διάστημα παραμονής του χρήστη στην πλατφόρμα (dwell time). Το Reinforcement Learning παρέχει έναν τρόπο για να βελτιστοποιούνται τόσο οι άμεσες όσο και οι μακροπρόθεσμες μετρικές, μέσω του σχεδιασμού μιας συνάρτησης ανταμοιβής.

Η συνάρτηση ανταμοιβής R: S x A → |R μπορεί να σχεδιαστεί με διαφορετικούς τρόπους. Εδώ ορίζεται γραμμικά, υποθέτοντας πως η ανταμοιβή όσον αφορά την αφοσίωση του χρήστη $r_t(m_t)$ σε κάθε διακριτή χρονική στιγμή είναι ένα άθροισμα από μετρικές με διαφορετικά βάρη: $r_t = \omega^T$ x m_t, όπου m_t είναι ένας πίνακας-στήλη που αποτελείται από διαφορετικές μετρικές, και ω είναι ο πίνακας με τα βάρη.

Άμεσες μετρικές: Στην άμεση αφοσίωση χρήστη, υπάγονται μετρικές όπως τα clicks, οι αγορές, κλπ. Το κοινό στοιχείο όλων των μετρικών της κατηγορίας αυτής είναι πως μετρώνται άμεσα μόλις ο χρήστης κάνει κάποια ενέργεια. Αν χρησιμοποιηθούν ως παράδειγμα τα clicks, ο αριθμός των clicks στο t-οστό feedback ορίζεται από την μετρική για click m_t^c: m_t^c = #clicks(f_t).

Μακροπρόθεσμες μετρικές: Οι μακροπρόθεσμες μετρικές συμπεριλαμβάνουν το βάθος αναζήτησης, το dwell time στο σύστημα, τις φορές που ο χρήστης επανέρχεται στην πλατφόρμα, κλπ. Τέτοιου είδους παράμετροι μετρώνται από προηγούμενες συμπεριφορές. Παρακάτω δίνονται δύο παραδείγματα από συναρτήσεις ανταμοιβής για μακροπρόθεσμες μετρικές:

Depth Metric: Το βάθος εξερεύνησης είναι μια μετρική που στο συγκεκριμένο σενάριο του feed streaming διαφέρει σε σχέση με άλλους τύπους συστήματος συστάσεων, λόγω του έμφυτου μηχανισμού δυνατότητας προσπέλσης άπειρων αντικειμένων. Αφού εμφανιστεί η t-οστή σύσταση, το σύστημα πρέπει να ανταμοίβει τη σύσταση, αν ο χρήστης παρέμεινε στο σύστημα και έκανε scroll down. Έτσι, η μετρική του βάθους εξερεύνησης m_t^d μπορεί να οριστεί ως εξής: m_t^d = #scans(f_t), όπου #scans(f_t) είναι ο αριθμός των σαρώσεων της t-οστής σύστασης.

Return Time Metric: Ο χρήστης θα χρησιμοποιεί το σύστημα πιο συχνά όταν είναι ικανοποιημένος από το σύστημα συστάσεων. Συνεπώς, ο χρόνος ανάμεσα σε δύο επισκέψεις μπορεί να αντικατοπτρίσει την ικανοποίηση του χρήστη από το σύστημα. Ο χρόνος επιστροφής

$$m_t^r = \frac{\beta}{v^r},$$

Εικόνα 2.1: Τύπος χρόνου επιστροφής

όπου το u^r αναπαριστά το χρόνο ανάμεσα σε δύο επισκέψεις και το β είναι παράμετρος.

Από τα παραπάνω παραδείγματα (clicks, depth metric, return time metric), προκύπτει εύκολα ότι m_t = $[m_t^c, m_t^d, m_t^r]^T$. Σημειώνεται ότι στο μοντέλο MDP, οι σωρευτικές ανταοιβές θα μεγιστοποιηθούν, που σημαίνει ότι στην πραγματικότητα βελτιστοποιούνται το συνολικό browsing depth και η συχνότητα των μελλοντικών επισκέψεων, που είναι ουσιαστικά η μελλοντική αφοσίωση του χρήστη.

2.5 ΕΚΜΑΘΗΣΗ ΠΟΛΙΤΙΚΗΣ ΓΙΑ ΣΥΣΤΗΜΑΤΑ ΣΥΣΤΑΣΕΩΝ

Για να εκτιμηθεί η μελλοντική ανταμοιβή, η αναμενόμενη μακροπρόθεσμη αφοσίωση του χρήστη για σύσταση i_t παρουσιάζεται ως:

$$Q^{\pi}(s_t, i_t) = \mathbb{E}_{i_k \sim \pi}[\underbrace{r_t}_{\text{current rewards}} + \underbrace{\sum_{k=1}^{T-t} \gamma^k r_{t+k}}_{\text{future rewards}}],$$

Εικόνα 2.2: Τύπος υπολογισμού μακροπρόθεσμης αφοσίωσης χρήστη

όπου το γ είναι ο συντελεστής μείωσης που ρόλο έχει να ισορροπήσει τη σημασία των τωρινών ανταμοιβών και των μακροπρόθεσμων. Το βέλτιστο $Q^*(s_t, i_t)$, που έχει τη μέγιστη αναμενόμενη ανταμοιβή που γίνεται να αποκτηθεί από τη βέλτιστη πολιτική, πρέπει να επαληθεύει τη βέλτιστη εξίσωση Bellman:

$$Q^*(s_t, i_t) = \mathbb{E}_{s_{t+1}}\left[r_t + \gamma \max_{i'} Q^*\left(s_{t+1}, i'\right) | s_t, i_t\right].$$

Εικόνα 2.3: Εξίσωση Bellman

Με δοσμένο το Q*, η σύσταση i_t επιλέγεται με το μέγιστο $Q^*(s_t, i_t)$. Παρόλα αυτά, σε συστάσεις που αφορούν real world συστήματα, με πάρα πολλούς χρήστες και αντικείμενα, η εκτίμηση της συνάρτησης $Q^*(s_t, i_t)$ είναι ακατόρθωτη. Αντ' αυτού, είναι πολύ πιο πρακτικό και εύκολο να χρησιμοποιείται προσεγγιστική συνάρτηση, όπως για παράδειγμα κάποιο νευρωνικό δίκτυο, ώστε να εκτιμηθεί η συνάρτηση $Q^*(s_t, i_t)$. Στην πράξη, τα νευρωνικά δίκτυα είναι απολύτως κατάλληλα στο να εντοπίζουν τα ενδιαφέροντα του χρήστη όσον αφορά τις συστάσεις. Σε αυτό το paper, γίνεται αναφορά σε μία προσεγγιστική συνάρτηση βασισμένη σε νευρωνικό δίκτυο με παράμετρο θ_q ως ένα Q-Network. Το Q-Network μπορεί να εκπαιδευτεί ελαχιστοποιώντας τη συνάρτηση απώλειας μέσου τετραγώνου που ορίζεται ως ακολούθως:

$$\begin{aligned} \ell(\theta_q) &= \mathbb{E}_{(s_t, i_t, r_t, s_{t+1}) \sim \mathcal{M}}\left[(y_t - Q(s_t, i_t; \theta_q))^2\right] \\ y_t &= r_t + \gamma \max_{i_{t+1} \in I} Q(s_{t+1}, i_{t+1}; \theta_q), \end{aligned}$$

Εικόνα 2.4: Συνάρτηση απώλειας μέσου τετραγώνου

όπου το $M = \{(s_t, i_t, r_t, s_{t+1})\}$ είναι ένας μεγάλος buffer που αποθηκεύει όλες τις προηγούμενες ροές, από τον οποίο παίρνονται δείγματα ως σύνολα για σύντομες επανεκπαιδεύσεις του συστήματος.

2.5.1 ΤΟ Q-NETWORK

Ο σχεδιασμός του Q-Network είναι πολύ σημαντικός για την απόδοση του συστήματος. Στη μακροπρόθεσμη βελτιστοποίηση της αφοσίωσης του χρήστη, οι συμπεριφορές αλληλεπίδρασης είναι πολύπλευρες (όχι μόνο τα clicks και οι αγορές, αλλά και το dwell time, οι φορές που ο χρήστης επαναχρησιμοποιεί το σύστημα, το skip, κλπ.), κάτι που κάνει τη μοντελοποίηση μη τετριμμένη. Για να βελτιστοποιηθεί αποτελεσματικά μια τέτοια σχέση, πρέπει πρώτα να δοθούν ως δεδομένες τέτοιες πληροφορίες στο Q-Network.

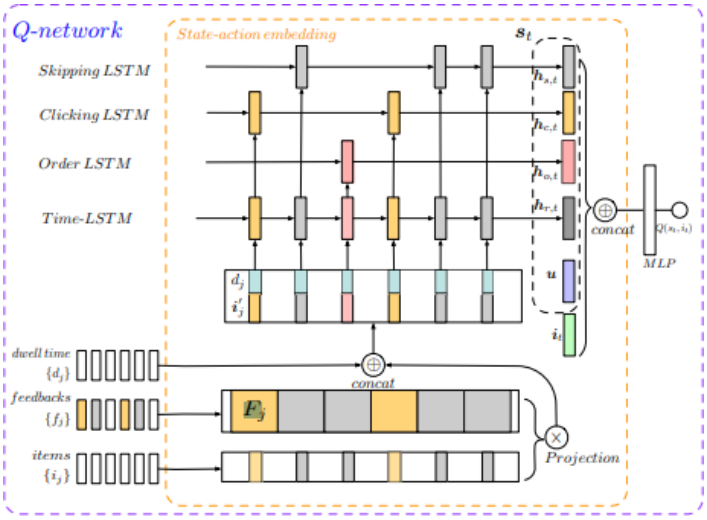
Στρώμα Ενσωμάτωσης Ακατέργαστης πληροφορίας

Ο σκοπός αυτού του στρώματος είναι να πάρει όλη την ακατέργαστη πληροφορία που σχετίζεται με την μακροπρόθεσμη σχέση με τον χρήστη, προς περαιτέρω βελτίωση. Δεδομένης μιας κατάστασης $s_t=\{u,(i_1, f_1, d_1),\ldots, (i_{t-1}, f_{t-1}, d_{t-1})\}$, θεωρείται οτι το f_t είναι όλοι οι πιθανοί τύποι συμπεριφοράς χρηστών στο i_t, συμπεριλαμβανομένων των clicks, των αγορών, του skip, της αποχώρησης από τη σελίδα κλπ. και το d_t το dwell time της συμπεριφοράς.

Ιεραρχικό Στρώμα Συμπεριφοράς

Για να αποτυπωθεί η πληροφορία των πολύπλευρων συμπεριφορών των χρηστών, όλες οι συμπεριφορές εισέρχονται στο Ιεραρχικό Στρώμα Συμπεριφοράς ανεξαιρέτως. Ρεαλιστικά, οι αριθμοί για συγκεκριμένες ενέργειες χρήστη είναι αρκετά άνισοι (π.χ. τα clicks είναι πολύ λιγότερα από τα skips, και οι αγορές είναι πολύ σπανιότερες). Σαν αποτέλεσμα, η απευθείας χρησιμοποίηση της εξόδου του Στρώματος Ενσωμάτωσης Ακατέργαστης Πληροφορίας θα κάνει το Q-Network να χάσει την πληροφορία που αφορά τις σπάνιες συμπεριφορές χρήστη, δηλαδή για παράδειγμα οι πληροφορίες που αφορούν αγορές θα χαθούν εντελώς μέσα στις πληροφορίες που αφορούν τα skips. Επιπλέον, κάθε τύπος συμπεριφοράς χρήστη έχει τα δικά του χαρακτηριστικά: το click σε κάποιο αντικείμενο, συνήθως αναπαριστά τις προτιμήσεις του χρήστη, η αγορά ενός αντικειμένου μπορεί να υποδεικνύει την αλλαγή στα ενδιαφέροντά του, και ο λόγος που κάνει skip μπορεί να είναι λίγο πιο περίπλοκος. Μπορεί να σημαίνει ότι ο χρήστης κάνει απλές αναζητήσεις χωρίς να ενδιαφέρεται για κάτι συγκεκριμένο, ή να είναι ενοχλημένος από κάποια διαφήμιση, κλπ.

Για να αναπαρασταθεί καλύτερα η κατάσταση του χρήστη, όπως φαίνεται και στο παρακάτω σχήμα (Εικόνα 2.5), προτείνεται ένα Ιεραρχικό Στρώμα Συμπεριφοράς που προστίθεται στο Στρώμα Ενσωμάτωσης Ακατέργαστης Πληροφορίας, όπου οι κυριότερες συμπεριφορές χρήστη, όπως τα clicks, τα skips και οι αγορές, αντιμετωπίζονται ξεχωριστά.



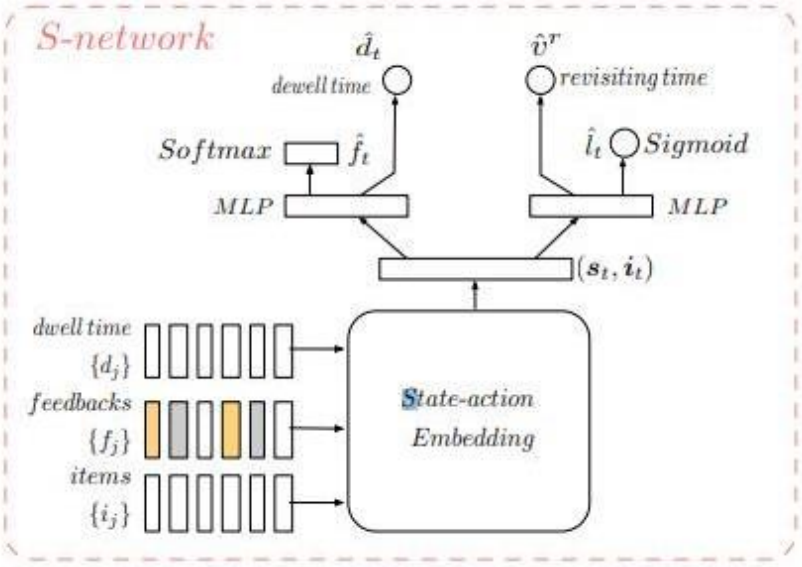
Εικόνα 2.5: Το Q-Network

2.5.2 ΕΚΜΑΘΗΣΗ ΕΚΤΟΣ ΠΟΛΙΤΙΚΗΣ (OFF-POLICY)

Με το προτεινόμενο Q-Learning framework, μπορούν να εκπαιδευτούν οι παράμετροι του μοντέλου μέσω της Trial and Error αναζήτησης [21] προτού βρεθεί μια σταθερή και πιο

αξιόπιστη πολιτική. Ωστόσο, λόγω του κόστους και του ρίσκου που υπάρχει στην ανάπτυξη μη-ικανοποιητικών πολιτικών, είναι σχεδόν αδύνατον να καταλήξει το σύστημα σε κάποια online πολιτική. Ένας εναλλακτικός τρόπος είναι να μάθει μια λογική πολιτική χρησιμοποιώντας τα ιστορικά δεδομένα D, που συλλέγονται από μια πολιτική π_b, πριν την ανάπτυξη πολιτικής. Δυστυχώς, το Q-Learning framework υποφέρει από το πρόβλημα Deadly Triad, δηλαδή το πρόβλημα της αστάθειας και γι'αυτό το λόγο υπάρχει απόκλιση όποτε συνδυάζονται τεχνικές όπως προσεγγιστικές συναρτήσεις, bootstrapping και offline εκπαίδευση.

Για να αποφευχθεί το πρόβλημα της αστάθειας και της απόκλισης στο offline Q-Learning, προτείνεται περαιτέρω ένας προσομοιωτής χρήστη (στον οποίο γίνεται αναφορά ως S-Network), ο οποίος προσομοιώνει το περιβάλλον και βοηθά την εκμάθηση πολιτικής. Ειδικότερα, σε κάθε γύρο συστάσεων, το S-Network (βλ. Εικόνα 2.6), ανταποκρίνεται με πραγματικά user feedbacks, δημιουργώντας την ανταπόκριση f_t του χρήστη, το dwell time d_t του και μια δυαδική μεταβλητή l_t, η οποία καταδεικνύει το αν ο χρήστης αποχωρεί από την πλατφόρμα ή όχι. Όπως φαίνεται στο παρακάτω σχήμα, η δημιουργία προσομοιώσεων ανταποκρίσεων χρηστών, επιτυγχάνεται με τη χρήση του S-Network S (θ_S) που είναι ένα νευρωνικό δίκτυο με πολλούς νευρώνες. Η λογική της κατάστασης-ενέργειας είναι σχεδιασμένη όπως στο Q-Network, αλλά έχει διαφορετικές παραμέτρους. Το στρώμα (s_t, i_t) είναι διαμοιρασμένο σε όλα τα tasks, ενώ τα υπόλοιπα στρώματα (πάνω από το (s_t, i_t) στην παρακάτω φωτογραφία) είναι συγκεκριμένα για κάθε task. Καθώς τα dwell time και user feedback είναι συμπεριφορές που αφορούν το κάθε session.



Εικόνα 2.6: Το S-Network

2.6 ΠΕΙΡΑΜΑΤΙΚΑ ΑΠΟΤΕΛΕΣΜΑΤΑ

Σύγκριση με πιο παραδοσιακά συστήματα: Συγκρίθηκε το FeedRec με πιο παραδοσιακά συστήματα. Τα αποτελέσματα όλων των μεθόδων πάνω σε πραγματικά δεδομένα όσον αφορά τρεις συγκεκριμένες μετρικές φαίνονται παρακάτω:

Agents	Average Clicks per Session	Average Depth per Session	Average Return Time
FM	1.9829	11.2977	16.5349
NCF	1.9425	11.1973	18.2746
GRU4Rec	2.1154	13.8060	14.0268
NARM	2.3030	15.3913	11.0332
DQN	1.8211	15.2508	6.2307
DEER	2.2773	18.0602	5.7363
DDPG-KNN(k=1)	0.6659	9.8127	15.4012
DDPG-KNN(k=0.1N)	2.5569	16.0936	7.3918
DDPG-KNN(k=N)	2.5090	14.6689	14.1648
S-Network	2.5124	16.1745	10.1846
FeedRec(C)	2.6194	18.1204	6.9640
FeedRec(D)	2.8217	21.8328	4.8756
FeedRec(R)	3.7194	23.4582	3.9280
FeedRec(All)	**4.0321***	**25.5652***	**3.9010***

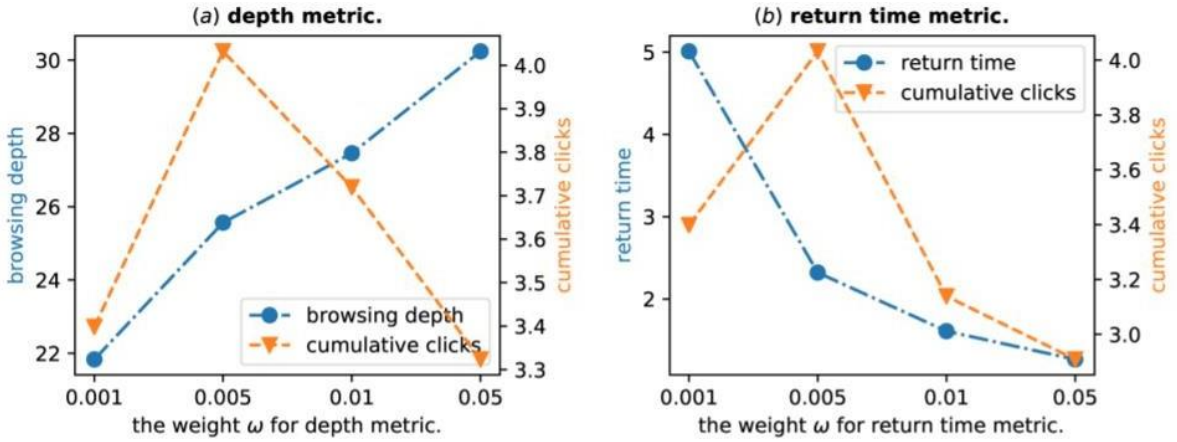
" $*$ " indicates the statistically significant improvements (*i.e.*, two-sided t-test with $p < 0.01$) over the best baseline.

Εικόνα 2.7: Αποτελέσματα πειραμάτων

Από τα αποτελέσματα φαίνεται ξεκάθαρα πως το FeedRec είχε πολύ καλύτερη επίδοση συνολικά από οποιοδήποτε παραδοσιακό σύστημα.

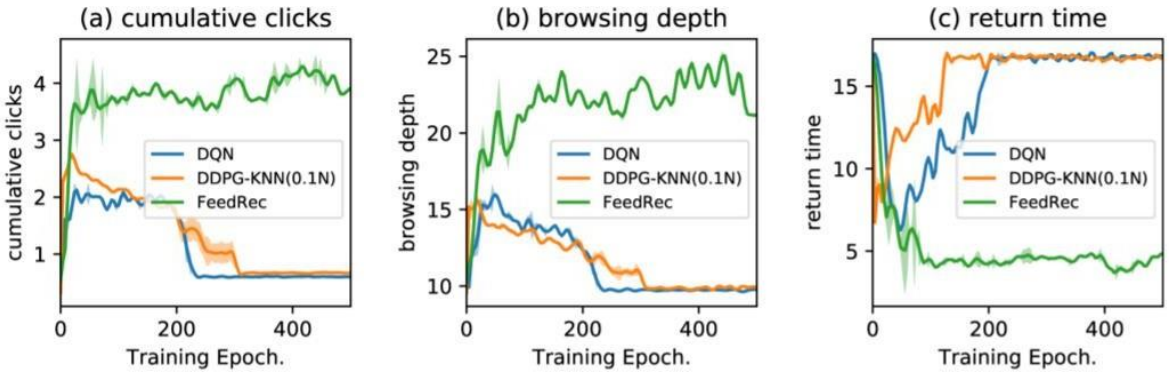
Η επιρροή του ω: Το βάρος ω ελέγχει τη σχετική σημασία των διαφορετικών μετρικών που καταδεικνύουν τη σχέση του χρήστη και της πλατφόρμας ως προς την συνάρτηση ανταμοιβής. Εξετάστηκε η επιρροή των βαρών ω. Στην Εικόνα 2.8, τα (a) και (b) δείχνουν την ευαισθησία των παραμέτρων ως προς το ω, πιο συγκεκριμένα, των μετρικών depth και return time. Φαίνεται πως η αύξηση του βάρους ω για τις δύο αυτές μετρικές, οδηγεί σε περισσότερες αναζητήσεις αντικειμένων και σε αυξάνει τις φορές που ο χρήστης επισκέπτεται τη σελίδα (μπλε γραμμή). Επίσης, τόσο στο (a) όσο και στο (b), το μοντέλο επιτυγχάνει καλύτερα αποτελέσματα στη μετρική που αφορά τα clicks (πορτοκαλί γραμμή) όταν το ω βρίσκεται στο 0.005. Αν αυξηθεί πολύ το βάρος σε αυτές τις μετρικές, θα αυξηθεί αυτομάτως και η επιρροή των clicks στις

ανταμοιβές, που δείχνει ότι μέτριες τιμές βαρών στο depth και στο return, μπορούν να βελτιώσουν αισθητά την απόδοση στα συνολικά clicks.



Εικόνα 2.8: Η επιρροή του ω στην απόδοση

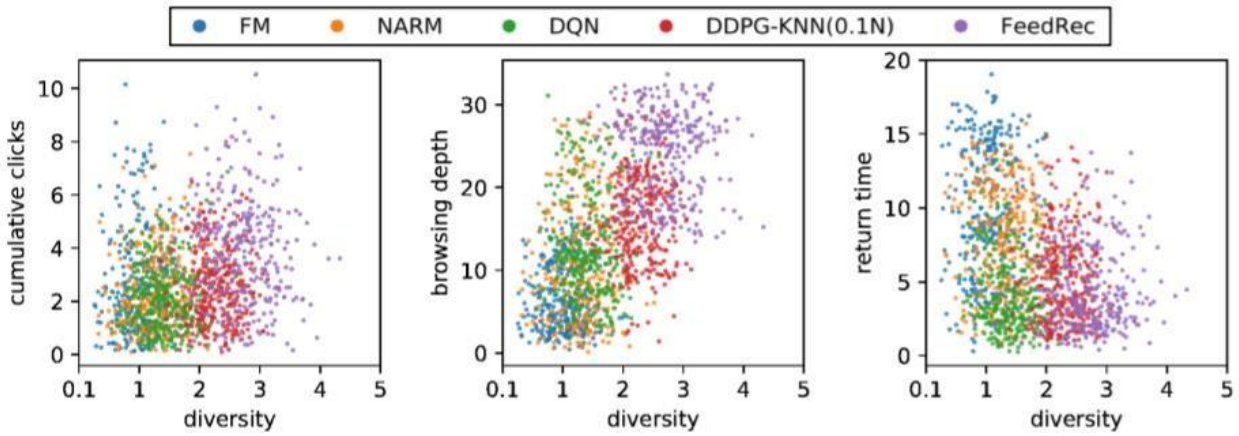
Το γνωστό Deadly Triad Problem προκαλέι τον κίνδυνο της αστάθειας και της απόκλισης στις περισσότερες διαδικασίες εκμάθησης εκτός πολιτικής, ακόμα και στο Q-Learning. Για να εξεταστεί το πλεονέκτημα του προτεινόμενου αλληλεπιδραστικού μοντέλου εκπαίδευσης, έγινε σύγκριση του FeedRec με τα DQN [22], DDPG-KNN [23] με τα ίδια δεδομένα εισόδου. Στην Εικόνα 2.9, φαίνονται διαφορετικές μετρικές με επανάληψη της διαδικασίας εκπαίδευσης. Φαίνεται πως τα DQN και DDPG-KNN επιτυγχάνουν το μέγιστο κοντά στις 40 επαναλήψεις, ενώ οι επιδόσεις μειώνονται σημαντικά, όσο αυξάνονται οι επαναλήψεις (βλ. πορτοκαλί και μπλε γραμμή). Αντιθέτως, το FeedRec επιτυγχάνει καλύτερες επιδόσεις σε αυτές τις τρεις μετρικές και οι επιδόσεις είναι σταθερές στην μέγιστη τιμή τους (βλ. πράσινη γραμμή). Αυτές οι παρατηρήσεις καταδεικνύουν πως το FeedRec είναι σταθερό και κατάλληλο όσον αφορά την αποφυγή του Deadly Triad Problem στην off-policy διαδικασία εκμάθησης πολιτικών για συστάσεις.



Εικόνα 2.9: Σύγκριση DQN, DDPG-KNN, FeedRec

Η σχέση μεταξύ ποικιλίας συστάσεων και ικανοποίησης του χρήστη: Πολλές από τις έρευνες που έχουν γίνει πάνω στα συστήματα συστάσεων σε πλατφόρμες τύπου e-commerce υποθέτουν πως η ικανοποίηση του χρήστη είναι άμεσα συνυφασμένη με την ποικιλία των αντικειμένων προς σύσταση, και μάλιστα με σχέση αναλογική, δηλαδή όσο αυξάνεται η ποικιλία, τόσο μεγαλύτερη είναι η ικανοποίηση του χρήστη. Το πείραμα που έγινε είχε στόχο να δει αν το FeedRec, το οποίο βελτιστοποιεί άμεσα την ικανοποίηση του χρήστη, έχει την ικανότητα να βελτιώσει την ποικιλία των αντικειμένων προς πρόταση. Για κάθε πολιτική, πάρθηκε ένα δείγμα από 300 ζευγάρια κατάστασης-ενέργειας και σχηματίστηκαν τα διαγράμματα της Εικόνας 2.10:

Οι οριζόντιοι άξονες δείχνουν την ποικιλία/διαφορετικότητα μεταξύ των προς σύσταση αντικειμένων, ενώ οι κάθετοι άξονες συμβολίζουν κάποιες μετρικές που δείχνουν την ικανοποίηση του χρήστη (π.χ. βάθος αναζήτησης, χρόνος μεταξύ δύο διαδοχικών επισκέψεων στην πλατφόρμα κλπ.). Φαίνεται πως η πολιτική του FeedRec η οποία μαθαίνεται και βελτιστοποιεί αμέσως την ικανοποίηση του χρήστη, ευνοεί την πρόταση περισσότερων αντικειμένων (βλ. μωβ bullets). Συνεπώς, η βελτιστοποίηση της ικανοποίησης του χρήστη μπορεί να αυξήσει την ποικιλία των προτεινόμενων αντικειμένων, και η διεύρυνση της ποικιλίας των προτεινόμενων αντικειμένων μπορεί να οδηγήσει στην αύξηση της ικανοποίησης του χρήστη.



Εικόνα 2.10: Μεταβολή μετρικών ως προς ποικιλία αντικειμένων προς σύσταση

2.7 ΣΥΝΟΨΗ – ΑΞΙΟΛΟΓΗΣΗ

Είναι σημαντικό να βελτιστοποιείται μακροπρόθεσμα η ικανοποίηση του χρήστη και το δέσιμό του με το σύστημα συστάσεων, ειδικά σε e-commerce περιβάλλοντα. Παρόλο που το Reinforcement Learning ταιριάζει εξ ορισμού στο πρόβλημα αυτό, αφού είναι κύριο χαρακτηριστικό του η βελτιστοποίηση μακροπρόθεσμων ανταμοιβών, υπάρχουν μερικές σημαντικές προκλήσεις στην εφαρμογή του. Αρχικά είναι δύσκολο να μοντελοποιηθούν τα διάφορα feedbacks του χρήστη (clicks, dwell time, χρόνος μεταξύ διαδοχικών επισκέψεων κλπ.), όπως επίσης το να γίνει αποτελεσματικά η off-policy εκμάθηση πολιτικής του συστήματος συστάσεων.

Για να αντιμετωπιστούν τα προβλήματα αυτά, προτάθηκε στο πλαίσιο του συγκεκριμένου paper ένα framework βασισμένο στο Reinforcement Learning, το FeedRec, ώστε να βελτιστοποιήσει τη μακροπρόθεσμη σχέση του χρήστη με το σύστημα. Πρώτον, το FeedRec χρησιμοποιεί ιεραρχικά Q-Networks, τα οποία στόχο έχουν να συνθέσουν τις συμπεριφορές των χρηστών. Στη συνέχεια, για να αποφευχθεί η αστάθεια στη σύγκλιση στην διαδικασία εκμάθησης πολιτικής, σχεδιάστηκε ένα S-Network για να προσομοιώσει το περιβάλλον και να βοηθήσει στο Q-Network.

Φάνηκε πως συγκεκριμένα στα συστήματα συστάσεων για e-commerce πλατφόρμες, η τεχνική εφαρμογής του Reinforcement Learning λειτούργησε με εξαιρετικά αποτελέσματα, αφού στο πειραματικό στάδιο αποδείχθηκε ότι οποιοσδήποτε άλλος πιο παραδοσιακός αλγόριθμος απείχε πολύ από το να ανταγωνιστεί το FeedRec.

ΚΕΦΑΛΑΙΟ 3

ΣΥΣΤΗΜΑΤΑ ΣΥΣΤΑΣΕΩΝ ΓΙΑ ΕΙΔΗΣΕΙΣ

3.1 ΠΑΡΟΥΣΙΑΣΗ ΠΡΟΒΛΗΜΑΤΟΣ

Σε αυτό το κεφάλαιο, θα παρουσιάσουμε ένα Σύστημα Συστάσεων, το οποίο βασίζεται στην Ενισχυτική Μάθηση και φτιάχτηκε για Συστάσεις Ειδήσεων. Λόγω της δυναμικής φύσης των ειδήσεων και των προτιμήσεων των χρηστών, το πρόβλημα αυτό είναι αρκετά απαιτητικό και δύσκολο. Παρόλο που έχουν δημιουργηθεί διάφοροι αλγόριθμοι που μοντελοποιούν αυτό το πρόβλημα, υπάρχουν τρία βασικά ζητήματα:

1. Όλοι οι αλγόριθμοι λαμβάνουν υπόψιν μόνο την τρέχουσα ανταμοιβή (CTR).

2. Σε πολύ λίγες έρευνες λαμβάνουν υπόψιν κάποιο άλλο feedback από τους χρήστες, πέρα από την επιλογή ή την απόρριψη ενός αντικειμένου. Για παράδειγμα, το πόσο συχνά ένας χρήστης επιστρέφει στην εφαρμογή είναι μια παράμετρος που θα βοηθούσε να βελτιωθεί το σύστημα.

3. Αυτοί οι αλγόριθμοι συνηθίζουν να προτείνουν παρόμοιες ειδήσεις στους χρήστες, πράγμα το οποίο μπορεί να τους κάνει να κουραστούν και να βαρεθούν τη χρήση της εφαρμογής.

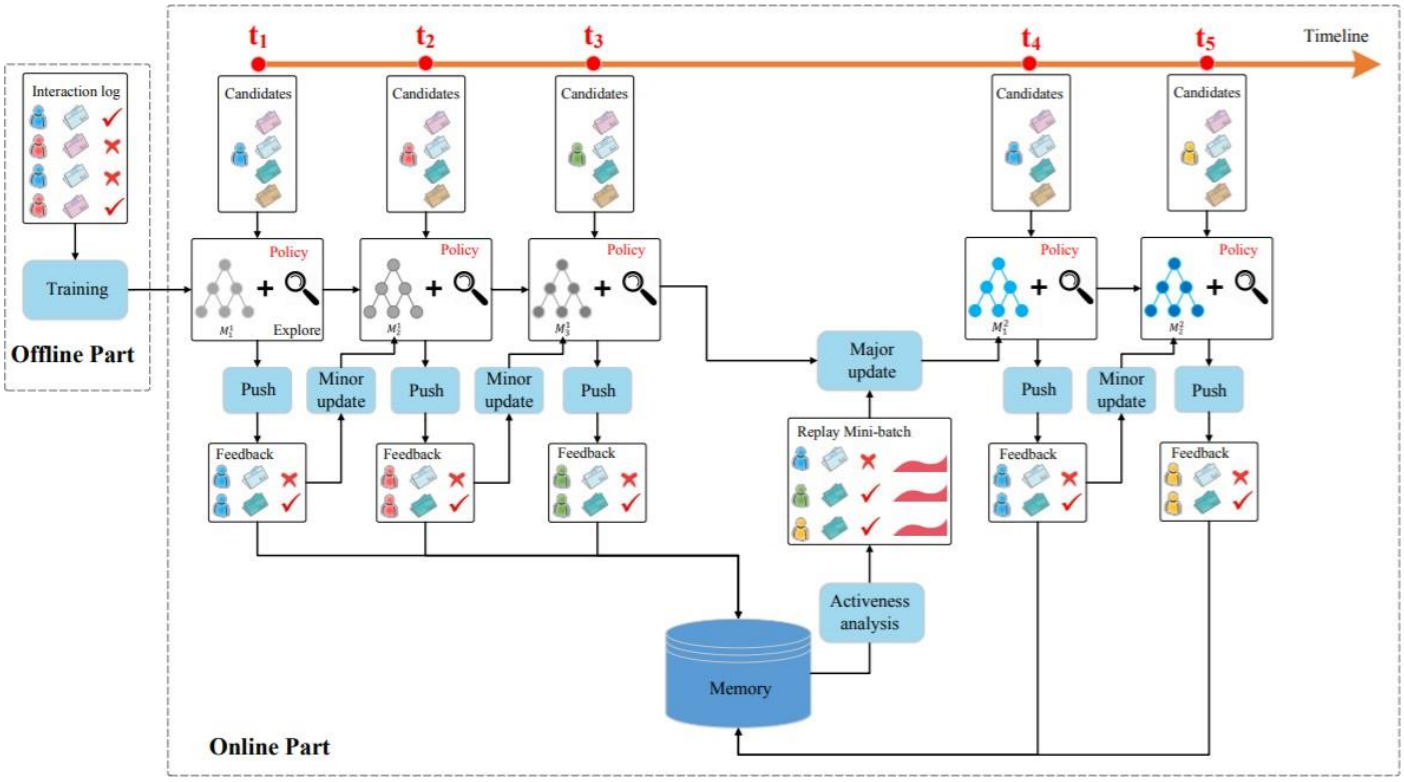
Στην παρούσα υλοποίηση [24] προτείνεται ένας αλγόριθμος που χρησιμοποιεί τεχνική Q-Learning, με τη βοήθεια της οποίας μπορεί να μοντελοποιηθεί η μελλοντική ανταμοιβή του χρήστη. Επιπλέον, λαμβάνεται υπόψιν η συχνότητα επιστροφής του χρήστη στην εφαρμογή ως συμπληρωματικό στοιχείο της επιλογή ή της απόρριψης ενός αντικειμένου για το feedback. Τέλος, ενσωματώνεται μια αποτελεσματική τεχνική εξερεύνησης, η οποία βοηθάει στην εύρεση νέων ελκυστικών ειδήσεων προκειμένου να προταθούν στον χρήστη.

3.2 ΥΛΟΠΟΙΗΣΗ

Προκειμένου να αντιμετωπιστούν τα τρία προβλήματα που αναφέραμε στην παράγραφο 3.1, προτείνεται η χρήση ενός Deep Reinforcement Learning συστήματος βασισμένο στο Deep Q-Network (DQN) [22] για online εξατομικευμένη σύσταση ειδήσεων. Ειδικότερα χρησιμοποιήθηκε μια συνεχής αναπαράσταση των χρηστών και των αντικειμένων ως είσοδο σε ένα πολυεπίπεδο Q-Network [17] που στόχο έχει να προβλέψει την ανταμοιβή και, συνεπώς, το

αν ο χρήστης θα επιλέξει κάποια συγκεκριμένη είδηση. Αρχικά, αυτό το σύστημα μπορεί να ανταποκριθεί στην δυναμική φύση των ειδήσεων, λόγω του online update που γίνεται συνεχώς στο DQN. Παράλληλα, το DQN λειτουργεί διαφορετικά σε σχέση με άλλες online μεθόδους, αφού έχει τη δυνατότητα να προσεγγίσει τη μελλοντική αλληλεπίδραση ανάμεσα στο χρήστη και τις ειδήσεις. Δεύτερον, σε συνδυασμό με την πάντα αξιοσημείωτη και σημαντική μετρική της επιλογής ή απόρριψης ενός αντικειμένου,προτείνεται η χρήση του User-activeness. Το User-activeness είναι μια μετρική που δείχνει τον αριθμό των χρηστών που είχαν αλληλεπίδραση με ένα συγκεκριμένο αντικείμενο σε κάποια ορισμένη χρονική περίοδο (π.χ. ώρα, μέρα, μήνας, ...). Αυτό μπορεί για παράδειγμα να αναπαρασταθεί από τη συχνότητα επιστροφής του χρήστη στην εφαρμογή μετά από κάποια σύσταση που του έγινε. Τρίτον, χρησιμοποιείται η τεχνική εξερεύνησης Dueling Bandit Gradient Descent [25], ώστε οι συστάσεις να παραμένουν αξιόπιστες και καίριες σε αντίθεση με αλγορίθμους όπως ο ε-greedy [26] και ο Upper Confidence Bound [27].

3.2.1 ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΥΛΟΠΟΙΗΣΗΣ



Εικόνα 3.1: Αρχιτεκτονική μοντέλου

Όπως φαίνεται στην Εικόνα 3.1, το μοντέλο μας αποτελείται από ένα online και ένα offline μέρος. Στο offline μέρος αντλούνται τέσσερα χαρακτηριστικά από τους χρήστες και από τις ειδήσεις, τα οποία θα αναλυθούν στην επόμενη παράγραφο (3.2.2). Με βάση αυτά τα τέσσερα χαρακτηριστικά, ένα Q-Network πολλαπλών επιπέδων προσπαθεί να προβλέψει την ανταμοιβή. Όσον αφορά το online μέρος, εκεί το σύστημα αλληλεπιδρά με τους χρήστες και θα ενημερώνει το δίκτυο με τον παρακάτω τρόπο:

1. **Push:** σε κάθε χρονική στιγμή (t_1, t_2, …), που ο χρήστης στέλνει αίτημα για ειδήσεις στο σύστημα, αυτό παίρνει ως είσοδο τα χαρακτηριστικά του χρήστη και των διαθέσιμων ειδήσεων και παράγει μια λίστα με τις k-καλύτερες ειδήσεις για να προτείνει. Αυτές οι k-καλύτερες ειδήσεις προκύπτουν ως συνδυασμός της εκμετάλλευσης του δεδομένου μοντέλου και της εξερεύνησης νέων αντικειμένων.

2. **Feedback:** Ο χρήστης που λαμβάνει τη λίστα με τις προτεινόμενες ειδήσεις παρέχει ένα feedback με τα clicks του.

3. **Minor Update:** Μετά από κάθε χρονική στιγμή, δηλαδή μετά από κάθε σύσταση το σύστημα θα ενημερώνεται συγκρίνοντας την απόδοση του δικτύου Q (exploitation network) που αναπαριστά το μοντέλο με τις ήδη γνωστές ειδήσεις και του δικτύου Q’ (exploration network) που αναπαριστά τις νέες ειδήσεις που ανακάλυψε το σύστημα. Αν το Q’ δίνει καλύτερα αποτελέσματα, τότε το Q θα ενημερωθεί σύμφωνα με το Q’, διαφορετικά το Q θα μείνει ως έχει.

4. **Major Update:** Μετά από μια συγκεκριμένη χρονική περίοδο (π.χ. κάθε τρεις χρονικές στιγμές), το σύστημα θα χρησιμοποιεί το feedback και το user activeness που θα έχει αποθηκεύσει στη μνήμη, ώστε να ενημερωθεί το δίκτυο Q. Πιο συγκεκριμένα, το σύστημα διατηρεί μια μνήμη όπου αποθηκεύονται το ιστορικό των clicks και στοιχεία για το user activeness. Το major update γίνεται συνήθως ανά προκαθορισμένο χρονικό διάστημα, όπως για παράδειγμα κάθε μια ώρα, ώστε να έχει συλλεχθεί επαρκής όγκος δεδομένων για την επίδραση των συστάσεων στους χρήστες.

3.2.2 ΚΑΤΑΣΚΕΥΗ ΧΑΡΑΚΤΗΡΙΣΤΙΚΩΝ

Για να μπορέσει να γίνει σωστή πρόβλεψη για το αν ο χρήστης θα επιλέξει μια συγκεκριμένη διαφήμιση, χρειάζονται και παράγονται οι εξής τέσσερις κατηγορίες χαρακτηριστικών:

1. **Χαρακτηριστικά Ειδήσεων:** περιέχει 417 διαφορετικά χαρακτηριστικά που αφορούν την είδηση, όπως τίτλος, πάροχος, αξιολόγηση, κατηγορία είδησης και μετρητής των click για την τελευταία ώρα, εξάωρο, εικοσιτετράωρο, εβδομάδα, χρόνο.

2. **Χαρακτηριστικά Χρηστών:** κυρίως περιέχει τα χαρακτηριστικά ειδήσεων για τα οποία ενδιαφέρθηκε (έκανε click) ο χρήστης την τελευταία ώρα, εξάωρο, εικοσιτετράωρο, εβδομάδα, χρόνο. Συνεπώς, περιέχει $417 \times 5 = 2065$ διαφορετικές πληροφορίες.

3. **Χαρακτηριστικά Ειδήσεων-Χρηστών:** αυτά είναι 25 διαφορετικά χαρακτηριστικά που περιγράφουν την αλληλεπίδραση μεταξύ ενός χρήστη και ενός συγκεκριμένου είδους ειδήσεων. Για παράδειγμα, η συχνότητα με την οποία ένα είδος ειδήσεων εμφανίζεται στο ιστορικό ενός χρήστη.

4. **Χαρακτηριστικά Περιβάλλοντος:** περιέχουν 32 διαφορετικά χαρακτηριστικά σχετικά με το περιβάλλον όταν συμβαίνει κάποια αίτηση για ειδήσεις, όπως ο χρόνος ανταπόκρισης, η μέρα και το πόσο πρόσφατη είναι η είδηση.

3.3 ΣΥΣΤΑΣΗ ΜΕ ΒΑΘΙΑ ΕΝΙΣΧΥΤΙΚΗ ΜΑΘΗΣΗ (DEEP REINFORCEMENT LEARNING)

Προκειμένου να μοντελοποιηθεί η πιθανότητα ένας χρήστης να επιλέξει ένα συγκεκριμένο είδος ειδήσεων και κατ' επέκταση τη μελλοντική ανταμοιβή, χρησιμοποιείται ένα Deep Q-Network (DQN), πάντα λαμβάνοντας υπόψιν τα δυναμικά χαρακτηριστικά που αναφέραμε παραπάνω. Υπό το πρίσμα της Ενισχυτικής Μάθησης, η πιθανότητα ενός χρήστη να επιλέξει κάποια είδηση, αντιστοιχίζεται στην ανταμοιβή που θα μπορούσε να πάρει το σύστημά μας. Έτσι, η συνολική ανταμοιβή μπορεί να μαθηματικοποιηθεί με τον εξής τύπο:

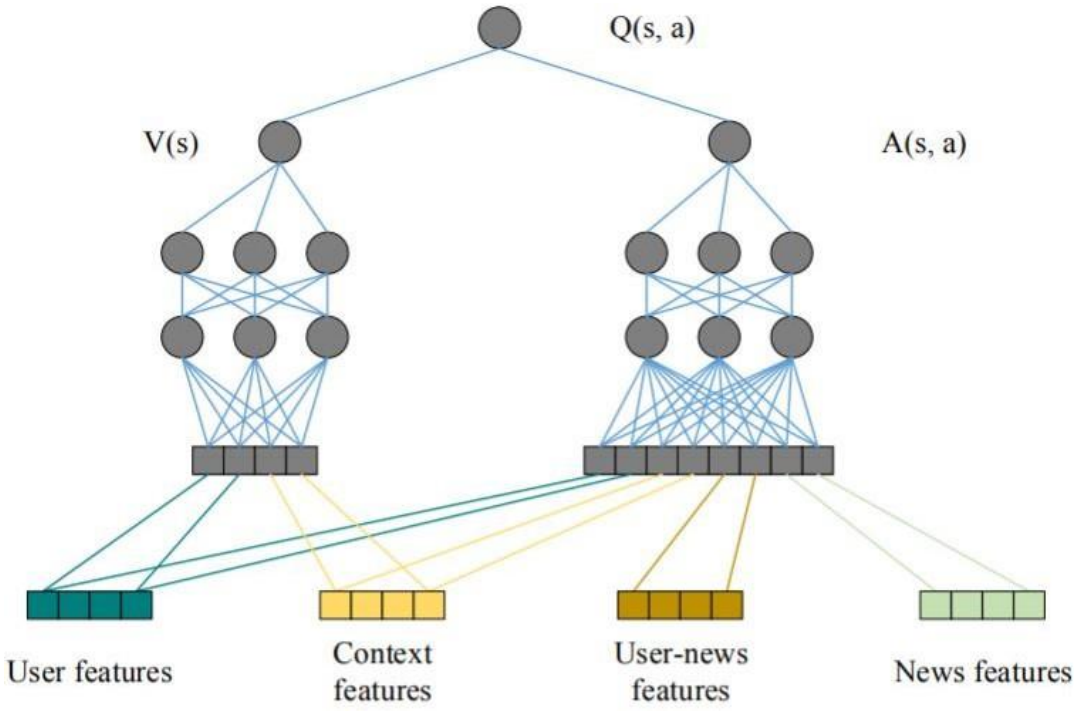
$$y_{s,a} = Q(s,a) = r_{immediate} + \gamma \times r_{future}$$

Όπου η κατάσταση s αναπαρίσταται από τα χαρακτηριστικά περιβάλλοντος και τα χαρακτηριστικά των χρηστών. Αντίστοιχα η ενέργεια a αναπαρίσταται από τα χαρακτηριστικά ειδήσεων και τα χαρακτηριστικά ειδήσεων-χρηστών. Το $r_{immediate}$ αναπαριστά τις ανταμοιβές για την τρέχουσα κατάσταση και το r_{future} την πρόβλεψη για τη μελλοντική ανταμοιβή. Τέλος, το γ αποτελεί μια σταθερά που ορίζεται από τους διαχειριστές του συστήματος, ώστε να ρυθμίζει την βαρύτητα της πρόβλεψης σε σχέση με την άμεση ανταμοιβή.

Ειδικότερα, αν από μια τρέχουσα κατάσταση s, πραγματοποιηθεί η ενέργεια a, τη χρονική στιγμή t, τότε η συνολική ανταμοιβή υπολογίζεται από την εξίσωση:

$$y_{s,a,t} = r_{a,t+1} + \gamma \times Q(s_{a,t+1}, \arg\max_{a'} Q(s_{a,t+1}, a'; W_t); W'_t),$$

όπου το $r_{a,t+1}$ αναπαριστά την ανταμοιβή για την τρέχουσα κατάσταση, επιλέγοντας την ενέργεια a. Το t+1 συμβολίζει απλώς την καθυστέρηση, καθώς η ανταμοιβή αργεί πάντα μια χρονικη στιγμή. Τα W, W' αναπαριστούν δύο διαφορετικά σύνολα παραμέτρων του DQN. Το σύστημά μας θα υπολογίσει την επόμενη κατάσταση $s_{a,t+1}$ με δεδομένο ότι έχει επιλεγεί η ενέργεια a. Με βάση αυτά, δεδομένου ενός συνόλου πιθανών ενεργειών {a'}, επιλέγεται η ενέργεια a' που δίνει τη μέγιστη μελλοντική ανταμοιβή σύμφωνα με τις παραμέτρους W_t'. Τα W_t, W_t' εναλλάσσονται ανά κάποιες επαναλήψεις. Αυτή η στρατηγική έχει αποδειχθεί ότι περιορίζει στο ελάχιστο τις υπερβολικά αισιόδοξες προβλέψεις του Q-Network.



Εικόνα 3.2: Το Q-Network

Στην Εικόνα 3.2 βλέπουμε ένα διάγραμμα που απεικονίζει τη λειτουργία του Q-Network. Όπως φαίνεται, δίνουμε τις τέσσερις κατηγορίες χαρακτηριστικών στο δίκτυο. Τα **Χαρακτηριστικά Χρήστη** και τα **Χαρακτηριστικά Περιβάλλοντος** χρησιμοποιούνται ως χαρακτηριστικά κατάστασης (s), ενώ τα **Χαρακτηριστικά Χρήστη-Ειδήσεων** και τα **Χαρακτηριστικά Ειδήσεων**

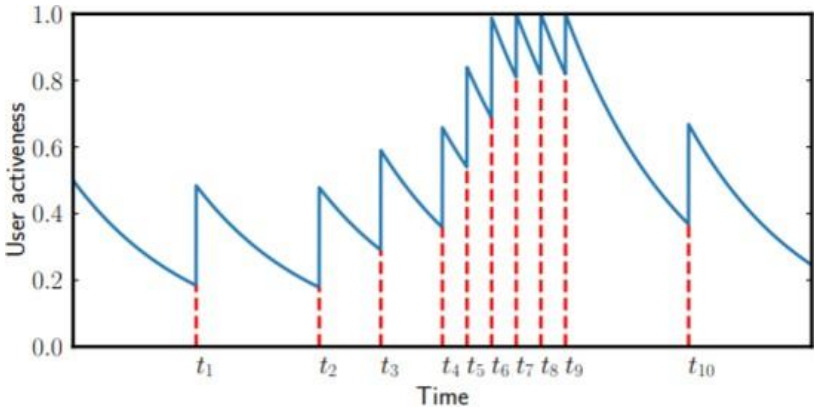
χρησιμοποιούνται ως χαρακτηριστικά ενέργειας (a). Από τη μία πλευρά, η ανταμοιβή που παίρνουμε όταν επιλέγουμε την ενέργεια a σε μια συγκεκριμένη κατάσταση s είναι στενά συνδεδεμένη με όλα τα χαρακτηριστικά. Από την άλλη πλευρά, η ανταμοιβή που καθορίζεται απο τα χαρακτηριστικά του χρήστη και μόνο (π.χ. το αν είναι ενεργός, το αν ο εν προκειμένω χρήστης έχει διαβάσει αρκετές ειδήσεις σήμερα κ.α.) επηρεάζεται από τις καταστάσεις του χρήστη και του περιβάλλοντος και μόνο. Με βάση αυτά, διαιρούμε την Q-function σε δύο υποσυναρτήσεις V(s) και A(s,a). Η V(s) εξαρτάται μόνο από τα χαρακτηριστικά κατάστασης, ενώ η A(s,a) εξαρτάται και από τα χαρακτηριστικά κατάστασης και από τα χαρακτηριστικά ενέργειας.

3.4 USER ACTIVENESS

Τα συνηθισμένα Συστήματα Συστάσεων εστιάζουν στην κατά το δυνατόν βελτίωση των μετρικών CTR, όπως η επιλογή/απόρριψη. Αυτό, όπως έχουμε ήδη επισημάνει, αποτελεί μόνο ένα μέρος από το συνολικό feedback που είμαστε σε θέση να συλλέξουμε από τους χρήστες. Η απόδοση του Συστήματος Συστάσεων επηρεάζει και το αν οι χρήστες θα θελήσουν μελλοντικά να ξαναχρησιμοποιήσουν την εφαρμογή. Αν γίνονται σωστές και εύστοχες συστάσεις, θα αυξήσουν την συχνότητα αλληλεπίδρασης των χρηστών με την εφαρμογή. Για αυτόν τον λόγο, η αλλαγή του user activeness θα έπρεπε να μελετάται και να λαμβάνεται σοβαρά υπόψιν.

Ο χρήστης κάνει αίτημα για ειδήσεις σύμφωνα με ένα καθόλου ομοιόμορφο μοτίβο. Συνήθως διαβάζουν ειδήσεις για ένα σύντομο χρονικό διάστημα, κατά το οποίο είτε θα κάνουν αίτημα για ειδήσεις είτε θα επιλέξουν κάποια είδηση, με μεγάλη συχνότητα. Μετά θα αποχωρήσουν από την εφαρμογή και θα ξαναγυρίσουν σε αυτήν μετά από κάποιες ώρες όταν θα θέλουν να ξαναδιαβάσουν τα νέα. Σαν επιστροφή θεωρούμε τη στιγμή που ο χρήστης θα κάνει αίτημα για ειδήσεις, αφού πάντα ο χρήστης κάνει αίτημα για ειδήσεις, πρωτού επιλέξει κάποια είδηση. Φυσικά, ενώ μελετάμε το user activeness δεν πρέπει να παραμελούμε την επιλογή/απόρριψη γιατί αυτή παραμένει ίσως η σπουδαιότερη παράμετρος που πρέπει να εξετάζεται.

Επειδή η ανάγνωση ειδήσεων είναι κάτι καθημερινό, ορίζουμε μια περίοδο T = 24 ώρες και έναν δείκτη S(t) που αυξάνεται όσο ο χρήστης επανέρχεται στην εφαρμογή με αιτήματα για ειδήσεις. Πιό συγκεκριμένα, κάθε φορά που επιστρέφει ο δείκτης αυξάνεται κατά σταθερό αριθμό S_a κι έτσι η τιμή διαμορφώνεται ως εξής: $S(t) = S(t) + S_a$ για τον συγκεκριμένο χρήστη.



Εικόνα 3.3: Διάγραμμα User Activeness-Χρόνου

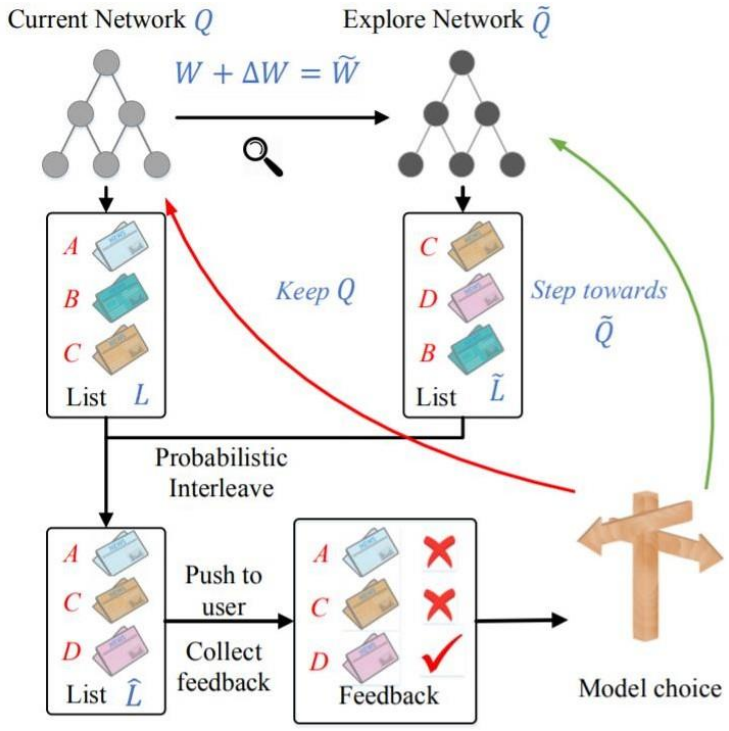
Όπως φαίνεται στην Εικόνα 3.3, η τιμή του S δεν ξεπερνά ποτέ το 1. Οι χρονικές στιγμές t_1, t_2, … είναι οι στιγμές που ο χρήστης επανέρχεται με νέα αιτήματα. Βλέπουμε ότι στις στιγμές αυτές υπάρχει αύξηση της τιμής κατά S_a, αλλά ανάμεσα σε αυτές η τιμή μειώνεται. Επίσης, στις στιγμές από t_4 έως t_9 που ο χρήστης επανέρχεται πολύ συχνά, η τιμή δεν ξεπερνά ποτέ το 1. Στην εφαρμογή μας, το λ, που είναι η παράμετρος μείωσης του S(t) ορίζεται στο 1.2×10^{-5} $second^{-1}$, ενώ το S_a στο 0.32.

Η πρόβλεψη για επιλογή/απόρριψη και αυτή για το user activeness συνδυάζονται με την εξής σχέση:

$$r_{total} = r_{click} + \beta \times r_{active}$$

3.5 ΕΞΕΡΕΥΝΗΣΗ

Οι πιο συνηθισμένες και άμεσες τεχνικές για εξερεύνηση (exploration) στην Ενισχυτική Μάθηση είναι μέσω των αλγορίθμων ε-greedy και UCB [28]. Ο ε-greedy προτείνει τυχαία αντικείμενα στον χρήστη με πιθανότητα ε, ενώ ο UCB διαλέγει αντικείμενα που δεν έχουν εξερευνηθεί αρκετά, στη λογική πως εκεί θα βρει μεγάλη ποικιλία. Όταν όμως πρόκειται για Συστήματα Συστάσεων είναι εύκολα κατανοητό πως οι τεχνικές αυτές θα βλάψουν την απόδοση του συστήματος, καθώς εδώ απαιτείται βραχυπρόθεσμη βελτίωση και επιτυχής πρόταση αντικειμένων στον χρήστη. Για αυτόν τον λόγο, αντί του random exploration [29] εφαρμόζεται ένας Dueling Bandit Gradient Descent αλγόριθμο [25] για την εξερεύνηση.



Εικόνα 3.4: Λειτουργία Q-Network για εξερεύνηση νέων αντικειμένων

Σύμφωνα με την Εικόνα 3.4, το σύστημά μας παράγει μια λίστα L με αντικείμενα διαθέσιμα προς σύσταση χρησιμοποιώντας το διαθέσιμο Q-Network και μια δεύτερη λίστα L' χρησιμοποιώντας ένα δίκτυο εξερεύνησης Q'. Οι παράμετροι W' του Q' μπορούν να παρθούν αν προσθέσουμε ένα μικρό ΔW στις υπάρχουσες παραμέτρους W του Q, ως εξής:

$$\Delta W = \alpha \times \text{rand}(-1,1) \times W$$

Το α είναι ο συντελεστής εξερεύνησης και η rand μια συνάρτηση που διαλέγει τυχαίο αριθμό ανάμεσα στο -1 και το 1. Στη συνέχεια το σύστημα θα παράξει μια λίστα L" από τις λίστες L και L'. Αρχικά επιλέγει τυχαία μια από τις λίστες L, L'. Έστω πως επιλέγει την L. Στη συνέχεια θα επιλέξει ένα στοιχείο της L με μια πιθανότητα επιλογής που εξαρτάται από τη θέση που έχει το στοιχείο στην αρχική λίστα L (τα στοιχεία που είναι πιο ψηλά έχουν μεγαλύτερη πιθανότητα να επιλεχθούν). Όταν η λίστα L" είναι έτοιμη, θα προταθεί στον χρήστη και το σύστημα θα πάρει ένα feedback. Εάν τα στοιχεία που προτάθηκαν στον χρήστη από το Q' πάρουν καλύτερο feedback, τότε το Q θα ενημερωθεί σύμφωνα με το Q' με τις παραμέτρους W' να ενημερώνονται σύμφωνα με τη συνάρτηση:

$$W'' = W + n \times W'.$$

Διαφορετικά, το Q θα παραμείνει ως έχει. Με αυτόν τον τρόπο εξασφαλίζουμε ότι παράλληλα με την εξερεύνηση, το Σύστημα Συστάσεων μας δε θα χάνει την αποδοτικότητα του, καθώς η λίστα που προτείνεται στον χρήστη δεν αποτελείται μόνο από αντικείμενα που βρέθηκαν μέσω εξερεύνησης.

3.6 ΣΥΝΟΨΗ – ΑΞΙΟΛΟΓΗΣΗ

Στο κεφάλαιο αυτό, προτάθηκε ένας αλγόριθμος Ενισχυτικής Μάθησης βασισμένος στα Deep Q-Networks για να κάνει online και εξατομικευμένη Σύσταση Ειδήσεων. Παλιότεροι και πιο κλασικοί αλγόριθμοι αποτυγχάνουν στο να ανταποκριθούν αποτελεσματικά στη δυναμική φύση των ειδήσεων σε συνδυασμό και με τη διαφορετική προτίμηση των χρηστών. Ο αλγόριθμος αυτός όχι μόνο απέδειξε με πειράματα ότι μπορεί να ανταποκριθεί σε αυτό, αλλά λόγω της Ενισχυτικής Μάθησης καταφέρνει να βελτιώνεται μακροπρόθεσμα και να εξασφαλίζει καλύτερη ανταμοιβή (clicks).

ΚΕΦΑΛΑΙΟ 4

ΣΥΣΤΗΜΑΤΑ ΣΥΣΤΑΣΕΩΝ ΓΙΑ ΜΟΥΣΙΚΗ

4.1 ΠΑΡΟΥΣΙΑΣΗ ΠΡΟΒΛΗΜΑΤΟΣ

Η μουσική είναι μία από τα πιο διαδεδομένες εκφράσεις της ανθρώπινης κουλτούρας. Έχει συνοδεύσει τον άνθρωπο κατά το πέρασμα των χρόνων και το άκουσμά της είναι μια από τις πιο κοινές ασχολίες. Όταν κανείς ακούει μουσική, συνήθως ακούει μια σειρά από τραγούδια, και όχι ένα συγκεκριμένο τραγούδι σε επανάληψη. Κάτι σημαντικό είναι πως έχει αποδειχθεί ότι η μουσική είναι βιώσιμη ανάλογα με το χρονικό πλαίσιο μέσα στο οποίο την ακούς. Αυτό σημαίνει αφενός ότι ένα τραγούδι επηρεάζει με διαφορετικό τρόπο ανάλογα με το πότε θα το ακούσει κανείς και με τη διάθεση στην οποία βρίσκεται τότε. Αφετέρου, δείχνει ότι η ευχαρίστηση που μπορεί κάποιος να πάρει ακούγοντας ένα τραγούδι, επηρεάζεται άμεσα από την θέση στην οποία βρίσκεται το τραγούδι μέσα σε μια σειρά από τραγούδια (playlist). Σύμφωνα με αυτήν την παραδοχή κατασκευάζουν τις playlists οι DJ, και όντως, η έρευνα που έχει γίνει για την αυτόματη κατασκευή playlist έχει φανεί ότι παράγει πολύ αρεστά αποτελέσματα. Παρόλα αυτά, η δουλειά που έχει γίνει δεν αφορά την κατασκευή εξατομικευμένων και προσαρμοσμένων playlist στις προτιμήσεις του εκάστοτε χρήστη.

Όσον αφορά τα συστήματα συστάσεων, η μουσική είναι ένας τομέας που έχει ιδιαίτερο ενδιαφέρον, τόσο ακαδημαϊκά, όσο και εμπορικά. Η έρευνα στον τομέα των συστάσεων έχει επικεντρωθεί στην πρόβλεψη της προτίμησης των χρηστών σε συγκεκριμένα τραγούδια και όχι σε playlists.

Γενικά, έχει γίνει μικρή προσπάθεια στο να συσχετιστούν η μάθηση των προτιμήσεων ενός χρήστη με τη δημιουργία ολιστικών λιστών αναπαραγωγής. Στο paper που θα μελετήσουμε [30] σε αυτό το κεφάλαιο, η προσπάθεια συγκεντρώθηκε στο να καλυφθεί αυτό το κενό και στην παρουσίαση του DJ-MC, ενός νέου πλαισίου για προσαρμοσμένη και εξατομικευμένη σύσταση λιστών μουσικής. Στο DJ-MC το πρόβλημα σύστασης λιστών μουσικής αντιμετωπίζεται ως ένα σειριακό πρόβλημα λήψης αποφάσεων, ενώ χρησιμοποιούνται και εργαλεία από το reinforcement learning ώστε να μελετώνται οι προτιμήσεις του χρήστη σε τραγούδια, αλλά και μεταβάσεις από τραγούδι σε τραγούδι ή από είδος μουσικής σε κάποιο άλλο είδος μουσικής. Τα θέματα που μελετήθηκαν στο συγκεκριμένο paper είναι τα εξής:

1. Αρχικά διατυπώνεται το πρόβλημα επιλογής μιας λίστας αναπαραγωγής ως μια Markov Decision Process και προτείνεται η χρήση του Reinforcement Learning ως μια πιθανώς αποτελεσματική προσθήκη στην διαδικασία.

2. Στη συνέχεια ελέγχεται αν ισχύει η υπόθεση ότι η λίστα αναπαραγωγής έχει σημαντική επιρροή στην εμπειρία του χρήστη.

3. Τέλος, δείχνεται εμπειρικά ότι το DJ-MC μπορεί να αναλάβει την τοποθέτηση των τραγουδιών σε σειρά (δημιουργία λίστας) και μάλιστα με καλύτερα αποτελέσματα από συστάσεις που βασίζονται αυστηρά στις ατομικές προτιμήσεις όσον αφορά τα τραγούδια, αφού αυτές οι προτιμήσεις μπορούν να εξαχθούν, ακόμα και με πολύ περιορισμένες πληροφορίες για το χρήστη. Συγκεκριμένα, δείχνεται ότι ακόμα και σε έναν νέο χρήστη για τον οποίο το σύστημα δε γνωρίζει απολύτως τίποτα, μπορεί με μόνο ένα session 25-50 τραγουδιών να εξάγει τις απαραίτητες πληροφορίες ώστε να παράξει με επιτυχία εξατομικευμένες λίστες.

4.2 ΤΟ ΠΡΟΒΛΗΜΑ ΩΣ MARKOV DECISION PROCESS (MDP)

Το πρόβλημα δημιουργίας μιας playlist εδώ θεωρείται ως μια Markov Decision Process (MDP). Ο αλγόριθμος MDP αποτελείται από τα στοιχεία (S, A, P, R, T), όπου το S είναι ένα σετ από καταστάσεις, το A είναι ένα σετ από ενέργειες και το P : S x A x S → [0,1] είναι η συνάρτηση πιθανότητας μετάβασης μεταξύ δύο καταστάσεων. Δηλαδή το P(s,a,s') = r δείχνει ότι η πιθανότητα να μεταβεί το σύστημα από την κατάσταση s στην s' μέσω της ενέργειας a είναι r. Το R : S x A → |R είναι η συνάρτηση ανταμοιβής, δηλαδή το R(s,a) = r είναι η ανταμοιβή που θα πάρει το σύστημα αν στην κατάσταση s επιλεχθεί η ενέργεια a. Το T είναι το σύνολο των τερματικών καταστάσεων, που τερματίζουν δηλαδή τη διαδικασία.

Τώρα προσαρμόζοντας τα παραπάνω στο εν προκειμένω πρόβλημα των Συστημάτων Συστάσεων για Μουσική και θεωρώντας πως υπάρχει ένα σύνολο με n συνολικά τραγούδια M = {α_1, α_2, ..., α_n} καθώς επίσης και ότι οι λίστες αναπαραγωγής έχουν μήκος k (αποτελούνται από k τραγούδια), προκύπτουν τα εξής:

1. Επειδή είναι χρήσιμο να αποτυπώνεται κάπως η εμπειρία του χρήστη από τα τραγούδια που έχει ήδη ακούσει, πρέπει να περιέχεται μια ταξινομημένη λίστα με όλα τα προηγούμενα τραγούδια στην λίστα αναπαραγωγής. Έτσι, η λίστα S θα κάνει ακριβώς αυτό: θα αποθηκεύει τα τραγούδια που έχουν αναπαραχθεί. Θα είναι, λοιπόν,

$S = \{(a_1, a_2, \dots, a_i) | 1 \leq i \leq k; \forall j \leq i, a_j \in M\}$.

Συνεπώς το κάθε $s \in S$ είναι μια λίστα από τραγούδια με μήκος από 0 όταν επιλέγεται το πρώτο τραγούδι έως k όταν η λίστα αναπαραγωγής έχει ολοκληρωθεί.

2. Το σύνολο των ενεργειών A είναι η επιλογή του επόμενου τραγουδιού που θα αναπαραχθεί, $a_k \in A$. Αυτό σημαίνει ότι το σύνολο των πιθανών ενεργειών ταυτίζεται με το σετ των τραγουδιών (A = M).

3. Σύμφωνα με τα (1) και (2) προκύπτει και η συνάρτηση μετάβασης P. Προφανώς, αφού οι ενέργειες a δηλώνουν επιλογές τραγουδιών, για κάθε δεδομένο συνδυασμό s και a θα υπάρχει μόνο ένα s' το οποίο θα ισχύει P(s,a,s') = 1, ενώ για όλα τα s" που είναι διαφορετικά από το s' θα ισχύει P(s,a,s") = 0. Έτσι, θα χρησιμοποιούμε τον συμβολισμό P(s,a) = s' για ευκολία.

4. Το R(s,a) που είναι η συνάρτηση ανταμοιβής, προφανώς εξαρτάται από τον κάθε χρήστη. Η εύρεση της μοναδικής συνάρτησης ανταμοιβής για τον κάθε χρήστη είναι από τα σημαντικότερα ζητήματα που προσπαθεί να λυθεί στο paper που μελετάμε, και θα αναλυθεί περισσότερο στη συνέχεια.

5. $T = \{(a_1, a_2, \dots, a_k)\}$: Το σετ των λιστών αναπαραγωγής μήκους k.

Γενικά, για να λυθεί ένα πρόβλημα MDP πρέπει να βρεθεί μια πολιτική $\pi : S \rightarrow A$, τέτοια ώστε από κάθε δεδομένη κατάσταση s να διαλέγεται η ενέργεια π(s) που θα εξασφαλίσει πως στο τέλος θα συλλεχθεί το μεγαλύτερο δυνατό άθροισμα από rewards. Εδώ, αφού η συνάρτηση P είναι ντετερμινιστική, η π* (η καλύτερη δυνατή πολιτική π) αναφέρεται στην μοναδική λίστα από τραγούδια που θα είναι η πιο «συναρπαστική» για τον χρήστη. Παρόλα αυτά η διαδικασία ξεκινά με δεδομένο ότι η συνάρτηση ανταμοιβής R του χρήστη είναι άγνωστη. Άρα η βέλτιστη λύση του προβλήματος στην προκειμένη περίπτωση είναι ισοδύναμη με τη βέλτιστη εύρεση της συνάρτησης ανταμοιβής του εκάστοτε χρήστη.

Στη θεωρία του Reinforcement Learning υπάρχουν δύο υψηλού επιπέδου προσεγγίσεις για την εύρεση της π*: η model-free και η model-based. Οι model-free προσεγγίσεις μαθαίνουν την αξία του να διαλέξουν την ενέργεια a από την κατάσταση s απευθείας. Δυο παραδείγματα είναι οι αλγόριθμοι Q-Learning και SARSA [31], οι οποίοι είναι αρκετά αποδοτικοί, αλλά για να λειτουργήσουν σωστά χρειάζονται αρκετά εμπειρικά δεδομένα προς μάθηση. Οι model-based προσεγγίσεις μαθαίνουν τις συναρτήσεις μετάβασης και ανταμοιβής (P και R) έτσι ώστε να είναι σε θέση να προσομοιώσουν αυθαίρετο αριθμό εμπειρικών δεδομένων προκειμένου να βρουν

την κατά προσέγγιση καλύτερη λύση του MDP προβλήματος. Συγκριτικά, οι model-based προσεγγίσεις απαιτούν περισσότερους υπολογιστικούς πόρους από τους model-free, ειδικά όταν καλούνται να ξαναλύνουν το MDP κάθε φορά που αλλάζουν τα δεδομένα του μοντέλου. Παρόλα αυτά, σε πολλές εφαρμογές, συμπεριλαμβανομένης και της playlist recommendation, που τα δεδομένα δεν απαιτούν μεγάλη υπολογισιμότητα, η ανταλλαγή του υπολογιστικού κόστους με την αποδοτικότητα των δεδομένων είναι θεμιτή. Έτσι, θα δούμε στη συνέχεια ότι σε αυτό το paper έχει αποφασιστεί η χρήση model-based προσέγγισης.

4.3 ΜΟΝΤΕΛΟΠΟΙΗΣΗ ΑΛΓΟΡΙΘΜΟΥ

Όπως καταλαβαίνουμε και από την προηγούμενη παράγραφο, η εύρεση της συνάρτησης προτιμήσεων ενός χρήστη μέσα από ένα μεγάλο σύνολο τραγουδιών και λιστών απαιτεί μια συμπαγή αναπαράσταση των τραγουδιών που είναι αρκετά πλούσια ώστε να δείχνει τις σημαντικές διαφορές στο πως αυτά γίνονται αντιληπτά από τον χρήστη. Σε αυτή τη λογική, το κάθε τραγούδι αναπαρίσταται σαν ένα διάνυσμα από song descriptors (περιγραφητές τραγουδιού).
Ειδικότερα, ο DJ-MC χρησιμοποιεί περιγραφητές φασματικής ακουστικής που συμπεριλαμβάνουν λεπτομέρειες σχετικά με το αποτύπωμα που μπορεί να αφήσει το τραγούδι στον χρήστη, όπως ρυθμικά χαρακτηριστικά, τη γενική έντασή του και την εναλλαγή αυτής κατά τη διάρκεια του τραγουδιού. Αυτοί οι περιγραφητές επιτρέπουν μια μεγάλη ευκαμψία (για παράδειγμα, στο να δείξουν ομοιότητες μεταξύ τραγουδιών με αρκετά διαφορετικό υπόβαθρο ή την ικανότητα της μοντελοποίησης τραγουδιών σε άγνωστες γλώσσες). Στην επόμενη παράγραφο θα δούμε λεπτομερώς του περιγραφητές που χρησιμοποιούνται από το DJ-MC.

Προκειμένου να επιταχυνθεί η διαδικασία μάθησης, δημιουργείται μια δεύτερη αναπαράσταση στην λογική ότι η συνάρτηση ανταμοιβής R του χρήστη μπορεί να αναπαρασταθεί ως το άθροισμα δύο παραγόντων:

1. Η προτίμηση του χρήστη σχετικά με τα τραγούδια (Songs), R_s= A → |R και

2. Η προτίμησή του σχετικά με τις μεταβάσεις (Transitions) από προηγούμενα τραγούδια σε ένα νέο, R_t: S x A → |R.

Έτσι, ισχύει ότι $R(s,a) = R_s(a) + R_t(s,a)$.

4.3.1 ΜΟΝΤΕΛΟΠΟΙΗΣΗ ΤΡΑΓΟΥΔΙΩΝ

Όπως αναφέρθηκε παραπάνω, τα τραγούδια θα αναπαρίστανται ως διανύσματα περιγραφητών που δίνουν προσεγγιστικά το στίγμα που μπορεί να δώσει το κάθε τραγούδι στον χρήστη, τα ρυθμικά χαρακτηριστικά του, τη γενική του ένταση και τις εναλλαγές στην ένταση κατά τη διάρκειά του. Για το σκοπό αυτό χρησιμοποιήθηκαν τα χαρακτηριστικά από το Million Song Dataset ώστε να εξαχθούν 12 χρήσιμοι περιγραφητές, από τους οποίους οι 2 είναι 12-διάστατοι. Δίνοντας έτσι ένα διάνυσμα που συνολικά είναι 34-διάστατο.

Στο Σχήμα 4.1 φαίνεται το ολοκληρωμένο σύνολο των περιγραφητών που χρησιμοποιούνται για τα τραγούδια.

Descriptors	Descriptor Indices
10th and 90th percentiles of tempo	1,2
average and variance of tempo	3,4
10th and 90th percentiles of loudness	5,6
average and variance of loudness	7,8
pitch dominance	9–20
variance of pitch dominance	21
average timbre weights	22–33
variance in timbre	34

Εικόνα 4.1: Περιγραφητές τραγουδιών

4.3.2 ΜΟΝΤΕΛΟΠΟΙΗΣΗ ΣΥΝΑΡΤΗΣΗΣ ΑΝΤΑΜΟΙΒΗΣ ΧΡΗΣΤΗ (R)

Παρά την αφθονία της βιβλιογραφίας για την ψυχολογία των ανθρώπων και για την αντίληψή τους για τη μουσική, δεν υπάρχει κάποιο μοντέλο για την εμπειρία των ακουσμάτων τους. Σε αυτό το paper η ακρόαση μοντελοποιείται σαν να είναι εξαρτώμενη όχι μόνο από τις προτιμήσεις των χρηστών σε σχέση με τα χαρακτηριστικά που περιγράψαμε, αλλά και από τα χαρακτηριστικά μεταβάσεων. Έτσι, παρακάτω αναλύεται η επιμέρους μοντελοποίηση των δύο συστατικών μερών του R: R_s και R_t.

ΣΥΝΑΡΤΗΣΗ ΑΝΤΑΜΟΙΒΗΣ ΧΡΗΣΤΗ ΣΥΜΦΩΝΑ ΜΕ ΤΑ ΤΡΑΓΟΥΔΙΑ (R_S)

Για να μοντελοποιηθεί η R_S, χρησιμοποιείται μια κωδικοποίηση των περιγραφητών του τραγουδιού για να παραχθεί ένας πίνακας δυαδικών στοιχείων. Έτσι, η R_S είναι μια γραμμική συνάρτηση αυτού του πίνακα χαρακτηριστικών. Κάθε ένα από αυτά τα χαρακτηριστικά συμβάλλει ανεξάρτητα στο κατά πόσο ένα τραγούδι αρέσει στον χρήστη.

Ειδικότερα, για κάθε περιγραφητή τραγουδιού μαζεύονται στατιστικά από ολόκληρη τη βάση δεδομένων της εφαρμογής και ο περιγραφητής ποσοτικοποιείται σε κάδους (bins) 10 εκατοστημορίων. Στη συνέχεια, προκειμένου να εφαρμοστεί σωστά το reinforcement learning συμβολίζουμε τον πίνακα χαρακτηριστικών για το τραγούδι α, ως $\theta_s(\alpha)$. Είναι ένας πίνακας μεγέθους #bins x #descriptors = 10 x 34 = 340 που αποτελείται από #descriptors που περιέχουν 1 στις συντεταγμένες που αντιστοιχούν στους κάδους που «ζει» το τραγούδι α, και 0 σε κάθε άλλη περίπτωση, που σημαίνει ότι το $\theta_S(\alpha)$ συμπεριφέρεται σαν συνάρτηση-δείκτης.

Για κάθε χαρακτηριστικό, υπάρχει μια τιμή που αναπαριστά την ευχαρίστηση που παίρνει ο χρήστης από τραγούδια που έχουν ενεργό (δηλαδή με τιμή 1), το εκάστοτε χαρακτηριστικό. Αυτές οι τιμές αναπαρίστανται σαν ένας πίνακας βαρών $\phi_S(u)$. Οπότε, ισχύει $R_S(\alpha) = \phi_S(u) \times \theta_S(\alpha)$. Προφανώς οι παράμετροι του $\phi_S(u)$ πρέπει να μαθαίνονται εκ νέου για κάθε καινούριο χρήστη.

ΣΥΝΑΡΤΗΣΗ ΑΝΤΑΜΟΙΒΗΣ ΧΡΗΣΤΗ ΣΥΜΦΩΝΑ ΜΕ ΤΙΣ ΜΕΤΑΒΑΣΕΙΣ (R_t)

Όπως είπαμε και νωρίτερα, ο αλγόριθμος που μελετάμε, εκτός από την ευχαρίστηση που παίρνει ο χρήστης ακούγοντας ένα τραγούδι αυτό καθαυτό, λαμβάνει υπόψιν και την επιρροή που έχει η θέση του τραγουδιού σε μια σειρά από τραγούδια (playlist). Για να μαθηματικοποιηθεί αυτή η εξάρτηση, χρησιμοποιείται η εξίσωση:

$$E[R_t((\alpha_1,...,\alpha_{t-1}), \alpha_t)] = \sum_{i=1}^{t-1} \frac{1}{i^2} r_t(a_{t-i}, a_t)$$

Όπου το $r_t(\alpha_i, \alpha_j)$ αναπαριστά την ευχαρίστηση του χρήστη όταν ακούει το τραγούδι α_j κάποια στιγμή, αφού έχει ακούσει το α_i. Ο όρος $\frac{1}{i^2}$ δείχνει ότι ένα τραγούδι που είχε αναπαραχθεί πριν από i τραγούδια έχει πιθανότητα $\frac{1}{i}$ να επηρεάσει την συνάρτηση μετάβασης, αφού πρακτικά ο χρήστης θα έχει ξεχάσει ότι το άκουσε αν έχει περάσει πολλή ώρα. Αν έχει πιθανότητα να επηρεάσει τη συνάρτηση, τότε ο δεύτερος παράγοντας $\frac{1}{i}$ δείχνει την επιρροή που θα έχει. Φαίνεται λοιπόν ότι όσο περνάνε τα τραγούδια η επιρροή του, όπως ήταν λογικό, μειώνεται.

Όπως στην περίπτωση του R_s, μπορεί να περιγραφεί και η R_t σαν μια γραμμική συνάρτηση ενός δυαδικού πίνακα χαρακτηριστικών: $R_t(\alpha_i, \alpha_j) = \phi_t(u) \times \theta_t(\alpha_i, \alpha_j)$, όπου $\phi_t(u)$ είναι ένας πίνακας βαρών που εξαρτάται από τον χρήστη και θ_t είναι ένας δυαδικός πίνακας χαρακτηριστικών.

Αν θέλαμε να λάβουμε υπόψιν τις μεταβάσεις μεταξύ όλων των 340 χαρακτηριστικών των τραγουδιών α_i και α_j, τότε η θ_t θα ήταν μεγέθους $340^2 > 100.000$. Αυτό θα μείωνε πολύ την αποδοτικότητα του αλγορίθμου, οπότε περιορίζονται οι θ_t και ϕ_t στο να αναπαριστούν τις μεταβάσεις μεταξύ των κάδων 10-εκατοστημορίων που προαναφέραμε. Με αυτό τον τρόπο, για κάθε έναν από τους 34 περιγραφητές τραγουδιών, υπάρχουν 100 χαρακτηριστικά, ένα από τα οποία έχει τιμή 1, ενώ τα άλλα 99 έχουν τιμή 0, δείχνοντας ποιο ζευγάρι κάδων υπήρχε τόσο στο τραγούδι α_i όσο και στο τραγούδι α_j. Οπότε, είναι εύκολα κατανοητό πως το θ_t αποτελείται από 3.400 δυαδικά χαρακτηριστικά, 34 από τα οποία έχουν την τιμή 1.

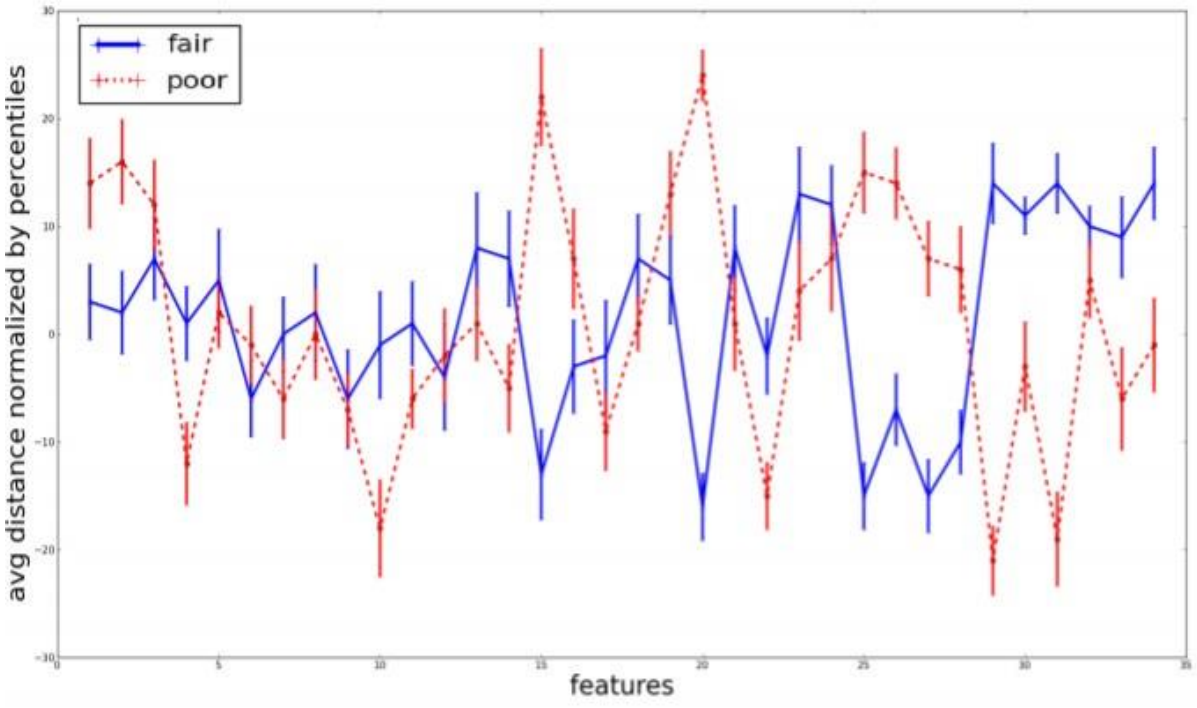
Είναι ξεκάθαρο, πως αυτή η αναπαράσταση που περιγράφουμε, έχει περιορισμένες δυνατότητες και δεν μπορεί να λάβει υπόψιν την ευχαρίστηση που μπορεί να λάβει ο χρήστης από την συνδυαστική μετάβαση μεταξύ πολλαπλών περιγραφητών. Παρόλα αυτά, θυσιάζεται συνειδητά η δυνατότητα να συμπεριλάβουμε αυτή την παράμετρο υπόψιν, προκειμένου να μπορεί να γίνεται η εκπαίδευση του συστήματος με εμφανώς λιγότερα δεδομένα. Εμπειρικά, έχει παρατηρηθεί πως η αναπαράσταση αυτή μπορεί να αποτυπώσει την πλειοψηφία της συνάρτησης μετάβασης πραγματικών ανθρώπων και να αναβαθμίσει την ποιότητα της πρότασης τραγουδιών.

Όπως στο $\phi_S(u)$, οι παράμετροι του $\phi_t(u)$ πρέπει να μαθαίνονται εκ νέου για κάθε νέο χρήστη. Συνολικά οι παράμετροι βάρους που πρέπει να μαθαίνει το σύστημα για κάθε νέο χρήστη είναι 3740. Με τόσες πολλές παραμέτρους, είναι ακατόρθωτο σε μια λίστα αναπαραγωγής 25 τραγουδιών να μπορούν να αναπαραχθούν τραγούδια που να τις έχουν όλες ενεργές. Παρόλα αυτά ο DJ-MC είναι ικανός να εκμαιεύσει γνώση μόνο από μερικά παραδείγματα μετάβασης, ώστε να προγραμματίσει μια μελλοντική σειρά τραγουδιών που συγκλίνει προς τις παραμέτρους που είναι θετικές και κατά αυτών που είναι αρνητικές.

4.3.3 ΕΚΦΡΑΣΤΙΚΟΤΗΤΑ ΤΟΥ ΜΟΝΤΕΛΟΥ ΧΡΗΣΤΗ

Η παραπάνω συνάρτηση ευχαρίστησης του χρήστη ως ένας 3740-διάστατος πίνακας δυαδικών χαρακτηριστικών είναι απλώς μία από τις πολλές δυνατές αναπαραστάσεις. Μια απαραίτητη ικανότητα που θα πρέπει να έχει κάθε χρήσιμη αναπαράσταση είναι το να είναι σε θέση τα χαρακτηριστικά του να μπορούν να ξεχωρίζουν τις λίστες αναπαραγωγής που κοινώς θεωρούνται «καλές» από αυτές που κοινώς θεωρούνται «κακές». Με βάση αυτό, ο DJ-MC

μπορεί να μοντελοποιήσει αποτελεσματικότερα τη συνάρτηση ανταμοιβής του χρήστη. Για να αξιολογηθεί το αν ο αλγόριθος αυτός έχει χαρακτηριστικά που είναι αρκετά εκφραστικά, εξετάστηκε το προφίλ μετάβασης μεταξύ δύο τύπων μετάβασης: την «μη ικανοποιητική» και την «ικανοποιητική», που παράχθηκαν από το ίδιο σύνολο τραγουδιών. Η «ικανοποιητική» παράχθηκε παίρνοντας σαν δείγμα ζευγάρια τραγουδιών που εμφανίζονταν σε μια πραγματική λίστα αναπαραγωγής. Η «μη-ικανοποιητική» παράχθηκε δημιουργώντας ζευγάρια τραγουδιών που βάσει των χαρακτηριστικών τους είναι αρκετά διαφορετικά. Για παράδειγμα, ένα τραγούδι με γρήγορο ρυθμό και υψηλή ένταση, να ακολουθείται από ένα τραγούδι με αργό ρυθμό και αρκετά χαμηλή ένταση. Η διαφορά μεταξύ των δύο αυτών προφίλ μπορεί να φανεί στην παρακάτω εικόνα. Όπου «fair» αντιστοιχεί στην «ικανοποιητική» μετάβαση, ενώ «poor» στην «μη-ικανοποιητική». Η ομαδοποίηση της «ικανοποιητικής» και της «μη-ικανοποιητικης» έγινε από το ίδιο σύνολο 20 τραγουδιών. Η Εικόνα 4.2 δείχνει τη μέση μετάβαση για κάθε χαρακτηριστικό. Τα 20 τραγούδια έχουν παρθεί από 5 διαφορετικά άλμπουμ. Για την «ικανοποιητική», έχει χρησιμοποιηθεί για τα τραγούδια η σειρά με την οποία αναπαράγονται και στο original άλμπουμ, ενώ για τη «μη-ικανοποιητική» τα τραγούδια έχουν μπει σε εντελώς τυχαία σειρά.

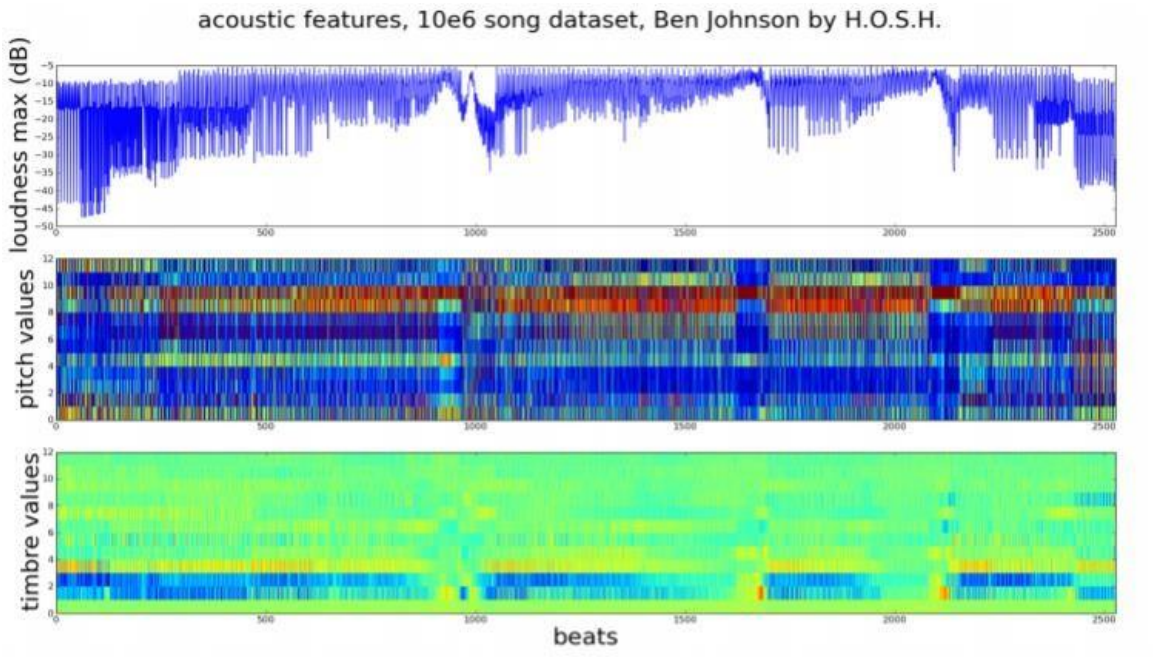


Εικόνα 4.2: Μέση μετάβαση ανά χαρακτηριστικό

4.4 ΔΕΔΟΜΕΝΑ

Ένα πολύ σημαντικό στοιχείο της δουλειάς που έχει γίνει στην συγκεκριμένη εφαρμογή είναι η εξαγωγή πραγματικών δεδομένων τόσο για τα τραγούδια, όσο και τις λίστες αναπαραγωγής προκειμένουν να ελέγχεται αυστηρά η εγκυρότητα της προσέγγισης. Σε αυτό το κεφάλαιο θα δούμε τις διαφορετικές πηγές που χρησιμοποιήθηκαν ώστε να μοντελοποιηθούν τα τραγούδια και οι λίστες αναπαραγωγής. Για τα τραγούδια, χρησιμοποιήθηκε η δωρεάν συλλογή Million Song Dataset (http://millionsongdataset.com) που περιέχει χαρακτηριστικά και δεδομένα για ένα εκατομμύριο διάσημα
τραγούδια. Το σύνολο των δεδομένων καλύπτει 44.745 καλλιτέχνες και 10^6 διαφορετικά τραγούδια.

Στην Εικόνα 4.3 δίνεται ένα παράδειγμα εισόδου ήχου για ένα τραγούδι:



Εικόνα 4.3: Είσοδος ήχου

Για να ελεγχθεί αρχικά η προσέγγιση σε προσομοίωση χρειάζονται επίσης και πραγματικές λίστες αναπαραγωγής, ώστε να εξαχθούν δεδομένα σχετικά με μεταβάσεις από τραγούδι σε τραγούδι. Στο paper που μελετάμε χρησιμοποιήθηκαν δύο πηγές για να ανακτηθούν λίστες αναπαραγωγής. Η πρώτη ήταν η βιβλιοθήκη του Yes.com και το Last.fm από όπου συλλέχθηκαν δεδομένα από τον Δεκέμβρη του 2010 έως τον Μάιο του 2011. Συνολικά υπήρξαν σαν δεδομένα 75.262 τραγούδια και 2.840.553 μεταβάσεις. Η δεύτερη πηγή είναι το «The Art of the Mix

Archive» (http://www.artofthemix.org), το οποίο είναι μια βάση δεδομένων/κοινότητα για χομπίστες που ασχολούνται με τη μουσική και με λίστες αναπαραγωγής. Από εκεί συνολικά πάρθηκαν 29.000 playlists, οι οποίες φτιάχνονται από αληθινούς χρήστες και όχι από εμπορικούς ραδιοφωνικούς παραγωγούς ή από κάποιο σύστημα συστάσεων, πράγμα που σημαίνει ότι μας δίνουν ό,τι πιο ρεαλιστικό γίνεται να έχουμε προκειμένου να κάνουμε μοντελοποίηση των προτιμήσεων πραγματικών χρηστών.

4.5 DJ-MC

Σε αυτό το σημείο θα μιλήσουμε για το DJ-MC, μια νέα reinforcement learning προσέγγιση σε ένα playlist-oriented, εξατομικευμένο σύστημα συστάσεων μουσικής. Η αρχιτεκτονική του DJ-MC περιέχει δύο κύρια στάδια:

1. Εκμάθηση των παραμέτρων χρήστη (ϕ_s και ϕ_t) και
2. Δημιουργία μιας λίστας αναπαραγωγής.

Το μέρος της εκμάθησης χωρίζεται από μόνο του σε δύο μέρη: την αρχικοποίηση και την εκμάθηση στην πορεία. Η αρχικοποίηση είναι πολύ σημαντική, προκειμένου ο χρήστης να νιώσει οικεία εξαρχής και να μην χάσει το ενδιαφέρον του μέχρι το σύστημα να καταφέρει να φτιάξει το ρεαλιστικό εξατομικευμένο μοντέλο χρήστη. Η εκμάθηση στην πορεία επιτρέπει στο σύστημα να βελτιώνεται συνεχώς μέχρι να συγκλίνει σε ένα όσο το δυνατόν πιο αξιόπιστο μοντέλο χρήστη. Στην προσομοίωση, θεωρείται πως ο χρήστης είναι σε θέση να προσδιορίσει μια αρχική λίστα από τραγούδια που του αρέσουν. Παρόλα αυτά, στη συνέχεια αποδεικνύεται πως αυτό το στάδιο μπορεί να αντικατασταθεί από τυχαία εξερεύνηση, διατηρώντας το ίδιο ικανοποιητικά αποτελέσματα στο στάδιο εκμετάλλευσης της υπάρχουσας γνώσης.

Το μέρος της δημιουργίας της λίστας επιτρέπει την επιλογή του επόμενου τραγουδιού που θα αναπαραχθεί. Όπως γράφτηκε νωρίτερα, το καθαρό πεδίο εφαρμογής του προβλήματος εκμάθησης, ακόμα και μετά από πολλά στάδια αφαίρεσης πιθανών τραγουδιών, είναι η λύση του MDP προβλήματος που είναι αρκετά πολύπλοκο. Για αυτό το λόγο, πρέπει να βρούμε μια προσεγγιστική λύση. Από πρακτικής άποψης, από κάθε δεδομένη κατάσταση, ο στόχος είναι να βρεθεί ένα τραγούδι που είναι «αρκετά καλό» ώστε να αναπαραχθεί στη συνέχεια. Για την επίτευξη αυτού του σκοπού, χρησιμοποιείται το Monte Carlo Tree Search [32].

Στη συνέχεια αναλύονται τα βήματα αρχικοποίησης του DJ-MC.

4.5.1 ΑΡΧΙΚΗ ΕΚΤΙΜΗΣΗ ΠΡΟΤΙΜΗΣΕΩΝ ΤΡΑΓΟΥΔΙΩΝ

Για να αρχικοποιηθεί το μοντέλο προτιμήσεων του χρήστη, ο DJ-MC δημοσκοπεί τον χρήστη για τα k_S αγαπημένα του τραγούδια στη βάση δεδομένων και τα περνάει σαν είσοδο στον αλγόριθμο της Εικόνας 4.4:

Input: Song corpus, $\mathcal{M}$
Number of preferred songs to be provided by listener, k_s
initialize all coordinates of ϕ_s to $1/(k_s + \#bins)$
$preferredSet = \{a_1, \ldots, a_{k_s}\}$ *(chosen by the listener)*
for $i = 1$ **to** k_s **do**
    $\phi_s = \phi_s + \frac{1}{(k_s+1)} \cdot \theta_s(a_i)$
end for

Εικόνα 4.4: Αλγόριθμος αρχικής εκτίμησης προτιμήσεων σε τραγούδια

Σαν τρόπος εξομάλυνσης, κάθε στοιχείο του $\phi_S(u)$ αρχικοποιείται ως $1/(k_S + \#κάδοι)$, όπου #κάδοι είναι ο βαθμός λεπτομέρειας της διακριτικής ευχέρειας του κάθε περιγραφητη τραγουδιού. Στην προκειμένη περίπτωση είναι 10. Έπειτα, για κάθε ένα από τα αγαπημένα τραγούδια (a), το $\phi_S(u)$ αυξάνεται κατά $1/(k + \#κάδοι) \times \theta_S(a)$. Στο τέλος αυτής της διαδικασίας, τα βάρη της $\phi_S(u)$ που αντιστοιχούν στον κάθε περιγραφητή του τραγουδιού, έχουν συνολικό άθροισμα 1.

4.5.2 ΑΡΧΙΚΗ ΕΚΤΙΜΗΣΗ ΠΡΟΤΙΜΗΣΕΩΝ ΜΕΤΑΒΑΣΗΣ

Στο δεύτερο στάδιο, ο χρήστης ερωτάται για τις προτιμήσεις του όσον αφορά τις μεταβάσεις, ακολουθώντας τη διαδικασία που περιγράφεται από τον αλγόριθμο της Εικόνας 4.5:

1: **Input:** Song corpus $\mathcal{M}$
Number of transitions to poll the listener, k_t
2: initialize all coordinates of ϕ_t to $1/(k_t + \#bins)$
3: Select upper median of $\mathcal{M}$, $\mathcal{M}^*$, based on R_s
4: δ = 10th percentile of all pairwise distances between songs in $\mathcal{M}$
5: representative set $\mathcal{C} = \delta$ -medoids $(\mathcal{M}^*)$
6: $song_0$ = choose a song randomly from $\mathcal{C}$
7: **for** $i = 1$ **to** k_t **do**
8: $song_i \leftarrow$ *chosen by the listener from* $\mathcal{C}$
9: $\phi_t = \phi_t + \frac{1}{(k_t+1)} \cdot \theta_t(song_{i-1}, song_i)$
10: **end for**

Εικόνα 4.5: Αλγόριθμος αρχικής εκτίμησης προτιμήσεων σε μεταβάσεις

Όπως και στην περίπτωση της αρχικοποίησης των προτιμήσεων τραγουδιών, η προβλεπόμενη τιμή της μετάβασης από τον κάδο i στον κάδο j για κάθε χαρακτηριστικό αρχικοποιείται στο $1/(k_t$ + #κάδοι), όπου k_t είναι ο αριθμός των μεταβάσεων που ερωτήθηκαν και #κάδοι ο αριθμός των κάδων.

Δε θα είχε πολύ νόημα να ερωτηθούν οι χρήστες σχετικά με μεταβάσεις σε ένα πολύ μικρό υποσύνολο όπως είναι τα προτιμώμενα τραγούδια, γιατί αυτό δε θα αποκάλυπτε αρκετά σχετικά με τις επιθυμητές μεταβάσεις. Για αυτό τον λόγο, εξερευνώνται οι προτιμήσεις των χρηστών μέσα από ένα στοχευμένο σύνολο, όπου τους παρουσιάζονται διαφορετικές πιθανές μεταβάσεις που συμπεριλαμβάνουν τις ποικίλες μορφές τραγουδιών και μεταβάσεων που υπάρχουν στη βάση, και όχι μόνο το υποσύνολο των προτιμώμενων τραγουδιών. Από την άλλη πλευρά, θα ήταν χρήσιμο να εξαχθούν συμπεράσματα για περιοχές του συνόλου δεδομένων (dataset), όπου οι προβλέψεις για την ανταμοιβή των τραγουδιών είναι χαμηλές.

Για να τα καλύψει όλα αυτά, ο DJ-MC αρχικά διαλέγει το 50% των τραγουδιών από το σύνολο M*. Το M είναι το σύνολο των τραγουδιών και το M* είναι το υποσύνολο του M, το οποίο περιέχει τα τραγούδια με τη μεγαλύτερη ανταμοιβή, όπως φαίνεται στη γραμμή 3 του παραπάνω αλγορίθμου. Στη συνέχεια, χρησιμοποιεί το feedback των χρηστών για αυτό το υποσύνολο. Αυτό το κάνει μέσω του αλγορίθμου δ-medoids [33] (γραμμή 3 του αλγορίθμου), δηλαδή μιας νέας μεθόδου για αντιπροσωπευτική επιλογή. Ο συγκεκριμένος αλγόριθμος

επιστρέφει ένα υποσύνολο, τέτοιο ώστε κανένα δείγμα του δεν απέχει περισσότερο από δ από μια επιθυμητή τιμή της συνάρτησης ανταμοιβής. Το δ αρχικοποιείται στην $4^{η}$ γραμμή σύμφωνα με το $10^{ο}$ εκατοστημόριο των αποστάσεων που έχουν στο ιστόγραμμα όλα τα ζεύγη τραγουδιών, αλλά ο τρόπος υπολογισμού και οι τεχνικές/μαθηματικές επεξηγήσεις πίσω από αυτό δεν χρειάζεται ούτε είναι απαραίτητο να εξηγηθούν στο πλαίσιο της παρούσας διπλωματικής. Το τελικό και αντιπροσωπευτικό υποσύνολο λοιπόν που χρησιμοποιεί ο DJ-MC συμβολίζεται με C (γραμμή 5) και ο DJ-MC ρωτάει τους χρήστες ποιο από τα τραγούδια του υποσυνόλου αυτού θα ήθελαν να ακούσουν στη συνέχεια (γραμμή 8). Για λόγους μοντελοποίησης, θεωρείται εδώ ότι ο χρήστης διαλέγει το επόμενο τραγούδι προσομοιώνοντας την ακουστική εμπειρία, συμπεριλαμβάνοντας την εξάρτηση από το ιστορικό ανταμοιβής μεταβάσεων, και διαλέγοντας το τραγούδι με τη μέγιστη συνολική ανταμοιβή. Ο DJ-MC συνεχίζει κάνοντας update στα χαρακτηριστικά της μετάβασης αυτής, αυξάνοντας το βάρος των χαρακτηριστικών μετάβασης κατά 1/(k+#κάδοι) (γραμμή 9), παρόμοια με το πώς έκανε update στο μοντέλο της προτίμησης τραγουδιών. Οπότε και σε αυτή την περίπτωση το άθροισμα των βαρών θα είναι 1 όπως και πριν.

4.5.3 ΣΥΝΕΧΗΣ ΕΚΜΑΘΗΣΗ

Μετά την αρχικοποίηση, ο DJ-MC ξεκινά να αναπαράγει τραγούδια για τον χρήστη, ζητώντας feedback, και ανανεώνοντας συνεχώς τις συναρτήσεις ϕ_S και ϕ_t. Για να μην είναι δύσχρηστος, ο DJ-MC δεν απαιτεί ξεχωριστά feedbacks για τα τραγούδια και για τις μεταβάσεις, αλλά αξιολογεί κάθε έναν από τους 2 παράγοντες από ένα και μόνο σήμα του χρήστη που αφορά το τραγούδι. Αυτό το κάνει με το να υπολογίζει τη σχετική συνεισφορά της ανταμοιβής του τραγουδιού και της μετάβασης στη συνολική ανταμοιβή, όπως προβλέπεται από το μοντέλο. Αυτή η διαδικασία περιγράφεται στον αλγόριθμο της Εικόνας 4.6.

1: **Input:** Song corpus, $\mathcal{M}$
Planned playlist duration, K
2: **for** $i \in \{1, \ldots, K\}$ **do**
3: Use Algorithm 4 to select song a_i, obtaining reward r_i
4: let $\bar{r} = average(\{r_1, \ldots, r_{i-1}\})$
5: $r_{incr} = log(r_i/\bar{r})$
weight update:
6: $w_s = \frac{R_s(a_i)}{R_s(a_i)+R_t(a_{i-1},a_i)}$
7: $w_t = \frac{R_t(a_{i-1},a_i)}{R_s(a_i)+R_t(a_{i-1},a_i)}$
8: $\phi_s = \frac{i}{i+1} \cdot \phi_s + \frac{1}{i+1} \cdot \theta_s \cdot w_s \cdot r_{incr}$
9: $\phi_t = \frac{i}{i+1} \cdot \phi_t + \frac{1}{i+1} \cdot \theta_t \cdot w_t \cdot r_{incr}$
10: Per $d \in$ descriptors, normalize ϕ_s^d, ϕ_t^d
(where ϕ_x^d denotes coordinates in ϕ_x corresponding to 10-percentile bins of descriptor d)
11: **end for**

Εικόνα 4.6: Αλγόριθμος υπολογισμού συνεισφοράς ανταμοιβής τραγουδιού και μετάβασης

Για την ακρίβεια, έστω πως r είναι η ανταμοιβή που δίνει ο χρήστης αφού ακούσει το τραγούδι a στην κατάσταση s, και r^- η μέση τιμή των ανταμοιβών που έχει δώσει ο χρήστης ως τώρα (γραμμή 4). Ορίζεται το $r_{incr} = \log(r/r^-)$ (γραμμή 5). Αυτός ο παράγοντας καθορίζει τόσο την κατεύθυνση όσο και το μέγεθος για το update στην ανταμοιβή (αρνητικό αν $r < r^-$, αλλιώς θετικό, και όσο μεγαλύτερη η διαφορά του r από τον μέσο όρο των μέχρι τώρα ανταμοιβών τόσο μεγαλύτερη και η ανταμοιβή). Έστω οι $R_S(\alpha_i)$ και $R_\tau(\alpha_{i-1}, \alpha_i)$ οι αναμενόμενες ανταμοιβές για το τραγούδι και την μετάβαση, αντίστοιχα, οι οποίες προκύπτουν από το μοντέλο που έχει αναλυθεί παραπάνω. Ο DJ-MC χρησιμοποιεί τα ποσοστά αυτών των τιμών για να θέσει βάρη. Συγκεκριμένα, ορίζονται τα βάρη για τα updates στα τραγούδια και στις μεταβάσεις ως εξής (γραμμές 5,6):

1. $w_s = \frac{R_s(a_i)}{R_s(a_i)+R_t(a_{i-1},a_i)}$

2. $w_t = \frac{R_t(a_{i-1}, a_i)}{R_s(a_i) + R_t(a_{i-1}, a_i)}$

Στη συνέχεια, το σύστημα χρησιμοποιεί τα βάρη αυτά για να μοιράσει όσο το δυνατόν αποδοτικότερα την ανταμοιβή του χρήστη ανάμεσα στα τραγούδια και τις μεταβάσεις και να ανανεώσει τις τιμές τους (γραμμές 8,9). Τέλος, ο DJ-MC κανονικοποιεί τα μοντέλα ανταμοιβής του χρήστης ώστε αυτά να αθροίζουν στο 1 (γραμμή 10).

4.5.4 ΔΗΜΙΟΥΡΓΙΑ ΛΙΣΤΑΣ ΑΝΑΠΑΡΑΓΩΓΗΣ

1: **Input:** Song corpus $\mathcal{M}$, planning horizon q
2: Select upper median of $\mathcal{M}$, $\mathcal{M}^*$, based on R_s
3: $BestTrajectory = null$
4: $HighestExpectedPayoff = -\infty$
5: **while** computational power not exhausted **do**
6: $trajectory = []$
7: **for** $1, \ldots, q$ **do**
8: $song \leftarrow$ selected randomly from $\mathcal{M}^*$ *(avoiding repetitions)*
9: optional: $song\_type \leftarrow$ selected randomly from $song\_types(\mathcal{M}^*)$ *(avoiding repetitions, $song\_types(\cdot)$ reduces the set to clusters)*
10: add $song$ to $trajectory$
11: **end for**
12: expectedPayoffForTrajectory $= R_s(song_1) + \sum_{i=2}^{q}(R_t((song_1, \ldots, song_{i-1}), song_i) + R_s(song_i))$
13: **if** expectedPayoffForTrajectory > HighestExpectedPayoff **then**
14: HighestExpectedPayoff = expectedPayoffForTrajectory
15: $BestTrajectory = trajectory$
16: **end if**
17: **end while**
18: optional: if planning over song types, replace $BestTrajectory[0]$ with concrete song.
19: return $BestTrajectory[0]$

Εικόνα 4.7: Αλγόριθμος δημιουργίας λίστας αναπαραγωγής

Αφού έχει εκτιμήσει τις συναρτήσεις R_s και R_t που δείχνουν τις προτιμήσεις του χρήστη όσον αφορά τα τραγούδια και τις μεταβάσεις και καθορίζουν την συνάρτηση ανταμοιβής $R(s,a) = R_S(a) + R_t(s,a)$, ο DJ-MC χρησιμοποιεί μια αναζήτηση δέντρου με ευριστικές τιμές. Όπως στην περίπτωση αρχικοποίησης των βαρών μετάβασης, ο DJ-MC διαλέγει ένα υποσύνολο που αποτελείται από τα μισά τραγούδια της βάσης, τα οποία με βάση την R_S συλλέγουν την μεγαλύτερη ανταμοιβή, όπως φαίνεται στη γραμμή 2 του αλγόριθμου στην Εικόνα 4.7. Στις

γραμμές 7-11 διαλέγει τυχαία τραγούδια από το υποσύνολο αυτό και έτσι δημιουργεί μια λίστα με τραγούδια. Στη συνέχεια, με τις συναρτήσεις R_S και R_t υπολογίζει την εκτιμώμενη ανταμοιβή της λίστας αυτής. Επαναλαμβάνει τη διαδικασία όσες περισσότερες φορές γίνεται, και τελικά επιλέγει την λίστα που δίνει τη μεγαλύτερη συνολική ανταμοιβή. Τότε, αναπαράγει το πρώτο τραγούδι της λίστας αυτής ως το επόμενο τραγούδι που θα αναπαραχθεί. Διαλέγει μόνο το πρώτο, και όχι ολόκληρη τη λίστα, γιατί όσο συσσωρεύεται ο θόρυβος από τη μοντελοποίηση, οι εκτιμήσεις του DJ-MC αποκλίνουν από τις πραγματικές τιμές ανταμοιβής. Επιπλέον, όπως εξηγήθηκε νωρίτερα οι συναρτήσεις ϕ_t και ϕ_S ανανεώνονται συνεχώς online με βάση το feedback των χρηστών. Συνεπώς, το να επαναπροσδιορίζει τις εκτιμήσεις του σε κάθε βήμα, θεωρείται απαραίτητο.

Αν η βάση με τα τραγούδια είναι πολύ μεγάλη ή ο διαθέσιμος χρόνος δεν είναι αρκετός, μπορεί να είναι αδύνατον ή μη αποδοτικό να φτιαχτούν λίστες που να ξεκινούν με όλα τα πιθανά τραγούδια. Για αυτόν τον λόγο, στην γραμμή 9 υπάρχει ένα προαιρετικό βήμα που είναι η ομαδοποίηση των τραγουδιών ανάλογα με το είδος τους. Έτσι, ο προγραμματισμός μπορεί να γίνει με βάση το είδος μουσικής, φτιάχνοντας λίστες όχι με τραγούδια, αλλά με είδος. Μόλις βρεθεί μια λίστα που είναι πολλά υποσχόμενη, ο DJ-MC διαλέγει ένα τραγούδι τυχαία από το είδος μουσικής που ήταν πρώτο στη λίστα.

Συνδυάζοντας τις διαδικασίες αρχικοποίησης, εκμάθησης στην πορεία και δημιουργίας μιας λίστας αναπαραγωγής, συνολικά η αρχιτεκτονική του DJ-MC παρουσιάζεται στον παρακάτω αλγόριθμο της Εικόνας 4.8.

1: **Input:** $\mathcal{M}$ - song corpus, K - planned playlist duration, k_s - number of steps for song preference initialization, k_t - the number of steps for transition preference initialization

Initialization:
2: Call Algorithm 1 with corpus $\mathcal{M}$ and parameter k_s to initialize song weights ϕ_s.
3: Call Algorithm 2 with corpus $\mathcal{M}$ and parameter k_t to initialize transition weights ϕ_t.

Planning and Model Update:
4: Run Algorithm 3 with corpus $\mathcal{M}$ and parameter K (Algorithm 3 iteratively selects the next song to play by calling algorithm 4, and then updates R_s and R_t. This is repeated for K steps.)

Εικόνα 4.8: Περιγραφικός αλγόριθμος λειτουργίας DJ-MC

4.6 ΣΥΝΟΨΗ-ΑΞΙΟΛΟΓΗΣΗ

Στο paper που μελετήσαμε στο Κεφάλαιο 4 παρουσιάστηκε ο DJ-MC, ένα πλήρες framework για δημιουργία λιστών αναπαραγωγής μουσικής. Το framework αυτό μαθαίνει τις προτιμήσεις του εκάστοτε χρήστη online και παράγει μια κατάλληλη εξατομικευμένη λίστα. Σε πειράματα που έγιναν στο πλαίσιο του paper αυτού, αποδείχθηκε ότι η προσέγγιση αυτή προσφέρει αξιοσημείωτα καλύτερα αποτελέσματα από τις καθιερωμένες μεθόδους, καθώς συνήθως μελετάται μόνο η προτίμηση των χρηστών σε τραγούδια. Εδώ μελετήθηκε η προτίμηση σε μεταβάσεις και το κατά πόσο αυτές είναι σημαντικές για τη δημιουργία μιας ολοκληρωμένης λίστας. Επίσης, ο αλγόριθμος αυτός δεν αρκείται στο να συλλέξει πληροφορίες για τις προτιμήσεις στα τραγούδια, αλλά τις προεκτείνει στα χαρακτηριστικά των τραγουδιών. Δηλαδή μπορεί να προτείνει ένα τραγούδι σε κάποιον χρήστη, ενώ ο χρήστης δεν το έχει ακούσει ποτέ, απλώς επειδή έχει παρόμοια ή ίδια χαρακτηριστικά με κάποια άλλα τραγούδια για τα οποία έχει δώσει καλό feedback.

ΚΕΦΑΛΑΙΟ 5

ΣΥΣΤΗΜΑΤΑ ΣΥΣΤΑΣΕΩΝ ΓΙΑ ΔΙΑΦΗΜΙΣΕΙΣ

5.1 ΠΑΡΟΥΣΙΑΣΗ ΠΡΟΒΛΗΜΑΤΟΣ

Στα εξατομικευμένα συστήματα συστάσεων για διαφημίσεις, ο στόχος είναι να μαθευτεί μια στρατηγική (από τα ιστορικά δεδομένα), η οποία για κάθε χρήστη της ιστοσελίδας, διαλέγει μια διαφήμιση με τη μεγαλύτερη πιθανότητα να προβληθεί από τον εκάστοτε χρήστη. Σχεδόν όλα αυτά τα συστήματα σήμερα χρησιμοποιούν Επιβλεπόμενη Μάθηση ή contextual bandit αλγορίθμους (κυρίως τους contextual bandit αλγορίθμους που κάνουν και εξερεύνηση, εκτός από εκμετάλλευση της υπάρχουσας γνώσης). Αυτοί οι αλγόριθμοι, όπως αναφέρθηκε και στην Εισαγωγή (Κεφ. 1), δεν διαχωρίζουν την επίσκεψη με τον επισκέπτη. Κάθε επίσκεψη αντιμετωπίζεται σαν ένας νέος επισκέπτης. Έτσι, δεν λαμβάνεται υπόψιν ποτέ η μακροπρόθεσμη επιρροή της εκάστοτε ιστοσελίδας και των διαφημίσεων στον χρήστη. Οι χρήστες όμως τείνουν συνεχώς να επιστρέφουν στις ιστοσελίδες και να αναπτύσσουν μακροπρόθεσμες σχέσεις με αυτές, με αποτέλεσμα να καταπατάται η κύρια υπόθεση, ότι δηλαδή επισκέπτης και επίσκεψη δεν διαφέρουν. Στο paper που πρόκειται να μελετήσουμε [34] και να αναλύσουμε λοιπόν, γίνεται μια προσπάθεια να λυθεί το πρόβλημα αυτό.

Οι αλγόριθμοι Reinforcement Learning που προσπαθούν να βελτιστοποιήσουν την μακροπρόθεσμη απόδοση του συστήματος, φαντάζουν άκρως κατάλληλοι για αυτού του είδους Personalized Ad Recommendation (PAR) συστήματα. Η τιμή Click Through Rate (CTR) είναι αυτή που συνήθως λαμβάνουν υπόψιν και προσπαθούν να μεγιστοποιήσουν οι κοινοί greedy αλγόριθμοι. Η φύση όμως των αλγορίθμων που λειτουργούν με Reinforcement Learning τους επιτρέπει να λαμβάνουν υπόψιν όλη τη διαθέσιμη γνώση για το χρήστη με σκοπό να μεγιστοποιήσουν την τιμή LTV (life-time value), δηλαδή τον αριθμό των click που θα κάνει ο χρήστης σε διαφημίσεις στις πολλαπλές επισκέψεις του στη σελίδα. Η κύρια διαφορά των RL προσεγγίσεων είναι ότι ξεχωρίζουν την επίσκεψη από τον επισκέπτη και ομαδοποιούν όλες τις επισκέψεις του κάθε χρήστη. Πρακτικά, αυτό επιτρέπει να υπάρχει μια «μνήμη» που καταγράφει το ιστορικό και τις προτιμήσεις του εκάστοτε χρήστη, ώστε να μην υπάρχει κάποιου είδους cold start κάθε φορά που ο χρήστης επισκέπτεται τη σελίδα. Έτσι, παρόλο που αξιολογούμε την αποδοτικότητα των RL αλγορίθμων με βάση τη μετρική CTR, δεν είναι αυτή η τιμή η οποία προσπαθούν να μεγιστοποιήσουν, και θα ήταν καταλληλότερο να τις αξιολογούμε βάσει του αριθμού των απαιτούμενων clicks ανά χρήστη, δηλαδή το LTV.

Συνεπώς, το πρόβλημα ανάγεται στην βελτιστοποίηση της διαδικασίας, προκειμένου να υπάρξει η υψηλότερη δυνατή LTV τιμή. Εδώ όμως συναντώνται δύο εμπόδια:

1) Ο υπολογισμός μιας κατάλληλης και λειτουργικής LTV πολιτικής και

2) Η αξιολόγηση της απόδοσης μιας πολιτικής ενός RL αλγορίθμου χωρίς να χρειαστεί να τον τρέξουμε. Δηλαδή η αξιολόγησή του αποκλειστικά και μόνο με βάση τα ιστορικά δεδομένα που έχουν παραχθεί από μία ή περισσότερες άλλες πολιτικές.

Ένα δεύτερο πρόβλημα που προκύπτει είναι γνωστό ως off-policy evaluation είναι πολύ σημαντικό όχι μόνο στη σύσταση διαφημίσεων, αλλά και σε άλλους τομείς όπως η υγεία και τα οικονομικά. Μπορεί η λύση του να βοηθήσει και στο πρώτο πρόβλημα στο να επιλεχθεί η σωστή αναπαράσταση για τον RL αλγόριθμο και στο να βελτιστοποιηθούν οι παράμετροί του. Κατά συνέπεια θα είναι εφικτό να αναπτυχθεί ένας πιο κλιμακούμενος αλγόριθμος και να παράγονται καλύτερες πολιτικές. Δυστυχώς, δεν έχουν εφαρμοστεί και άρα δεν έχουν θεωρητικά βρεθεί μέθοδοι που να εγγυώνται ότι μια RL πολιτική μπορεί να λειτουργήσει με μεγάλη αποδοτικότητα σε ένα πραγματικό σύστημα χωρίς να χρειαστεί να εκτελεστεί ο αλγόριθμος.

5.2 ΠΡΟΕΤΟΙΜΑΣΙΑ ΑΛΓΟΡΙΘΜΟΥ

Αρχικά το πρόβλημα μοντελοποιείται ως μια Markov Decision Process (MDP). Συμβολίζεται με s_t ο πίνακας χαρακτηριστικών που περιγράφει την t-οστή επίσκεψη ενός χρήστη στην ιστοσελίδα (κατάσταση) και με α_t την t-οστή διαφήμιση που εμφανίζεται στον χρήστη (ενέργεια). Με r_t συμβολίζεται η ανταμοιβή, η οποία παίρνει τιμή 1 αν ο χρήστης επιλέξει τη διαφήμιση α_t, ή 0 αν την αγνοήσει. Εδώ υπάρχει η υπόθεση ότι ο χρήστης επισκέπτεται το πολύ T φορές και επιλέγει (ή όχι) κάποια διαφήμιση επίσης T φορές. Έτσι προκύπτει το σύνολο τ, όπου $\tau = \{ s_1, \alpha_1, r_1, s_2, \alpha_2, r_2, \dots , s_T,\alpha_T,r_T\}$, το οποίο δείχνει το ιστορικό των αλληλεπιδράσεων με κάποιον χρήστη και καλείται «δεξαμενή τ». Αυτό που προκύπτει από την δεξαμενή τ λοιπόν είναι το μειωμένο άθροισμα των ανταμοιβών, $R(\tau):= \sum_{t=1}^{T} \gamma^{t-1} r_t$, όπου το $\gamma \in [0,1]$ είναι ένας παράγοντας μείωσης.

Μια πολιτική π χρησιμοποιείται για να καθορίσει την πιθανότητα που έχει να εμφανιστεί η κάθε διαφήμιση. Έστω π(α/s) ο συμβολισμός της πιθανότητας του να εμφανιστεί η διαφήμιση α στην κατάσταση s, η οποία είναι ανεξάρτητη του χρόνου t. Ο στόχος είναι να βρεθεί μια πολιτική που μεγιστοποιεί τον εκτιμώμενο συνολικό αριθμό clicks ανά χρήστη: $\rho(\pi):= E[R(\tau)|\pi]$. Τα ιστορικά δεδομένα είναι ένα σύνολο από δεξαμενές, μία για κάθε χρήστη. Πιο συγκεκριμένα, το D είναι τα ιστορικά δεδομένα που περιέχουν n δεξαμενές $\{\tau_i\}$ για i=1,2,…,n, όπου η κάθε μία χαρακτηρίζεται από μια πολιτική συμπεριφοράς π_i, η οποία την δημιούργησε. Επιπλέον δίνεται

μια πολιτική π_e που παράχθηκε από τον αλγόριθμο Ενισχυτικής Μάθησης, και της οποίας πρέπει να υπολογισθεί η αποδοτικότητα.

5.3 ΑΞΙΟΛΟΓΗΣΗ ΕΚΤΟΣ ΠΟΛΙΤΙΚΗΣ (OFF-POLICY) ΜΕ ΠΙΘΑΝΟΤΙΚΕΣ ΕΓΓΥΗΣΕΙΣ

Οι High Confidence Off-Policy Evaluation (HCOPE) [35] μέθοδοι είναι μια οικογένεια μεθόδων που χρησιμοποιούν τα ιστορικά δεδομένα D προκειμένου να βρουν ένα κατώτατο όριο για την απόδοση της πολιτικής αξιολόγησης π_e με βαθμό εμπιστοσύνης 1-δ. Σε αυτό το paper χρησιμοποιούνται τρεις διαφορετικές προσεγγίσεις HCOPE, που όλες βασίζονται στο importance sampling. Ο εκτιμητής του importance sampling:

$$\hat{\rho}(\pi_e|\tau_i,\pi_i) := \underbrace{R(\tau_i)}_{\text{return}} \underbrace{\prod_{t=1}^{T} \frac{\pi_e(a_t^{\tau_i}|s_t^{\tau_i})}{\pi_i(a_t^{\tau_i}|s_t^{\tau_i})}}_{\text{importance weight}},$$

Εικόνα 5.1: Εκτιμητής importance sampling

είναι ένας αμερόληπτος εκτιμητής του ρ(π) αν το τ_i έχει παραχθεί από την πολιτική π_i.
Παρόλο που ο εκτιμητής του importance sampling είναι ευκολότερα κατανοητός, στα περισσότερα πειράματα της έρευνας που έγινε στο συγκεκριμένο paper χρησιμοποιήθηκε περισσότερο ο ανά βήμα εκτιμητής important sampling:

$$\hat{\rho}(\pi_e|\tau_i,\pi_i) := \sum_{t=1}^{T} \gamma^{t-1} r_t \left(\prod_{j=1}^{t} \frac{\pi_e(a_j^{\tau_i}|s_j^{\tau_i})}{\pi_i(a_j^{\tau_i}|s_j^{\tau_i})} \right)$$

Εικόνα 5.2: Ανά βήμα εκτιμητής importance sampling

Ο όρος στην παρένθεση είναι το βάρος της ανταμοιβής που παράχθηκε τη στιγμή t. Αυτός ο εκτιμητής έχει μικρότερη διακύμανση από τον πρώτο, αλλά είναι και αυτός αμερόληπτος.

Για συντομία, οι προσεγγίσεις του HCOPE περιγράφονται με τους όρους ενός συνόλου από μη-αρνητικές ανεξάρτητες τυχαίες μεταβλητές $X=\{X_i\}$, όπου $i=1,2,\ldots,n$. Έτσι, χρησιμοποιείται ο όρος $X_i = \rho(\pi_e/\tau_i, \pi_i)$, όπου το ρ υπολογίζεται είτε με την πρώτη, είτε με την δεύτερη σχέση. Οι τρεις προσεγγίσεις που θα χρησιμοποιηθούν είναι:

1) **Ανισότητα συγκέντρωσης** [36]**:** Εδώ χρησιμοποιείται η ανισότητα συγκέντρωσης (Concentration Inequality – CI) και αποκαλείται CI προσέγγιση. Το 1-δ κατώτατο όριο που παράγεται από την CI προσέγγιση το συμβολίζουμε με ρ^{CI}. Το όφελος από αυτή την ανισότητα συγκέντρωσης είναι ότι το κατώτατο όριό της είναι ένα πραγματικό κατώτατο όριο, δηλαδή δεν κάνει λάθος υπόθεση ή εκτίμηση, και έτσι θεωρείται ασφαλής.

2) **Student's t-test** [37]**:** Ένας τρόπος να γίνει πιο ακριβές το κατώτατο όριο που παράχθηκε από την CI προσέγγιση είναι να εισαχθεί μια ψευδής, πλην όμως λογική υπόθεση. Πιο συγκεκριμένα, εδώ επιστρατεύεται το θεώρημα του κεντρικού ορίου, σύμφωνα με το οποίο το $X:=\frac{1}{n}\sum_{i=1}^{n} Xi$ έχει κανονική κατανομή αν το n είναι μεγάλο. Υπό την υπόθεση ότι το X έχει κανονική κατανομή, μπορεί να εφαρμοστεί το Student's t-test για να παράξει το $\rho^{TT}(X,\delta)$, ένα 1-δ κατώτατο όριο του $\rho(\pi_e)$. Από τις 3 προσεγγίσεις, αυτή είναι η μοναδική που μπορεί να εξηγηθεί πλήρως με 3 σχέσεις, οπότε παρατίθεται εδώ:

$$\hat{X} := \frac{1}{n}\sum_{i=1}^{n} X_i, \qquad \sigma := \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left(X_i - \hat{X}\right)^2},$$

$$\rho_{-}^{\mathrm{TT}}(\mathbf{X},\delta) := \hat{X} - \frac{\sigma}{\sqrt{n}} t_{1-\delta,n-1},$$

Εικόνα 5.3: Σχέσεις που περιγράφουν την προσέγγιση Student's t-test

όπου με $t_{1-\delta,\nu}$ συμβολίζεται η αντίστροφη της συνάρτησης κατανομής t με ν βαθμούς ελευθερίας του Student t-test, υπολογισμένη με πιθανότητα 1-δ.

Επειδή το ρ^{TT} βασίζεται σε μια ψευδή (αλλά λογική) υπόθεση, η προσέγγιση θεωρείται ημι-ασφαλής. Παρόλο που η προσέγγιση TT παράγει στενότερα κατώτατα όρια από την CI, τείνει να είναι αρκετά συντηρητική για την εφαρμογή αυτήν (ad recommendation).

3) **Μεροληπτικά διορθωμένο και επιταχυνόμενο Bootstrap:** Ένας τρόπος να διορθωθεί η αρκετά συντηρητική φύση του TT αλγορίθμου είναι η χρήση boostrapping [38] για να εκτιμηθεί προσεγγιστικά αλλά με ακρίβεια η κατανομή του X, και στη συνέχεια να υποτεθεί ότι αυτή η εκτίμηση είναι και η σωστή. Η πιο γνωστή τέτοιου είδους προσέγγιση ονομάζεται "Bias Corrected and accelerated (BCa) bootstrap". Το κατώτατο όριο που παράγεται από τον αλγόριθμο BCa συμβολίζεται με $\rho^{BCa}(X,\delta)$.

Παρότι ημι-ασφαλής, η BCa προσέγγιση παράγει κατώτατα όρια το $\rho(\pi_e)$ που είναι στην πραγματικότητα μικρότερα από $\rho(\pi_e)$ για περίπου το (1-δ) τοις εκατό του χρόνου. Ενώ λοιπόν είναι ημι-ασφαλής η BCa, θεωρείται αρκετά αξιόπιστη για να χρησιμοποιηθεί σε πολλές εφαρμογές, και ιδίως στον ιατρικό τομέα.

5.4 CTR-LTV ΜΕΤΡΙΚΕΣ

Κάθε εξατομικευμένη πολιτική για σύσταση διαφημίσεων μπορεί να αξιολογηθεί είτε με βάση την άπληστη/βραχυπρόθεσμη απόδοσή του, είτε με βάση την μακροπρόθεσμη απόδοσή του. Για την άπληστη απόδοση, Click Through Rate (CTR) είναι μια λογική μετρική, ενώ η Life-Time Value (LTV) φαίνεται να είναι η σωστή επιλογή για τη μακροπρόθεσμη απόδοση. Αυτές οι δύο μετρικές ορίζονται ως:

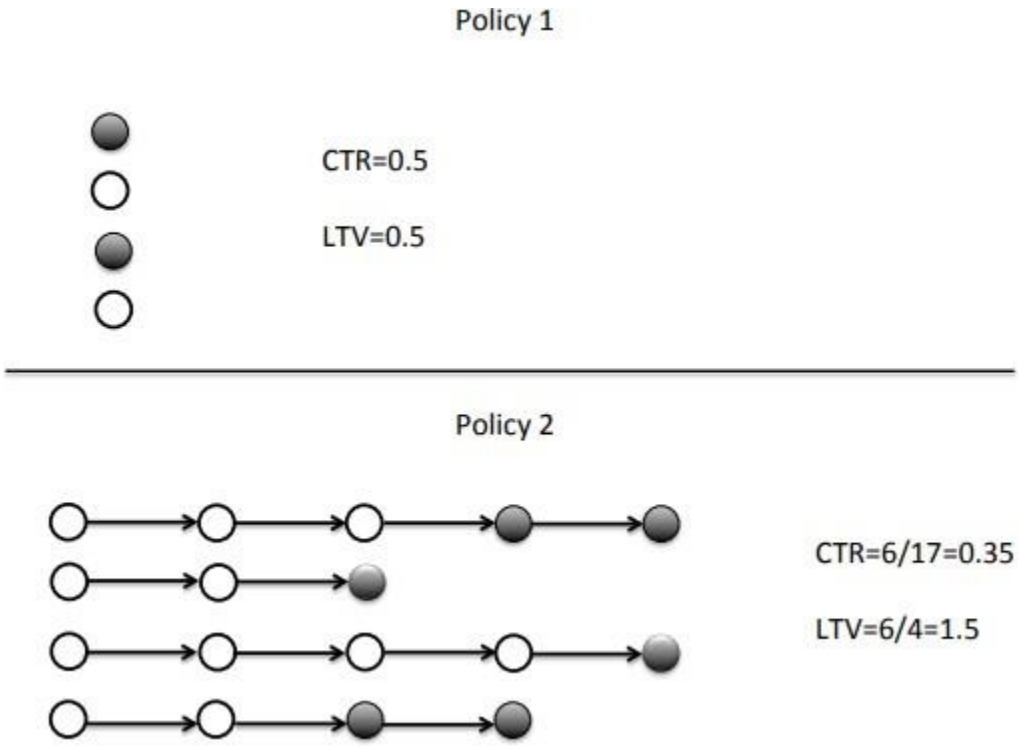
$$\text{CTR} = \frac{\text{Total \# of Clicks}}{\text{Total \# of } \mathbf{Visits}} \times 100,$$

$$\text{LTV} = \frac{\text{Total \# of Clicks}}{\text{Total \# of } \mathbf{Visitors}} \times 100.$$

Εικόνα 5.4: Ορισμός μετρικών CTR, LTV

Η CTR είναι μια καλά καθορισμένη μετρική στην ψηφιακή διαφήμιση και μπορεί να εκτιμηθεί από τα ισορικά δεδομένα (off-policy) τόσο με αμερόληπτες όσο και με μεροληπτικές μεθόδους. Στο συγκεκριμένο paper, επεκτείνεται η πρακτική προσέγγιση που έχει γίνει για LTV εκτίμηση, με την αντικατάσταση της ανισότητας συγκέντρωσης με τον t-test και με τον BCa, και εφαρμόζοντάς τους για πρώτη φορά σε πραγματικά online διαφημιστικά δεδομένα. Ο κύριος λόγος για τον οποίο χρησιμοποιείται LTV είναι ότι η CTR δεν είναι μια καλή μετρική για να εκτιμηθεί η μακροπρόθεσμη απόδοση και θα μπορούσε να οδηγήσει σε παραπλανητικά συμπεράσματα. Για παράδειγμα, μια άπληστη διαφημιστική στρατηγική σε κάποια ιστοσελίδα που εμφανίζει απευθείας μια διαφήμιση σχετική με το τελικό πρϊόν που θα μπορούσε να αγοράσει ένας χρήστης. Άλλο υποθετικό παράδειγμα, θα ήταν στην ιστοσελίδα της BMW να εμφανίζεται απευθείας διαφήμιση που λέει στον χρήστη πως αν αγοράσει κάποιο αυτοκίνητο, θα κερδίσει κάποια έκπτωση. Αν εμφανιστεί μια τέτοια διαφήμιση ο χρήστης είτε θα ψωνίσει κατευθείαν, είτε θα αποχωρήσει από την ιστοσελίδα. Από την άλλη πλευρά, μια άλλη στρατηγική marketing που προσπαθεί να ρίξει τον χρήστη σε ένα «χωνί» πωλήσεων, προτού του παρουσιάσει την έκπτωση, θα μπορούσε να είναι πιο επιτυχημένη. Για παράδειγμα, στην ιστοσελίδα της BMW θα μπορούσε αρχικά να εμφανιστεί σε κάποιον μια ελκυστική οικονομική προσφορά και μια καταπληκτική συμφωνία για τα service, πριν προταθεί η τελική έκπτωση. Τέτοιες μακροπρόθεσμες τακτικές θα κέρδιζαν περισσότερη αλληλεπίδραση με τον χρήστη και συνεπώς θα εκμαίευαν περισσότερα clicks ανά χρήστη και περισσότερες πωλήσεις. Το κρίσιμο στοιχείο εδώ είναι πως μια τέτοια πολιτική θα μπορούσε να αλλάξει τον αριθμό των φορών που

μια διαφήμιση θα εμφανιστεί σε κάποιον χρήστη. Στην Εικόνα 5.5 φαίνεται μια αναπαράσταση όσων εξηγήθηκαν:



Εικόνα 5.5: Σύγκριση άπληστης πολιτικής 1 με μακροπρόθεσμη πολιτική 2

Οι κύκλοι αναπριστούν επισκέψεις χρηστών. Οι μαύροι κύκλοι αναπαριστούν τα clicks. Η πολιτική 1 είναι άπληστη και οι χρήστες δεν επιστρέφουν. Η πολιτική 2 προσπαθεί να βελτιώσει την απόδοση μακροπρόθεσμα και οι χρήστες επιστρέφουν αρκετές φορές, και επιλέγουν αντικείμενα περισσότερες φορές. Παρόλο που η πολιτική 2 έχει μικρότερη CTR σε σχέση με την πολιτική 1, οδηγεί σε μεγαλύτερο κέρδος, αφού έχει υψηλότερη LTV. Έτσι, φαίνεται πως η LTV είναι αρκετά καλύτερη μετρική όσον αφορά τη Σύσταση Διαφημίσεων.

5.5 ΑΛΓΟΡΙΘΜΟΙ ΣΥΣΤΑΣΗΣ ΔΙΑΦΗΜΙΣΕΩΝ

Για άπληστη βελτιστοποίηση, χρησιμοποιήθηκε ένας Αλγόριθμος Random Forest (RF) [39] για να χαρτογραφήσει μια σχέση από χαρακτηριστικά σε ενέργειες. Οι αλγόριθμοι Ενισχυτικής Μάθησης αποτελούν μεθόδους που ενδείκνυνται για ταξινόμηση, όντας ικανοί να υπερταξινομήσουν (με την καλή έννοια) πολλά δεδομένα, συνεπώς είναι απολύτως κατάλληλοι

για προβλήματα σχετικά με Big Data. Το σύστημα εκπαιδεύεται με έναν αλγόριθμο Ενισχυτικής Μάθησης για κάθε μια από τις προτάσεις/ενέργειες ώστε να προβλέπει την άμεση ανταμοιβή. Κατά την εκτέλεση, χρησιμοποιείται μια ε-greedy [26] στρατηγική, κατά την οποία διαλέγεται με πιθανότητα 1-ε η πρόταση με την μεγαλύτερη προβλεπόμενη ανταμοιβή από την RF ή οι υπόλοιπες προτάσεις, η καθεμία με πιθανότητα ε/(|A|-1) (βλ. αλγόριθμο Εικόνας 5.6).

1: $y = \mathbf{X}_{\text{train}}(\text{reward})$
2: $x = \mathbf{X}_{\text{train}}(\text{features})$
3: $\bar{x} = \text{informationGain}(x, y)$ {feature selection}
4: $\text{rf}_a = \text{randomForest}(\bar{x}, y)$ {for each action}
5: $\pi_e = \text{epsilonGreedy}(\text{rf}, \mathbf{X}_{\text{test}})$
6: $\pi_b = \text{randomPolicy}$
7: $W = \hat{\rho}(\pi_e | \mathbf{X}_{\text{test}}, \pi_b)$ {importance weighted returns}
8: **return** $(\rho^{\dagger}_{-}(W, \delta), \text{rf})$ {bound and random forest}

Εικόνα 5.6: ε-greedy στρατηγική αλγορίθμου

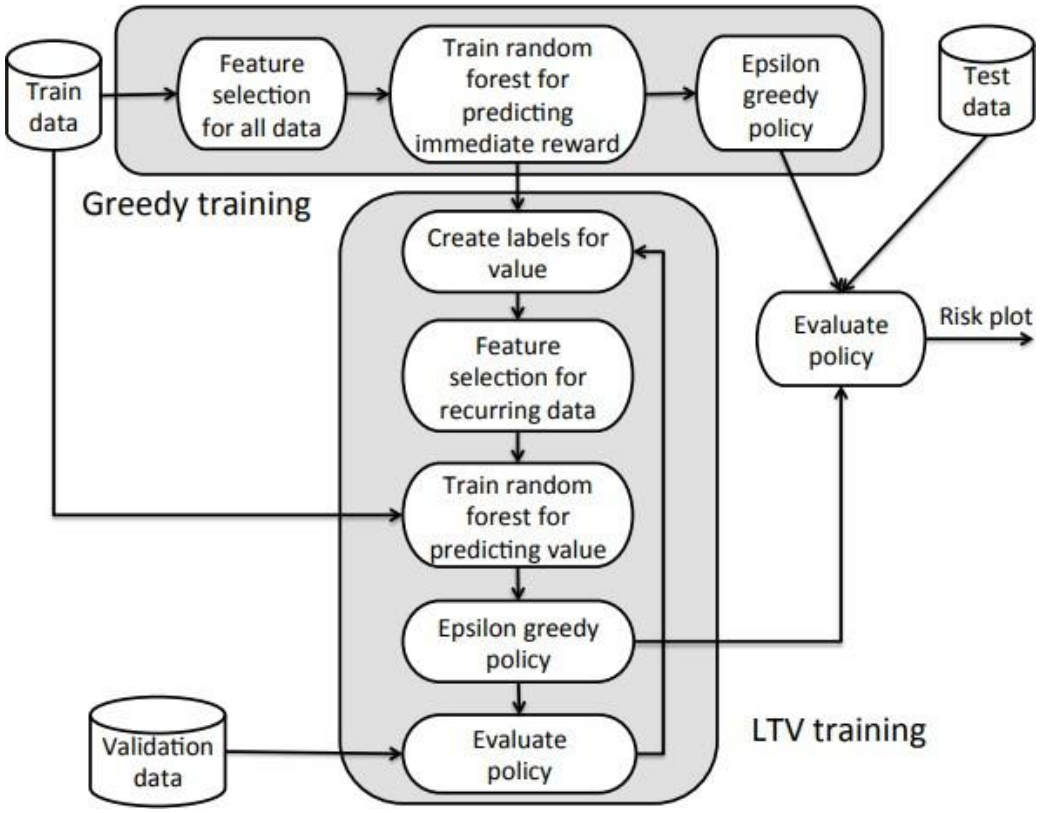
Για τη βελτιστοποίηση του LTV, χρησιμοποιήθηκε ένας υπερσύγχρονος αλγόριθμος Ενισχυτικής μάθησης, που ονομάζεται FQI [40], που επιτρέπει το χειρισμό μεταβλητών μεγάλων διαστάσεων που είναι συνεχείς και διακριτές. Όταν μια αυθαίρετη συνάρτηση προσέγγισης χρησιμοποιείται στον αλγόριθμο FQI, δε συγκλίνει μονοτονικά, αλλά αυξομειώνεται κατά τη διάρκεια διαδοχικών επαναληπτικών εκπαιδεύσεων. Για να εξομαλυνθεί το πρόβλημα της αυξομείωσης του FQI και για καλύτερη επιλογή χαρακτηριστικών διαφήμισης, χρησιμοποιήθηκε κατά τη διάρκεια της εκπαιδευτικής διαδικασίας η off-policy δομή αξιολόγησης HCOPE [35]. Η διαδικασία κρατάει ιστορικό των αποτελεσμάτων του FQI, ώστε να διαλέξει το καλύτερο με βάση ένα σύνολο δεδομένων επανάληψης (βλ. αλγόριθμο Εικόνας 5.7).

1: $\pi_b = \text{randomPolicy}$
2: $Q = \text{RF.GREEDY}(\mathbf{X}_{\text{train}}, \mathbf{X}_{\text{test}}, \delta)$ {start with greedy value function}
3: **for** $i = 1$ **to** K **do**
4: $r = \mathbf{X}_{\text{train}}(\text{reward})$ {use recurrent visits}
5: $x = \mathbf{X}_{\text{train}}(\text{features})$
6: $y = r_t + \gamma \max_{a \in A} Q_a(x_{t+1})$
7: $\bar{x} = \text{informationGain}(x, y)$ {feature selection}
8: $Q_a = \text{randomForest}(\bar{x}, y)$ {for each action}
9: $\pi_e = \text{epsilonGreedy}(Q, \mathbf{X}_{\text{val}})$
10: $W = \hat{\rho}(\pi_e | \mathbf{X}_{\text{val}}, \pi_b)$ {importance weighted returns}
11: $\text{currBound} = \rho^{\dagger}_{-}(W, \delta)$
12: **if** currBound $>$ prevBound **then**
13: prevBound = currBound
14: $Q_{\text{best}} = Q$
15: **end if**
16: **end for**
17: $\pi_e = \text{epsilonGreedy}(Q_{\text{best}}, \mathbf{X}_{\text{test}})$
18: $W = \hat{\rho}(\pi_e | \mathbf{X}_{\text{test}}, \pi_b)$
19: **return** $\rho^{\dagger}_{-}(W, \delta)$ {lower bound}

Εικόνα 5.7: Λειτουργία FQI αλγορίθμου

Και οι δύο αλγόριθμοι περιγράφονται γραφικά στο σχήμα της Εικόνας 5.8. Και για τους δύο αλγορίθμους στην αρχή υπάρχουν τρία σύνολα δεδομένων: X_{TRAIN}, X_{VAL}, X_{TEST}. Το κάθε ένα από αυτά φτιάχνεται από πλήρεις δεξαμενές χρηστών. Ένας χρήστης εμφανίζεται μόνο σε ένα από αυτά τα αρχεία. Τα X_{VAL} και X_{TEST} περιέχουν χρήστες που έχουν επιλεχθεί με τυχαία πολιτική. Η άπληστη προσέγγιση συνεχίζει κάνοντας πρώτα επιλογή χαρακτηριστικών στο X_{TRAIN}, εκπαιδεύει ένα τυχαίο δάσος (Forest), μετατρέπει την πολιτική σε ε-greedy στο X_{TEST} και μετά αξιολογεί αυτή την πολιτική χρησιμοποιώντας off-policy τεχνικές αξιολόγησης. Η LTV προσέγγιση ξεκινά από το τυχαίο μοντέλο δάσους της άπληστης προσέγγισης. Στη συνέχεια υπολογίζει τις ετικέτες (labels) όπως φαίνεται στο βήμα 6 του αλγορίθμου 2 από το X_{TRAIN}. Η πολιτική ελέγχεται και στη συνέχεια η διαδικασία αυτή επαναλαμβάνεται κατά έναν προκαθορισμένο αριθμό και με βάση

τα αποτελέσματα επιλέγεται η βέλτιστη πολιτική που θα εφαρμοστεί στο X_{TEST}. Τα τελικά αποτελέσματα είναι «risk plots», δηλαδή κάποια γραφήματα που δείχνουν το κατώτατο όριο του αναμενόμενου αθροίσματος από μειωμένη ανταμοιβή της πολιτικής για διαφορετικές μεταβλητές εμπιστοσύνης.



Εικόνα 5.8: Συνολική λειτουργία μοντέλου με greedy και LTV trainings

5.6 ΣΥΝΟΨΗ - ΑΞΙΟΛΟΓΗΣΗ

Σε αυτό το κεφάλαιο, παρουσιάστηκε ένα framework για εκπαίδεση και εκτίμηση εξατομικευμένων στρατηγικών πρότασης διαφημίσεων σε χρήστες. Αυτό το framework βασίζεται κυρίως σε μια οικογένεια τεχνικών off-policy εκτίμησης που αναπτύχθηκαν. Το μεγαλύτερο μέρος αυτού του κεφαλαίου αφορά στη χρήση αυτών των τεχνικών (HCOPE) σε συνδυασμό με αλγορίθμους Ενισχυτικής Μάθησης ώστε να αναπτυχθούν Συστήματα Συστάσεων που στόχο έχουν τη μεγιστοποίηση του Life-Time Value του εκάστοτε χρήστη (LTV). Παρόλα αυτά, αυτές οι HCOPE τεχνικές μπορούν επίσης να χρησιμοποιηθούν για να εκτιμήσουν την απόδοση μιας πιο βραχυπρόθεσμης προσέγγισης που βελτιστοποιεί το Click Through Rate

(CTR), και για να παρέχουν υψηλής ακρίβειας όρια για αυτήν. Έγιναν πειράματα που αφορούσαν σύνολα δεδομένων (datasets) που παράχθηκαν από πραγματικές διαφημίσεις και από πραγματικά συστήματα συστάσεων, ώστε να φανεί η αποδοτικότητα της στρατηγικής και για να φανούν κάποια από τα θέματα που συζητήθηκαν, όπως η LTV έναντι της βραχυπρόθεσμης βελτιστοποίησης, οι μετρήσεις αποδοτικότητας με βάση της μετρικής LTV έναντι των μετρήσεων με βάση την CTR, και η αξία της χρήσης μεγάλης ακρίβειας off-policy αξιολόγησης στην εκμάθηση και στην αξιολόγηση πολιτικών Ενισχυτικής Μάθησης.

Συγκεντρωτικά, η συνεισφορά της συγκεκριμένης έρευνας μπορεί να συνοψιστεί ως εξής:

1. Σε αντίθεση με την πλειοψηφία της υπάρχουσας γνώσης σε συστήματα συστάσεων διαφημίσεων, δείχθηκε πόσο σημαντική είναι η εστίαση της προσοχής στην βελτίωση της LTV μετρικής και πως η βελτίωση αυτή μπορεί να οδηγήσει σε επιθυμητά αποτελέσματα. Για παράδειγμα, μπόρεσαν να παραχθούν εξαιρετικά αποτελέσματα από μια πραγματική καμπάνια διαφημίσεων με ένα σχετικά μικρό σύνολο ιστορικών δεδομένων για τους χρήστες και τις διαφημίσεις.
2. Καθορίστηκε η σχέση μεταξύ των μετρικών CTR και LTV και εξηγήθηκε εμπειρικά το γιατί η CTR μπορεί να μην είναι μια απολύτως σωστή μετρική για να αξιολογηθεί η απόδοση ενός συστήματος συστάσεων διαφημίσεων με πολλούς χρήστες να κάνουν επαναληπτικές επισκέψεις στην εκάστοτε ιστοσελίδα.
3. Είναι η πρώτη έρευνα που έγινε και χρησιμοποίησε πραγματικά δεδομένα πάνω στα οποία προσπάθησε να βελτιστοποιήσει την LTV πολιτική και με αυτόν τον τρόπο δείχνει πως η προτεινόμενη τεχνική δίνει εγγυήσεις για την εγκυρότητά της και είναι πολλά υποσχόμενη.
4. Συνδυάστηκαν υπερσύγχρονες τεχνικές, όπως οι μέθοδοι HCOPE ώστε να σχεδιαστούν αλγόριθμοι που μαθαίνουν αποδοτικά μια πολιτική σύστασης διαφημίσεων με μια καλή CTR ή LTV μέτρηση αποδοτικότητας.

Όπως και στις προηγούμενες περιπτώσεις φάνηκε ότι το να μελετώνται μακροπρόθεσμα τα Συστήματα Συστάσεων οποιουδήποτε τομέα αποφέρει πολύ καλύτερα αποτελέσματα από τις περιπτώσεις όπου το Σύστημα Συστάσεων λειτουργεί με άπληστη (greedy) πολιτική και στοχεύει στο καλύτερο βραχυπρόθεσμο κάθε φορά αποτέλεσμα. Φαίνεται πως η επιστροφή του χρήστη στη σελίδα και η επιρροή που ασκείται από τον χρήστη στη σελίδα και αντίστροφα, είναι υψίστης σημασίας για την αποδοτικότητα του αλγορίθμου συστάσεων και, συνεπώς, πρέπει να μελετάται και να αναλύεται ενδελεχώς.

ΚΕΦΑΛΑΙΟ 6

ΣΥΜΠΕΡΑΣΜΑΤΑ – ΓΕΝΙΚΗ ΑΞΙΟΛΟΓΗΣΗ

6.1 ΕΠΙΣΚΟΠΗΣΗ

Στα προηγούμενα κεφάλαια είδαμε συγκεκριμένους αλγορίθμους που εφάρμοζαν την τεχνική της Ενισχυτικής Μάθησης σε Συστήματα Συστάσεων σε διάφορους τομείς, όπως η μουσική, οι ειδήσεις, το ηλεκτρονικό εμπόριο (e-commerce) και η διαφήμιση. Παρόλο που σε κάθε τομέα, η εφαρμογή της Ενισχυτικής Μάθησης είχε ορισμένες διαφοροποιήσεις, σύμφωνα και με τη φύση της εκάστοτε πλατφόρμας, υπάρχουν ορισμένα κοινά θετικά που αποκομίστηκαν. Επίσης, εντοπίστηκαν και κάποιες κοινές δυσκολίες στην εφαρμογή της Ενισχυτικής Μάθησης και στη λειτουργία της.

Στη συνέχεια θα γίνει μια σύνοψη των κοινών θετικών αυτών στοιχείων, των δυσκολιών, αλλά και των ωφελών της εφαρμογής της Ενισχυτικής Μάθησης σε κάθε έναν τομέα ξεχωριστά.

6.2 ΚΟΙΝΑ ΩΦΕΛΗ ΚΑΙ ΔΥΣΚΟΛΙΕΣ

6.2.1 ΘΕΤΙΚΑ

Αρχικά, το σημαντικότερο κέρδος, που φάνηκε σε όλες τις εφαρμογές ήταν το γεγονός ότι αυτές που χρησιμοποιούν ως εργαλείο την Ενισχυτική Μάθηση λειτουργούν μακροπρόθεσμα, καλύτερα από κάθε άλλη εφαρμογή με την οποία συγκρίθηκαν. Αυτό συμβαίνει κυρίως, διότι η Ενισχυτική Μάθηση μπορεί εξ ορισμού να διορθώσει στην πορεία λάθη που έγιναν κατά τη διαδικασία εκμάθησης και δε πορεύεται εσαεί με βάση την αρχική εκπαίδευση του συστήματος. Επίσης, κάθε λάθος που διορθώνεται από το μοντέλο, έχει απειροελάχιστες πιθανότητες να επαναληφθεί στο μέλλον. Φαίνεται για αυτό το λόγο, πως το Reinforcement Learning είναι ο τύπος Μηχανικής Μάθησης που είναι πιο κοντά στη διαδικασία εκμάθησης και διόρθωσης λαθών από τον άνθρωπο. Συνεπώς, είναι ο πιο συνεπής και πλησιάζει κατά το μέγιστο την τελειότητα.

Επιπλέον, από τη διαφορετικότητα της κάθε εφαρμογής, καταλαβαίνουμε πως η Ενισχυτική Μάθηση είναι εξαιρετικά προσαρμοστική, κι έτσι έχει τη δυνατότητα να φτιάξει το καταλληλότερο μοντέλο συστάσεων σε κάθε περίπτωση. Είναι η πιο κατάλληλη μορφή Μηχανικής Μάθησης για χρήση σε τέτοιου είδους εφαρμογές, αφού ένα από τα κύρια χαρακτηριστικά της είναι η δυνατότητά της να συλλέγει πληροφορίες για το περιβάλλον, μέσω

της αλληλεπίδρασης που έχει με αυτό. Η αλληλεπίδραση με τον χρήστη είναι πολύ βασικό στοιχείο των Συστημάτων Συστάσεων.

Σημαντικό είναι πως ένα Σύστημα Ενισχυτικής Μάθησης ακόμα και εν τη απουσία ενός συνόλου δεδομένων για εκπαίδευση, μπορεί να μάθει απο την εμπειρία που αποκομίζει μόνο του. Προφανώς σε αυτή την περίπτωση θα υπάρχει κόστος σε συνέπεια, ειδικά στο πρώιμα στάδια λειτουργίας μιας εφαρμογής (όπου η εμπειρία είναι προφανώς μηδαμινή) αλλά το γεγονός πως οι αλγόριθμοι που είναι βασισμένοι σε Ενισχυτική Μάθηση μπορούν να ανταπεξέλθουν και κάτω από τέτοιες συνθήκες, μπορεί μόνο ως πλεονέκτημα να θεωρηθεί.

Τέλος, ένα στοιχείο της Ενισχυτικής Μάθησης που την καθιστά άκρως κατάλληλη για εφαρμογές σε Συστήματα Συστάσεων, είναι η ισορροπία που μπορεί να επιτύχει ανάμεσα στην αναζήτηση (exploration) νέων αντικειμένων και την εκμετάλλευση (exploitation) της ήδη υπάρχουσας γνώσης. Η αναζήτηση είναι η διαδικασία δοκιμής νέων αντικειμένων προκειμένου να φανεί αν μπορούν να λειτουργήσουν καλύτερα από όσα έχουν ήδη δοκιμαστεί, ενώ η αξιοποίηση είναι η πιο ασφαλής διαδικασία επαναχρημοποίησης γνωστών αντικειμένων, με γνωστή απήχηση όταν επιλέγονται προς σύσταση. Πολλοί αλγόριθμοι Μηχανικής Μάθησης χρησιμοποιούν τις δύο αυτές τεχνικές, παρόλα αυτά, η Ενισχυτική Μάθηση είναι αποδεδειγμένα η μοναδική που επιτυγχάνει τόσο καλή ισορροπία ανάμεσά τους [41].

6.2.2 ΑΡΝΗΤΙΚΑ

Τα αρνητικά της εφαρμογής της Ενισχυτικής Μάθησης σε Συστήματα Συστάσεων έχουν να κάνουν περισσότερο με τις δυσκολίες που προκύπτουν στην εφαρμογή της και στη σωστή χρησιμοποίησή της, και όχι με την αποτελεσματικότητα, που όπως αναφέρθηκε παραπάνω είναι ανώτερη, τουλάχιστον στα πεδία εφαρμογής που αναλύθηκαν.

Βασικό είναι ότι πρέπει η Ενισχυτική Μάθηση να εφαρμόζεται ελεγχόμενα, αφού η παρατεταμένη χρήση του μπορεί να οδηγήσει στη δημιουργία πολλών καταστάσεων (states), το οποίο μπορεί να φανεί καταστροφικό ως προς τα αποτελέσματα. Για αυτό το λόγο, δεν είναι καθόλου κατάλληλο για εφαρμογή σε πολύ απλά προβλήματα: με την εφαρμογή του Reinforcement Learning γίνονται πολύ πιο σύνθετα, δίχως να υπάρχει ουσιαστικός λόγος.

Ένα στοιχείο που δημιουργεί μεγάλη δυσκολία είναι το γεγονός ότι η Ενισχυτική Μάθηση είναι framework που χρειάζεται πολλά δεδομένα για να λειτουργήσει σωστά. Γι’ αυτό είδαμε να εφαρμόζεται με μεγάλη αποτελεσματικότητα σε εφαρμογές με συνεχή και μεγάλη ροή δεδομένων, όπως η σύσταση ειδήσεων, η σύσταση μουσικής, η σύσταση στο ηλεκτρονικό

εμπόριο (e-commerce) κλπ. Παρόλα αυτά, όταν δεν υπάρχουν δεδομένα δημιουργείται πρόβλημα.

Επίσης, το framework αυτό, θεωρεί πάντα ότι ο κόσμος είναι Μαρκοβιανός (Markovian), που δεν ισχύει [42]. Το Μαρκοβιανό μοντέλο περιγράφει μια σειρά από πιθανές ενέργειες, που η πιθανότητα της κάθε ενέργειας εξαρτάται μόνο από την κατάστση που βρισκόμαστε, η οποία με τη σειρά της εξαρτάται από την αμέσως προηγούμενη ενέργεια. Αυτό δεν ανταποκρίνεται απόλυτα στην πραγματικότητα, αφού ενέργειες και επόμενες καταστάσεις συχνά εξαρτώνται από εξωγενείς, μη προβλέψιμους παράγοντες.

Συνήθως η Ενισχυτική Μάθηση δεν μπορεί να σταθεί μόνη της, καθώς χρειάζεται βοήθεια άλλων τεχνικών για να δημιουργήσει το εκάστοτε κατάλληλο μοντέλο. Είδαμε συχνά να χρησιμοποιούνται τα Q-Networks, S-Networks αλλά και το Deep Learning.

Στον Πίνακα 6.1 φαίνονται συγκεντρωτικά τα θετικά και αρνητικά της εφαρμογής Ενισχυτικής Μάθησης σε Συστήματα Συστάσεων.

ΘΕΤΙΚΑ	ΑΡΝΗΤΙΚΑ
Βελτιστοποίηση μακροπρόθεσμων μετρικών και σχέσης χρήστη-πλατφόρμας	Υπεβολική χρήση οδηγεί στην υπερανάλυση του προβλήματος και δημιουργία πολλών καταστάσεων που αυξάνουν την πολυπλοκότητα του αλγορίθμου και μπορούν να λειτουργήσουν πολύ αρνητικά ως προς την αποδοτικότητά του
Διόρθωση (μέσω εμπειρίας) λαθών που έγιναν κατά τη διαδικασία εκπαίδευσης	Μη κατάλληλο για εφαρμογή σε απλά προβλήματα
Σχεδόν μηδενική πιθανότητα επανάληψης λάθους	Ανάγκη για πολλά δεδομένα
Προσαρμοστικότητα στην εκάστοτε εφαρμογή	Υποθέτει πάντα ότι ο κόσμος λειτουργεί σύμφωνα με το Μαρκοβιανό πρότυπο
Δυνατότητα εκμάθησης, ακόμα και ελλείψει Training Dataset	Συνήθως χρειάζεται συνεισφορά άλλων τεχνικών (π.χ. Q-Network)
Ισορροπία μεταξύ exploration και exploitation	

Εικόνα 6.1: Γενικά θετικά και αρνητικά εφαρμογής Ενισχυτικής Μάθησης σε Συστήματα Συστάσεων

6.3 ΩΦΕΛΗ ΚΑΙ ΔΥΣΚΟΛΙΕΣ ΑΝΑ ΤΟΜΕΑ

6.3.1 ΣΥΣΤΗΜΑΤΑ ΣΥΣΤΑΣΕΩΝ ΗΛΕΚΤΡΟΝΙΚΟΥ ΕΜΠΟΡΙΟΥ

Στον τομέα του Ηλεκτρονικού Εμπορίου, η βελτιστοποίηση των μακροπρόθεσμων μετρικών ήταν ύψιστης σημασίας, αφού είναι πολύ σημαντικό ο χρήστης να επανέρχεται συχνά στην πλατφόρμα γιατί υπάρχει μεγάλη ανανέωση πληροφορίας και νέα αντικείμενα εμφανίζονται συνεχώς. Έτσι, είναι σαφές πως χρειάζεται και εξερεύνηση νέων αντικειμένων, που καθιστά την Ενισχυτική Μάθηση άκρως κατάλληλη, με το κόστος φυσικά να είναι η αύξηση της πολυπλοκότητας στη δημιουργία της εφαρμογής και την επιστράτευση άλλων τεχνικών για υποβοήθηση στην Ενισχυτική Μάθηση.

ΘΕΤΙΚΑ	ΑΡΝΗΤΙΚΑ
Βελτιστοποίηση μακροπρόθεσμων μετρικών	Δυσκολία στη μοντελοποίηση των μακρπρόθεσμων μετρικών (dwell time, ικανοποίηση χρήστη κλπ.)
Διαρκής εκμάθηση, που είναι απαραίτητη με τόσα νέα δεδομένα σε e-commerce συστήματα	Δυσκολία στη βελτιστοποίηση των μακροπρόθεσμων μετρικών
Ευνοεί τις συνεχείς επισκέψεις του χρήστη, οι οποίες είναι απαραίτητες με τόσο πυκνή ροή δεδομένων	Ανάγκη χρήσης πολλαπλών επιπέδων Q-Learning
Μέσω του exploration βρίσκει συνεχώς νέα αντικείμενα, που είναι επίσης απαραίτητο γιατί διαρκώς εμφανίζονται καινούργια	Ανάγκη δημιουργίας S-Network για προσομοίωση συμπεριφοράς χρήστη (Υπάρχει δυσκολία στη μοντελοποίηση του Feedback)

Εικόνα 6.2: : Θετικά και αρνητικά εφαρμογής Ενισχυτικής Μάθησης σε e-commerce Συστήματα Συστάσεων

6.3.2 ΣΥΣΤΗΜΑΤΑ ΣΥΣΤΑΣΕΩΝ ΓΙΑ ΕΙΔΗΣΕΙΣ

Οι πλατφόρμες ανάγνωσης ειδήσεων είναι φυσικό να έχουν συνεχώς νέα αντικείμενα (άρθρα ειδήσεων) προς σύσταση. Συνεπώς, είναι πολύ σημαντικό όχι μόνο να γίνεται συνεχώς εξερεύνηση νέων αντικειμένων, αλλά και να γίνεται σωστή επιλογή των προς σύσταση ειδήσεων, ανάλογα με το profile του εκάστοτε χρήστη, ώστε η σύσταση να μην είναι άστοχη και ανούσια. Εδώ, εμφανίζονται και πάλι τα γνωστά προβλήματα εφαρμογής Ενισχυτικής Μάθησης όπως αναλύθηκαν και προηγουμένως, και αφορούν στην δυσκολία στη μοντελοποίηση και στην αναγκαστική χρησιμοποίηση άλλων αλγορίθμων που βοηθούν την υλοποίηση.

ΘΕΤΙΚΑ	ΑΡΝΗΤΙΚΑ
Λαμβάνονται υπόψιν πολλές περισσότερες μετρικές (και μακροπρόθεσμες) απ' ότι σε αλγορίθμους που έχουν μελετηθεί για το ίδιο πρόβλημα	Χρήση Q-Learning για την πρόβλεψη μελλοντικής ανταμοιβής από το χρήστη. Συνεπώς αυξάνεται η συνολική πολυπλοκότητα του αλγορίθου
Γίνεται πολύ σημαντικό exploration σε νέα άρθρα ειδήσεων	Δυσκολία στη μοντελοποίηση της συμπεριφοράς του χρήστη
Δημιουργείται profile χρήστη, ώστε να ξεχωρίζονται οι ενδιαφέρουσες ειδήσεις από τις αδιάφορες	

Εικόνα 6.3: Θετικά και αρνητικά εφαρμογής Ενισχυτικής Μάθησης σε Συστήματα Συστάσεων για Ειδήσεις

6.3.3 ΣΥΣΤΗΜΑΤΑ ΣΥΣΤΑΣΕΩΝ ΓΙΑ ΜΟΥΣΙΚΗ

Η μουσική, όπως και οι ειδήσεις, είναι ένας τομέας στον οποίο ο αλγόριθμος πρέπει να λαμβάνει υπόψιν την τεράστια ανάγκη για εξερεύνηση, όπως επίσης και το profile του κάθε χρήστη, αφού αν οι ειδήσεις πρέπει να είναι εξατομικευμένες, καταλαβαίνουμε πόσο πιο ιδιαίτερη είναι η ανάγκη για εξατομίκευση στη μουσική. Φυσικά, τα τραγούδια δεν μπορούν να προτείνονται μόνο βάσει είδους μουσικής (π.χ. rock, pop, jazz κλπ.). Υπάρχει ανάγκη για ανάλυση των τραγουδιών με πολλά περισσότερα χαρακτηριστικά όσον αφορά το ρυθμό και πολλές άλλες παραμέτρους, αφού ο σκοπός του αλγορίθμου δεν είναι η απλή και παραδοσιακή σύσταση τραγουδιών, αλλά λιστών αναπαραγωγής. Συνεπώς, έχει σημασία όχι μόνο ποια τραγούδια θα προταθούν, αλλά και σε ποια σειρά.

ΘΕΤΙΚΑ	ΑΡΝΗΤΙΚΑ
Δίνεται η δυνατότητα δημιουργίας δυναμικών λιστών προς σύσταση και όχι απλώς τραγουδιών	Δυσκολία στη μοντελοποίηση των τραγουδιών, λόγω της ανάγκης διάσπασής τους σε πολλά χαρακτηριστικά
Μέσω των πολλών χαρακτηριστικών στα οποία αναλύεται το κάθε τραγούδι, μπορεί να μελετηθεί η μετάβαση από τραγούδι σε τραγούδι	Δυσκολία μοντελοποίησης της συνάρτησης ανταμοιβής του χρήστη
Δημιουργείται profile για κάθε χρήστη ώστε να γίνει η καλύτερη δυνατή εξατομίκευση	

Εικόνα 6.4: Θετικά και αρνητικά εφαρμογής Ενισχυτικής Μάθησης σε Συστήματα Συστάσεων για Μουσική

6.3.4 ΣΥΣΤΗΜΑΤΑ ΣΥΣΤΑΣΕΩΝ ΓΙΑ ΔΙΑΦΗΜΙΣΕΙΣ

Στον τομέα των διαφημίσεων ήταν απαραίτητο να διαμορφωθεί το profile του εκάστοτε χρήστη, αφού οι διαφημίσεις είναι ένα αντικείμενο που συνήθως έχει μικρή απήχηση. Ο περισσότερος κόσμος τις αγνοεί, ή ακόμα χειρότερα δυσανασχετεί με αυτές. Συνεπώς, η διαφήμιση πρέπει να είναι απόλυτα στοχευμένη αν θέλει να πετύχει το σκοπό της. Και στην περίπτωση αυτή, ο αλγόριθμος εστίασε στη βελτίωση μακροπρόθεσμων μετρικών που δεν είχαν μελετηθεί σε προηγούμενες έρευνες και επίσης κρατώντας ιστορικό για κάθε χρήστη, κατάφερε να διαχωρίσει την επίσκεψη από τον επισκέπτη όπως εξηγήθηκε στο Κεφάλαιο 5.

Βέβαια, τα αναπόφευκτα προβλήματα της μοντελοποίησης και της αναγκαστικής χρήσης (πολλών μάλιστα) επιπρόσθετων αλγορίθμων δεν έλειψαν ούτε από τη συγκεκριμένη εφαρμογή.

ΘΕΤΙΚΑ	ΑΡΝΗΤΙΚΑ
Δημιουργία profile χρήστη για εξατομικευμένες συστάσεις	Δυσκολία στη μοντελοποίηση των δεδομένων που αναπαριστούν τις διαφημίσεις
Δόθηκε η δυνατότητα για βελτίωση μακροπρόθεσμων μετρικών	Δυσκολία στον υπολογισμό της ανταμοιβής του χρήστη
Λόγω του ιστορικού κάθε χρήστη, ήταν εφικτό να γίνει διαχωρισμός μεταξύ επισκέπτη και επίσκεψης	Χρήση (πολλών) επιπλέον αλγορίθμων για να καταστεί δυνατή η μοντελοποίηση των δεδομένων (π.χ. Random Forest Algorithm)

Εικόνα 6.5: Θετικά και αρνητικά εφαρμογής Ενισχυτικής Μάθησης σε Συστήματα Συστάσεων για Διαφημίσεις

6.3.5 ΑΞΙΟΛΟΓΗΣΗ

Μετά από την αναλυτική και εκτενή ανάλυση που κάναμε στον κάθε έναν από τους αλγορίθμους που αναλύσαμε, και αφού στο Κεφάλαιο 6 σημειώσαμε τα θετικά και τα αρνητικά της εφαρμογής της Ενισχυτικής Μάθησης σε κάθε περίπτωση, είμαστε σε θέση να κρίνουμε συνολικά την συμμετοχή της Ενισχυτικής Μάθησης στα Συστήματα Συστάσεων.

Είδαμε πως σε όλες τις περιπτώσεις οι αλγόριθμοι που χρησιμοποίησαν ως εργαλείο την Ενισχυτική Μάθηση λειτούργησαν πολύ αποδοτικότερα από οποιονδήποτε αλγόριθμο συγκρίθηκαν. Συνεπώς, μπορούμε να συμπεράνουμε ότι η αρχική υπόθεση, πως η Ενισχυτική Μάθηση είναι μάλλον η πιο κατάλληλη μορφή Μηχανικής Μάθησης για χρήση σε εφαρμογές με μεγάλο όγκο δεδομένων, είναι σωστή για τους τομείς που μελετήθηκαν.

Ουσιαστικά, τα μοναδικά «μελανά» σημεία που καταγράψαμε, αφορούν στην δυσκολία υλοποίησης και όχι σε αρνητικά αποτελέσματα κατά την εκτέλεση. Ομολογουμένως, η απόφαση για χρήση Ενισχυτικής Μάθησης, περιπλέκει αρκετά τα πράγματα από άποψη υλοποίησης, αλλά αφού πρόκειται για τόσο σημαντική διαφορά στα αποτελέσματα, μάλλον η εύκολη υλοποίηση είναι ένα στοιχείο που αξίζει να θυσιαστεί.

Bibliography

[1] P. G. Roetzel, "Information overload in the information age: a review of the literature from business administration, business psychology, and related disciplines with a bibliometric approach and framework development," 2019.

[2] F. R. a. L. R. a. B. Shapira, Introduction to Recommender Systems Handbook, 2011.

[3] M. Mohri, "Foundations of Machine Learning," *The MIT Press,* 2012.

[4] G. Hinton and T. Sejnowski, "Unsupervised Learning: Foundations of Neural Computation," *MIT Press,* 1999.

[5] L. P. Kaelbling, M. L. Littman and A. W. Moore, "Reinforcement Learning: A Survey," *Journal of Artificial Intelligence Research,* 2001.

[6] P. W. Farris, N. T. Bendle, P. E. Pfeifer and D. J. Reibstein, Marketing Metrics: The Definitive Guide to Measuring Marketing Performance., 2010.

[7] J. Bobadilla, F. Ortega, A. Hernando and J. Bernal, A collaborative filtering approach to mitigate the new user cold start problem, 2012.

[8] L. X. Z. D. J. S. W. L. D. Y. Lixin Zou, "Reinforcement Learning to Optimize Long-term User Engagement in Recommender Systems," 2019.

[9] C. Rickman, The Digital Business Start-Up Workbook: The Ultimate Step-by-Step Guide to Succeeding Online from Start-up to Exit, 2012.

[10] D. P. Kroese, T. Brereton, T. Taimre and Z. I. Botev, " Why the Monte Carlo method is so important today," 2014.

[11] D. Hubbard and D. A. Samuelson, "Modeling Without Measurements," 2009.

[12] R. B. A. Sutton, "Time Derivative Models of Pavlovian Reinforcement," 1990.

[13] L. Breiman, " Bagging Predictors," 1994.

[14] Y. D. F. S. M. H. N. S. J. M. Hado van Hasselt, "Deep Reinforcement Learning and the Deadly Triad," 2018.

[15] R. Bellman, "A Markovian Decision Process," *Journal of Mathematics,* 1957.

[16] R. A. Howard, " Dynamic Programming and Markov Processes," *The M.I.T. Press,* 1960.

[17] T. Matiisen, "Demystifying Deep Reinforcement Learning," *Computational Neuroscience Lab,* 2018.

[18] C. Watkins and P. Dayan, "Q-learning," 1992.

[19] S. Hochreiter and J. Schmidhuber, Long short-term memory, 1997.

[20] D. Bouneffouf, A. Bouzeghoub and A. L. Gançarski, A Contextual-Bandit Algorithm for Mobile Context-Aware Recommender System, 2012.

[21] W. R. Ashby, Design for a Brain, 1960.

[22] T. Seno, "Welcome to Deep Reinforcement Learning Part 1 : DQN," *towardsdatascience.com,* 2017.

[23] H. D. X. C. H. Z. W. W. X. T. R. Z. Y. &. Y. Y. Chen, " Large-Scale Interactive Recommendation with Tree-Structured Policy Gradient. Proceedings of the AAAI Conference on Artificial Intelligence, 33(01), 3312-3320.," 2019.

[24] F. Z. Z. Z. Y. X. N. J. Y. X. X. Z. L. Guanjie Zheng, "DRN: A Deep Reinforcement Learning Framework for News Recommendation," 2018.

[25] Y. Yue, J. Broder, R. Kleinberg and T. Joachims, "The K-armed dueling bandits problem," *Journal of Computer and System Sciences,* 2012.

[26] M. Tokic, "Adaptive ε-greedy exploration in reinforcement learning based on value differences," 2010.

[27] P. Auer, "Using upper confidence bounds for online learning," 2000.

[28] L. Kocsis, Discounted UCB, 2006.

[29] J. Hu, H. Niu, J. Carrasco, B. Lennox and F. Arvin, "Voronoi-Based Multi-Robot Autonomous Exploration in Unknown Environments via Deep Reinforcement Learning," 2020.

[30] M. S.-T. P. S. Elad Liebman, "DJ-MC: A Reinforcement-Learning Agent for Music Playlist Recommendation," 2015.

[31] M. Wiering and J. Schmidhuber, "Fast Online Q(λ)," 1998.

[32] M. Enzenberger and M. Mūller, "Fuego – An Open-Source Framework for Board Games and Go Engine Based on Monte Carlo Tree Search," 2008.

[33] A. Struyf, M. Hubert and P. Rousseeuw, "Clustering in an Object-Oriented Environment," *Journal of Statistical Software,* 1997.

[34] P. S. T. M. G. Georgios Theocharous, "Personalized Ad Recommendation Systems for Life-Time Value Optimization with Guarantees," 2015.

[35] G. T. M. G. Philip S. Thomas, "High Confidence Off-Policy Evaluation," 2015.

[36] F. Chung and L. Lu, "Old and new concentration inequalities," 2010.

[37] E. Weisstein, "Student's t-Distribution," *mathworld.wolfram.com.*

[38] J. A. Aslam, R. A. Popa and R. L. and Rivest, "On Estimating the Size and Confidence of a Statistical Audit," 2007.

[39] T. K. Ho, "Random Decision Forests," *Proceedings of the 3rd International Conference on Document Analysis and Recognition,,* 1995.

[40] M. Riedmiller, "Neural Fitted Q Iteration - First Experiences with a Data Efficient Neural Reinforcement".

[41] M. Coggan, "Exploration and Exploitation in Reinforcement Learning," 2004.

[42] M. L. L. A. W. M. L. P. Kaelbling, "Reinforcement Learning: A Survey," 1996.