ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ
ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
"ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ & ΥΠΗΡΕΣΙΕΣ"
ΚΑΤΕΥΘΥΝΣΗ " ΠΡΟΗΓΜΕΝΑ ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ"

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

" Συνεργατικές προσεγγίσεις μάθησης σε συστήματα συστάσεων. "

Παναγιώτα Παπαευσταθίου

ΜΕ1834

Επιβλέπουσα καθηγήτρια: Μαρία Χαλκίδη

ΑΘΗΝΑ, ΟΚΤΩΒΡΙΟΣ 2020

Περιεχόμενα

Πίνακας Εικόνων

Κατάλογος Πινάκων

Περίληψη

Η συγκεκριμένη εργασία αποτελεί προϊόν μεταπτυχιακής διατριβής που πραγματοποιήθηκε στα πλαίσια του «Μεταπτυχιακού Προγράμματος Σπουδών στα Πληροφορικά Συστήματα & Υπηρεσίες» με κατεύθυνση τα Προηγμένα Πληροφοριακά Συστήματα του Πανεπιστημίου Πειραιώς. Σκοπός της είναι η μελέτη των διαφορετικών τεχνικών federated learning σε συστήματα συστάσεων, η υλοποίηση μιας από αυτές και η διεξαγωγή πειραμάτων.

Στο πρώτο κεφάλαιο γίνεται εισαγωγή στη συνεργατική προσέγγιση μάθησης (federated learning). Στη συνέχεια, γίνεται εκτενής αναφορά στις κατηγορίες της και στην αρχιτεκτονική της. Το επόμενο κεφάλαιο περιλαμβάνει τον ορισμό των συστημάτων συστάσεων, τα αίτια χρήσης τους, τη διαδικασία σύστασης και τις μορφές της. Επίσης, αναλύονται οι διαφορετικοί τύποι των συστημάτων συστάσεων. Στο τρίτο κεφάλαιο, γίνεται αναφορά στην βασική ιδέα παρόμοιων σχετικών δημοσιεύσεων πάνω στην συνεργατική προσέγγιση μάθησης σε συστήματα συστάσεων και παρουσιάζεται ένας συγκεντρωτικός πίνακας. Τέλος, στο τέταρτο κεφάλαιο περιγράφεται η υλοποίηση της federated learning εφαρμογής και παρουσιάζονται τα πειραματικά αποτελέσματα.

Λέξεις-κλειδιά: συνεργατικές προσεγγίσεις μάθησης (federated learning), συστήματα συστάσεων (recommender systems)

Abstract

This work is a product of a postgraduate dissertation conducted within the framework of the "Postgraduate Program in Information Systems & Services" in the direction of Advanced Information Systems of the University of Piraeus. Its purpose is to study the different federated learning techniques in recommender systems, to implement one of them and to conduct experiments.

The first chapter introduces the federated learning approach. Then, there is an extensive reference to its categories and its architecture.The next chapter includes the definition of recommendation systems, the reasons for their use, the recommendation process and its forms. Also, the different types of recommendation systems are analyzed. In the third chapter, reference is made to the basic idea of similar relevant publications on the collaborative learning approach in referral systems and a summary table is presented. Finally, the fourth chapter describes the implementation of the federated learning application and presents the experimental results.

Keywords: federated learning, recommender systems

Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά την καθηγήτρια κα. Χαλκίδη Μαρία για την εμπιστοσύνη που μου έδειξε, και για την καθοδήγησή της κατά τη διάρκεια υλοποίησης της διπλωματικής εργασίας. Ακόμα, θα ήθελα να ευχαριστήσω όλους του καθηγητές του Πανεπιστημίου Πειραιώς για τις πολύτιμες γνώσεις που μου προσέφεραν. Τέλος, θέλω να εκφράσω ένα τεράστιο ευχαριστώ στην οικογένεια μου, για την στήριξη κατά τη διάρκεια των σπουδών μου.

ΚΕΦΑΛΑΙΟ 1: Συνεργατική Προσέγγιση Μάθησης (Federated Learning)

Στο παρόν κεφάλαιο, πραγματοποιείται μια σύντομη εισαγωγή στον Γενικό Κανονισμό Προστασίας Δεδομένων (GDPR) και συνοπτική αναφορά στη συνεργατική προσέγγιση μάθησης (federated learning). Στην συνέχεια, δίνεται ο ορισμός της, οι κατηγορίες στις οποίες διαχωρίζεται και παρουσιάζεται η αρχιτεκτονική της. Το κεφάλαιο αυτό καταλήγει με σχετικές μελέτες πάνω στη συνεργατική προσέγγιση μάθησης (federated learning).

1.1 Σκοπός της εργασίας

Λόγω της ταχείας ανάπτυξης του Διαδικτύου σε συνδυασμό με το πρόβλημα της συσσώρευσης πληροφοριών, η χρήση των συστημάτων συστάσεων (recommender systems) γίνεται απαραίτητη και για τις ηλεκτρονικές επιχειρήσεις και για τους πελάτες. Ωστόσο, οι χρήστες κατανοούν επίσης την ανάγκη εμπιστευτικότητας των δεδομένων τους η οποία στα συστήματα συστάσεων επιτυγχάνεται με μια πρόσφατη κατανεμημένη τεχνική μηχανικής μάθησης που είναι το federated learning. [1]

Η συγκεκριμένη εργασία ασχολείται με την μελέτη των διαφορετικών τεχνικών federated learning για συστήματα συστάσεων, την υλοποίηση μιας federated learning τεχνικής σε συστήματα συστάσεων και τη διεξαγωγή πειραμάτων.

1.2 Ο Γενικός Κανονισμός Προστασίας Δεδομένων (GDPR)

Με την πρόοδο των μεγάλων δεδομένων, την έμφαση στην προστασία των δεδομένων προσωπικού χαρακτήρα και την ασφάλεια, κάθε διαρροή δημόσιων δεδομένων θα προκαλέσει μεγάλη ανησυχία στα μέσα ενημέρωσης και το κοινό. [2] Χαρακτηριστικό παράδειγμα αποτελεί ο Γενικός Κανονισμός για την Προστασία Δεδομένων (GDPR), ο οποίος τέθηκε σε εφαρμογή την 25η Μαΐου 2018. Πιο συγκεκριμένα, αυτός ο κανονισμός επιδιώκει την εξισορρόπηση μεταξύ του δικαιώματος της προστασίας των προσωπικών δεδομένων από τη μία πλευρά και του δικαιώματος στην πληροφόρηση, διαφάνεια και δημόσια ασφάλεια από την άλλη, με τρόπο που να προάγει την ελεύθερη και ανεμπόδιστη οικονομική ανάπτυξη και επιχειρηματική δραστηριότητα. [3]

Ο Γενικός Κανονισμός για την Προστασία Δεδομένων (GDPR) έχει σχεδόν απαγορεύσει όλα τα είδη αυτόνομων δραστηριοτήτων συλλογής, μεταφοράς και χρήσης δεδομένων των χρηστών. Αυτό σημαίνει, ότι δεν είναι πλέον αποδεκτή η απλή συλλογή πηγών δεδομένων και η ενσωμάτωσή τους σε μια τοποθεσία χωρίς άδεια

χρήστη. Επίσης, πολλές κανονικές λειτουργίες στον τομέα των μεγάλων δεδομένων (big data), όπως η συγχώνευση δεδομένων χρήστη από διάφορες πηγές για την κατασκευή ενός μοντέλου τεχνητής νοημοσύνης χωρίς καμιά συμφωνία με το χρήστη, πρέπει να θεωρούνται παράνομες στο νέο ρυθμιστικό πλαίσιο.

1.3 Μια επισκόπηση στη συνεργατική προσέγγιση μάθησης (federated learning)

Όπως έγινε αντιληπτό και από τα παραπάνω, πολλές προσπάθειες ανταλλαγής δεδομένων που πραγματοποιούνταν στο παρελθόν, σήμερα απαιτούν δραστικές αλλαγές για να είναι συμβατές με τον Κανονισμό.

Η προσέγγιση συνεργατικής μάθησης, σκοπεύει, οι βιομηχανίες να χρησιμοποιούν με ακρίβεια τα δεδομένα σε οργανισμούς, ενώ τηρούν την ιδιωτικότητα, την ασφάλεια και τις απαιτήσεις των κανονισμών, όχι μόνο από την δημιουργία πιο ευέλικτων και ισχυρών μοντέλων που επιτρέπουν την επιχειρηματική συνεργασία χρησιμοποιώντας συλλογικά δεδομένα αλλά και χωρίς απευθείας ανταλλαγή δεδομένων.

Το federated learning είναι ένα σύστημα που:

- Τα δεδομένα που διανέμονται βρίσκονται σε κάθε οντότητα δεδομένων, χωρίς αποκάλυψη της ιδιωτικότητας και παραβίαση της συμμόρφωσης στους κανονισμούς.
- Πολλαπλά μέρη δεδομένων δημιουργούν ένα εικονικό κοινό μοντέλο κάτω από ένα federated σύστημα δεδομένων, κερδίζοντας αμοιβαίο όφελος από το σύστημα.
- Κάτω από έναν τέτοιο federated μηχανισμό, η ταυτότητα και η κατάσταση κάθε συμμετέχοντα είναι η ίδια
- Αυτό το εικονικό μοντέλο έχει τις ίδιες ή σχεδόν ίδιες επιδόσεις με το μοντέλο που δημιουργείται συγκεντρώνοντας όλα μαζί τα δεδομένα.

Οι προσεγγίσεις συνεργατικής μάθησης επιτρέπουν η εκμάθηση να γίνεται ενώ πολλαπλά σύνολα δεδομένων δεν μετακινούνται – δηλαδή δεν απαιτείται καμία ανταλλαγή δεδομένων στα ακατέργαστα δεδομένα για την προστασία της ιδιωτικότητας και του απορρήτου. Με αυτόν τον τρόπο, παρέχεται μια εφικτή λύση στο πρόβλημα των απομονωμένων δεδομένων. Αυτό το πρόβλημα παρουσιάζεται συνήθως σε πολλές βιομηχανίες. Εκεί, τα δεδομένα υπάρχουν υπό τη μορφή απομονωμένων νησιών (isolated islands). Λόγω του ανταγωνισμού της βιομηχανίας, της ασφάλειας της ιδιωτικής ζωής και των περίπλοκων διοικητικών διαδικασιών, η ίδια η εταιρεία αντιμετωπίζει μεγάλη αντίσταση ακόμη και στην ενσωμάτωση δεδομένων μεταξύ των διαφόρων τμημάτων της. Είναι σχεδόν αδύνατο να ενσωματωθούν τα δεδομένα τα οποία βρίσκονται διάσπαρτα σε όλη τη χώρα ή το κόστος είναι απαγορευτικό.

Ας υποθέσουμε ότι υπάρχουν δύο εταιρείες Α και Β με διαφορετικά δεδομένα η καθεμιά. Για παράδειγμα, η Εταιρεία Α έχει δεδομένα που αφορούν τα προφίλ των χρηστών. Η εταιρεία Β έχει δεδομένα σχετικά με τα χαρακτηριστικά προϊόντων και τις ετικέτες. Σύμφωνα με τον Γενικό Κανονισμό Προστασίας Δεδομένων (GDPR), οι δύο

εταιρείες δεν μπορούν συνδυάσουν τα δεδομένα τους, επειδή οι αρχικοί πάροχοι των δεδομένων δεν συμφώνησαν σε αυτό. Γίνεται η υπόθεση ότι κάθε συμβαλλόμενο μέλος δημιουργεί ένα μαθησιακό μοντέλο για μια εργασία ταξινόμησης ή πρόβλεψης αντίστοιχα, και ότι αυτές οι εργασίες ήδη αναγνωρίζονται από τους αντίστοιχους χρήστες τους κατά την απόκτηση των δεδομένων. Τώρα το ερώτημα είναι πώς θα επιτευχθεί η κατασκευή μοντέλων υψηλότερης ποιότητας και για τις δύο εταιρείες Α και Β. Ωστόσο, επειδή τα δεδομένα είναι ελλιπή (για παράδειγμα, η εταιρεία Α δεν διαθέτει δεδομένα ετικέτας ενώ η Β δεν διαθέτει δεδομένα χαρακτηριστικών) ή τα δεδομένα είναι ανεπαρκή (δηλαδή η ποσότητα των δεδομένων είναι ανεπαρκής για την κατασκευή ενός καλού μοντέλου) τα αποτελέσματα των μοντέλων μπορεί να μην είναι ικανοποιητικά.

Επομένως, σκοπός του federated learning είναι να λύσει αυτό το πρόβλημα στοχεύοντας στη δημιουργία ενός μοντέλου κατά μήκος των οργανισμών. Σε αυτό το μοντέλο, μεμονωμένα δεδομένα κάθε οργανισμού παραμένουν στο τοπικό τους περιβάλλον και οι παράμετροι του μοντέλου ανταλλάσσονται με μηχανισμούς κρυπτογράφησης σε ένα συνεργατικό (federated) σύστημα. Δηλαδή, ένα εικονικό διαμοιραζόμενο μοντέλο δημιουργείται χωρίς να παραβιάζονται οι κανονισμοί περί απορρήτου των δεδομένων. Αυτό το εικονικό μοντέλο έχει τις ίδιες επιδόσεις με το μοντέλο που δημιουργείται τοποθετώντας όλα τα δεδομένα μαζί. Αλλά όταν δημιουργείται ένα εικονικό μοντέλο, τα ίδια τα δεδομένα δεν μετακινούνται, ούτε αποκαλύπτεται η ιδιωτικότητά τους, ούτε επηρεάζονται οι προδιαγραφές των δεδομένων (data specifications). Με αυτόν τον τρόπο, το μοντέλο που κατασκευάστηκε εξυπηρετεί μόνο τοπικές εργασίες στις αντίστοιχες περιοχές του. Κάτω από έναν τέτοιο συνεργατικό (federated) μηχανισμό, η ταυτότητα και η κατάσταση κάθε συμμετέχοντα είναι η ίδια και το συνεργατικό (federated) σύστημα βοηθά στη δημιουργία μιας στρατηγικής "κοινού πλούτου", για αυτό και αυτό το σύστημα ονομάζεται "federated learning". [2]

1.4 Ο ορισμός της συνεργατικής προσέγγισης μάθησης (federated learning)

Ορίζονται πολλαπλοί κάτοχοι δεδομένων F_i, $i = 1 \dots N$ οι οποίοι επιθυμούν να εκπαιδεύσουν ένα μοντέλο μηχανικής μάθησης συγχωνεύοντας τα αντίστοιχα δεδομένα τους D_i. Μια κλασική μέθοδος είναι να τοποθετηθούν όλα τα δεδομένα μαζί και να χρησιμοποιηθεί το $D = \{D_i, i = 1 \dots N\}$ ώστε να εκπαιδευτεί ένα μοντέλο M_{sum}. Ωστόσο, αυτή η λύση δεν είναι δυνατόν να εφαρμοστεί εξαιτίας νομικών ζητημάτων που προκύπτουν παραδείγματα των οποίων αποτελούν η ιδιωτικότητα και η ασφάλεια των δεδομένων. Για να λυθεί αυτό το πρόβλημα, προτείνεται η συνεργατική προσέγγιση μάθησης (federated learning). Πιο συγκεκριμένα, το federated learning είναι μια διαδικασία εκμάθησης στην οποία οι κάτοχοι των δεδομένων εκπαιδεύουν συνεργατικά ένα μοντέλο M_{FED} και σε αυτήν την διαδικασία κανένας κάτοχος δεδομένων F_i δεν εκθέτει τα δεδομένα D_i. [2]

1.5 Κατηγοριοποίηση της συνεργατικής προσέγγισης μάθησης (federated learning)

Σε αυτήν την ενότητα, κατηγοριοποιείται η συνεργατική προσέγγιση μάθησης με βάση τα γνωρίσματα και την κατανομή του δείγματος των χαρακτηριστικών στα δεδομένα των απομονωμένων νησιών.

Θεωρώντας ότι υπάρχουν πολλοί ιδιοκτήτες δεδομένων, το σύνολο δεδομένων D_i το οποίο έχει στην κατοχή του κάθε ιδιοκτήτης δεδομένων μπορεί να αναπαρασταθεί από έναν πίνακα. Κάθε γραμμή του πίνακα αντιπροσωπεύει έναν χρήστη και κάθε στήλη αντιπροσωπεύει ένα χαρακτηριστικό του χρήστη. Ταυτόχρονα, ορισμένα σύνολα δεδομένων μπορεί να περιέχουν δεδομένα ετικετών. Προκειμένου να δημιουργηθεί ένα μοντέλο πρόβλεψης της συμπεριφοράς των χρηστών, απαιτείται η ύπαρξη δεδομένων ετικέτας. Τα χαρακτηριστικά των χρηστών καλούνται ως X ενώ των ετικετών ως Y. Για παράδειγμα, στον οικονομικό τομέα, οι οφειλές ενός χρήστη αποτελούν την ετικέτα Y που πρέπει να προβλεφθεί. Στο μάρκετινγκ, η ετικέτα Y είναι η επιθυμία αγοράς του χρήστη ενώ, στον τομέα της εκπαίδευσης Y είναι ο βαθμός γνώσης ενός μαθητή. Το χαρακτηριστικό του χρήστη X και η ετικέτα Y αποτελούν το πλήρες σύνολο δεδομένων εκπαίδευσης (X, Y). Παρ 'όλα αυτά, στην πραγματικότητα, οι χρήστες των διαφόρων συνόλων δεδομένων δεν είναι όμοιοι ή τα χαρακτηριστικά τους δεν είναι ολόιδια. Πιο συγκεκριμένα, θεωρώντας ως παράδειγμα τη συνεργατική προσέγγιση μάθησης (federated learning) με δύο κατόχους δεδομένων, η κατανομή των δεδομένων μπορεί να διαχωριστεί στις τρεις περιπτώσεις που ακολουθούν:

Η επικάλυψη των χαρακτηριστικών (X1, X2, ...) είναι μεγάλη, ενώ η επικάλυψη των χρηστών (U1,U2 ...) είναι μικρή.

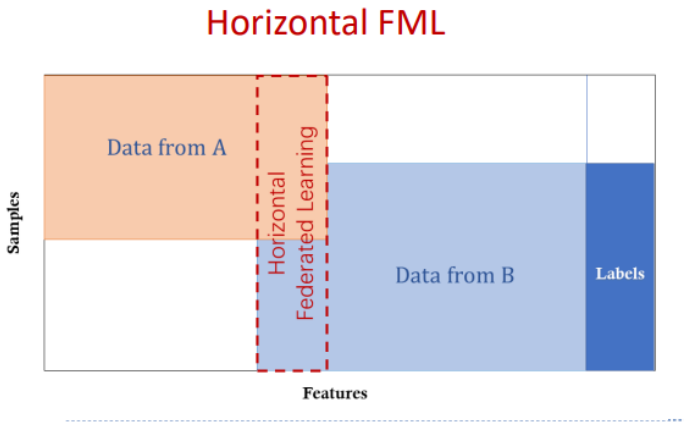
Η επικάλυψη των χρηστών (U1, U2 ...) είναι μεγάλη, ενώ η επικάλυψη των χαρακτηριστικών (X1, X2, ...) είναι μικρή.

Η επικάλυψη των χρηστών (U1, U2 ...) και η επικάλυψη των χαρακτηριστικών (X1, X2, ...) είναι και οι δύο μικρές.

Προκειμένου να υπάρξουν λύσεις για τις τρεις παραπάνω περιπτώσεις, το federated learning κατηγοριοποιείται σε: horizontally federated learning (οριζόντια συνεργατική προσέγγιση μάθησης), vertically federated learning (κάθετη συνεργατική προσέγγιση μάθησης) και federated transfer learning (μεταφορά γνώσης με συνεργατική προσέγγιση μάθησης). [2]

1.5.1 Horizontally Federated Learning

Στις περιπτώσεις όπου, τα δύο σύνολα δεδομένων μοιράζονται τον ίδιο χώρο χαρακτηριστικών αλλά διαφορετικό χώρο δειγμάτων, ένα σύστημα συνεργατικής προσέγγισης μάθησης ονομάζεται horizontally federated learning. Παραδείγματος χάριν, δύο περιφερειακές τράπεζες ενδέχεται να έχουν πολύ διαφορετικές ομάδες χρηστών από τις αντίστοιχες περιφέρειές τους. Επομένως, η τομή των χρηστών είναι πολύ μικρή όπως παρουσιάζεται στην Εικόνα 1. Ωστόσο, οι επιχειρήσεις τους είναι πολύ παρόμοιες, οπότε οι λειτουργίες κάθε χρήστη είναι οι ίδιες. Σε αυτή την περίπτωση, ένα horizontally federated learning μοντέλο μάθησης μπορεί να κατασκευαστεί. Το 2017, η Google πρότεινε μια horizontally federated learning λύση για τις ενημερώσεις των μοντέλων του τηλεφώνου Android: Ένας χρήστης που χρησιμοποιεί συνεχώς ένα τηλέφωνο Android ενημερώνει τοπικά τις παραμέτρους του μοντέλου και ανεβάζει τις παραμέτρους στο Android cloud. Επομένως, παρέχεται από κοινού εκπαίδευση του κεντρικού μοντέλου μαζί με άλλους ιδιοκτήτες δεδομένων. Τέλος, εισάγεται ένα ασφαλές σχήμα συνάθροισης προκειμένου να προστατευτεί το απόρρητο των συγκεντρωτικών ενημερώσεων των χρηστών στο πλαίσιο του federated learning πλαισίου. [2]



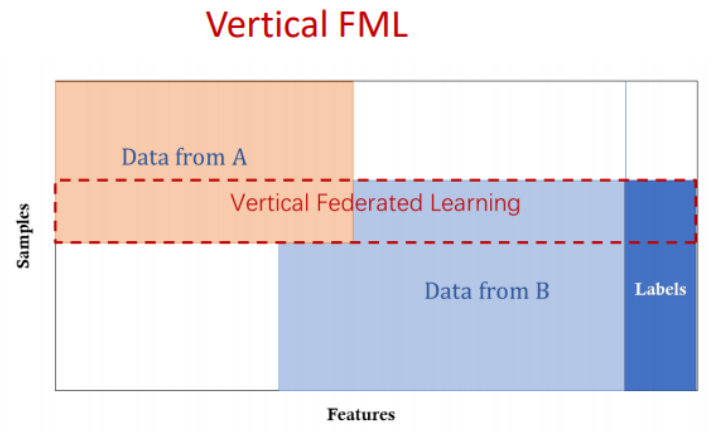
Εικόνα 1: Horizontally Federated Learning: Μεγάλη επικάλυψη χαρακτηριστικών των δύο συνόλων δεδομένων.[1]

1.5.2 Vertically Federated Learning

Το vertically federated learning είναι εφαρμοστέο στις περιπτώσεις όπου τα δύο σύνολα δεδομένων μοιράζονται τον ίδιο χώρο χρηστών αλλά διαφορετικό χώρο χαρακτηριστικών όπως φαίνεται στην Εικόνα 2. Για παράδειγμα, υπάρχουν δύο διαφορετικές εταιρείες στην ίδια πόλη, η μία είναι μια τράπεζα και η άλλη μια εταιρεία ηλεκτρονικού εμπορίου. Η βάση των χρηστών τους είναι πιθανό να περιέχει το μεγαλύτερο μέρος των κατοίκων της περιοχής, έτσι το μέγεθος των κοινών χρηστών είναι μεγάλο. Ωστόσο, στην τράπεζα καταγράφονται τα εισοδήματα και οι δαπάνες του χρήστη ενώ στην εταιρεία ηλεκτρονικού εμπορίου υπάρχει το ιστορικό αγορών του χρήστη. Επομένως, είναι ευρέως αντιληπτό, ότι οι λειτουργίες του χρήστη είναι πολύ διαφορετικές όπως παρουσιάζεται στην Εικόνα 3. Το vertically federated learning είναι η διαδικασία συγκέντρωσης αυτών των διαφορετικών χαρακτηριστικών σε μια

[1] https://img.fedai.org.cn/fedweb/1552916850679.pdf

κρυπτογραφημένη κατάσταση και ο υπολογισμός των απωλειών εκπαίδευσης και των ανάδελτα (συναρτήσεων που υπολογίζουν μια βελτίωση του μοντέλου με βάση τις απώλειες) με τρόπο που διατηρείται η ιδιωτικότητα για την κατασκευή ενός μοντέλου συλλογικών δεδομένων. Προς το παρόν, τα μοντέλα μηχανικής μάθησης, όπως τα μοντέλα λογιστικής παλινδρόμησης, τα δενδρικά μοντέλα και τα μοντέλα που βασίζονται σε νευρωνικά δίκτυα έχουν αποδειχθεί ότι μπορούν να ενσωματωθούν σε αυτό το συνεργατικό σύστημα (federated system). [2]



Εικόνα 2: Vertically Federated Learning:Μεγάλη επικάλυψη των ίδιων IDs (χρηστών) των δύο συνόλων δεδομένων.[2]

ID	X1	X2	X3
U1	9	80	600
U2	4	50	550
U3	2	35	520
U4	10	100	600
U5	5	75	600
U6	5	75	520
U7	8	80	600

Retail A Data

ID	X4	X5	Y
U1	6000	600	No
U2	5500	500	Yes
U3	7200	500	Yes
U4	6000	600	No
U8	6000	600	No
U9	4520	500	Yes
U10	6000	600	No

Bank B Data

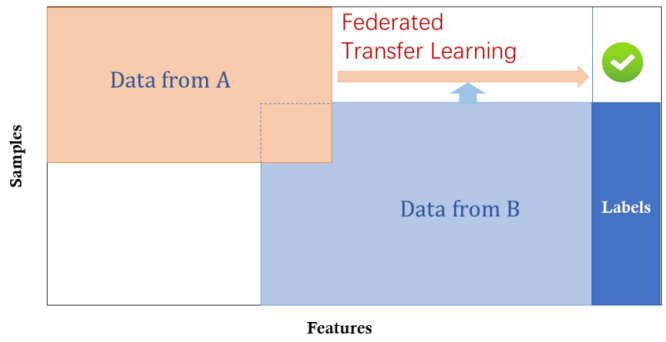
Εικόνα 3: Παράδειγμα Vertically Federated Learning[3]

1.5.3 Federated Transfer Learning

Το federated transfer learning εφαρμόζεται σε περιπτώσεις όπου τα δύο σύνολα δεδομένων διαφέρουν και σε δείγματα και σε χαρακτηριστικά. Σε αυτή την περίπτωση, τεχνικές μεταφοράς γνώσης μπορούν να εφαρμοστούν για να ξεπεραστεί η έλλειψη δεδομένων ή ετικετών. Για παράδειγμα, υπάρχει μια τράπεζα που βρίσκεται στην Κίνα

[2] https://img.fedai.org.cn/fedweb/1552916850679.pdf
[3] https://img.fedai.org.cn/fedweb/1552916850679.pdf

και μια επιχείρηση ηλεκτρονικού εμπορίου που βρίσκεται στις Ηνωμένες Πολιτείες Αμερικής. Λόγω γεωγραφικών περιορισμών, οι ομάδες χρηστών των δύο ιδρυμάτων έχουν ένα μικρό σημείο τομής. Από την άλλη πλευρά, λόγω των διαφορετικών επιχειρήσεων, μόνο ένα μικρό μέρος των χαρακτηριστικών των δύο εταιρειών αλληλεπικαλύπτεται. Σε αυτή την περίπτωση, προκειμένου να πραγματοποιηθεί αποτελεσματικά η συνεργατική προσέγγιση μάθησης, είναι απαραίτητο να εισαχθεί η μεταφορά γνώσης για την επίλυση του προβλήματος του μικρού μεγέθους των δεδομένων και της αδύναμης εποπτείας, βελτιώνοντας έτσι την απόδοση του μοντέλου.



Εικόνα 4: Federated Transfer Learning[4]:Μικρή επικάλυψη των ίδιων IDs (χρηστών) και των χαρακτηριστικών των δύο συνόλων δεδομένων

1.6 Η αρχιτεκτονική του Federated Learning

Προκειμένου να πραγματοποιηθεί η εισαγωγή της αρχιτεκτονικής της συνεργατικής προσέγγισης μάθησης (federated learning) θα χρησιμοποιηθεί το vertically federated learning ως παράδειγμα. Αρχικά, γίνεται η υπόθεση της ύπαρξης δύο κατόχων δεδομένων (π.χ. εταιρείες Α και Β) ως παράδειγμα για να γίνει η εισαγωγή στην αρχιτεκτονική του συστήματος της συνεργατικής προσέγγισης μάθησης (federated learning) το οποίο μπορεί να επεκταθεί και σε περιπτώσεις που αφορούν πολλούς κατόχους δεδομένων. Ακόμη, εικάζεται ότι οι εταιρείες Α και Β θέλουν να εκπαιδεύσουν από κοινού ένα μοντέλο μηχανικής μάθησης και ότι τα επιχειρησιακά τους συστήματα έχουν τα δικά τους δεδομένα. Επιπρόσθετα, η εταιρεία Β διαθέτει επίσης δεδομένα ετικετών που το μοντέλο θα πρέπει να προβλέψει. Επιπλέον, εξαιτίας ζητημάτων ιδιωτικότητας και ασφάλειας των δεδομένων, οι εταιρείες Α και Β δεν μπορούν να ανταλλάξουν άμεσα δεδομένα. Σε αυτό το σημείο, το μοντέλο μπορεί να κατασκευαστεί χρησιμοποιώντας το σύστημα της συνεργατικής προσέγγισης μάθησης (federated learning), το οποίο αποτελείται από δύο μέρη, όπως φαίνεται στην Εικόνα 4α.

Μέρος 1: Ευθυγράμμιση κρυπτογραφημένης οντότητας (Encrypted entity alignment). Εφόσον οι ομάδες χρηστών των δύο εταιρειών δεν είναι οι ίδιες, το σύστημα

[4] https://img.fedai.org.cn/fedweb/1552916850679.pdf

χρησιμοποιεί την τεχνολογία ευθυγράμμισης η οποία βασίζεται στην κρυπτογράφηση των ID των χρηστών προκειμένου να επιβεβαιώνονται οι κοινοί χρήστες και των δύο μερών, χωρίς οι εταιρείες A και B να εκθέτουν τα αντίστοιχα δεδομένα και το σύστημα να μην εκθέτει χρήστες που δεν αλληλεπικαλύπτονται μεταξύ τους.

Μέρος 2: Κρυπτογραφημένο μοντέλο εκπαίδευσης. Μετά τον καθορισμό των κοινών οντοτήτων, τα δεδομένα τους μπορούν να χρησιμοποιηθούν για να εκπαιδεύσουν το μοντέλο μηχανικής μάθησης. Προκειμένου να διασφαλιστεί η εμπιστευτικότητα των δεδομένων κατά τη διάρκεια της εκπαίδευσης, είναι απαραίτητο να χρησιμοποιηθεί ένα τρίτο μέλος (συνεργάτης) C για κρυπτογράφηση. Λαμβάνοντας το μοντέλο γραμμικής παλινδρόμησης ως ένα παράδειγμα, η διαδικασία της εκπαίδευσης μπορεί να χωριστεί στα ακόλουθα τέσσερα βήματα όπως παρουσιάζεται στην Εικόνα 4β:

Βήμα 1: Ο συνεργάτης C δημιουργεί ζεύγη κρυπτογράφησης και στέλνει το δημόσιο κλειδί στις A και B.

Βήμα 2: Οι A και B κρυπτογραφούν και ανταλλάσσουν τα ενδιάμεσα αποτελέσματα για τα ανάδελτα (συναρτήσεις που υπολογίζουν μια βελτίωση του μοντέλου με βάση τις απώλειες) και τον υπολογισμό των απωλειών

Βήμα 3: Οι A και B υπολογίζουν τα κρυπτογραφημένα ανάδελτα (gradients) αντίστοιχα, και επίσης η B υπολογίζει τις κρυπτογραφημένες απώλειες. Οι A και B στέλνουν τις κρυπτογραφημένες τιμές στο C.

Βήμα 4: Ο C αποκρυπτογραφεί και στέλνει τα αποκρυπτογραφημένα ανάδελτα (gradients) και τις απώλειες πίσω στις A και B.

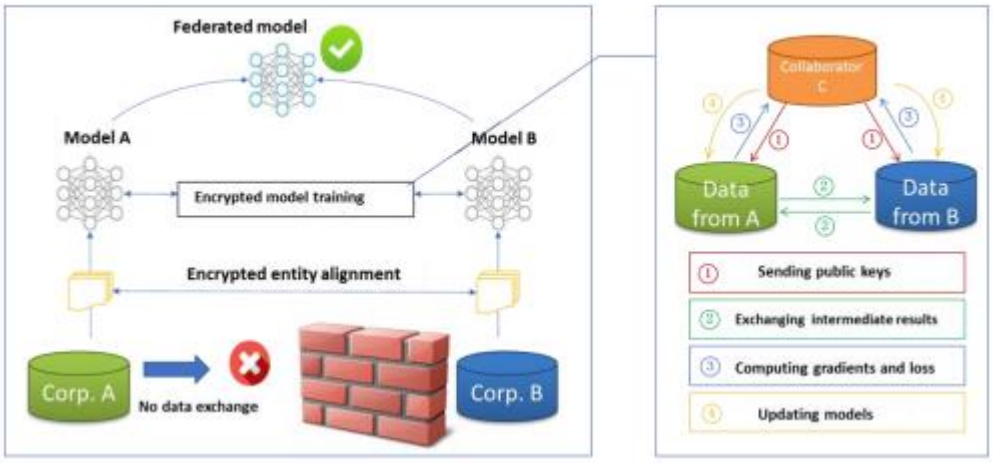
Οι A και B ενημερώνουν τις παραμέτρους του μοντέλου αντίστοιχα.

Οι επαναλήψεις μέσω των παραπάνω βημάτων συνεχίζονται μέχρι να συγκλίνει η συνάρτηση απωλειών και έτσι ολοκληρώνεται ολόκληρη η διαδικασία της εκπαίδευσης. Κατά την διάρκεια της ευθυγράμμισης των οντοτήτων και της εκπαίδευσης του μοντέλου, τα δεδομένα των A και B διατηρούνται τοπικά και η αλληλεπίδραση των δεδομένων στην εκπαίδευση δεν οδηγεί σε διαρροή δεδομένων προσωπικού χαρακτήρα. Επομένως, τα δύο μέρη επιτυγχάνουν μια κοινή εκπαίδευση μοντέλου συνεργατικά με τη βοήθεια της συνεργατικής προσέγγισης μάθησης (federated learning).

Μέρος 3: Μηχανισμός κινήτρων. Ένα σημαντικό χαρακτηριστικό του federated learning είναι ότι λύνει το πρόβλημα γιατί οι διαφορετικοί οργανισμοί χρειάζεται να δημιουργήσουν από κοινού ένα μοντέλο. Μετά την δημιουργία του μοντέλου, η απόδοση του μοντέλου θα εκδηλωθεί στις πραγματικές εφαρμογές και θα καταγραφεί σε έναν μόνιμο μηχανισμό καταγραφής δεδομένων (όπως είναι το blockchain). Οι οργανισμοί που παρέχουν περισσότερα δεδομένα θα είναι καλύτεροι και η αποτελεσματικότητα του μοντέλου θα εξαρτάται από την συμβολή του παρόχου των δεδομένων στο σύστημα. Η αποτελεσματικότητα αυτών των μοντέλων κατανέμεται στα μέλη βάσει federated μηχανισμών και συνεχίζει να παρακινεί περισσότερους οργανισμούς να γίνουν μέλη στο federation των δεδομένων.

Η εφαρμογή των παραπάνω τριών βημάτων όχι μόνο λαμβάνει υπόψη μόνο την προστασία της ιδιωτικής ζωής και την αποτελεσματικότητα της συνεργατικής

μοντελοποίησης μεταξύ πολλών οργανισμών, αλλά εξετάζει επίσης με ποιόν τρόπο θα γίνεται η επιβράβευση των οργανισμών που συνεισφέρουν περισσότερα δεδομένα και με ποιόν τρόπο θα εφαρμόζονται τα κίνητρα με βάση ένα μηχανισμό συναίνεσης. Επομένως, η συνεργατική προσέγγιση μάθηση (feaderated learning) είναι ένας μηχανισμός εκμάθησης «κλειστού βρόχου».



α *β*

Εικόνα 5: Η αρχιτεκτονική του Federated Learning συστήματος

1.7 Σχετικές Μελέτες

Η συνεργατική προσέγγιση μάθησης (federated learning) προστατεύει το απόρρητο των δεδομένων των χρηστών με πολύ διαφορετικό τρόπο από ότι οι τεχνικές επίτευξης ανωνυμίας όπως η k-anonymity (μοντέλο για τη διαφύλαξη ευαίσθητων δεδομένων). Πρώτον, το federated learning προστατεύει την ιδιωτικότητα των δεδομένων των χρηστών μέσω της ανταλλαγής παραμέτρων κάτω από έναν μηχανισμό κρυπτογράφησης. Το σχήμα κρυπτογράφησης περιλαμβάνει την ομομορφική κρυπτογράφηση (homomorphic encryption). Πιο συγκεκριμένα, η ομομορφική κρυπτογράφηση αφορά την εκτέλεση υπολογισμών σε κρυπτογραφημένα δεδομένα, επιτρέποντας τον συνδυασμό κρυπτοκειμένων με τέτοιο τρόπο, ώστε το κρυπτοκείμενο που προκύπτει, να αντιστοιχεί στον υπολογισμό κάποιας (άλλης) συνάρτησης στα μηνύματα. Έτσι το αποτέλεσμα μπορεί να ανακτηθεί με μία απλή αποκρυπτογράφηση. Φυσικά, όλα τα κρυπτοκείμενα πρέπει να έχουν κρυπτογραφηθεί με το ίδιο δημόσιο κλειδί, για να μπορέσει να έχει νόημα ο συνδυασμός τους. [4] Στο federated learning, τα δεδομένα και το ίδιο το μοντέλο δεν μεταδίδονται, ούτε μπορούν να υπολογισθούν από τα δεδομένα της άλλης οντότητας. Επομένως, δεν υπάρχει καμία πιθανότητα διαρροής στο επίπεδο των ακατέργαστων δεδομένων, ούτε παραβίαση των αυστηρότερων νόμων προστασίας των δεδομένων, όπως το GDPR. Η διαφοροποίηση των μεθόδων επίτευξης ανωνυμίας όπως η k-anonymity περιλαμβάνει την χρήση

γενικευμένων μεθόδων προκειμένου να γίνει απόκρυψη ορισμένων ευαίσθητων χαρακτηριστικών μέχρις ότου η τρίτη οντότητα να μην μπορεί να διακρίνει την άλλη οντότητα, κάνοντας έτσι αδύνατη την αποκατάσταση των δεδομένων προκειμένου να προστατευτεί η ιδιωτικότητα των χρηστών. Ωστόσο, αυτή η μέθοδος εξακολουθεί να μεταδίδει ανεπεξέργαστα δεδομένα, επομένως υπάρχει πιθανότητα πιθανής επίθεσης, και η προστασία της ιδιωτικότητας των δεδομένων ενδέχεται να μην είναι εφαρμοστέα σε αυστηρότερους κανονισμούς που αφορούν την προστασία των δεδομένων όπως είναι το GDPR. Επομένως, γίνεται ευκόλως διακριτό, ότι το federated learning είναι ένα πιο ισχυρό εργαλείο προστασίας της ιδιωτικότητας των δεδομένων των χρηστών. [2] [5]

Το horizontally federated learning με την πρώτη ματιά είναι κάπως παρόμοιο με την κατανεμημένη μηχανική μάθηση (distributed machine learning). Η κατανεμημένη μηχανική μάθηση καλύπτει πολλές πτυχές, συμπεριλαμβανομένης της κατανεμημένης αποθήκευσης δεδομένων εκπαίδευσης, λειτουργίας υπολογιστικών εργασιών και κατανομής των αποτελεσμάτων του μοντέλου κλπ. Ο parameter server είναι ένα τυπικό στοιχείο στην κατανεμημένη μηχανική μάθηση. Ως ένα εργαλείο το οποίο επιταχύνει τη διαδικασία της εκπαίδευσης, ο parameter server αποθηκεύει δεδομένα σε κατανεμημένους κόμβους εργασίας και διανέμει δεδομένα και υπολογιστικούς πόρους μέσω ενός κεντρικού κόμβου έτσι ώστε η εκπαίδευση του μοντέλου να γίνεται πιο αποτελεσματικά. Στο horizontally federated learning, ο κόμβος εργασίας αναπαριστά τον κάτοχο των δεδομένων. Έχει πλήρη αυτονομία για τα τοπικά δεδομένα και μπορεί να αποφασίσει πότε και πώς θα λάβει μέρος στη συνεργατική προσέγγιση μάθησης (federated learning). Στον parameter server, ο κεντρικός κόμβος έχει πάντα τον έλεγχο, έτσι το federated learning αντιμετωπίζει ένα πιο περίπλοκο μαθησιακό περιβάλλον. Δεύτερον, το federated learning δίνει έμφαση στην προστασία της ιδιωτικότητας των δεδομένων του κατόχου των δεδομένων κατά τη διάρκεια της εκπαίδευσης του μοντέλου. Τα αποτελεσματικά μέτρα για την προστασία της ιδιωτικότητας των δεδομένων μπορούν να διαχειριστούν καλύτερα το όλο και πιο αυστηρό ρυθμιστικό πλαίσιο για την προστασία της ιδιωτικότητας και της ασφάλειας των δεδομένων στο μέλλον. [2] [6]

Το federated database system είναι ένα σύστημα που ενσωματώνει πολλαπλές μονάδες βάσεων δεδομένων και διαχειρίζεται το ενσωματωμένο σύστημα στο σύνολό του. Προτείνεται η επίτευξη της διαλειτουργικότητας με πολλαπλές ανεξάρτητες βάσεις δεδομένων. Το federated database system συχνά χρησιμοποιεί κατανεμημένη αποθήκευση για τις μονάδες των βάσεων δεδομένων και στην πράξη τα δεδομένα σε κάθε μονάδα βάσης δεδομένων είναι ετερογενή. Επομένως, έχει πολλές ομοιότητες με το federated learning όσον αφορά τον τύπο και την αποθήκευση των δεδομένων. Ωστόσο, το federated database system δεν εμπεριέχει κανένα μηχανισμό προστασίας της ιδιωτικότητας στη διαδικασία της αλληλεπίδρασης και όλες οι μονάδες των βάσεων δεδομένων είναι εντελώς ορατές στο σύστημα διαχείρισης. Επιπλέον, το επίκεντρο του federated database system είναι οι βασικές λειτουργίες των δεδομένων, συμπεριλαμβανομένης της εισαγωγής, της διαγραφής, της αναζήτησης και της συγχώνευσης κλπ., ενώ ο σκοπός της συνεργατικής προσέγγισης μάθησης (federated learning) είναι να καθιερώσει ένα κοινό μοντέλο για κάθε ιδιοκτήτη δεδομένων βάσει της προϋπόθεσης της προστασίας της ιδιωτικότητας των δεδομένων. [2] [7]

ΚΕΦΑΛΑΙΟ 2: Συστήματα Συστάσεων

Στις ενότητες που ακολουθούν, παρουσιάζονται διεξοδικά τα συστήματα συστάσεων. Πιο συγκεκριμένα, στην πρώτη ενότητα γίνεται συνοπτική αναφορά στα συστήματα συστάσεων. Έπειτα, δίνεται ο ορισμός τους, προσδιορίζονται τα αίτια χρήσης τους και οι μορφές της σύστασης. Τέλος, αναλύονται οι διάφοροι τύποι των συστημάτων συστάσεων.

2.1 Λίγα λόγια για τα συστήματα συστάσεων

Σήμερα, υπάρχει μια εκτενής κατηγορία εφαρμογών Web που περιλαμβάνει την πρόβλεψη σχετικά με τις απαντήσεις του χρήστη σε διάφορες επιλογές του. Μια τέτοια εγκατάσταση ονομάζεται σύστημα συστάσεων. Παράδειγμα συστήματος συστάσεων αποτελεί, η προσφορά ειδησεογραφικών άρθρων σε αναγνώστες εφημερίδων οι οποίοι βρίσκονται σε απευθείας σύνδεση (on line), με βάση προβλέψεις που βασίζονται στα ενδιαφέροντά τους. Τα συστήματα συστάσεων χρησιμοποιούν διάφορες τεχνολογίες.

Σε μια εφαρμογή ενός συστήματος συστάσεων υπάρχουν δύο κατηγορίες οντοτήτων, οι χρήστες και τα αντικείμενα. Οι χρήστες έχουν προτιμήσεις για ορισμένα αντικείμενα, και αυτές οι προτιμήσεις πρέπει να εξαχθούν από τα δεδομένα. Τα ίδια τα δεδομένα αντιπροσωπεύονται ως πίνακας, δίνοντας για κάθε ζεύγος χρήστη-αντικειμένου μια τιμή που αντιπροσωπεύει τι είναι γνωστό για το βαθμό προτιμήσεως του χρήστη για το αντικείμενο αυτό. Οι τιμές προέρχονται από ένα ταξινομημένο σύνολο, π.χ. ακέραιοι αριθμοί από το 1 έως το 5 που αντιπροσωπεύουν τον αριθμό των αστεριών που ο χρήστης έδωσε ως βαθμολογία για αυτό το αντικείμενο. Όταν ο πίνακας είναι αραιός, αυτό σημαίνει ότι οι περισσότερες καταχωρήσεις είναι "άγνωστες". Άγνωστη βαθμολογία συνεπάγεται ότι δεν υπάρχουν ρητές πληροφορίες σχετικά με την προτίμηση του χρήστη για το αντικείμενο.

Πίνακας 1: Πίνακας με την βαθμολογία των χρηστών για κάθε ταινία

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

Στον Πίνακα 1 υπάρχει ένα παράδειγμα ενός πίνακα, στον οποίο αναπαριστώνται οι αξιολογήσεις των χρηστών σχετικά με βαθμολογίες ταινιών σε κλίμακα 1-5, με το 5 να αποτελεί την υψηλότερη βαθμολογία. Τα κενά αντιπροσωπεύουν την κατάσταση κατά την οποία ο χρήστης δεν έχει αξιολογήσει την ταινία. Τα ονόματα των ταινιών είναι HP1, HP2 και HP3 για το Harry Potter I, II και III, TW για το Twilight και SW1, SW2 και SW3 για τα επεισόδια Star Wars 1, 2 και 3. Οι χρήστες εκπροσωπούνται με κεφαλαία γράμματα από Α έως Δ.

Ο στόχος ενός συστήματος συστάσεων είναι η πρόβλεψη των κενών στον πίνακα.

Για παράδειγμα, θα γίνει προσπάθεια να απαντηθεί το παρακάτω ερώτημα: αν άρεσε στον χρήστη Α το SW2. Παρατηρώντας τον Πίνακα 1 διαπιστώνεται ότι υπάρχουν ελάχιστα στοιχεία. Το σύστημα συστάσεων μπορεί να σχεδιαστεί έτσι ώστε να ληφθούν υπόψη οι ιδιότητες των ταινιών, όπως ο παραγωγός, ο σκηνοθέτης, τα αστέρια, ή ακόμα και η ομοιότητα των ονομάτων τους. Εφόσον ισχύει αυτό, αξίζει να σημειωθεί, η ομοιότητα μεταξύ των SW1 και SW2. Έτσι, εφόσον του χρήστη Α δεν του άρεσε το SW1, είναι πολύ πιθανό να μην του αρέσει και το SW2. Εναλλακτικά, με πολύ περισσότερα δεδομένα, παρατηρείται το γεγονός ότι οι άνθρωποι που βαθμολόγησαν τόσο τα SW1 όσο και τα SW2 τείνουν να δίνουν παρόμοιες αξιολογήσεις. Έτσι, συμπεραίνεται ότι ο χρήστης Α θα έδινε επίσης και στο SW2 μια χαμηλή βαθμολογία, παρόμοια με την βαθμολογία που έδωσε ο ίδιος χρήστης και για το SW1. [8]

2.2 Ορισμός των συστημάτων συστάσεων

Στην προηγούμενη ενότητα έγινε αναφορά ότι τα συστήματα συστάσεων χρησιμοποιούντα για την πρόταση αντικειμένων σε χρήστες. Ένας ακόμη ορισμός των συστημάτων συστάσεων παρέχεται παρακάτω:

Τα συστήματα συστάσεων είναι συστήματα φιλτραρίσματος πληροφοριών που αντιμετωπίζουν το πρόβλημα της υπερφόρτωσης των πληροφοριών, φιλτράροντας κρίσιμες πληροφορίες από τη μεγάλη ποσότητα δυναμικά παραγόμενων πληροφοριών

σύμφωνα με τις προτιμήσεις, τα ενδιαφέροντα ή τη συμπεριφορά του χρήστη όσον αφορά κάποιο αντικείμενο. [9]

Ένας περισσότερο επίσημος ορισμός είναι ο εξής:

Έστω C το σύνολο όλων των χρηστών και S το σύνολο όλων των πιθανών αντικειμένων που μπορούν να προταθούν, όπως βιβλία ή ταινίες. Ο χώρος S των πιθανών αντικειμένων μπορεί να είναι πολύ μεγάλος, να κυμαίνεται δηλαδή σε εκατοντάδες χιλιάδες ή ακόμη και εκατομμύρια αντικείμενα σε ορισμένες εφαρμογές, όπως είναι η σύσταση βιβλίων ή CD. Παρομοίως, και ο χώρος των χρηστών μπορεί επίσης να είναι πολύ μεγάλος έως και εκατομμύρια σε ορισμένες περιπτώσεις. Ορίζεται το u να είναι μια συνάρτηση ωφέλειας (utility function) που μετρά τη χρησιμότητα των αντικειμένων στο χρήστη c δηλ.

$u: C \times S \to R$ όπου R είναι ένα σύνολο που έχει ταξινομηθεί (π.χ. μη αρνητικοί ακέραιοι αριθμοί ή πραγματικοί αριθμοί εντός ενός συγκεκριμένου εύρους). Στη συνέχεια, για κάθε χρήστη $c \in C$ επιλέγονται τέτοια αντικείμενα $s' \in S \backslash S_c$ που μεγιστοποιούν τη χρησιμότητα του χρήστη. Πιο συγκεκριμένα:

$$\forall c \in C, {s'}_c = arg \max_{s \in S} u(c, s)$$ [10]

2.3 Αίτια χρήσης των συστημάτων συστάσεων

Στην πραγματικότητα, υπάρχουν διάφοροι λόγοι για τους οποίους οι πάροχοι υπηρεσιών ενδέχεται να θέλουν να εκμεταλλευτούν αυτή την τεχνολογία:

- **Αύξηση του αριθμού των πωληθέντων αντικειμένων:** Αυτή είναι ίσως η πιο σημαντική λειτουργία για ένα εμπορικό σύστημα συστάσεων, δηλαδή να είναι σε θέση οι πάροχοι υπηρεσιών να πουλήσουν ένα πρόσθετο σύνολο αντικειμένων σε σύγκριση με εκείνα που συνήθως πωλούνται χωρίς καμία σύσταση. Αυτός ο στόχος επιτυγχάνεται επειδή τα στοιχεία για τα οποία γίνεται σύσταση είναι πιθανό να ταιριάζουν στις ανάγκες και τις επιθυμίες του χρήστη. Οι μη εμπορικές εφαρμογές έχουν παρόμοιους στόχους, ακόμη και αν δεν υπάρχει κόστος για τον χρήστη που σχετίζεται με την επιλογή ενός αντικειμένου. Γενικά, από την άποψη του παρόχου υπηρεσιών, ο πρωταρχικός στόχος για την εισαγωγή ενός συστήματος συστάσεων είναι η αύξηση του ποσοστού μετατροπής (conversion rate), δηλ. του αριθμού των χρηστών που αποδέχονται τη σύσταση και καταναλώνουν ένα αντικείμενο, σε σύγκριση με τον αριθμό των απλών επισκεπτών που περιηγούνται απλώς στις πληροφορίες.

- **Πώληση διαφοροποιημένων ειδών:** Μια άλλη σημαντική λειτουργία ενός συστήματος συστάσεων είναι να επιτρέψει στον χρήστη να επιλέξει στοιχεία που μπορεί να είναι δύσκολο να εντοπιστούν χωρίς συγκεκριμένες συστάσεις. Για παράδειγμα, σε ένα σύστημα σύστασης ταινιών όπως είναι το Netflix, ο πάροχος υπηρεσιών ενδιαφέρεται να μισθώσει όλα τα DVD στον κατάλογο, όχι μόνο τα πιο δημοφιλή. Αυτό θα μπορούσε να είναι δύσκολο χωρίς το σύστημα συστάσεων, αφού ο πάροχος υπηρεσιών δεν θα είχε την οικονομική δυνατότητα

του ρίσκου της διαφήμισης ταινιών που είναι πιθανό να μην ταιριάζουν με τα ενδιαφέροντα ενός συγκεκριμένου χρήστη.

- **Αύξηση της ικανοποίησης των χρηστών:** Ένα καλά σχεδιασμένο σύστημα συστάσεων μπορεί επίσης να βελτιώσει την εμπειρία του χρήστη με τον διαδικτυακό τόπο ή την εφαρμογή. Ο χρήστης με αυτόν τον τρόπο, βρίσκει τις συστάσεις ενδιαφέρουσες, σχετικές και με σωστά σχεδιασμένη την αλληλεπίδραση ανθρώπου υπολογιστή, απολαμβάνοντας τη χρήση του συστήματος. Ο συνδυασμός αποτελεσματικών διεπαφών αυξάνει την υποκειμενική αξιολόγηση του χρήστη από το σύστημα. Αυτή με τη σειρά της αυξάνει τη χρήση του συστήματος και την πιθανότητα να γίνουν αποδεκτές οι συστάσεις.

- **Αύξηση της αφοσίωσης των χρηστών:** Ένας χρήστης πρέπει να είναι πιστός σε έναν διαδικτυακό τόπο και όταν τον επισκέπτεται αυτός ο διαδικτυακός τόπος χρειάζεται να τον αναγνωρίζει ως παλιό πελάτη και τον αντιμετωπίζει ως πολύτιμο επισκέπτη. Αυτό είναι ένα τυπικό χαρακτηριστικό ενός συστήματος συστάσεων, δεδομένου ότι πολλά συστήματα συστάσεων υπολογίζουν τις συστάσεις, αξιοποιώντας πληροφορίες που αποκτήθηκαν από τον χρήστη σε προηγούμενες αλληλεπιδράσεις.

- **Καλύτερη κατανόηση των επιθυμιών του χρήστη:** Μια άλλη σημαντική λειτουργία ενός συστήματος συστάσεων, η οποία μπορεί να χρησιμοποιηθεί σε πολλές άλλες εφαρμογές, είναι η περιγραφή των προτιμήσεων του χρήστη, η οποία είτε συλλέγεται ρητά είτε προβλέπεται από το σύστημα. Ο φορέας παροχής υπηρεσιών μπορεί στη συνέχεια να αποφασίσει να επαναχρησιμοποιήσει αυτές τις γνώσεις για έναν αριθμό άλλων ζητημάτων, όπως η βελτίωση της διαχείρισης του αποθέματος ή της παραγωγής του αντικειμένου. Για παράδειγμα, όσον αφορά τον ταξιδιωτικό τομέα, οι οργανισμοί διαχείρισης προορισμών μπορούν να αποφασίσουν να διαφημίσουν μια συγκεκριμένη περιοχή σε νέους πελάτες ή να διαφημίσουν ένα συγκεκριμένο τύπο διαφημιστικού μηνύματος που προκύπτει από την ανάλυση των δεδομένων που συλλέγονται από το σύστημα συστάσεων.

Ακόμη, δημοφιλείς εργασίες την υλοποίηση των οποίων βοηθά το σύστημα συστάσεων είναι οι παρακάτω:

- **Εύρεση κάποιων καλών αντικειμένων:** Προτείνονται στο χρήστη ορισμένα αντικείμενα ταξινομημένα σε λίστα μαζί με προβλέψεις, βάσει του πόσο θα τα ήθελε.

- **Εύρεση όλων των καλών αντικειμένων:** Συστήνονται όλα τα αντικείμενα που μπορούν να ικανοποιήσουν κάποιες ανάγκες των χρηστών. Σε τέτοιες περιπτώσεις είναι ανεπαρκές να βρεθούν μερικά καλά αντικείμενα. Αυτό ισχύει ιδιαίτερα όταν ο αριθμός των αντικειμένων είναι σχετικά μικρός ή όταν το σύστημα συστάσεων ανήκει σε μια κρίσιμη αποστολή, όπως σε ιατρικές ή

χρηματοοικονομικές εφαρμογές. Σε αυτές τις περιπτώσεις, επιπρόσθετα από το όφελος που προκύπτει από την προσεκτική εξέταση όλων των πιθανοτήτων, ο χρήστης μπορεί να ωφεληθεί από την κατάταξη των αντικειμένων αυτών στο σύστημα συστάσεων ή από πρόσθετες επεξηγήσεις που παράγει το σύστημα συστάσεων.

- **Σχολιασμός που συνοδεύει τις προτάσεις:** Λαμβάνοντας υπόψη ένα υπάρχον πλαίσιο, π.χ. αντικειμένων, δίνεται έμφαση σε αυτά ανάλογα με τις μακροπρόθεσμες προτιμήσεις του χρήστη. Για παράδειγμα, ένα σύστημα συστάσεων σχετικό με την τηλεόραση θα μπορούσε να σχολιάσει ποιες από τις τηλεοπτικές εκπομπές που παρουσιάζονται στον ηλεκτρονικό οδηγό προγραμμάτων αξίζει να παρακολουθήσουν οι χρήστες.

- **Σύσταση ακολουθίας:** Αντί να γίνει εστίαση στη δημιουργία μιας μόνο σύστασης, προτείνεται μια σειρά αντικειμένων που να είναι στο σύνολό τους ευχάριστα. Τυπικά παραδείγματα περιλαμβάνουν τη σύσταση τηλεοπτικών σειρών ή μια συλλογή από μουσικά κομμάτια.

- **Σύσταση συνδυασμού:** Προτείνεται στο χρήστη μια ομάδα αντικειμένων τα οποία να ταιριάζουν μεταξύ τους. Για παράδειγμα, ο σχεδιασμός ενός ταξιδιού μπορεί να αποτελείται από διάφορα αξιοθέατα, προορισμούς και υπηρεσίες διαμονής που βρίσκονται σε μια οριοθετημένη περιοχή. Από την οπτική γωνία του χρήστη, αυτές οι διάφορες εναλλακτικές λύσεις μπορούν να εξεταστούν και να επιλεγούν ως προορισμός του ταξιδιού.

- **Απλή περιήγηση:** Εδώ, ο χρήστης περιηγείται στον κατάλογο χωρίς καμία άμεση πρόσθεση αγοράς ενός αντικειμένου. Η εργασία της σύστασης είναι να βοηθήσει τον χρήστη να περιηγηθεί στα στοιχεία που είναι πιο πιθανό να εμπίπτουν στο πεδίο των ενδιαφερόντων του χρήστη για αυτή τη συγκεκριμένη περίοδο περιήγησης.

- **Εύρεση αξιόπιστου συστήματος συστάσεων:** Κάποιοι χρήστες δεν εμπιστεύονται τα συστήματα συστάσεων, επομένως τα δοκιμάζουν για να ελέγξουν την ποιότητα των παραγόμενων συστάσεων. Έτσι, κάποιο σύστημα μπορεί επίσης να προσφέρει συγκεκριμένες λειτουργίες οι οποίες να επιτρέπουν στους χρήστες να δοκιμάσουν τη συμπεριφορά τους επιπροσθέτως εκείνων που απαιτούνται για τη λήψη συστάσεων.

- **Βελτίωση του προφίλ:** Αυτό σχετίζεται με την ικανότητα του χρήστη να παρέχει πληροφορίες (εισροές) στο σύστημα σχετικά με τις προτιμήσεις του. Αυτή είναι μια βασική θεμελιώδης αρχή που είναι απολύτως απαραίτητη για την παροχή εξατομικευμένων συστάσεων. Εάν το σύστημα δεν έχει συγκεκριμένες γνώσεις σχετικά με τον ενεργό χρήστη, μπορεί απλώς να του

παρέχει τις ίδιες συστάσεις οι οποίες θα παραδίδονταν και σε έναν "μέσο" χρήστη.

- **Έκφραση άποψης (Express self):** Ορισμένοι χρήστες μπορεί να μην ενδιαφέρονται καθόλου να λάβουν κάποια σύσταση. Αντιθέτως, αυτό που είναι σημαντικό για αυτούς είναι να τους επιτραπεί να συνεισφέρουν με τις αξιολογήσεις τους και να εκφράζουν τις απόψεις και τις πεποιθήσεις τους. Η ικανοποίηση των χρηστών για τη συγκεκριμένη δραστηριότητα μπορεί να λειτουργήσει ως μοχλός για τη στενή ενασχόληση του χρήστη με την εφαρμογή.

- **Παροχή βοήθειας σε άλλους:** Μερικοί χρήστες έχουν την ευχαρίστηση να συνεισφέρουν με πληροφορίες, π.χ. αξιολογήσεις, επειδή πιστεύουν ότι η κοινότητα επωφελείται από τη συμβολή τους. Αυτό θα μπορούσε να αποτελέσει σημαντικό κίνητρο για την εισαγωγή πληροφοριών σε ένα σύστημα συστάσεων που δεν χρησιμοποιείται συστηματικά. Για παράδειγμα, όσον αφορά ένα σύστημα συστάσεων για αυτοκίνητα, ένας χρήστης που έχει ήδη αγοράσει το καινούργιο του αυτοκίνητο γνωρίζει ότι η βαθμολογία που παρέχεται στο σύστημα είναι πιο πιθανό να είναι χρήσιμη για άλλους χρήστες και όχι για τον ίδιο (όταν θα θελήσει να αγοράσει ξανά ένα αυτοκίνητο).

- **Επίδραση σε άλλους χρήστες:** Υπάρχουν χρήστες με κύριο στόχο την επιρροή άλλων χρηστών στην αγορά συγκεκριμένων προϊόντων. Ως αποτέλεσμα, υπάρχουν και κάποιοι κακόβουλοι χρήστες που μπορούν να χρησιμοποιήσουν το σύστημα μόνο για να προωθήσουν ή να επιβάλλουν κυρώσεις σε ορισμένα αντικείμενα

Συνεπώς, ο ρόλος ενός συστήματος συστάσεων σε ένα πληροφοριακό σύστημα μπορεί να είναι αρκετά διαφορετικός. [11]

2.4 Η διαδικασία σύστασης και οι μορφές της

Τα δημογραφικά στοιχεία για το χρήστη όπως το φύλο, η ηλικία κ.λπ. αλλά και οι γνώσεις σχετικά με τις προτιμήσεις του συνθέτουν ένα “προφίλ χρήστη”. Παράλληλα, κάθε αντικείμενο περιγράφεται με τη βοήθεια χαρακτηριστικών γνωρισμάτων, τα οποία συμβάλλουν στη διαμόρφωση ενός προφίλ αντικειμένου, που ονομάζεται “περιεχόμενο”

Η καταγραφή των δεδομένων του χρήστη γίνεται με δύο τρόπους: είτε άμεσα (explicit) είτε έμμεσα (implicit). Καταγραφή δεδομένων άμεσα σημαίνει ότι ο χρήστης αλληλεπιδρά με το σύστημα ειδικά για τον σκοπό της παροχής των απαραίτητων πληροφοριών σε αυτό. Αντίθετα, καταγραφή δεδομένων έμμεσα σημαίνει ότι το σύστημα καταγράφει τα δεδομένα ενώ ο χρήστης αλληλεπιδρά με αυτό για άλλο σκοπό.

Η διαδικασία της σύστασης γίνεται σε διάφορες μορφές. Αρχικά, μπορεί να γίνει σύσταση ενός συγκεκριμένου χρήστη η οποία στοχεύει σε έναν άλλο χρήστη (συμπεριλαμβανομένου του εαυτού του) (ένας προς έναν). Αυτή η προσέγγιση πραγματοποιείται όταν ένας χρήστης έχει χρησιμοποιήσει ένα συγκεκριμένο τύπο προϊόντος το οποίο προτείνει σε έναν άλλο χρήστη όπως συμβαίνει στη συνομιλία μεταξύ δύο φίλων ή μεταξύ των καταναλωτών οι οποίοι αναφέρονται στην εμπειρία τους όσον αφορά τη χρήση ενός συγκεκριμένου προϊόντος.

Επιπρόσθετα, υπάρχει η σύσταση ενός συγκεκριμένου χρήστη η οποία απευθύνεται σε πολλούς χρήστες (έναν έως πολλούς). Χαρακτηριστικό παράδειγμα αυτής της σύστασης αποτελεί η εφαρμογή LivingSocial.com η οποία επιτρέπει στους χρήστες να επιλέγουν προϊόντα που τους αρέσουν, να κάνουν κριτική πάνω σε αυτά και να μοιράζονται αυτές τις κριτικές με φίλους τους σε μέσα κοινωνικής δικτύωσης (όπως είναι το Facebook).

Ακόμη, γίνονται συστάσεις από πολλούς χρήστες οι οποίες συγκεντρώνονται και συναθροίζονται μόνο για έναν συγκεκριμένο χρήστη. Σε όλους τους ανθρώπους δεν μπορούν να αρέσουν τα ίδια προϊόντα οπότε οι προτιμήσεις ορισμένων ανθρώπων δεν θα συμφωνούν με την πλειοψηφία. Για αυτό, υπάρχουν αλγόριθμοι που καλύπτουν αυτό το γεγονός, λαμβάνοντας έμμεσα ή και άμεσα καταγεγραμμένες συστάσεις χρηστών, προτείνοντάς τες σε χρήστες με παρόμοιο προφίλ. Αυτή η διαδικασία αναφέρεται συχνά ως συνεργατικό φιλτράρισμα.

Τέλος, γίνονται συστάσεις πολλών χρηστών σε πολλούς χρήστες (πολλοί σε πολλούς). Χαρακτηριστικό παράδειγμα αποτελούν οι αξιολογήσεις μη καθορισμένων χρηστών οι οποίες συναθροίζονται (π.χ. μέσος όρος) για ένα συγκεκριμένο προϊόν και διατίθενται σε πολλούς άλλους μη καθορισμένους χρήστες. [12]

2.5 Τύποι συστημάτων συστάσεων

Σημαντικότερες προσεγγίσεις των συστημάτων συστάσεων αποτελούν: η προσέγγιση βάσει περιεχομένου (content based filtering) και η προσέγγιση συνεργατικού φιλτραρίσματος (collaborative filtering). Υπάρχουν όμως και άλλες συμπληρωματικές προσεγγίσεις, όπως η προσέγγιση βασισμένη στη γνώση (knowledge-based filtering) και η προσέγγιση βασισμένη σε δημογραφικά στοιχεία (demography-based filtering). Επίσης, είναι εφικτός ο συνδυασμός δύο ή περισσότερων προσεγγίσεων των συστημάτων συστάσεων δημιουργώντας ένα υβριδικό σύστημα. [13] [14]

2.5.1 Συνεργατικό φιλτράρισμα (collaborative filtering)

Το συνεργατικό φιλτράρισμα είναι η διαδικασία φιλτραρίσματος ή αξιολόγησης αντικειμένων χρησιμοποιώντας τις απόψεις άλλων ανθρώπων. Η ύπαρξη αυτού του όρου, προέρχεται από την ανταλλαγή απόψεων μεταξύ τους. [15]

Τα πρώιμα συνεργατικά συστήματα φιλτραρίσματος σχεδιάστηκαν για να παρέχουν ρητά στους χρήστες πληροφορίες σχετικά με αντικείμενα. Δηλαδή, οι χρήστες επισκέπτονταν έναν διαδικτυακό τόπο με σκοπό τη λήψη συστάσεων από το σύστημα συνεργατικού φιλτραρίσματος. Αργότερα, οι διαδικτυακοί τόποι άρχισαν να χρησιμοποιούν συστήματα συνεργατικού φιλτραρίσματος πίσω από τα παρασκήνια για την προσαρμογή του περιεχομένου τους στους χρήστες. Χαρακτηριστικό παράδειγμα αποτελεί η επιλογή των άρθρων ειδήσεων που πρέπει να παρουσιάζει ένας διαδικτυακός τόπος εμφανώς στους χρήστες του. [15]

Οι πάροχοι πληροφοριών στο διαδίκτυο πρέπει να αντιμετωπίσουν την περιορισμένη προσοχή του χρήστη και τον περιορισμένο χώρο στην οθόνη. Το συνεργατικό φιλτράρισμα μπορεί να προβλέψει ποιες πληροφορίες είναι πιθανό οι χρήστες να θέλουν να δουν, επιτρέποντας στους παρόχους να επιλέγουν υποσύνολα πληροφοριών για εμφάνιση στον περιορισμένο χώρο της οθόνης. Η τοποθέτηση αυτών των πληροφοριών σε εμφανή θέση, επιτρέπει στον χρήστη να μεγιστοποιήσει την περιορισμένη προσοχή του. Με αυτόν τον τρόπο, το συνεργατικό φιλτράρισμα επιτρέπει στο διαδίκτυο την προσαρμογή στις ανάγκες κάθε μεμονωμένου χρήστη. [15]

Πιο συγκεκριμένα, το συνεργατικό φιλτράρισμα δημιουργεί ένα μοντέλο με βάση την προηγούμενη συμπεριφορά ενός χρήστη, τις δραστηριότητες ή τις προτιμήσεις του και διατυπώνει συστάσεις στον χρήστη βάσει παρόμοιων δραστηριοτήτων ή προτιμήσεων με άλλους χρήστες. [16]

2.5.1.1 Κατηγορίες συνεργατικού φιλτραρίσματος

Το συνεργατικό φιλτράρισμα μπορεί να χωριστεί σε δύο κατηγορίες: σε εκείνο που βασίζεται στη μνήμη (memory-based) και σε εκείνο που βασίζεται στο μοντέλο (model-based).

Οι αλγόριθμοι συνεργατικού φιλτραρίσματος προσανατολισμένοι στο μοντέλο παρέχουν συστάσεις αντικειμένων αναπτύσσοντας πρώτα ένα μοντέλο αξιολογήσεων χρηστών. Οι αλγόριθμοι σε αυτήν την κατηγορία ακολουθούν μια πιθανολογική προσέγγιση και υπολογίζεται η αναμενόμενη τιμή της πρόβλεψης του χρήστη δεδομένης της αξιολόγησής του σε άλλα αντικείμενα. Η διαδικασία της δημιουργίας του μοντέλου εκτελείται από διαφορετικούς αλγόριθμους μηχανικής μάθησης όπως είναι: το Bayesian δίκτυο, η συσταδοποίηση (clustering) και οι προσεγγίσεις βάσει κανόνων (rule-based approaches). Το Bayesian μοντέλο δικτύου σχηματίζει ένα πιθανολογικό μοντέλο για το πρόβλημα του συνεργατικού φιλτραρίσματος. Το μοντέλο συσταδοποίησης αντιμετωπίζει το συνεργατικό φιλτράρισμα ως ένα πρόβλημα κατηγοριοποίησης (classification). Λειτουργεί συγκεντρώνοντας παρόμοιους χρήστες στην ίδια κατηγορία και εκτιμώντας την πιθανότητα ότι ένας συγκεκριμένος χρήστης είναι σε μια συγκεκριμένη κατηγορία C, και από εκεί υπολογίζει την πιθανότητα των αξιολογήσεων με βάση κάποιες συνθήκες. Η προσέγγιση βάσει κανόνων εφαρμόζει αλγόριθμους εύρεσης κανόνων συσχέτισης προκειμένου να βρει τη συσχέτιση μεταξύ αντικειμένων που έχουν αγοραστεί από κοινού και στη συνέχεια δημιουργεί προτάσεις αντικειμένων με βάση την ισχύ της συσχέτισης μεταξύ αντικειμένων. [17]

Οι αλγόριθμοι που βασίζονται στη μνήμη χρησιμοποιούν όλη την βάση δεδομένων αντικειμένου-χρήστη για να δημιουργήσουν μια πρόβλεψη. Αυτά τα συστήματα χρησιμοποιούν τεχνικές της στατιστικής για την εύρεση ενός συνόλου χρηστών, γνωστών ως «γειτόνων», που έχουν ιστορικό συμφωνίας με τον χρήστη-στόχο (δηλαδή, είτε βαθμολογούν διαφορετικά αντικείμενα δίνοντάς τους παρόμοιες αξιολογήσεις είτε τείνουν να αγοράσουν ένα παρόμοιο σύνολο αντικειμένων). Όταν μια «γειτονιά» χρηστών σχηματίζεται, αυτά τα συστήματα χρησιμοποιούν διαφορετικούς αλγόριθμους για να συνδυάσουν τις προτιμήσεις των «γειτόνων» για την παραγωγή μιας πρόβλεψης ή μιας κορυφαίας σύστασης για τον ενεργό χρήστη. Οι τεχνικές, οι οποίες είναι επίσης γνωστές ως πλησιέστερος γείτονας (nearest-neighbor) ή το συνεργατικό φιλτράρισμα με βάση τον χρήστη (user-based collaborative filtering), είναι πιο δημοφιλείς και χρησιμοποιούνται ευρέως στην πράξη. [17]

2.5.1.2 Χρήσεις συνεργατικού φιλτραρίσματος

Οι εργασίες του χρήστη για τις οποίες είναι χρήσιμο το συνεργατικό φιλτράρισμα περιλαμβάνουν [15]:

- **Βοήθεια προκειμένου ο χρήστης να βρει νέα αντικείμενα που μπορεί να του αρέσουν:** Ο χρήστης δεν μπορεί να αξιολογήσει όλα τα αντικείμενα, σε έναν κόσμο ο οποίος υφίσταται συσσώρευση μεγάλου όγκου πληροφοριών. Για αυτό, το σύστημα συνεργατικού φιλτραρίσματος παρουσιάζει μερικά αντικείμενα στο χρήστη για να διαλέξει. Αυτό εφαρμόζεται πιο συχνά σε καταναλωτικά είδη (μουσική, βιβλία, ταινίες), αλλά μπορεί επίσης να εφαρμοστεί σε ερευνητικές εργασίες, ιστοσελίδες ή άλλα αντικείμενα με δυνατότητα αξιολόγησης.

- **Συμβουλές για ένα συγκεκριμένο αντικείμενο:** Ο χρήστης σκέπτεται ένα συγκεκριμένο αντικείμενο και ζητάει από την κοινότητα να τον ενημερώσει αν γνωρίζει κάτι σχετικά με αυτό.

- **Βοήθεια σε έναν χρήστη ώστε να βρει άλλον χρήστη (ή μερικούς χρήστες) που θα ήθελε:** Μερικές φορές, η γνώση σε ποιόν χρήστη χρειάζεται ένας άλλος χρήστης να στρέψει την προσοχή του είναι εξίσου σημαντική, με το να γνωρίζει σε τι να επικεντρωθεί. Αυτό μπορεί να βοηθήσει στον σχηματισμό ομάδων συζήτησης, στη σύνδεση χρηστών έτσι ώστε να είναι εφικτή η ανταλλαγή κοινωνικών συστάσεων.

- **Βοήθεια στις ομάδες ώστε να βρουν κάτι νέο που μπορεί να τους αρέσει:** Το συνεργατικό φιλτράρισμα μπορεί να βοηθήσει ομάδες ανθρώπων να βρουν αντικείμενα που μεγιστοποιούν την αξία. Για παράδειγμα, ένα ζευγάρι που επιθυμεί να δει μια ταινία μαζί ή μια ερευνητική ομάδα που θέλει να διαβάσει ένα επιστημονικό άρθρο.

- **Βοήθεια σχετικά με συγκεκριμένες εργασίες σε έναν τομέα:** Για παράδειγμα, ένα σύστημα συστάσεων ερευνητικών άρθρων μπορεί επίσης να επιθυμεί να υποστηρίξει εργασίες όπως είναι η σύσταση άρθρων τα οποία θα μπορούσαν να αναφερθούν στην βιβλιογραφία.

- **Βοήθεια στην εύρεση «νέων» ή «παλαιών» αντικειμένων:** Για παράδειγμα, η αγορά ορισμένων αντικειμένων τα οποία εξυπηρετούν τις ανάγκες του χρήστη που είτε αγοράζει για πρώτη φορά είτε τα έχει αγοράσει στο παρελθόν.

2.5.1.3 Λειτουργικότητα Συνεργατικού Φιλτραρίσματος

Το συνεργατικό φιλτράρισμα υποστηρίζει επίσης ομάδες εργασιών. Δεν είναι τυχαίο το γεγονός ότι αυτή η λειτουργικότητα του συστήματος σχετίζεται με τις εργασίες του χρήστη στις οποίες έγινε αναφορά στην προηγούμενη ενότητα. Στην ιδανική περίπτωση, το σύστημα θα υποστηρίζει όλες τις εργασίες του χρήστη, αν και η χαρτογράφηση μιας πραγματικής εφαρμογής στη λειτουργικότητα ενός πραγματικού συστήματος συνεργατικού φιλτραρίσματος μπορεί να αποτελέσει μια δύσκολη υπόθεση. Σε κάθε περίπτωση, παρακάτω παρουσιάζονται οι ομάδες της λειτουργικότητας των συστημάτων συνεργατικού φιλτραρίσματος [15]:

Προτεινόμενα Αντικείμενα: Εμφάνιση μιας λίστας αντικειμένων σε έναν χρήστη, με βάση την χρησιμότητά τους για εκείνον. Συχνά, αυτό περιγράφεται ως μια πρόβλεψη όσον αφορά τον τρόπο με τον οποίο θα αξιολογήσει ο χρήστης τα αντικείμενα και στη συνέχεια, ο χρήστης πράγματι πραγματοποιεί την αξιολόγηση των αντικειμένων όπως είχε προβλεφθεί. Ωστόσο, ορισμένοι επιτυχείς αλγόριθμοι συστάσεων δεν υπολογίζουν καθόλου τις προβλεπόμενες τιμές των αξιολογήσεων. Για παράδειγμα, ο αλγόριθμος συστάσεων της Amazon συγκεντρώνει στοιχεία παρόμοια με αυτά των αγορών και των αξιολογήσεων ενός χρήστη χωρίς να υπολογίζεται ποτέ μια προβλεπόμενη βαθμολογία. Αντί να παρουσιάζεται μια εξατομικευμένη βαθμολογία πρόβλεψης στον χρήστη, η διεπαφή του χρήστη εμφανίζει την μέση τιμή αξιολόγησης των πελατών. Ως αποτέλεσμα, η λίστα των συστάσεων μπορεί να προβάλλεται χωρίς σειρά σε σχέση με την εμφανιζόμενη μέση τιμή αξιολόγησης. Σε πολλές εφαρμογές, η επιλογή των κορυφαίων στοιχείων είναι πολύ σημαντική ενώ η παραγωγή προβλεπόμενων τιμών είναι δευτερεύουσας σημασίας.

Πρόβλεψη ενός συγκεκριμένου αντικειμένου. Δεδομένου ενός συγκεκριμένου αντικειμένου, υπολογίζεται η προβλεπόμενη αξιολόγησή του. Σε αυτό το σημείο, αξίζει να σημειωθεί ότι η πρόβλεψη μπορεί να είναι πιο απαιτητική υπόθεση από τη σύσταση. Στην περίπτωση της σύστασης αντικειμένων, ένα σύστημα το μόνο που χρειάζεται να κάνει είναι να προετοιμαστεί προκειμένου να προσφέρει μερικές εναλλακτικές λύσεις. Ορισμένοι αλγόριθμοι το εκμεταλλεύονται αυτό σε σχέση με την κλιμάκωση, εξοικονομώντας υπολογιστικούς πόρους. Για να υπάρξουν προβλέψεις για ένα συγκεκριμένο αντικείμενο, το σύστημα απαιτείται να είναι έτοιμο να δώσει αποτέλεσμα για οποιοδήποτε ζητούμενο αντικείμενο, ακόμη και εάν η αξιολόγησή του έχει πραγματοποιηθεί ελάχιστες φορές.

Εξειδικευμένη σύσταση. Υπάρχει ένα δεδομένο σύνολο περιορισμών και η σύσταση πραγματοποιείται μέσα από αυτό το σύνολο.

2.5.2 Συστήματα συστάσεων με βάση το περιεχόμενο

Όπως είδαμε, τα συστήματα συστάσεων συνεργατικού φιλτραρίσματος δεν χρησιμοποιούν χαρακτηριστικά των αντικειμένων προκειμένου να προβούν σε προβλέψεις. Τα συστήματα που βασίζονται στο περιεχόμενο έχουν σχεδιαστεί για να εκμεταλλεύονται σενάρια στα οποία μπορούν να περιγράφουν αντικείμενα με περιγραφικά σύνολα χαρακτηριστικών. [14]

Πιο συγκεκριμένα, τα συστήματα συστάσεων βάσει περιεχομένου προσπαθούν να αντιστοιχήσουν τους χρήστες με αντικείμενα παρόμοια με αυτά που τους άρεσαν στο παρελθόν. Αυτή η ομοιότητα δεν βασίζεται απαραίτητα σε συσχετισμούς αξιολογήσεων μεταξύ των χρηστών, αλλά σε χαρακτηριστικά των αντικειμένων που άρεσαν στον χρήστη. Σε αντίθεση με τα συστήματα συνεργατικού φιλτραρίσματος, τα οποία αξιοποιούν ρητά τις αξιολογήσεις των άλλων χρηστών εκτός από αυτές του χρήστη-στόχου, τα συστήματα που βασίζονται στο περιεχόμενο επικεντρώνονται σε μεγάλο βαθμό στις αξιολογήσεις του χρήστη-στόχου και τα χαρακτηριστικά των αντικειμένων που άρεσαν σε αυτόν. Επομένως, οι άλλοι χρήστες παίζουν πολύ μικρό ή και καθόλου ρόλο στα συστήματα συστάσεων που βασίζονται στο περιεχόμενο. Δηλαδή, η μεθοδολογία που βασίζεται στο περιεχόμενο αξιοποιεί μια διαφορετική πηγή δεδομένων για τη διαδικασία της σύστασης. [14]

Στο πιο βασικό επίπεδο, τα συστήματα συστάσεων με βάση το περιεχόμενο εξαρτώνται από δύο πηγές δεδομένων:

Η πρώτη πηγή δεδομένων είναι μια περιγραφή διαφόρων αντικειμένων με όρους σχετικούς με χαρακτηριστικά/γνωρίσματα τα οποία στηρίζονται στο περιεχόμενο. Ένα παράδειγμα μιας τέτοιας αναπαράστασης θα μπορούσε να είναι η περιγραφή του κειμένου ενός προϊόντος από τον κατασκευαστή. [14]

Η δεύτερη πηγή δεδομένων είναι ένα προφίλ χρήστη, το οποίο δημιουργείται από τα σχόλια των χρηστών σχετικά με διάφορα αντικείμενα. Τα σχόλια των χρηστών μπορεί να είναι άμεσα (ρητά) ή έμμεσα. Τα άμεσα σχόλια μπορούν να αντιστοιχούν σε αξιολογήσεις, ενώ τα έμμεσα σχόλια σε ενέργειες των χρηστών. Οι αξιολογήσεις συλλέγονται με παρόμοιο τρόπο με εκείνων των συνεργατικών συστημάτων. [14]

Το προφίλ χρήστη συσχετίζει τα χαρακτηριστικά των διαφόρων αντικειμένων με τα ενδιαφέροντα του χρήστη (βαθμολογίες). Ένα βασικό παράδειγμα προφίλ χρήστη μπορεί απλώς να είναι ένα σύνολο δεδομένων εκπαίδευσης με ετικέτα των χαρακτηριστικών των αντικειμένων, οι αξιολογήσεις του χρήστη ως ετικέτες και ένα μοντέλο ταξινόμησης ή παλινδρόμησης το οποίο συσχετίζει τα χαρακτηριστικά των αντικειμένων με τις αξιολογήσεις του χρήστη. Το προφίλ του συγκεκριμένου χρήστη εξαρτάται σε μεγάλο βαθμό από τη μεθοδολογία που υπάρχει. Για παράδειγμα, μπορούν να χρησιμοποιηθούν άμεσες αξιολογήσεις σε μία ρύθμιση ενώ έμμεσα σχόλια σε άλλη. Επιπρόσθετα, είναι πιθανό ο χρήστης να καθορίσει το δικό του προφίλ σε

όρους λέξεων-κλειδιών ενδιαφέροντος και αυτή η προσέγγιση μοιράζεται ορισμένα χαρακτηριστικά με συστήματα συστάσεων που βασίζονται στη γνώση. [14]

Θα πρέπει να τονιστεί ιδιαίτερα, το γεγονός ότι οι αξιολογήσεις των άλλων χρηστών συνήθως δεν παίζουν ρόλο στους αλγόριθμους συστάσεων με βάση το περιεχόμενο. Αυτό μπορεί να χαρακτηριστεί και ως πλεονέκτημα και ως μειονέκτημα, ανάλογα με την περίπτωση. Από τη μία πλευρά, όσον αφορά την περίπτωση της ψυχρής εκκίνησης, στην οποία ελάχιστες πληροφορίες σχετικά με τις αξιολογήσεις των άλλων χρηστών είναι διαθέσιμες, μια τέτοια προσέγγιση μπορεί ακόμα να χρησιμοποιηθεί εφόσον επαρκείς πληροφορίες για τα ενδιαφέροντα του χρήστη είναι διαθέσιμες. Αυτό, τουλάχιστον, ανακουφίζει εν μέρει το πρόβλημα της ψυχρής εκκίνησης όταν ο αριθμός των άλλων χρηστών στο σύστημα σύστασης είναι μικρός. Επιπλέον, όταν ένα αντικείμενο είναι καινούργιο, δεν είναι δυνατό να ληφθούν οι αξιολογήσεις των άλλων χρηστών για αυτό το αντικείμενο. Οι μέθοδοι με βάση το περιεχόμενο επιτρέπουν συστάσεις σε τέτοιες ρυθμίσεις επειδή αυτές μπορούν να εξαγάγουν τα χαρακτηριστικά από το νέο αντικείμενο και να τα χρησιμοποιήσουν για να κάνουν προβλέψεις. Από την άλλη πλευρά, το πρόβλημα της ψυχρής εκκίνησης για τους νέους χρήστες δεν μπορεί να αντιμετωπιστεί με τα συστήματα συστάσεων με βάση το περιεχόμενο. Επιπρόσθετα, όταν δεν χρησιμοποιούνται οι αξιολογήσεις των άλλων χρηστών, μειώνεται η διαφορετικότητα και η καινοτομία των προτεινόμενων αντικειμένων. Σε πολλές περιπτώσεις, τα αντικείμενα που συνίστανται μπορεί να είναι προφανή για τον χρήστη ή μπορεί να είναι άλλα αντικείμενα τα οποία έχει καταναλώσει προηγουμένως. Αυτό συμβαίνει επειδή τα χαρακτηριστικά περιεχομένου θα προτείνουν πάντα αντικείμενα με παρόμοια χαρακτηριστικά με αυτά που έχει δει ο χρήστης στο παρελθόν. Ένα προτεινόμενο αντικείμενο με παρόμοια χαρακτηριστικά συχνά παρουσιάζει μικρή έκπληξη στον χρήστη. [14]

2.5.3 Σύγκριση συνεργατικού φιλτραρίσματος με το φιλτράρισμα βάσει περιεχομένου

Οι μέθοδοι που βασίζονται στο περιεχόμενο έχουν πολλά πλεονεκτήματα και μειονεκτήματα σε σύγκριση με τις μεθόδους συνεργατικού φιλτραρίσματος. Τα πλεονεκτήματα των μεθόδων που βασίζονται στο περιεχόμενο είναι τα παρακάτω [14]:

1. Όταν ένα νέο αντικείμενο προστίθεται σε έναν πίνακα αξιολογήσεων, δεν έχει βαθμολογίες από τους χρήστες. Καμία από τις μεθόδους συνεργατικού φιλτραρίσματος βάσει μνήμης και μοντέλου δεν θα προτείνονταν για ένα τέτοιο αντικείμενο, επειδή δεν υπάρχουν επαρκείς αξιολογήσεις για τους σκοπούς της σύστασης. Από την άλλη πλευρά, στην περίπτωση των μεθόδων βάσει περιεχομένου, τα προηγούμενα στοιχεία που έχουν αξιολογηθεί από έναν συγκεκριμένο χρήστη αξιοποιούνται για να κάνουν προτάσεις. Επομένως, εφόσον ο χρήστης δεν είναι «νέος» χρήστης, μπορούν πάντα να γίνουν σημαντικές προτάσεις υλοποίησης με τρόπο που να αντιμετωπίζει το «νέο» αντικείμενο με δίκαιο τρόπο σε σύγκριση με τα άλλα αντικείμενα. Τα συστήματα συνεργατικού φιλτραρίσματος αντιμετωπίζουν το πρόβλημα της ψυχρής εκκίνησης τόσο για τους «νέους» χρήστες όσο και για τα «νέα»

αντικείμενα, λαμβάνοντας υπόψη ότι τα συστήματα με βάση το περιεχόμενο υφίσταται το πρόβλημα της ψυχρής εκκίνησης μόνο για «νέους» χρήστες.

2. Οι μέθοδοι με βάση το περιεχόμενο παρέχουν εξηγήσεις σχετικά με τους όρους των χαρακτηριστικών των αντικειμένων. Αυτό συχνά δεν είναι δυνατό με τις συστάσεις συνεργατικού φιλτραρίσματος.

3. Οι μέθοδοι που βασίζονται στο περιεχόμενο μπορούν γενικά να χρησιμοποιηθούν ως τυποποιημένοι ταξινομητές κειμένου. Επιπλέον, κάθε πρόβλημα ταξινόμησης για κάθε χρήστη δεν είναι γενικά πολύ μεγάλο, όπως στην περίπτωση των συστημάτων συνεργατικού φιλτραρίσματος.

Από την άλλη πλευρά, οι μέθοδοι συστάσεων που βασίζονται στο περιεχόμενο έχουν επίσης πολλά μειονεκτήματα έναντι των μεθόδων συνεργατικού φιλτραρίσματος.

1. Τα συστήματα συστάσεων βάσει περιεχομένου τείνουν να βρίσκουν αντικείμενα παρόμοια με αυτά που έχει δει ο χρήστης. Αυτό το πρόβλημα αναφέρεται ως «υπερεξειδίκευση» (overspecialization). Είναι πάντα επιθυμητό να έχουμε μια ορισμένη ποσότητα καινοτομίας και ευνοϊκής συγκυρίας (serendipity) στις συστάσεις. Η καινοτομία αναφέρεται στο γεγονός ότι το αντικείμενο είναι διαφορετικό από αυτό που έχει δει ο χρήστης στο παρελθόν. Ομοίως, η έννοια της ευνοϊκής συγκυρίας υπονοεί ότι ο χρήστης θα ήθελε να ανακαλύψει «εκπληκτικά» αντικείμενα τα οποία σχετίζονται μεταξύ τους που διαφορετικά τους δεν έχουν βρεθεί. Αυτό αποτελεί πρόβλημα για τα συστήματα που βασίζονται στο περιεχόμενο στα οποία τα μοντέλα ταξινόμησης βάσει χαρακτηριστικών τείνουν να προτείνουν αντικείμενα τα οποία έχουν αρκετές ομοιότητες. Για παράδειγμα, εάν ένας χρήστης δεν έχει ακούσει ποτέ ή δεν έχει αξιολογήσει ποτέ του κλασική μουσική, το σύστημα συστάσεων που βασίζεται στο περιεχόμενο συνήθως δεν θα του προτείνει κάτι τέτοιο γιατί η κλασική μουσική θα περιγράφεται από πολύ διαφορετικές τιμές χαρακτηριστικών από αυτές που έχει αξιολογήσει ο χρήστης μέχρι τώρα. [14]

   Από την άλλη πλευρά, ένα σύστημα συνεργατικού φιλτραρίσματος μπορεί να προτείνει τέτοια στοιχεία αξιοποιώντας τα ενδιαφέροντα της όμοιας ομάδας του. Για παράδειγμα, ένα σύστημα συνεργατικού φιλτραρίσματος μπορεί να συμπεραίνει αυτόματα μια «εκπληκτική» σχέση μεταξύ συγκεκριμένων λαϊκών και κλασικών τραγουδιών και να προτείνει τα αντίστοιχα κλασικά τραγούδια σε έναν χρήστη που του αρέσει αρκετά η λαϊκή μουσική. Η υπερεξειδίκευση και η έλλειψη ευνοϊκής συγκυρίας είναι οι δύο πιο σημαντικές προκλήσεις των συστημάτων σύστασης βάσει περιεχομένου. [14]

2. Ακόμα κι αν τα συστήματα που βασίζονται στο περιεχόμενο βοηθούν στην επίλυση του προβλήματος της ψυχρής εκκίνησης για καινούργια αντικείμενα, δεν βοηθούν στην επίλυση αυτών των προβλημάτων όσον αφορά τους «νέους» χρήστες. Στην πραγματικότητα, σχετικά με τους «νέους» χρήστες, το πρόβλημα των συστημάτων βάσει περιεχομένου μπορεί να είναι πιο σοβαρό, επειδή το μοντέλο κατηγοριοποίησης κειμένου απαιτεί συνήθως επαρκή αριθμό δεδομένων εκπαίδευσης για να αποφευχθεί η υπερβολική τοποθέτηση.

Φαίνεται μάλλον υπερβολικό ότι τα δεδομένα εκπαίδευσης για όλους τους άλλους χρήστες απορρίπτονται και μόνο το (μικρό) σύνολο δεδομένων εκπαίδευσης που προορίζεται για έναν μόνο χρήστη αξιοποιείται. [14]

2.5.4 Συστήματα συστάσεων με βάση την γνώση (knowledge-based approach)

Τα συστήματα συστάσεων με βάση τη γνώση είναι κατάλληλα για τη σύσταση αντικειμένων που δεν αγοράζονται συχνά. Επιπλέον, σε αυτούς τους τομείς αντικειμένων, οι χρήστες είναι γενικά πιο σαφείς σχετικά με τις απαιτήσεις τους. Για παράδειγμα, ένας χρήστης μπορεί συχνά να είναι πρόθυμος να αποδεχτεί μια σύσταση για μια ταινία γνωρίζοντας ελάχιστες πληροφορίες, αλλά θα ήταν απρόθυμος να αποδεχτεί συστάσεις για ένα σπίτι ή ένα αυτοκίνητο χωρίς να διαθέτει λεπτομερείς πληροφορίες σχετικά με συγκεκριμένα χαρακτηριστικά του αντικειμένου. Επομένως, τα συστήματα συστάσεων που βασίζονται στη γνώση είναι κατάλληλα σε τύπους τομέων/πεδίων αντικειμένων διαφορετικών από εκείνων των συστημάτων συστάσεων με βάση το περιεχόμενο και τη συνεργασία (collaborative). Γενικά, τα συστήματα συστάσεων που βασίζονται στη γνώση είναι κατάλληλα στις ακόλουθες περιπτώσεις [14]:

1. Οι πελάτες θέλουν να προσδιορίσουν άμεσα τις απαιτήσεις τους. Επομένως, η διαδραστικότητα είναι κρίσιμο στοιχείο αυτών των συστημάτων. Αξίζει να σημειωθεί, ότι τα συστήματα συστάσεων συνεργατικού φιλτραρίσματος και τα συστήματα συστάσεων με βάσει το περιεχόμενο δεν επιτρέπουν αυτόν τον τύπο της λεπτομερής ανατροφοδότησης.

2. Είναι δύσκολο να αποκτηθούν αξιολογήσεις για έναν συγκεκριμένο τύπο αντικειμένου, λόγω της μεγαλύτερης πολυπλοκότητας στον τομέα του προϊόντος όσον αφορά τους τύπους των αντικειμένων και τις διαθέσιμες επιλογές.

3. Σε ορισμένους τομείς, όπως είναι εκείνος των υπολογιστών, οι αξιολογήσεις ενδέχεται να είναι ευαίσθητες στο χρόνο (time-sensitive). Για παράδειγμα, οι αξιολογήσεις σε ένα παλιό αυτοκίνητο ή υπολογιστή δεν είναι πολύ χρήσιμες για συστάσεις επειδή εξελίσσονται μεταβάλλοντας τη διαθεσιμότητα του προϊόντος και τις αντίστοιχες απαιτήσεις των χρηστών.

Ένα κρίσιμο τμήμα των συστημάτων που βασίζονται στη γνώση είναι ο μεγαλύτερος έλεγχος που έχει ο χρήστης στην καθοδήγηση της διαδικασίας σύστασης. Αυτός είναι ένα άμεσο αποτέλεσμα της ανάγκης να καθοριστούν λεπτομερείς απαιτήσεις σε έναν εγγενώς περίπλοκο τομέα προβλημάτων. Εδώ θα πρέπει να τονιστεί ιδιαίτερα το γεγονός ότι, οι συστάσεις των συστημάτων που βασίζονται στο περιεχόμενο και των συστημάτων συνεργατικού φιλτραρίσματος βασίζονται κυρίως σε ιστορικά δεδομένα, ενώ εκείνων που βασίζονται στη γνώση στηρίζονται στις άμεσες πληροφορίες των χρηστών σχετικά με το τι αυτοί θέλουν. Ένα σημαντικό χαρακτηριστικό το οποίο διακρίνει τα συστήματα που βασίζονται στη γνώση είναι ένα υψηλό επίπεδο εξατομίκευσης (customization) στον συγκεκριμένο τομέα. [14]

Αυτή η εξατομίκευση επιτυγχάνεται μέσω της χρήσης μιας βάσης γνώσεων που κωδικοποιεί σχετικές γνώσεις στον τομέα με τη μορφή περιορισμών ή μετρήσεων ομοιότητας. Ορισμένα συστήματα που βασίζονται στη γνώση ενδέχεται επίσης να χρησιμοποιούν χαρακτηριστικά των χρηστών (π.χ. δημογραφικά χαρακτηριστικά) εκτός από τα χαρακτηριστικά των αντικειμένων, τα οποία καθορίζονται την ώρα του ερωτήματος. Σε τέτοιες περιπτώσεις, είναι επίσης εφικτή η κωδικοποίηση σχέσεων μεταξύ χαρακτηριστικών του χρήστη και χαρακτηριστικών των αντικειμένων. Ωστόσο, η χρήση τέτοιων χαρακτηριστικών δεν είναι γενικευμένη στα συστήματα με βάση τη γνώση, τα οποία εστιάζουν κυρίως στις απαιτήσεις των χρηστών. [14]

Τα συστήματα συστάσεων που βασίζονται στη γνώση μπορούν να κατηγοριοποιηθούν με βάση τη διαδραστική μεθοδολογία του χρήστη και τις αντίστοιχες βάσεις γνώσης που χρησιμοποιούνται για να διευκολύνουν την αλληλεπίδραση.

Υπάρχουν δύο βασικοί τύποι συστημάτων σύστασης που βασίζονται στη γνώση [14]:

1. Συστήματα σύστασης βάσει περιορισμών (constraint-based recommender systems): Στα συστήματα βάσει περιορισμών, οι χρήστες συνήθως καθορίζουν απαιτήσεις ή περιορισμούς (π.χ. κατώτερα ή ανώτερα όρια) στα χαρακτηριστικά του αντικειμένου. Επιπλέον, κανόνες για συγκεκριμένους τομείς χρησιμοποιούνται ώστε να ταιριάζουν με τις απαιτήσεις του χρήστη ή με τα χαρακτηριστικά του αντικειμένου. Αυτοί οι κανόνες αντιπροσωπεύουν τη συγκεκριμένη γνώση του τομέα που χρησιμοποιείται από το σύστημα. Ακόμη, οι κανόνες αυτοί θα μπορούσαν να λάβουν τη μορφή συγκεκριμένων περιορισμών πεδίου σχετικά με τα χαρακτηριστικά του αντικειμένου (π.χ., "Τα αυτοκίνητα πριν από το έτος 1970 δεν έχουν cruise control."). Επιπλέον, τα συστήματα που βασίζονται σε περιορισμούς συχνά δημιουργούν κανόνες συσχετίζοντας χαρακτηριστικά των χρηστών με χαρακτηριστικά των αντικειμένων (π.χ. "Οι μεγαλύτεροι επενδυτές δεν επενδύουν σε προϊόντα υψηλού κινδύνου."). Σε τέτοιες περιπτώσεις, τα χαρακτηριστικά του χρήστη μπορούν επίσης να καθοριστούν στη διαδικασία αναζήτησης. Η εξάρτηση σχετικά με τον αριθμό και τον τύπο των επιστρεφόμενων αποτελεσμάτων, μπορεί να δώσει στον χρήστη την ευκαιρία να τροποποιήσει τις αρχικές απαιτήσεις του. Για παράδειγμα, ένας χρήστης μπορεί να χαλαρώσει μερικούς περιορισμούς όταν επιστρέφονται πολύ λίγα αποτελέσματα ή να προσθέσει περισσότερους περιορισμούς όταν επιστρέφονται πάρα πολλά αποτελέσματα Αυτή η διαδικασία αναζήτησης επαναλαμβάνεται διαδραστικά μέχρι ο χρήστης να φτάσει στα αποτελέσματα που επιθυμεί.

2. Συστήματα βασισμένα στην υπόθεση (case-based recommender systems:): Στα συστήματα συστάσεων τα οποία βασίζονται στην υπόθεση, συγκεκριμένες περιπτώσεις προσδιορίζονται από τον χρήστη ως στόχοι. Οι μετρικές ομοιότητας ορίζονται στα χαρακτηριστικά των αντικειμένων για την ανάκτηση παρόμοιων αντικειμένων με αυτούς τους στόχους. Οι μετρικές ομοιότητας συχνά καθορίζονται προσεκτικά για έναν συγκεκριμένο τομέα. Ως εκ τούτου, οι μετρικές ομοιότητας αποτελούν τη γνώση του πεδίου που χρησιμοποιείται σε τέτοια συστήματα. Τα αποτελέσματα που επιστρέφονται χρησιμοποιούνται

συχνά ως νέες περιπτώσεις στόχων με ορισμένες διαδραστικές τροποποιήσεις από τον χρήστη. Για παράδειγμα, όταν ένας χρήστης βλέπει ένα αποτέλεσμα που του έχει επιστραφεί, το οποίο είναι σχεδόν παρόμοιο με αυτό που θέλει, μπορεί να επανεκδώσει ξανά ένα ερώτημα με αυτόν τον στόχο, αλλά με μερικά από τα χαρακτηριστικά να έχουν μεταβληθεί σύμφωνα με τις προτιμήσεις του. Εναλλακτικά, μπορεί να καθοριστεί μια κατευθυντική κριτική ώστε να περικόψει αντικείμενα με συγκεκριμένες τιμές χαρακτηριστικών μεγαλύτερες (ή μικρότερες) από αυτές ενός συγκεκριμένου αντικειμένου ενδιαφέροντος. Αυτή η διαδραστική διαδικασία χρησιμοποιείται για να καθοδηγήσει τον χρήστη στην τελική σύσταση.

2.5.5 Συστήματα συστάσεων βασισμένα σε δημογραφικά στοιχεία

Στα συστήματα συστάσεων με βάση τα δημογραφικά στοιχεία, οι δημογραφικές πληροφορίες σχετικά με τον χρήστη αξιοποιούνται για να μάθουν κατηγοριοποιητές (classifiers), οι οποίοι να μπορούν να αντιστοιχίσουν συγκεκριμένα δημογραφικά στοιχεία σε αξιολογήσεις ή τάσεις της αγοράς. Ένα πρώιμο σύστημα σύστασης, που αναφέρεται ως Grundy [18], προτείνει βιβλία με βάση μια βιβλιοθήκη με μη αυτόματο τρόπο. Τα χαρακτηριστικά του χρήστη συλλέγονται με τη χρήση ενός διαδραστικού διαλόγου. Σύμφωνα με το AI Magazine [19], παρατηρήθηκε ότι οι δημογραφικές ομάδες από την έρευνα μάρκετινγκ μπορούν να χρησιμοποιηθούν για να προτείνουν αντικείμενα. Με βάση την [20], γίνονται συστάσεις σε ιστοσελίδες με βάση τα δημογραφικά χαρακτηριστικά των χρηστών, που έχουν αξιολογήσει μια συγκεκριμένη ιστοσελίδα με υψηλή βαθμολογία. Σε πολλές περιπτώσεις, οι δημογραφικές πληροφορίες μπορούν να συνδυαστούν με πρόσθετο πλαίσιο για την καθοδήγηση της διαδικασίας σύστασης. [14]

Οι πιο πρόσφατες τεχνικές έχουν επικεντρωθεί στη χρήση κατηγοριοποιητών (classifiers) για την δημιουργία συστάσεων. Ένα από τα ενδιαφέροντα συστήματα από αυτή την άποψη ήταν μια τεχνική που εξήγαγε χαρακτηριστικά από τις αρχικές σελίδες των χρηστών (user home pages) για να προβλέψει την πιθανότητα να τους αρέσουν ορισμένα εστιατόρια. Οι κατηγοριοποιητές βάσει κανόνων [21] [22] χρησιμοποιούνται συχνά για να συσχετίσουν το δημογραφικό προφίλ με την αγοραστική συμπεριφορά με έναν διαδραστικό τρόπο. Ενώ η προσέγγιση στις [21] [22] η οποία δεν χρησιμοποιήθηκε ειδικά για να προτείνει συγκεκριμένα αντικείμενα, μπορεί εύκολα να συνδυαστεί με ένα σύστημα σύστασης. Αν και τα συστήματα συστάσεων βάσει δημογραφικών στοιχείων συνήθως δεν παρέχουν τα καλύτερα αποτελέσματα σε αυτόνομη βάση, όμως η συνεισφορά τους είναι σημαντική, με τη δύναμη άλλων συστημάτων σύστασης, ως συστατικό των υβριδικών μοντέλων. Τα συστήματα συστάσεων βάσει δημογραφικών στοιχείων μερικές φορές συνδυάζονται με τα συστήματα συστάσεων βάσει γνώσεων για να αυξήσουν την αντοχή τους. [14]

2.5.6 Υβριδικά Συστήματα Συστάσεων

Στις προηγούμενες ενότητες έγινε αναφορά σε ορισμένες διαφορετικές κατηγορίες μεθόδων σύστασης. Οι μέθοδοι συνεργασίας (collaborative) χρησιμοποιούν τις αξιολογήσεις μιας κοινότητας χρηστών προκειμένου να κάνουν προτάσεις, ενώ οι μέθοδοι συστάσεων που βασίζονται στο περιεχόμενο χρησιμοποιούν τις αξιολογήσεις ενός μεμονωμένου χρήστη σε συνδυασμό με τις περιγραφές κεντρικών χαρακτηριστικών των αντικειμένων για την διεξαγωγή συστάσεων. Οι μέθοδοι συστάσεων που βασίζονται στη γνώση απαιτούν άμεσα τις απαιτήσεις των χρηστών για να κάνουν προτάσεις, χωρίς να χρειάζεται ιστορικό αξιολογήσεων. Επομένως, αυτές οι μέθοδοι χρησιμοποιούν διαφορετικές πηγές δεδομένων, και έχουν διαφορετικά πλεονεκτήματα και μειονεκτήματα. Για παράδειγμα, τα συστήματα συστάσεων που βασίζονται στη γνώση μπορούν να αντιμετωπίσουν ζητήματα του προβλήματος της ψυχρής εκκίνησης πολύ καλύτερα από ότι τα συστήματα συστάσεων βάσει περιεχομένου ή εκείνων της συνεργασίας (collaborative), επειδή δεν απαιτούν αξιολογήσεις. Από την άλλη πλευρά, είναι πιο αδύναμα από τα συστήματα συστάσεων που βασίζονται στο περιεχόμενο και από τα συστήματα με βάση τη συνεργασία (collaborative) όσον αφορά τη χρήση μόνιμης εξατομίκευσης από ιστορικά δεδομένα. Εάν ένας διαφορετικός χρήστης εισάγει τις ίδιες απαιτήσεις και δεδομένα σε μια διαδραστική διεπαφή με γνώσεις, μπορεί να έχει ακριβώς το ίδιο αποτέλεσμα. [14]

Σε γενικές γραμμές, θα ήταν ιδανικό να γίνει χρήση όλης της διαθέσιμης γνώσης η οποία προέρχεται από διαφορετικές πηγές δεδομένων και να μπορεί επίσης να χρησιμοποιηθεί η αλγοριθμική ισχύς των διαφόρων συστημάτων σύστασης δίνοντας ως αποτέλεσμα ισχυρά συμπεράσματα. Τα υβριδικά συστήματα σύστασης έχουν σχεδιαστεί για να τα εξερευνήσουν τέτοιες δυνατότητες. Υπάρχουν τρεις βασικοί τρόποι δημιουργίας υβριδικών συστημάτων συστάσεων [14]:

1. Σχεδιασμός συνόλου (ensemble design): Σε αυτόν τον σχεδιασμό, συνδυάζονται αποτελέσματα από off-the-shelf αλγόριθμους σε μία και πιο ισχυρή έξοδο. Για παράδειγμα, μπορούν να συνδυαστούν οι έξοδοι ενός συστήματος συστάσεων που βασίζεται στο περιεχόμενο και ενός συστήματος συνεργατικού φιλτραρίσματος σε μια έξοδο. Επιπρόσθετα, υπάρχει μια σημαντική παραλλαγή όσον αφορά τις συγκεκριμένες μεθοδολογίες που χρησιμοποιούνται για τη διαδικασία του συνδυασμού. Η βασική αρχή αυτής της εργασίας δεν διαφέρει πολύ από τον σχεδιασμό μεθόδων για σύνολα σε πολλές εφαρμογές της εξόρυξης δεδομένων όπως είναι η συσταδοποίηση (clustering), η ταξινόμηση (classification) και οι στατιστικοί έλεγχοι για σπάνια σημεία (outlier analysis).

2. Μονολιθικός σχεδιασμός: Σε αυτήν την περίπτωση, δημιουργείται ένας ολοκληρωμένος αλγόριθμος προτάσεων χρησιμοποιώντας διάφορους τύπους δεδομένων. Μερικές φορές μπορεί να μην υπάρχει σαφής διάκριση μεταξύ των διαφόρων τμημάτων (π.χ. με βάση το περιεχόμενο και με βάση τη συνεργασία (collaborative)) του αλγόριθμου. Σε άλλες περιπτώσεις, η ύπαρξη αλγόριθμων

συστάσεων συνεργατικού φιλτραρίσματος ή αλγόριθμων συστάσεων βασισμένων στο περιεχόμενο, ίσως χρειάζεται να τροποποιηθούν, ώστε να χρησιμοποιηθούν στο πλαίσιο της συνολικής προσέγγισης, ακόμη και όταν υπάρχουν σαφείς διακρίσεις μεταξύ των σταδίων εκείνων που βασίζονται στο περιεχόμενο και εκείνων του συνεργατικού φιλτραρίσματος.

3. Μικτά συστήματα: Όπως και τα σύνολα, αυτά τα συστήματα χρησιμοποιούν πολλαπλούς αλγόριθμους προτάσεων, αλλά τα στοιχεία που προτείνουν τα διάφορα συστήματα παρουσιάζονται μαζί δίπλα-δίπλα. Για παράδειγμα, το τηλεοπτικό πρόγραμμα μιας ολόκληρης μέρας είναι μια σύνθετη οντότητα που περιέχει πολλά στοιχεία. Δεν έχει νόημα η παρακολούθηση της σύστασης ενός μεμονωμένου αντικειμένου, είναι ο συνδυασμός των αντικειμένων που δημιουργεί τη σύσταση.

Επομένως, ο όρος «υβριδικό σύστημα» (hybrid system) χρησιμοποιείται σε ευρύτερο πλαίσιο από ότι ο όρος «σύνολο συστήματος» (ensemble system). Όλα τα συστήματα συνόλου είναι εξ ορισμού υβριδικά συστήματα, αλλά το αντίστροφο δεν είναι απαραίτητα αληθές. [14]

Αν και τα υβριδικά συστήματα συστάσεων συνήθως συνδυάζουν τη δύναμη διαφορετικών τύπων σύστασης (π.χ. σύσταση με βάση την γνώση), δεν υπάρχει κανένας λόγος για τον οποίο τα συστήματα αυτά να μην μπορούν να συνδυάσουν μοντέλα του ίδιου τύπου. Δεδομένου ότι τα μοντέλα τα οποία βασίζονται στο περιεχόμενο είναι ουσιαστικά ταξινομητές κειμένου, είναι πολύ γνωστό ότι υπάρχει μεγάλη ποικιλία μοντέλων συνόλων (ensemble models) για τη βελτίωση της ακρίβειας της ταξινόμησης. Επομένως, μπορεί να χρησιμοποιηθεί οποιοδήποτε σύστημα συνόλου βασισμένο στην κατηγοριοποίηση (classification-based ensemble system) για τη βελτίωση της αποτελεσματικότητας των μοντέλων που βασίζονται στο περιεχόμενο. [14]

Σε ευρύτερο επίπεδο, τα υβριδικά συστήματα συστάσεων μπορούν να συσχετιστούν στενά με το πεδίο της ανάλυσης συνόλου στην κατηγοριοποίηση. Για παράδειγμα, τα συνεργατικά μοντέλα είναι γενικεύσεις του μοντέλου ταξινόμησης. [14]

Σύμφωνα με τον Burke [23], τα υβριδικά συστήματα συστάσεων μπορούν να ταξινομηθούν στις ακόλουθες κατηγορίες:

- Εφαρμόζοντας ξεχωριστά τα διαφορετικά συστήματα και παρουσιάζοντας τα αποτελέσματά τους είτε μαζί, είτε σε ξεχωριστές λίστες.

- Σταθμίζοντας τα αποτελέσματα των επιμέρους συστημάτων για να εξάγουμε ένα ενιαίο αποτέλεσμα.

- Επιλέγοντας με βάση κάποιο κριτήριο να χρησιμοποιήσουμε τα αποτελέσματα από ένα μόνο σύστημα.

- Βελτιστοποιώντας τις εξαγόμενες συστάσεις του ενός συστήματος με χρήση των άλλων.

- Συνδυάζοντας δεδομένα από διαφορετικές πηγές και αναλύοντάς τα από ένα σύστημα συστάσεων.

- Χρησιμοποιώντας σαν είσοδο του ενός συστήματος τις εξαγόμενες συστάσεις του άλλου.

- Δημιουργώντας ένα μοντέλο βασισμένο σε ένα σύστημα, το οποίο μετά χρησιμοποιείται σαν είσοδο στο δεύτερο.

Εξαιτίας της ιδιότητας που έχουν οι υβριδικές μεθοδολογίες να ελαχιστοποιούν τα μειονεκτήματα των επιμέρους συστατικών τους, αναπτύσσονται όλο και περισσότερα υβριδικά συστήματα συστάσεων.

ΚΕΦΑΛΑΙΟ 3: Federated Learning In Recommender Systems

Στο παρόν κεφάλαιο πραγματοποιείται μια σύντομη εισαγωγή στο πρόβλημα του Federated Learning σε συστήματα συστάσεων που καλείται να αντιμετωπίσει η παρούσα εργασία και στην συνέχεια παρατίθενται η σχετική βιβλιογραφική επισκόπηση πάνω στο ζήτημα αυτό.

3.1 Federated Learning σε συστήματα συστάσεων

Μεταξύ της εκτεταμένης έρευνας για την προστασία της ιδιωτικότητας στα συστήματα συστάσεων υπάρχουν μέθοδοι που περιλαμβάνουν αλγόριθμους κρυπτογραφίας [24] [25]. Τα συμβατικά κατανεμημένα συστήματα συστάσεων επιταχύνουν τους υπολογισμούς κατά την διαίρεση των κρυπτογραφημένων δεδομένων σε κομμάτια και την επεξεργασία σε πολλαπλά νήματα ενός αξιόπιστου διακομιστή [26]. Μια πιο πρόσφατη κατανεμημένη τεχνική μηχανικής μάθησης είναι το federated learning. Όπως έγινε αντιληπτό από το πρώτο κεφάλαιο, το federated learning διασφαλίζει την ιδιωτικότητα των χρηστών αφού τα δεδομένα και το ίδιο το μοντέλο δεν μεταδίδονται, ούτε μπορούν να υπολογισθούν από τα δεδομένα της άλλης οντότητας.

3.2 Σχετική δουλειά

Στην δημοσίευση [27] ο αλγόριθμος που προτείνεται βασίζεται στην ιδέα του meta-learning. Θεωρείται το γεγονός ότι οι συσκευές έχουν αρκετά δεδομένα για να μάθουν ένα μοντέλο το οποίο βασίζεται μόνο σε τοπικά δεδομένα. Τότε, το federated learning χρησιμοποιείται για να βρει τις βέλτιστες υπερπαραμέτρους για τον αλγόριθμο, χρησιμοποιώντας τις συσκευές για να υπολογίσουν τα gradients των υπερπαραμέτρων.

Πιο συγκεκριμένα, προτείνεται ένα federated meta-learning πλαίσιο για σύσταση το οποίο διαμοιράζει πληροφορίες χρηστών σε επίπεδο αλγορίθμου ενώ διατηρείται παράλληλα το απόρρητο των χρηστών. Σε αυτό το πλαίσιο, υπάρχει ένας παραμετροποιημένος αλγόριθμος ο οποίος παραμετροποιείται από τις παραμέτρους του μοντέλου και εκπαιδεύει παραμετροποιημένα μοντέλα συστάσεων. Δηλαδή, τόσο ο αλγόριθμος όσο και τα μοντέλα παραμετροποιούνται και ως εκ τούτου θα πρέπει να βελτιστοποιηθούν. Αρχικά, στο επίπεδο του αλγορίθμου, ο διακομιστής στέλνει τον παραμετροποιημένο αλγόριθμο, σε ένα σύνολο δειγματοληπτούμενων χρηστών. Στη συνέχεια, στο επίπεδο του μοντέλου που λειτουργεί στις συσκευές των χρηστών, κάθε δειγματοληπτούμενος χρήστης εκπαιδεύει ένα συγκεκριμένο μοντέλο χρήστη χρησιμοποιώντας τον τρέχον αλγόριθμο (support set) και αξιολογεί το μοντέλο στην αντίστοιχη συσκευή (query set). Δηλαδή, σε αυτό το πλαίσιο, το meta-training πραγματοποιείται με κατανεμημένο τρόπο, όπου κάθε χρήστης έχει ένα συγκεκριμένο μοντέλο που εκπαιδεύεται χρησιμοποιώντας τοπικά δεδομένα. Επίσης, αυτό το μοντέλο αξιολογείται για την παροχή gradient απωλειών δοκιμών (test loss gradients) οι οποίες χρησιμοποιούνται για τη βελτίωση της ικανότητας του αλγορίθμου να

εκπαιδεύει μοντέλα. Τέλος, στο επίπεδο του αλγορίθμου που λειτουργεί στον διακομιστή, ο διακομιστής συλλέγει από αυτούς τους χρήστες τις gradient απώλειες δοκιμών για να ενημερώσει τις παραμέτρους του μοντέλου. [27]

Στην εργασία [28], προτείνεται ένα federated meta-learning πλαίσιο όπου ένας παραμετροποιημένος αλγόριθμος διαμοιράζεται. Πιο συγκεκριμένα, χρησιμοποιείται το federated meta-learning πλαίσιο που παρουσιάστηκε παραπάνω και αποτελεί μεταγενέστερο άρθρο του προηγούμενου. Εδώ, κάθε πελάτης (χρήστης) μεταχειρίζεται ως ένα task. Ειδικότερα, το meta-learning εξελίσσεται με βάση τα επεισόδια, όπου σε κάθε επεισόδιο μια παρτίδα από tasks δειγματοληπτείται από μια κατανομή tasks T πάνω σε ένα meta-training set. Σε αυτήν εδώ την περίπτωση όμως, η προσέγγιση εκτελείται στους meta-learning αλγορίθμους Model-Agnostic Meta-learning (MAML) και Meta-SGD. Στόχος είναι συνεργατικά να γίνει meta-train σε έναν αλγόριθμο χρησιμοποιώντας δεδομένα τα οποία κατανέμονται μεταξύ των πελατών (χρηστών). Λαμβάνοντας τον MAML ως παράδειγμα εκτέλεσης απώτερος σκοπός είναι η εκπαίδευση μιας αρχικοποίησης για το μοντέλο χρησιμοποιώντας όλα μαζί τα δεδομένα των πελατών (clients). Ο MAML περιλαμβάνει δύο επίπεδα βελτιστοποίησης: έναν εσωτερικό βρόχο για την εκπαίδευση μοντέλων συγκεκριμένων tasks και έναν εξωτερικό βρόχο για την ενημέρωση της αρχικοποίησης με απώλειες δοκιμών των tasks. Επομένως, η μεταδιδόμενη πληροφορία σε αυτήν την διαδικασία, αποτελείται από την αρχικοποίηση των παραμέτρων του μοντέλου (από τον διακομιστή στους πελάτες (χρήστες)) και από τις απώλειες δοκιμών (από τους πελάτες στον διακομιστή). Έτσι, δεν απαιτείται η συλλογή δεδομένων στον διακομιστή. Τέλος, είναι σημαντικό να τονιστεί το γεγονός ότι, στον Meta-SGD επιπρόσθετα με τις παραμέτρους του μοντέλου μεταδίδεται επίσης και ένα διάνυσμα ως τμήμα των παραμέτρων του αλγορίθμου το οποίο χρησιμοποιείται ως εσωτερικός βρόχος εκπαίδευσης του μοντέλου. Συμπερασματικά, με αυτό το πλαίσιο, βελτιώνεται η γενίκευση πάνω σε νέα tasks.

Στην δημοσίευση [29], χρησιμοποιείται το federated learning σε ένα κατανεμημένο σύστημα σύστασης με απλό σχεδιασμό, όπου νευρωνικά δίκτυα με την ίδια αρχιτεκτονική εκπαιδεύονται στην συσκευή κάθε χρήστη. Στόχος είναι η χρήση του federated learning σε ένα κατανεμημένο σύστημα συστάσεων με απλό σχεδιασμό, όπου νευρωνικά δίκτυα με την ίδια αρχιτεκτονική εκπαιδεύονται στις συσκευές των χρηστών. Οι χρήστες λαμβάνουν βελτιωμένες προτάσεις καθώς αντικαθιστούν επανειλημμένα τις παραμέτρους του μοντέλου το οποίο εκτελείται στις συσκευές τους με τις συναθροίσεις τους που λαμβάνονται από τον διακομιστή παραμέτρων. Με αυτόν τον τρόπο, οι αξιολογήσεις των χρηστών για διαφορετικά αντικείμενα παραμένουν στις συσκευές τους, ενώ τα μοτίβα των προτιμήσεών τους κοινοποιούνται με ασφάλεια μέσω ενός κεντρικού διακομιστή στο cloud.

Ειδικότερα, μια ομάδα συσκευών (edge-devices) συνδέεται στον parameter server προκειμένου να αρχικοποιήσει τα μοντέλα. Ο parameter server στέλνει το αρχικό διάνυσμα παραμέτρων σε κάθε συσκευή. Στην συγκεκριμένη μεθοδολογία υπάρχουν 3 διακριτές φάσεις: το global training, το local training και το test. Οι ροές αυτών των φάσεων που αρχίζουν με το global training, έπειτα ακολουθεί το local training και συνέχεια η δοκιμή (test) καλούνται κύκλοι (cycles). Επομένως, σε κάθε κύκλο, κάθε

συσκευή ξεκινά την φάση της εκπαίδευσης (global training). Σε αυτήν, η συσκευή στέλνει τα διανύσματα των παραμέτρων μετά από μερικά βήματα stochastic gradient descent (SGD). Τα διανύσματα των παραμέτρων συναθροίζονται στον parameter server και στέλνονται πίσω στις συσκευές. Αυτές χρησιμοποιούν το συναθροιζόμενο διάνυσμα παραμέτρων που έχουν λάβει για να αρχικοποιήσουν τις παραμέτρους του μοντέλου στα επόμενα βήματα του SGD. Μετά από έναν προκαθορισμένο αριθμό γύρων επικοινωνίας, οι συσκευές σταματούν να στέλνουν τις παραμέτρους τους και σώζουν ένα αντίγραφο του διανύσματος των παραμέτρων τους τοπικά. Ύστερα, αρχίζουν να εκπαιδεύουν το μοντέλο τους τοπικά (local training). Αυτή η εκπαίδευση (local training) βοηθά τις συσκευές να προσαρμόσουν το ολικό (global) διάνυσμα των παραμέτρων το οποίο εκπαιδεύεται σε όλες τις συσκευές έτσι ώστε το μοντέλο σε κάθε ατομική συσκευή να ταιριάζει στα τοπικά του δεδομένα. Έπειτα, μόλις η τοπική εκπαίδευση (local training ολοκληρωθεί, μετα-φορτώνουν αυτό το αντίγραφο πίσω στο μοντέλο. Τέλος, αφού ο επόμενος κύκλος ξεκινήσει, μπορεί εύκολα να πραγματοποιηθεί σύγκριση του πόσο διαφορετικά θα ήταν τα αποτελέσματα των δοκιμών εάν οι συσκευές επικοινωνούσαν για μεγαλύτερο χρονικό διάστημα πριν μεταβούν στην τοπική εκπαίδευση. [29]

Στην [30], χρησιμοποιείται η τεχνική του matrix factorization[5] προκειμένου να υλοποιηθεί η πρώτη federated collaborative filter μέθοδος. Όμως, αντιμετωπίζονται ορισμένες προκλήσεις. Οι ενημερώσεις των διανυσμάτων παραγόντων των αντικειμένων χρειάζονται ένα σύνολο δυαδικών τιμών για implicit feedback και μια παράμετρο εμπιστοσύνης τα οποία βρίσκονται διαθέσιμα στον διακομιστή. Επομένως, ο υπολογισμός των ενημερώσεων των παραγόντων αντικειμένων δεν θα γίνει στους πελάτες αλλά στον διακομιστή. Όμως, με αυτόν τον τρόπο, δεν διατηρείται η ιδιωτικότητα των χρηστών καθώς οι αλληλεπιδράσεις μεταξύ χρηστών-αντικειμένων θα πρέπει να παραμείνουν μόνο στην συσκευή του χρήστη. Για αυτό, υιοθετείται μια stochastic gradient descent προσέγγιση για να ενημερωθούν τα διανύσματα των αντικειμένων στον διακομιστή ενώ διατηρείται η ιδωτικότητα των χρηστών.

Αναλυτικότερα, όλα τα διανύσματα παραγόντων των αντικειμένων (item factor vectors) ενημερώνονται στον διακομιστή και έπειτα κατανέμονται σε κάθε πελάτη. Τα διανύσματα παραγόντων των χρηστών (user factor vectors) ενημερώνονται τοπικά σε κάθε πελάτη χρησιμοποιώντας τα δεδομένα του χρήστη και τα διανύσματα παραγόντων των αντικειμένων από τον διακομιστή. Έπειτα, οι ενημερώσεις μέσω των gradients των διανυσμάτων παραγόντων των αντικειμένων υπολογίζονται σε κάθε πελάτη και στέλνονται στον διακομιστή όπου τα gradients συναθροίζονται για να ενημερώσουν το κύριο μοντέλο (πίνακας παραγόντων αντικειμένων) . [30]

Η εργασία [31] αποσκοπεί στην σύγκριση των προσεγγίσεων gossip learning και federated learning σε ένα σύστημα συστάσεων με βάση το low-rank matrix decomposition. Αρχικά, περιγράφεται το πρόβλημα low rank matrix decomposition το οποίο διατυπώνεται ως πρόβλημα μηχανικής μάθησης και παρουσιάζονται οι βασικές

[5] Η παραγοντοποίηση πινάκων (Matrix Factorization, ή εν συντομία MF) αποτελεί μία ισχυρή τεχνική αποσύνθεσης πινάκων, η οποία προσεγγίζει έναν πίνακα παρατηρήσεων X ως το γινόμενο δύο επιμέρους χαμηλής τάξης πινάκων W και H. Γενικά, η παραγοντοποίηση πινάκων βασίζεται στην λογική ότι, κατά τον πολλαπλασιασμό πινάκων, οι πίνακες-παράγοντες μπορούν να έχουν πολύ μικρότερες διαστάσεις από τον πίνακα-αποτέλεσμα. Επομένως, από την παραγοντοποίηση πινάκων προκύπτουν πίνακες-παράγοντες με διαστάσεις σημαντικά ελαττωμένες συγκριτικά με τον αρχικό πίνακα παρατηρήσεων.

ιδέες για την επίλυση του με κατανεμημένο τρόπο. Στη συνέχεια περιγράφεται η federated learning προσέγγιση για την επίλυση του συγκεκριμένου προβλήματος. Έπειτα, παρουσιάζεται ο gossip learning αλγόριθμος προκειμένου να επιλυθεί το ίδιο πρόβλημα. Τέλος, αναφέρονται τα βασικά στοιχεία των learning αλγορίθμων τα οποία είναι κοινά και στις δύο προσεγγίσεις.

Πιο συγκεκριμένα, ο federated learning αλγόριθμος προσαρμόζεται στο πρόβλημα του rank-k matrix decomposition. Σε αυτό το πλαίσιο υπάρχει ένας master κόμβος και αρκετοί worker κόμβοι. Ο master κόμβος στην αρχή αρχικοποιεί το global μοντέλο. Μετά την αρχικοποίηση σε κάθε γύρο, ο master κόμβος στέλνει το global μοντέλο σε όλους τους worker κόμβους. Έπειτα, οι worker ενημερώνουν το global μοντέλο που έχουν λάβει και το δικό τους τοπικό μοντέλο χρήστη χρησιμοποιώντας τις τοπικές αξιολογήσεις. Ύστερα, οι worker κόμβοι στέλνουν το πιθανώς συμπιεσμένο gradient του μοντέλου στον master. Στο τέλος κάθε γύρου, ο διακομιστής ενημερώνει το global μοντέλο με τον μέσο όρο των ληφθέντων gradients.

Αξίζει να σημειωθεί, ότι στο federated learning πλαίσιο θεωρείται το γεγονός ότι οι worker συγχρονίζονται. Δηλαδή, ο master κόμβος πρέπει να περιμένει έως ότου όλοι (ή οι περισσότεροι) οι κόμβοι να στείλουν το gradient στον συγκεκριμένο γύρο. Και οι worker όμως θα πρέπει να περιμένουν μέχρι το επόμενο global συναθροιζόμενο μοντέλο να επεξεργαστεί από τον master. Τέλος, εφαρμόζονται τεχνικές συμπίεσης κατά τη μεταφόρτωση των ενημερώσεων στο διακομιστή.

Στην [32], παρουσιάζεται μια πρακτική μέθοδος για το federated leraning βαθιών δικτύων (deep networks) βάσει επαναλήψεων των μέσων όρων του μοντέλου. Υπάρχει ένα σύγχρονο σχήμα ενημέρωσης που προχωρά σε γύρους επικοινωνίας (rounds of communication) και ένα σταθερό σύνολο πελατών, ο καθένας με ένα σταθερό τοπικό σύνολο δεδομένων. Στην αρχή κάθε γύρου, επιλέγεται ένα τυχαίο κλάσμα των πελατών και ο διακομιστής στέλνει την τρέχουσα κατάσταση του ολικού (global) αλγορίθμου σε καθέναν από τους πελάτες (π.χ. τις τρέχουσες παραμέτρους του μοντέλου). Έπειτα, κάθε επιλεγμένος πελάτης εκτελεί τοπικό υπολογισμό με βάση την ολική (global) κατάσταση και το τοπικό του σύνολο δεδομένων και στέλνει μια ενημέρωση στο διακομιστή. Στη συνέχεια, ο διακομιστής εφαρμόζει αυτές τις ενημερώσεις στην ολική (global) κατάσταση και η διαδικασία επαναλαμβάνεται. Πιο αναλυτικά, παρουσιάστηκε ο FederatedSGD αλγόριθμος ο οποίος χρησιμοποιεί την μέθοδο large-batch synchronous SGD για βελτιστοποίηση επιλέγοντας ένα κλάσμα πελατών σε κάθε γύρο και υπολογίζοντας το gradient απωλειών για όλα τα δεδομένα που κατέχουν αυτοί οι πελάτες. Κατά την εφαρμογή του FederatedSGD χρησιμοποιώντας όλο το τοπικό dataset ως μονή παρτίδα και σταθερό ρυθμό εκμάθησης, κάθε πελάτης υπολογίζει το μέσο gradient στα τοπικά του δεδομένα στο τρέχον μοντέλο και ο κεντρικός server συναθροίζει αυτά τα gradients και εκτελεί την ενημέρωση. Δηλαδή, κάθε πελάτης κάνει τοπικά ένα βήμα gradient descent στο τρέχον μοντέλο χρησιμοποιώντας τα τοπικά του δεδομένα, και ο διακομιστής παίρνει έπειτα τον μέσο όρο των βαρών των μοντέλων που προκύπτουν. Σε αυτήν την περίπτωση, μπορεί να προστεθεί περισσότερος υπολογισμός σε κάθε πελάτη επαναλαμβάνοντας την τοπική ενημέρωση πριν το βήμα του μέσου όρου. Αυτή η προσέγγιση ονομάζεται Federated Averaging. Εδώ, η ποσότητα του υπολογισμού ελέγχεται από τρεις βασικές παραμέτρους: το κλάσμα των πελατών που εκτελεί υπολογισμό σε κάθε γύρο, τον αριθμό των

περασμάτων εκπαίδευσης που κάνει κάθε πελάτης στο τοπικό σύνολο δεδομένων του σε κάθε γύρο (αριθμός των local epoch, ένα epoch αποτελεί ένα πλήρες πέρασμα του training set) και το τοπικό μέγεθος παρτίδας.

3.2.1 Συγκεντρωτικός Πίνακας Προσεγγίσεων

Πίνακας 2: Συγκεντρωτικός πίνακας προσεγγίσεων.

Δημοσίευση	Αλγόριθμος	Προσέγγιση recommender system	Προσέγγιση federated learning	Προσέγγιση Federated meta-learning	Μοντέλο σύστασης
Federated Meta-Learning for recommendation/Federated Meta-Learning with Fast Convergence and Efficient Communication	Model-Agnostic Meta-learning (MAML) & Meta-SGD	content-based recommendation		Διαμοιρασμός (sharing) σε επίπεδο αλγορίθμου σε μικρά μοντέλα, διατηρώντας την ιδιωτικότητα	νευρωνικό δίκτυο & κατηγοριοποιητής: νευρωνικό δίκτυο ή λογιστική παλινδρόμιση
A Simple and Efficient Federated Recommender System	Federated learning for recommendation		Απλή επέκταση του Federated Learning		νευρωνικό δίκτυο
Federated Collaborative Filtering For Privacy-Preserving Personalized Recommendation System	Federated Collaborative Filter (FCF) Θεωρείται παραλλαγή του ALS αλγόριθμου	collaborative Filtering τεχνική: matrix factorization	Federated έκδοση της collaborative filtering μεθόδου για implicit σύνολα δεδομένων		factor matrix
Decentralized Recommendation based on Matrix Factorization: A Comparison of Gossip and Federated Learning⋆	Federated Learning Αλγόριθμος	collaborative Filtering τεχνική: low-rank matrix decomposition	Προσαρμογή της προσέγγισης low rank matrix Decomposition στο federated learning		low-rank matrix decomposition
Communication-Efficient Learning of Deep Networks from Decentralized Data	FederatedSGD/ Federated Averaging		federated leraning βαθιών δικτύων βάσει επαναλήψεων των μέσων όρων του μοντέλου		

Από τις παραπάνω δημοσιεύσεις προκύπτει ο παρακάτω συγκεντρωτικός πίνακας προσεγγίσεων.

Στον συγκεκριμένο πίνακα παρουσιάζεται το όνομα του αλγορίθμου, οι προσεγγίσεις συστημάτων συστάσεων, federated learning, federated meta-learning καθώς και το μοντέλο σύστασης κάθε δημοσίευσης, εφόσον αυτά υπάρχουν. Όσον αφορά την προσέγγιση των συστημάτων συστάσεων παρατηρούνται δύο από τις βασικές κατηγορίες των συστημάτων συστάσεων: τα συστήματα συστάσεων που είναι βασισμένα στο περιεχόμενο (content based) και το συνεργατικό φιλτράρισμα (collaborative filtering). Στην προσέγγιση federated learning γίνονται διακριτές ορισμένες παραλλαγές της όπως είναι η federated έκδοση της collaborative filtering μεθόδου για implicit σύνολα δεδομένων, η προσαρμογή της προσέγγισης low rank matrix decomposition στο federated learning και το federated leraning βαθιών δικτύων βάσει επαναλήψεων των μέσων όρων του μοντέλου εκτός της βασικής federated learning προσέγγισης. Μόνο μια δημοσίευση χρησιμοποιεί την προσέγγιση federated meta-learning. Τέλος, τα μοντέλα σύστασης ποικίλουν όντας στην πλειοψηφία τους νευρωνικά δίκτυα.

ΚΕΦΑΛΑΙΟ 4: Υλοποίηση Federated Learning εφαρμογής σε συστήματα συστάσεων και πειραματικά αποτελέσματα

Στο κεφάλαιο αυτό, αρχικά, αναφέρονται εν συντομία τα εργαλεία ανάπτυξης και οι βιβλιοθήκες που χρησιμοποιήθηκαν για την υλοποίηση της federated learning εφαρμογής σε συστήματα συστάσεων. Στην συνέχεια, παρουσιάζεται η προσέγγιση federated learning η οποία υλοποιήθηκε. Ύστερα, γίνεται σύντομη αναφορά του συνόλου δεδομένων το οποίο χρησιμοποιήθηκε, του ποσοστού των test και train data, των βασικών στοιχείων υλοποίησης και της federated εκπαίδευσης του μοντέλου. Τέλος, παρουσιάζονται τα πειραματικά αποτελέσματα.

4.1 Εργαλεία ανάπτυξης και βιβλιοθήκες

Τα βασικότερα εργαλεία ανάπτυξης και οι κυριότερες βιβλιοθήκες που χρησιμοποιήθηκαν περιγράφονται παρακάτω.

Ο κώδικας γράφτηκε σε γλώσσα προγραμματισμού Python. [33]

Το Keras είναι ένα API μηχανικής μάθησης υψηλού επιπέδου που επιτρέπει την δημιουργία, εκπαίδευση, εύκολη αξιολόγηση και εκτέλεση όλων των ειδών νευρωνικών δικτύων. Το documentation του είναι διαθέσιμο στο https://keras.io. Η εφαρμογή αναφοράς ονομάζεται απλά Keras. Υπάρχουν τρεις δημοφιλείς open source βιβλιοθήκες μηχανικής μάθησης: TensorFlow, Microsoft Cognitive Toolkit (CNTK) και Theano. Στην συγκεκριμένη εργασία χρησιμοποιήθηκε το TensorFlow. Το TensorFlow περιλαμβάνει την δικιά του Keras εφαρμογή που ονομάζεται tf.keras. Υποστηρίζει μόνο το TensorFlow ως backend, αλλά έχει το πλεονέκτημα ορισμένων πολύ χρήσιμων επιπλέον δυνατοτήτων. Για παράδειγμα, υποστηρίζει το API δεδομένων του TensorFlow που καθιστά πολύ εύκολη την φόρτωση και την προεπεξεργασία δεδομένων αποτελεσματικά. [34]

Η βιβλιοθήκη NumPy. Αποτελεί βασική υποδομή για επιστημονικές εφαρμογές εισάγοντας την δημιουργία N-διάστατων πινάκων. [35]

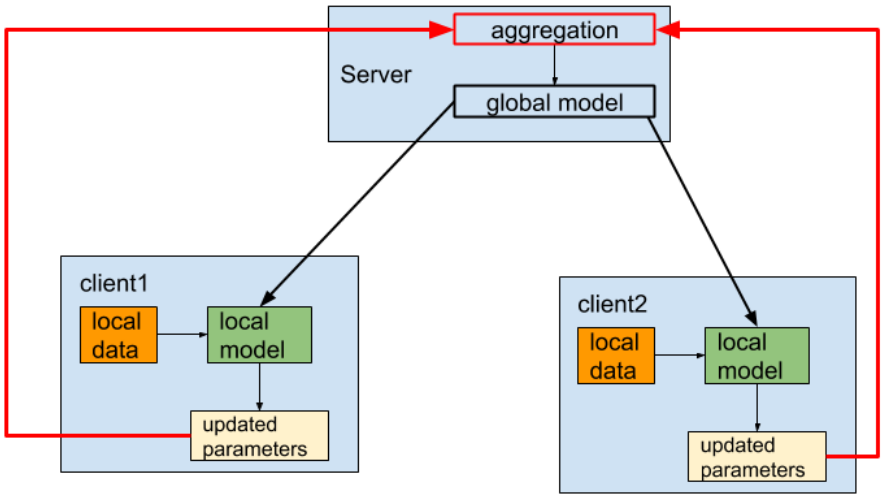
Η βιβλιοθήκη Pandas εισάγει την δομή δεδομένων Dataframes η οποία αποτελεί μια δισδιάστατη δομή δεδομένων όπου κάθε στήλη της δομής πρέπει να έχει τον ίδιο τύπο δεδομένων αλλά οι διαφορετικές στήλες μπορούν να έχουν διαφορετικούς τύπους. [36]

Η βιβλιοθήκη random εφαρμόζει ψευδοτυχαίους γεννήτορες για ποικίλες κατανομές. [37]

4.2 Προσέγγιση federated learning

Ο κάθε χρήστης (client) λαμβάνει τα βάρη του τρέχοντος ολικού (global) μοντέλου από τον διακομιστή και μαθαίνει το δικό του μοντέλο συστάσεων με βάση τα τοπικά

δεδομένα του και την πληροφορία που έρχεται από τον server δημιουργώντας ενημερωμένες παραμέτρους οι οποίες στη συνέχεια μεταφορτώνονται πίσω στο διακομιστή για συνάθροιση (aggregation) όπως παρουσιάζεται στην Εικόνα 6. Αυτός ο κύκλος επικοινωνίας συνεχίζεται έως ότου επιτευχθεί ένας προκαθορισμένος αριθμός epoch. Δηλαδή, ο στόχος είναι να δημιουργηθεί ένα βελτιωμένο ολικό μοντέλο χρησιμοποιώντας πληροφορία από τους τοπικούς χρήστες (clients).



Εικόνα 6:Federated learning προσέγγιση

4.3 Σύνολο δεδομένων

Το σύνολο δεδομένων που χρησιμοποιήθηκε στην συγκεκριμένη εργασία είναι το: MovieLens 1M Dataset (https://grouplens.org/datasets/movielens/1m/). Αυτό περιλαμβάνει αξιολογήσεις που έγιναν από ένα σύνολο χρηστών σε διαφορετικές ταινίες. Πιο συγκεκριμένα, περιέχει 1000209 αξιολογήσεις ταινιών από 6040 χρήστες σε 3706 ταινίες [38]. Αυτό είναι ένα πολύ μεγάλο σύνολο δεδομένων, κατάλληλο για να δοκιμάσουμε αλγορίθμους μηχανικής μάθησης. Η αξιολόγηση κάθε χρήστη είναι ένας ακέραιος από 1 ως 5.

4.4 Test και Train data

Γίνεται διαχωρισμός των δεδομένων σε train και test data. Το 90% των δεδομένων χρησιμοποιείται για training και το 10% για testing.

4.5 Βασικά στοιχεία υλοποίησης

Η συγκεκριμένη εργασία παρουσιάζει μια προσομοίωση του Federated Learning. Οι πελάτες αναπαρίστανται από τα κομμάτια στα οποία σπάνε τα δεδομένα και όλα τα τοπικά μοντέλα εκπαιδεύονται στο ίδιο μηχάνημα.

4.5.1 Δημιουργία Μοντέλου

Χρησιμοποιείται ένα embedding layer που αναμένει τον αριθμό των χρηστών ή ταινιών. Δηλαδή, δημιουργούνται τα embeddings τόσο για τους χρήστες όσο και για τις ταινίες. Στην συνέχεια, λαμβάνεται το εσωτερικό γινόμενο και των δύο ενσωματώσεων.

4.5.2 Συνάθροιση μοντέλου (Federated Averaging)

Πιο συγκεκριμένα με βάση τον Federated Averaging Algorithm [32] ο οποίος αναφέρθηκε και στο προηγούμενο κεφάλαιο στον οποίο έχει βασιστεί η συγκεκριμένη εργασία, η συνάθροιση σημαίνει απλώς μια λειτουργία μέσου όρου.

Υπολογίζεται ο μέσος όρος των παραμέτρων με βάση την αναλογία των δεδομένων που συνεισφέρει κάθε συμμετέχων πελάτης. Αυτή είναι η federated εξίσωση μέσου όρου που χρησιμοποιήθηκε στην συγκεκριμένη εργασία.
Ο αλγόριθμος εφαρμόζεται σε οποιαδήποτε αντικειμενική συνάρτηση του τύπου:

$$\min_{w \in \mathbb{R}^d} f(w) \qquad \text{where} \qquad f(w) \stackrel{\text{def}}{=} \frac{1}{n} \sum_{i=1}^{n} f_i(w).$$

Για ένα πρόβλημα μηχανικής μάθησης, τυπικά παίρνουμε το : $fi(w)=l(xi,yi;w)$ το οποίο είναι η απώλεια της πρόβλεψης στο (xi,yi) η οποία γίνεται με παραμέτρους μοντέλου w. Γίνεται η υπόθεση ότι υπάρχουν τουλάχιστον K πελάτες στους οποίους τα δεδομένα κατανέμονται, με το Pk να είναι το σύνολο των δεικτών των σημείων δεδομένων στον πελάτη k, με $nk=|Pk|$. Επομένως, η αντικειμενική μπορεί να ξαναγραφεί ως [32]:

$$f(w) = \sum_{k=1}^{K} \frac{n_k}{n} F_k(w) \quad \text{where} \quad F_k(w) = \frac{1}{n_k} \sum_{i \in \mathcal{P}_k} f_i(w).$$

Εδώ, στη δεξιά πλευρά, εκτιμώνται οι παράμετροι βάρους για κάθε πελάτη με βάση τις τιμές απωλειών που καταγράφονται στα δεδομένα με τα οποία έγινε η εκπαίδευση. Στην αριστερά, γίνεται scaled κάθε μία από αυτές τις παραμέτρους και αθροίζονται όλες μαζί.

Η συνάρτηση weight_scaling_factor υπολογίζει την αναλογία των τοπικών δεδομένων εκπαίδευσης ενός πελάτη με τα συνολικά δεδομένα εκπαίδευσης που κατέχουν όλοι οι πελάτες. Αρχικά, αποκτάται το μέγεθος της παρτίδας του πελάτη και χρησιμοποιείται για να υπολογιστεί ο αριθμός των δεδομένων του. Στη συνέχεια, αποκτάται το ολικό (global) μέγεθος των δεδομένων εκπαίδευσης. Έπειτα, υπολογίζεται ο συντελεστής κλιμάκωσης (scaling factor) ως κλάσμα. Εδώ, κάθε πελάτης αναμένεται να υποδείξει τον αριθμό των δεδομένων με τα οποία εκπαιδεύτηκε, ενώ ενημερώνει τον διακομιστή με νέες παραμέτρους μετά από κάθε τοπικό βήμα εκπαίδευσης.

Η συνάρτηση scale_model_weights μετράει κάθε ένα από τα βάρη του τοπικού μοντέλου βάσει της τιμής του συντελεστή που υπολογίζεται στην συνάρτηση weight_scalling_factor.

Η συνάρτηση sum_scaled_weights αθροίζει τα βάρη όλων των πελατών μαζί

4.6 Federated Εκπαίδευση του μοντέλου

Η λογική της εκπαίδευσης έχει δύο κύριους βρόχους, ο εξωτερικός βρόχος είναι για την ολική (global) επανάληψη και ο εσωτερικός είναι για επανάληψη μέσω της τοπικής εκπαίδευσης των πελατών.
Στον εξωτερικό βρόχο, πρώτα, λαμβάνονται τα αρχικά βάρη του ολικού μοντέλου και διασφαλίζεται η τυχαιότητα. Ύστερα, ξεκινάει η επανάληψη της εκπαίδευσης των πελατών.
Για κάθε πελάτη, δημιουργήθηκε ένα νέο αντικείμενο μοντέλου, έγινε compile και τέθηκαν τα βάρη της αρχικοποίησης του με τις τρέχουσες παραμέτρους του ολικού μοντέλου.
Ακολούθησε η εκπαίδευση του τοπικού μοντέλου των πελατών.
Μετά την εκπαίδευση, τα νέα βάρη των πελατών έγιναν scaled. Αυτή ήταν η τοπική εκπαίδευση. Επιστρέφοντας στον εξωτερικό βρόχο, αθροίστηκαν όλα τα scaled τοπικά βάρη και ενημερώθηκε το ολικό μοντέλο με αυτή την νέα συνάθροιση. Εδώ, ολοκληρώνεται ένα πλήρες epoch εκπαίδευσης.

Εκτελέστηκαν 5 ολικοί βρόχοι εκπαίδευσης και το εκπαιδευμένο ολικό μοντέλο έγινε tested στα tested δεδομένα μετά από κάθε γύρο.

4.7 Πειραματικά αποτελέσματα

Εδώ, θα παρουσιάσουμε ορισμένα πειράματα πάνω στην Federated Learning εφαρμογή σε συστήματα συστάσεων.

Αρχικά, αναφέρονται οι παράμετροι του αλγορίθμου που χρησιμοποιήθηκαν. Οι παράμετροι αυτοί είναι:

- Μέθοδος βελτιστοποίησης: Adam [39]
- Ρυθμός εκμάθησης (learning rate - LR): 0.001
- Embedding size = 50
- Αριθμός πελατών = 10
- Μετρική προς βελτιστοποίηση: Μέσο τετραγωνικό σφάλμα - Mean Squared Error (MSE) [40]
- Batch size: 1000
- Αριθμός γύρων επικοινωνίας: 5
- Αριθμός εποχών ανά γύρο επικοινωνίας (epochs): 1
- Βήματα ανά εποχή (Steps): 10

Σημειώνεται ότι ο αριθμός γύρων αναφέρεται στην συλλογή δεδομένων από όλους τους πελάτες για τον υπολογισμό των συνολικών βαρών. Σε έναν πελάτη, το epoch είναι ένα πέρασμα όλων των δεδομένων εκπαίδευσης από τον αλγόριθμο εκπαίδευσης, ο οποίος επεξεργάζεται τα δεδομένα σε batches(παρτίδες) όπως έχει ήδη αναφερθεί. Πιο αναλυτικά, όσον αφορά τα βήματα ανά epoch, παίρνουμε τα δεδομένα, τα διαιρούμε σε batches και για κάθε batch εκτελούμε τον Adam αλγόριθμο και

υπολογίζουμε τα βάρη. Αυτός ο υπολογισμός είναι ένα βήμα και επαναλαμβάνεται number_of_steps φορές πριν προχωρήσουμε στο επόμενο batch.

Στη συνέχεια, πραγματοποιούνται αλλαγές σε κάποιες από αυτές τις παραμέτρους για να βελτιωθεί το αποτέλεσμα. Έτσι, θα γίνει αντιληπτό πως η κάθε παράμετρος επηρεάζει το αποτέλεσμα.

Τέλος, συγκρίνεται το federated μοντέλο, με μια εκτέλεση έχοντας συγκεντρωμένα τα δεδομένα μαζί (απλή προσέγγιση). Στην τελευταία περίπτωση, ορίζεται και ένας αριθμός epoch.

4.7.1 Μετρικές αξιολόγησης

Για να προσδιοριστεί η ακρίβεια μιας πρόβλεψης χρησιμοποιούνται διάφορα μέτρα αξιολόγησης. Γενικά το σφάλμα πρόβλεψης θεωρείται η διαφορά μεταξύ της πραγματικής και της προβλεπόμενης τιμής και εκφράζεται ως:

$$e_t = Y_t - \hat{Y}_t$$

όπου Yt η πραγματική τιμή και $\hat{Y}t$ η προβλεπόμενη για τη χρονική στιγμή t.

Παρακάτω, θα αναλυθεί το μέσο τετραγωνικό σφάλμα.

4.7.1.1 MSE

Υπολογίζεται ως η μέση τιμή των τετραγώνων της διαφοράς μεταξύ των πραγματικών και των προβλεπόμενων τιμών. Η βέλτιστη τιμή είναι το 0.0. Μεγαλύτερα σφάλματα έχουν ως αποτέλεσμα περισσότερα λάθη παρά μικρότερα, το οποίο σημαίνει ότι το μοντέλο τιμωρείται για μεγαλύτερα λάθη.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2$$

4.7.2 Ανάλυση πειραμάτων στο Federated Model

Παρακάτω παρουσιάζονται 7 δοκιμές στις οποίες αλλάζουμε ορισμένες από τις παραμέτρους του μοντέλου.

Δοκιμή 1.

Παράμετροι: Γύροι=5, Epoch=1, LR=0.001, Steps=10, Μέθοδος βελτιστοποίησης: Adam

Αρχικά εκτελούμε ένα μοντέλο με τις αρχικές παραμέτρους. Η αξιολόγηση του μοντέλου με τα δεδομένα δοκιμής δίνει MSE = 0. 099296 (Εικόνα 7), ενώ ο χρόνος ήταν 147.997 sec.

Ωστόσο, ο χρόνος εκτέλεσης των 10 πελατών είναι συνολικά 139.638 sec, το οποίο σημαίνει ότι χρειαζόμαστε κατά μέσο όρο 13.9 sec ανά πελάτη

Στη συνέχεια αλλάζουμε τις τιμές των παραμέτρων και δίνουμε τα αντίστοιχα διαγράμματα για το MSE, ενώ από κάτω δίνουμε την τελική τιμή του καθώς και τον χρόνο εκτέλεσης. Τα αποτελέσματα δίνονται συγκεντρωτικά παρακάτω στον Πίνακα 1.

Εικόνα 7: Δοκιμή1

MSE: 0. 099296

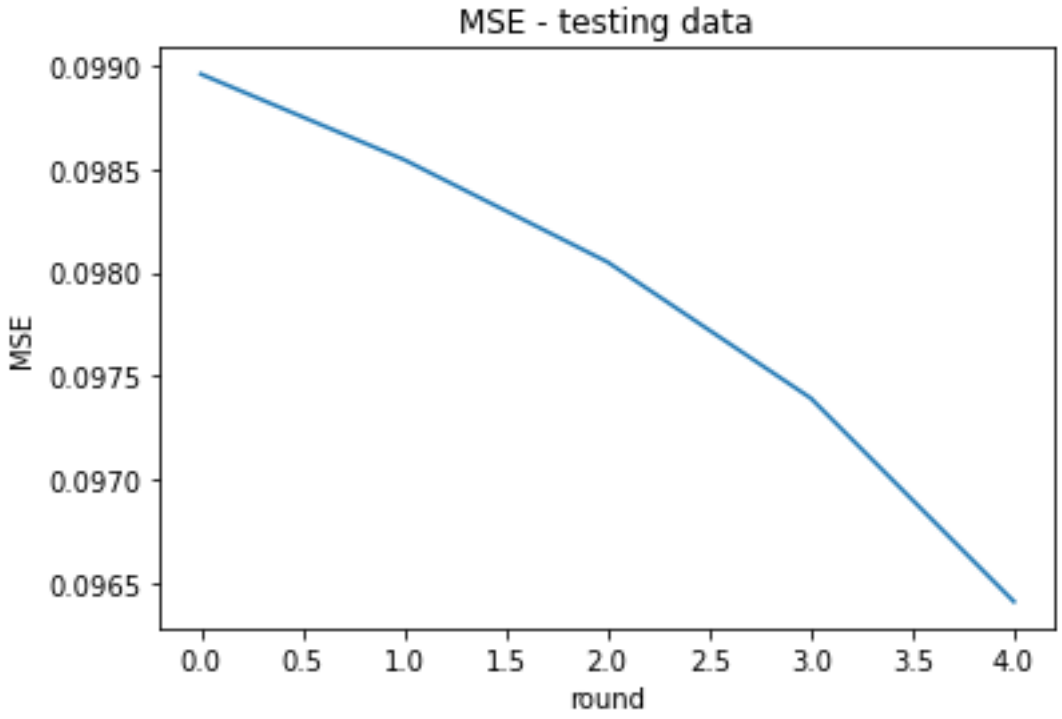
Time (sec): 147.997

Client Time (sec): 139.638

Δοκιμή 2.

Παράμετροι: Γύροι=5, Epoch=5, LR=0.001, Steps=10, Μέθοδος βελτιστοποίησης: Adam

Η αύξηση του αριθμού των epoch ανά γύρο επικοινωνίας βελτιώνει το τελικό MSE σε 0.096409 έναντι 0.099296 (Εικόνα 8), ενώ ο συνολικός χρόνος αυξάνει από 147.997 sec σε 167.368 sec.



Εικόνα 8: Δοκιμή 2

MSE: 0.096409

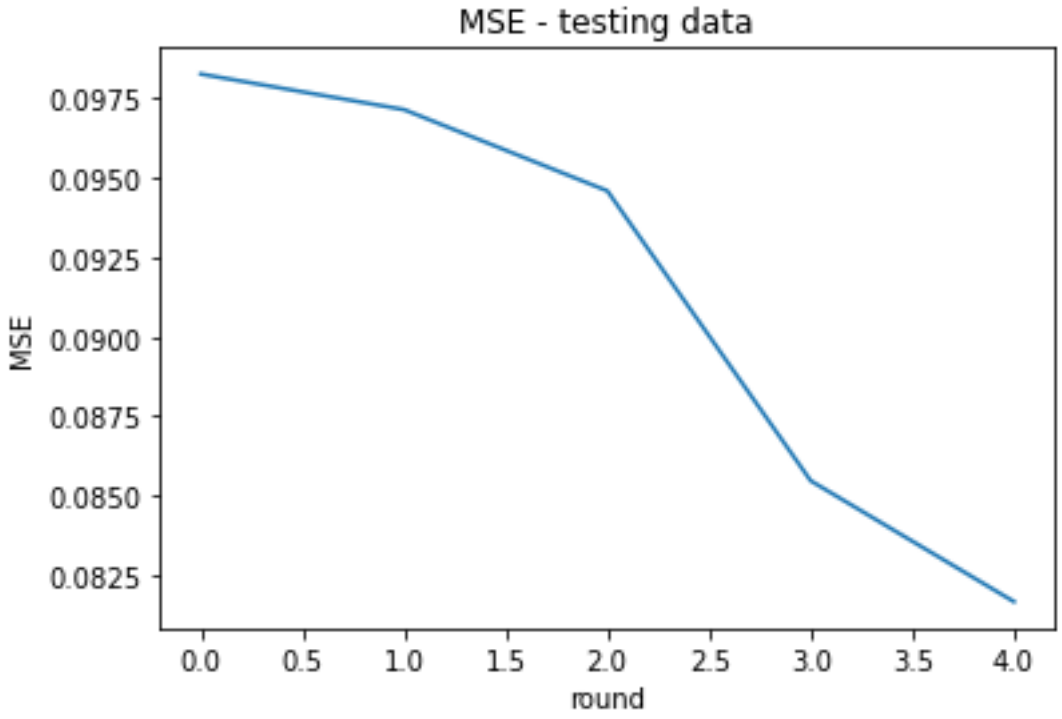
Time (sec): 167.368

Client Time (sec): 158.818

Δοκιμή 3.

Παράμετροι: Γύροι=5, Epoch=10, LR=0.001, Steps=10, Μέθοδος βελτιστοποίησης: Adam

Το MSE είναι 0.081667, άρα βελτιώνεται περαιτέρω σε σχέση με πριν (Εικόνα 9). Ωστόσο, αυξάνεται αντίστοιχα ο απαιτούμενος χρόνος, οπότε για να επιτύχουμε μια ισορροπία μεταξύ απόδοσης και χρόνου, στη συνέχεια θα σταθεροποιήσουμε τις Epoch σε 10.



Εικόνα 9: Δοκιμή 3

MSE: 0.081667

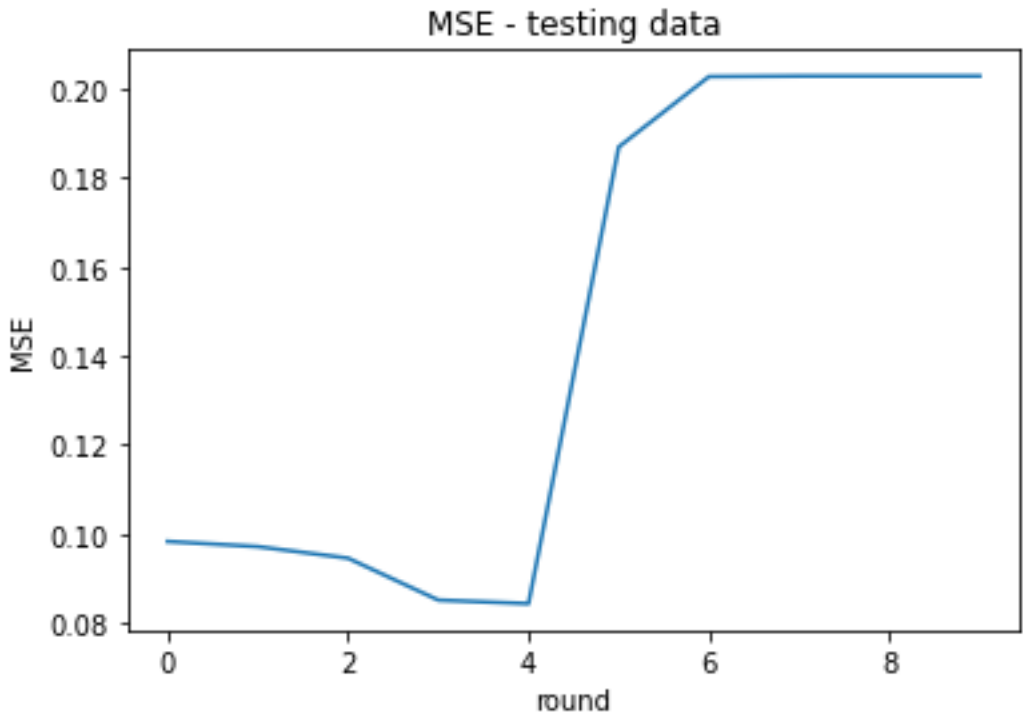
Time (sec): 180.171

Client Time (sec): 170.62

Δοκιμή 4.

Παράμετροι: Γύροι=10, Epoch=10, LR=0.001, Steps=10, Μέθοδος βελτιστοποίησης: Adam

Θα δοκιμάσουμε να αλλάξουμε τη μεταβλητή αριθμός γύρων επικοινωνίας, κρατώντας τον αριθμό των εποχών ίσο με 10. Αυξάνοντας τον αριθμό των γύρων επικοινωνίας σε 10, το MSE είναι 0.20283 και συγκεκριμένα από την Εικόνα 10 παρατηρούμε ότι από τον 4^{ο} γύρο αυξάνεται, δηλαδή οι προβλέψεις είναι χειρότερες, οπότε παρακάτω θα συνεχίσουμε με 5 γύρους. Αυτό πιθανόν να είναι αποτέλεσμα υπερεκπαίδευσης (over-training), δηλαδή το μοντέλο προσαρμόζεται πάρα πολύ στα δεδομένα εκπαίδευσης, οπότε δεν μπορεί να παράγει ικανοποιητικές προβλέψεις για διαφορετικά δεδομένα, όπως τα δεδομένα δοκιμής.



Εικόνα 10: Δοκιμή 4

MSE: 0.20283

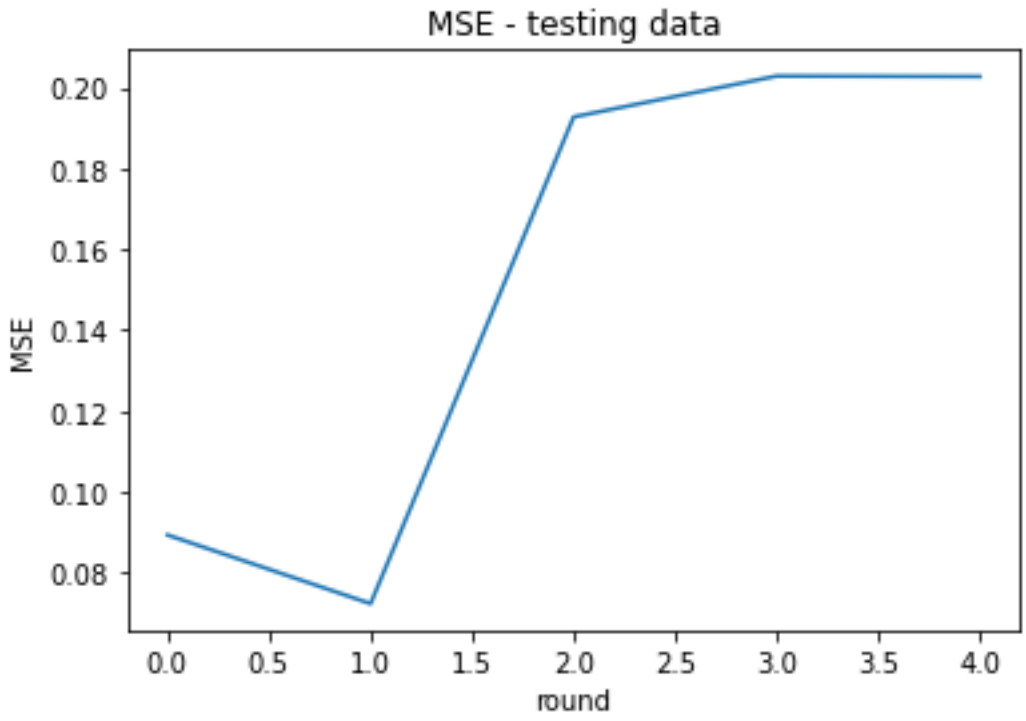
Time (sec): 333.824

Client Time (sec): 319.335

Δοκιμή 5.

Παράμετροι: Γύροι=5, Epoch=10, LR = 0.01, Steps = 10, Μέθοδος βελτιστοποίησης: Adam

Αλλάζοντας το ρυθμό μάθησης (αύξηση της τιμής) έχουμε MSE = 0.20269 (Εικόνα 11), δηλαδή πτώση της απόδοσης σε σχέση με την Δοκιμή 3 που οι υπόλοιπες παράμετροι είναι ίδιες. Οπότε στη συνέχεια ο ρυθμός μάθησης θα παραμείνει LR = 0.001. Αυτό συνέβη επειδή όταν ο ρυθμός μάθησης αυξηθεί, γίνονται μεγάλες μεταβολές στα βάρη, οπότε μπορεί ο αλγόριθμος να «υπερπηδήσει» τη βέλτιστη τιμή (ελάχιστο του MSE) και να μην συγκλίνει.



Εικόνα 11: Δοκιμή 5

MSE: 0.20269

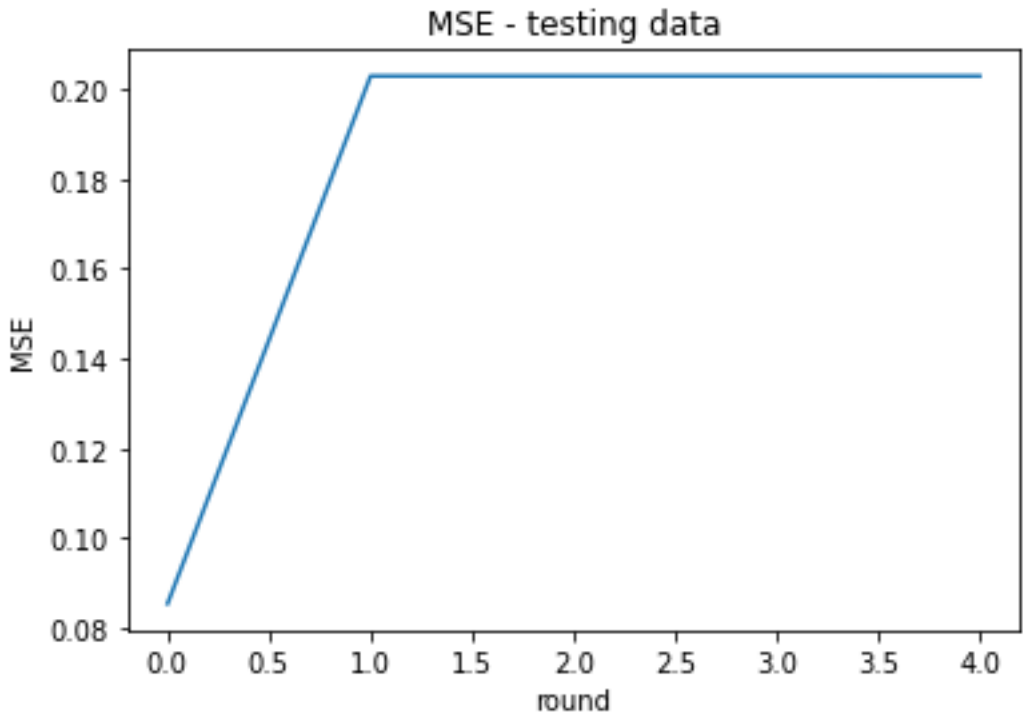
Time (sec): 158.688

Client Time (sec): 150.782

Δοκιμή 6.

Παράμετροι: Γύροι=5, Epoch=10, LR = 0.001, steps=100, Μέθοδος βελτιστοποίησης: Adam

Εδώ αυξάνουμε τον αριθμό βημάτων ανά εποχή από 10 σε 100. Αυτό οδηγεί σε MSE = 0.20283, το οποίο επίσης είναι χειροτέρευση της απόδοσης σε σχέση με πριν (Εικόνα 12), ενώ αυξάνεται και ο απαιτούμενος χρόνος. Πιθανότατα και εδώ έχουμε φαινόμενο υπερεκπαίδευσης.



Εικόνα 12: Δοκιμή 6

MSE: 0.20283

Time (sec): 367.097

Client Time (sec): 358.484

Δοκιμή 7.

Παράμετροι: Γύροι=5, Epoch=5, LR = 0.001, steps=10, Μέθοδος βελτιστοποίησης: SGD

Τέλος, δοκιμάζουμε να αλλάξουμε τη μέθοδο βελτιστοποίησης σε SGD [41]. Το αποτέλεσμα είναι MSE = 0.099368 (Εικόνα 13) το οποίο είναι χειρότερο από τη μέθοδο Adam. Επίσης, η μέθοδος SGD φαίνεται να μειώνει το MSE με πολύ αργό ρυθμό.

Εικόνα 13: Δοκιμή 7

MSE: 0.099368

Time (sec): 126.492

Client Time (sec): 119.479

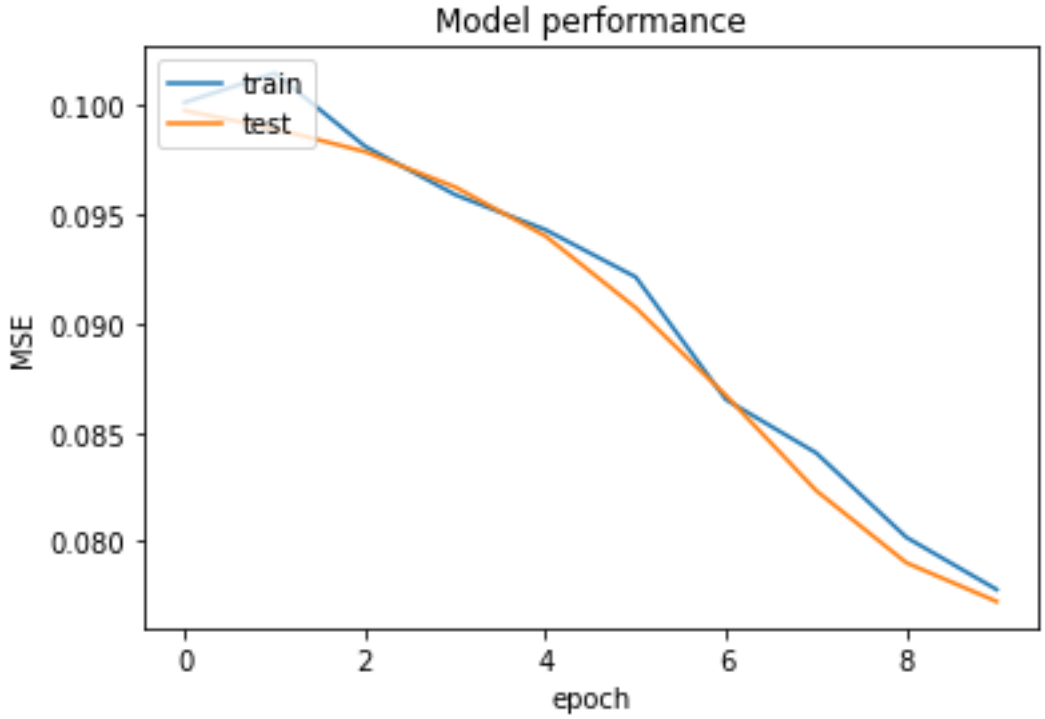
Όλες οι παραπάνω δοκιμές συνοψίζονται στον Πίνακα 3.

Πίνακας 3: Δοκιμές.

Δοκιμή	**Παράμετροι**	**MSE**	**Χρόνος (sec)**
1	Γύροι=5, Epoch=1, LR=0.001, Steps=10, Adam	0.099296	147.997
2	Γύροι=5, Epoch=5, LR=0.001, Steps=10, Adam	0.096409	167.368
3	Γύροι=5, Epoch=10, LR=0.001, Steps=10, Adam	0.081667	180.171
4	Γύροι=10, Epoch=10, LR=0.001, Steps=10, Adam	0.20283	333.824
5	Γύροι=5, Epoch=10 ,LR=0.01, Steps=10, Adam	0.20269	158.688
6	Γύροι=5, Epoch=10, LR=0.001, Steps=100, Adam	0.20283	367.097
7	Γύροι=5, Epoch=10, LR=0.001, Steps=10, SGD	0.099368	126.492

4.7.3 Σύγκριση federated learning και απλής προσέγγισης

Αρχικά, δοκιμάζουμε να εκπαιδεύσουμε ένα μοντέλο με όλα τα δεδομένα και τις ίδιες παραμέτρους με την αρχική δοκιμή. Ο αριθμός των epoch=10.



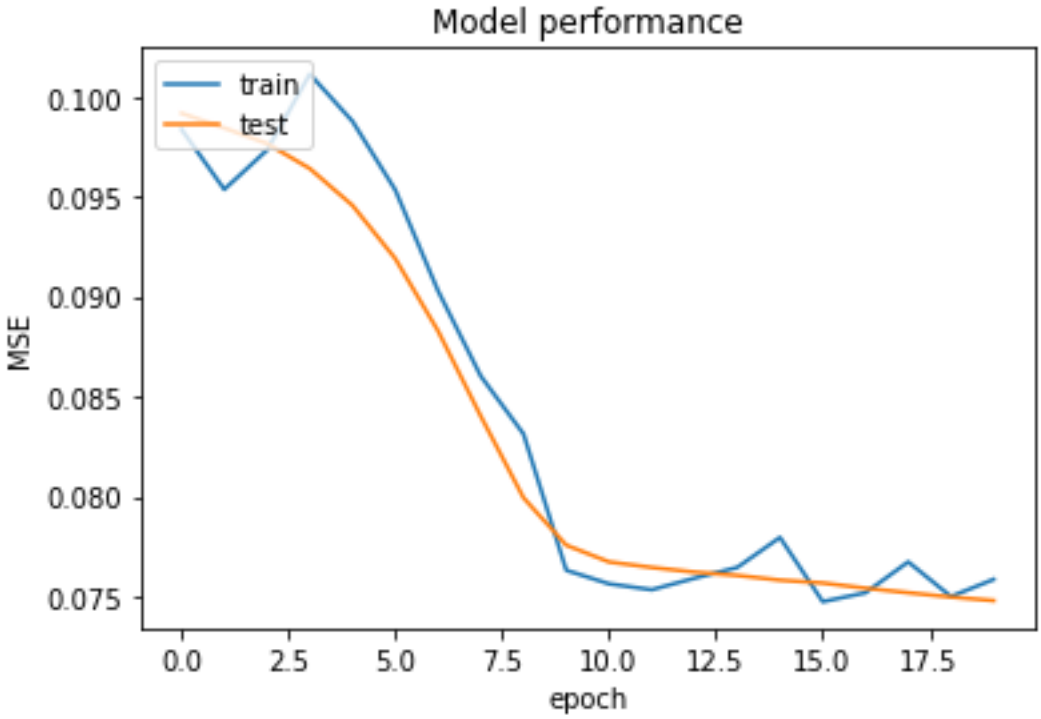
Εικόνα 14: Απόδοση απλής προσέγγισης, epoch=10

Time (sec): 9.518

MSE: 0.077

Η μετρική MSE έχει αρκετά καλύτερες τιμές σε σύγκριση με το federated model. Αυτό είναι λογικό, καθώς αυτό το μοντέλο έχει όλα τα δεδομένα διαθέσιμα, οπότε μπορεί να εκπαιδευτεί καλύτερα. Παρατηρούμε ότι το μοντέλο έχει περιθώρια βελτίωσης, καθώς οι καμπύλες δεν έχουν φτάσει σε σταθερό σημείο.

Στη συνέχεια, δοκιμάζουμε αριθμό epoch ίσο με 20. Όπως παρατηρούμε παρακάτω, το μοντέλο έχει φτάσει σε ένα σημείο όπου το μέσο τετραγωνικό σφάλμα δεν μπορεί να βελτιωθεί άλλο στην εκπαίδευση. Η αξιολόγηση του μοντέλου με τα δεδομένα δοκιμής δίνει MSE = 0.0746, ενώ ο χρόνος ήταν 14.630 sec. Αυτό είναι το βέλτιστο αποτέλεσμα και είναι 8.7% καλύτερο από την καλύτερη επίδοση του federated learning (0.081667). Ωστόσο, το federated learning είναι εφικτό και μπορεί να οδηγήσει σε ικανοποιητικά αποτελέσματα διατηρώντας την ιδωτικότητα των χρηστών.



Εικόνα 15: Απόδοση μοντέλου epoch=20

Time (sec): 14.6296

MSE: 0.074578

R2 : 0.042689

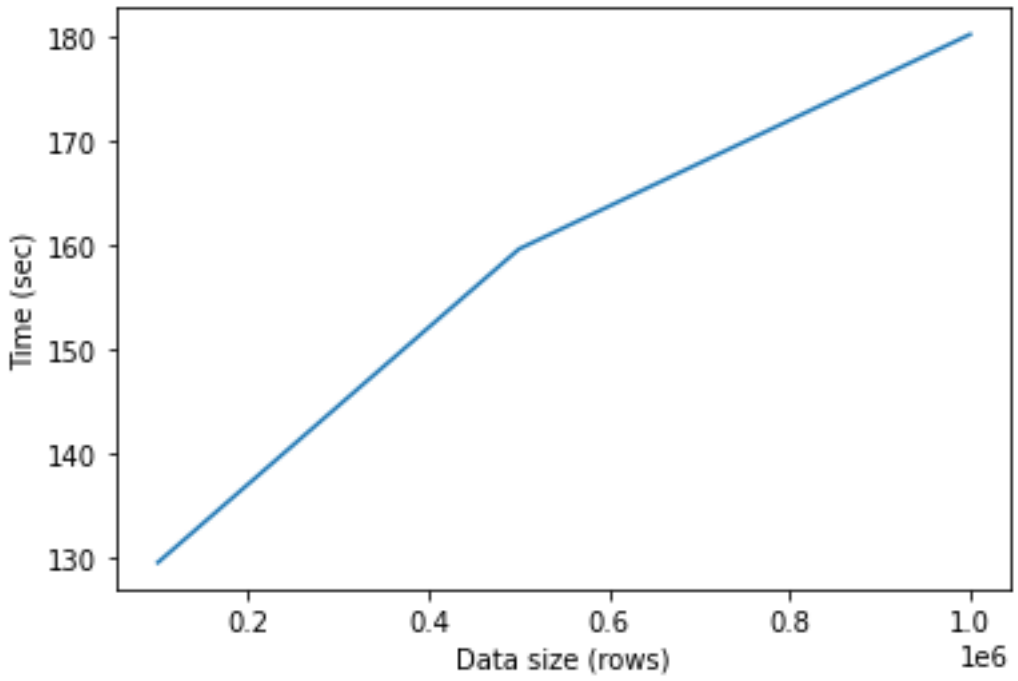
4.7.4 Χρονική πολυπλοκότητα

Για να εξετάσουμε τη χρονική πολυπλοκότητα του μοντέλου, δοκιμάζουμε την εκτέλεση στο 10% και στο 50% των δεδομένων. Οι παράμετροι είναι αυτές που οδήγησαν προηγουμένως στο καλύτερο αποτέλεσμα (Δοκιμή 3). Τα αποτελέσματα φαίνονται στον Πίνακα 4

Πίνακας 4: Χρονική Πολυπλοκότητα.

Μέγεθος	Χρόνος (sec)	Χρόνος πελατών (sec)	MSE
10%	129.445	122.02	0.075973
50%	159.546	151.181	0.077959
100%	180.171	170.62	0.081667

Όπως παρατηρούμε στην Εικόνα 16, η αύξηση στο χρόνο φαίνεται ότι είναι γραμμική ως προς το μέγεθος των δεδομένων.



Εικόνα 16: Διάγραμμα χρόνου σε σχέση με το μέγεθος των δεδομένων

Συμπέρασμα

Σε αυτή την εργασία αρχικά, ασχοληθήκαμε με την μελέτη των συνεργατικών προσεγγίσεων μάθησης (Federated Learning). Δόθηκε ο ορισμός τους, οι κατηγοριοποιήσεις τους και η αρχιτεκτονική τους. Στην συνέχεια, δόθηκε ιδιαίτερη έμφαση στα συστήματα συστάσεων όπου παρουσιάστηκε ο ορισμός τους, τα αίτια χρήσης τους και οι τύποι των συστημάτων συστάσεων. Έπειτα, μελετήθηκαν ορισμένες προσεγγίσεις federated learning σε συστήματα συστάσεων και υλοποιήθηκε μια από αυτές. Πιο συγκεκριμένα, δημιουργήσαμε και αξιολογήσαμε ένα μοντέλο federated learning. Η δοκιμή έγινε σε ένα μεγάλο σύνολο δεδομένων, το οποίο είχε περίπου 1000000 δείγματα από αξιολογήσεις ταινιών. Στο μοντέλο που δημιουργήθηκε, τροποποιήσαμε μια ευρεία γκάμα παραμέτρων και αξιολογήσαμε τα αποτελέσματα τόσο ως προς την απόδοση σε δεδομένα δοκιμής με τη μετρική MSE όσο και ως προς το συνολικό χρόνο εκτέλεσης. Βρήκαμε ότι οι περισσότερες παράμετροι οδηγούν σε μικρή βελτίωση της μετρικής απόδοσης, με εξαίρεση τον αριθμό βημάτων ανά epoch, το οποίο μπορεί να βελτιώσει αρκετά την απόδοση. Επίσης, τα αποτελέσματα ενός αντίστοιχου μοντέλου που εκτελείται κεντρικά πάνω σε όλα τα δεδομένα, έχει καλύτερη απόδοση, αλλά μόνο κατά 8.7%. Τέλος, ως προς την χρονική πολυπλοκότητα, φαίνεται να είναι γραμμική με το μέγεθος των δεδομένων (αριθμός δειγμάτων), το οποίο δείχνει ότι το μοντέλο μπορεί να εφαρμοστεί σε μεγάλα σύνολα δεδομένων χωρίς πρόβλημα.

Βιβλιογραφία

[1] «openmined,» 19 Μάρτιος 2020. [Ηλεκτρονικό]. Available: https://blog.openmined.org/federated-learning-recommendations-part1/. [Πρόσβαση 27 Σεπτέμβριος 2020].

[2] W. A. Group, «Federated Learning White Paper V1.0,» 2018. [Ηλεκτρονικό]. Available: https://www.fedai.org/static/flwp-en.pdf. [Πρόσβαση 5 Απρίλιος 2020].

[3] «ΣΕΒ,» [Ηλεκτρονικό]. Available: https://www.sev.org.gr/Uploads/Documents/51628/meleti_sev_GDPR_final.pdf. [Πρόσβαση 5 Απρίλιος 2020].

[4] Ε. ΖΑΧΟΣ, Α. ΠΑΓΟΥΡΤΖΗΣ και Π. ΓΡΟΝΤΑΣ, «kallipos.gr,» 2015. [Ηλεκτρονικό]. Available: https://repository.kallipos.gr/bitstream/11419/5439/1/main-ΚΟΥ.pdf. [Πρόσβαση 10 Απρίλιος 2020].

[5] L. SWEENEY, «k-anonymity: a model for protecting privacy,» *International Journal on Uncertainty,* τόμ. 10, αρ. 5, pp. 557-570, 2002 Μαίος 2002.

[6] C. J. C. H. e. a. Ho Q, «More effective distributed ml via a stale synchronous parallel parameter server,» *Advances in neural information processing systems,* pp. 1223-1231, 2013.

[7] L. J. A. Sheth A P, «Federated database systems for managing distributed, heterogeneous, and autonomous databases,» *ACM Computing Surveys (CSUR),* τόμ. 22, αρ. 3, pp. 183-236, 1990.

[8] J. Leskovec, A. Rajaraman και J. D. Ullman, «Mining of Massive Datasets,» [Ηλεκτρονικό]. Available: http://infolab.stanford.edu/~ullman/mmds/book.pdf. [Πρόσβαση 6 Απρίλιος 2020].

[9] F. Isinkaye, Y. Folajimi και B. Ojokoh , «Recommendation systems:Principles, methods and evaluation,» *Egyptian Informatics Journal,* αρ. 16, pp. 261-273, Αύγουστος 2015.

[10] . G. Adomavicius και A. Tuzhilin, «Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions,» *IEEE Transactions on Knowledge and Data Engineering,* τόμ. 17, αρ. 6, p. 734–749, Ιούνιος 2005.

[11] . F. Ricci, L. Rokach και B. Shapira, Recommender System Handbook, Springer, 2011.

[12] F. v. Reischach, F. Michahelles και A. Schmidt, «The Design Space of Ubiquitous Product Recommendation,» *MUM '09: Proceedings of the 8th International Conference on Mobile and Ubiquitous Multimedia,* pp. 1-10, Νοέμβριος 2009.

[13] H. M, «Recommender systems for e-shops,» *Vrije Universitei,* 2011 .

[14] A. C.C., «Recommender Systems: The Textbook,» [Ηλεκτρονικό]. Available: http://pzs.dstu.dp.ua/DataMining/recom/bibl/1aggarwal_c_c_recommender_systems_the_textbook.pdf. [Πρόσβαση 30 Απρίλιος 2020].

[15] D. F. J. H. S. S. J. Ben Schafer, «Collaborative Filtering Recommender Systems,» σε *The Adaptive Web*, In: P. Brusilovsky, A. Kobsa, and W. Nejdl (Eds.) επιμ., LNCS 4321, Springer-Verlag Berlin Heidelberg, 2007, p. 291 – 324.

[16] A. C. R., «Recommender: An analysis of collaborative filtering techniques,» 2014. [Ηλεκτρονικό]. Available: http://cs229.stanford.edu/proj2014/Christopher%20Aberger,%20Recommender.pdf. [Πρόσβαση 30 Απρίλιος 2020].

[17] G. K. J. A. K. J. T. R. B. M. Sarwar, «Item-Based Collaborative Filtering Recommendation,» Proceedings of the 10th international conference on World Wide Web,, 2001, pp. 285-295.

[18] E. Rich, «User modeling via stereotypes,» *Cognitive Science,* αρ. 3, p. 329–354, Οκτώβριος 1979.

[19] B. Krulwich, «Lifestyle finder: Intelligent user profiling using large-scale demographic,» *AI Magazine,* τόμ. 18, αρ. 2, p. 37–45, 1995.

[20] M. Pazzani, «A framework for collaborative, content-based and demographic filtering,» *Artificial Intelligence Review,* αρ. 13, pp. 5-6, Δεκέμβρος 1999.

[21] Z. S. P. Y. C. Aggarwal, «Online generation of profile association rules,» *KDD'98: Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining,* p. 129–133, Αύγουστος 1998.

[22] Z. S. P. Y. C. Aggarwal, «Online algorithms for finding profile association rules,» *CIKM '98: Proceedings of the seventh international conference on Information and knowledge management,* p. 86–95, Νοέμβριος 1998.

[23] R. Burke, «Hybrid Systems for Personalized Recommendations,» *Intelligent Techniques for Web Personalization,* 11 Αυγούστου 2003.

[24] J. Canny, «Collaborative Filtering with Privacy via Factor Analysis,» *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieva,* pp. 238-245, Αύγουστος 2002.

[25] S. I. U. W. M. J. N. T. Valeria Nikolaenko, «Privacy-preserving matrix factorization,» *CCS '13: Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security,* pp. 801-812, Νοέμβριος 2013.

[26] Y. W. ,. A. S. Ziqi Liu, «Fast Differentially Private Matrix Factorization,» *RecSys '15: Proceedings of the 9th ACM Conference on Recommender Systems,* pp. 171-178, Σεπτέμβριος 2015.

[27] Z. D. Z. L. X. H. Fei Chen, «Federated Meta-Learning for Recommendation,» 22 Φεβροαυάριος 2018.

[28] M. L. Z. D. Z. L. X. H. Fei Chen, «Federated Meta-Learning with Fast Convergence and Efficient Communication,» 14 Δεκέμβριος 2019.

[29] M. S. C. C. M. S. Amir Jalalirad, «A Simple and Efficient Federated Recommender System,» *BDCAT '19: Proceedings of the 6th IEEE/ACM International Conference on Big Data Computing, Applications and Technologies,* p. 53–58, 2-5 Δεκέμβριος 2019.

[30] E. I. S. A. K. W. O. Q. F. K. E. T. A. F. Muhammad Ammad-ud-din, «Federated Collaborative Filtering For Privacy-Preserving Personalized Recommendation System,» 29 Ιανουάριος 2019.

[31] G. D. M. J. István Hegedűs, «Decentralized Recommendation based on Matrix Factorization: A Comparison of Gossip and Federated Learning⋆,» σε *Machine Learning and Knowledge Discovery in Databases*, Springer, 2020, pp. 317-332.

[32] E. M. D. R. S. H. B. A. y. A. H. Brendan McMahan, «Communication-Efficient Learning of Deep Networks,» *International Conference on Artificial Intelligence and Statistics (AISTATS) ,* 28 Φεβρουάριος 2017.

[33] «Python,» [Ηλεκτρονικό]. Available: https://www.python.org/. [Πρόσβαση 12 Ιούλιος 2020].

[34] A. Géron, «Introduction to Artificial Neural Networks with Keras.,» σε *Hands-On Machine Learning with Scikit-Learn, Keras and Tensorflow Concepts, Tools and Techniques to Build Intelligent Systems*, O'Reilly, 2019, pp. 277-324.

[35] «NumPy,» [Ηλεκτρονικό]. Available: https://numpy.org/doc/stable/user/whatisnumpy.html. [Πρόσβαση 19 Ιούλιος 2020].

[36] «pandas.DataFrame,» [Ηλεκτρονικό]. Available: https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.html. [Πρόσβαση 19 Ιούλιος 2020].

[37] «random,» [Ηλεκτρονικό]. Available: https://docs.python.org/3/library/random.html. [Πρόσβαση 19 Ιούλιος 2020].

[38] F. M. &. K. J. A. Harper, «The movielens datasets: History and context,» *ACM Transactions on Interactive Intelligent Systems,* Δεκέμβριος 2015.

[39] D. P. K. a. J. Ba., «Adam: A method for stochastic optimization,» Δεκέμβριος 2014.

[40] «Mean squared error,» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Mean_squared_error. [Πρόσβαση 17 Σεπτέμβριος 2020].

[41] S. g. descent, «Stochastic gradient descent,» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Stochastic_gradient_descent. [Πρόσβαση 20 Σεπτέμβριος 2020].

Παράρτημα

Παρακάτω παρατίθενται ο κώδικας.

Imports

```
import pandas as pd
import numpy as np
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers
import matplotlib.pyplot as plt
import random
from sklearn.metrics import mean_squared_error
from tensorflow.keras import backend as K
import time
```

Προεπεξεργασία

```
df = pd.read_csv('ml-1m/ratings.dat', delimiter=";", header=None)
df.columns = ['userId','movieId','rating','timestamp']
userId_movieId_list=df[["userId", "movieId"]].values.tolist()
rating_list=df["rating"].values.tolist()
userIds_list_unique = df["userId"].unique().tolist()
user_to_user_enc= {x: i for i, x in enumerate(userIds_list_unique)}
user_enc_to_user = {i: x for i, x in enumerate(userIds_list_unique)}
movie_Ids_list_unique = df["movieId"].unique().tolist()
movie_to_movie_enc = {x: i for i, x in enumerate(movie_Ids_list_unique)}
movie_enc_to_movie = {i: x for i, x in enumerate(movie_Ids_list_unique)}
df["user"] = df["userId"].map(user_to_user_enc)
df["movie"] = df["movieId"].map(movie_to_movie_enc)
mumber_of_users = len(user_to_user_enc)
mumber_of_movies = len(movie_enc_to_movie)
df["rating"] = df["rating"].values.astype(np.float32)
# min and max ratings will be used to normalize the ratings later
minimum_rating = min(df["rating"])
maximum_rating = max(df["rating"])
print(
    "The number of users is: {}, the number of movies is: {}, the minimum rating is: {}, the maximium rating is: {}".format(
        mumber_of_users, mumber_of_movies, minimum_rating, maximum_rating
    )
)
```

Διαχωρισμός σε training και test data

```
df = df.sample(frac=1, random_state=42) #100% of the data
#df = df.sample(frac=0.1, random_state=42) #10% of data
#df = df.sample(frac=0.5, random_state=42) #50% of data


x = df[["user", "movie"]].values.tolist()
# Normalize the targets between 0 and 1. Makes it easy to train.
y = df["rating"].apply(lambda x: (x - minimum_rating) / (maximum_rating - minimum_rating)).values
# Assuming training on 90% of the data and validating on 10%.
train_indices = int(0.9 * df.shape[0])
x_train, x_test, y_train, y_test = (
    x[:train_indices],
    x[train_indices:],
    y[:train_indices],
    y[train_indices:],
)

# using np.array instead of list
x = np.array(df[["user", "movie"]].values)
y = np.array(y)

x_train, x_test, y_train, y_test = (
    x[:train_indices,0:2],
    x[train_indices:,0:2],
    y[:train_indices],
    y[train_indices:],
)
```

Ορισμός μοντέλου και βοηθητικών συναρτήσεων για το federated learning

```
def clients_creation(userId_movieId_list, rating_list, number_of_clients=10, initial='clients'):


    #create a list of client names
    names_of_clients = ['{}_{}'.format(initial, i+1) for i in range(number_of_clients)]

    #randomize the data
    data = list(zip(userId_movieId_list, rating_list))
    random.shuffle(data)

    #fragment data and place at each client
    size = len(data)//number_of_clients
    fragments = [data[i:i + size] for i in range(0, size*number_of_clients, size)]

    #number of clients must equal number of fragments
    assert(len(fragments) == len(names_of_clients))

    return {names_of_clients[i] : fragments[i] for i in range(len(names_of_clients))}


def split_into_batches(data_fragment, bs=32):


    userId_movieId_list, rating_list = zip(*data_fragment)
    dataset = tf.data.Dataset.from_tensor_slices((list(userId_movieId_list), list(rating_list)))
    return dataset.shuffle(len(rating_list)).batch(bs)
```

```
class My_Recommendation_Model(keras.Model):
    def __init__(self, number_of_users, number_of_movies, embedding_size, **kwargs):
        super(My_Recommendation_Model, self).__init__(**kwargs)
        self.number_of_users = number_of_users
        self.number_of_movies = number_of_movies
        self.embedding_size = embedding_size
        self.user_embedding = layers.Embedding(
            number_of_users,
            embedding_size,
            embeddings_initializer="he_normal",
            embeddings_regularizer=keras.regularizers.l2(1e-6),
        )
        self.user_bias = layers.Embedding(number_of_users, 1)
        self.movie_embedding = layers.Embedding(
            number_of_movies,
            embedding_size,
            embeddings_initializer="he_normal",
            embeddings_regularizer=keras.regularizers.l2(1e-6),
        )
        self.movie_bias = layers.Embedding(number_of_movies, 1)
```

```
def call(self, inputs):
    user_vector = self.user_embedding(inputs[:, 0])
    user_bias = self.user_bias(inputs[:, 0])
    movie_vector = self.movie_embedding(inputs[:, 1])
    movie_bias = self.movie_bias(inputs[:, 1])
    dot_user_movie = tf.tensordot(user_vector, movie_vector, 2)
    # Add all the components (including bias)
    x = dot_user_movie + user_bias + movie_bias
    # The sigmoid activation forces the rating to between 0 and 1
    return tf.nn.sigmoid(x)
```

```
def weight_scalling_factor(clients_trn_data, client_name):
    names_of_clients = list(clients_trn_data.keys())
    #get the bs
    bs = list(clients_trn_data[client_name])[0][0].shape[0]
    #first calculate the total training data points across clinets
    global_count = sum([tf.data.experimental.cardinality(clients_trn_data[client_name]).numpy() for client_name in names_of_clients])*bs
    # get the total number of data points held by a client
    local_count = tf.data.experimental.cardinality(clients_trn_data[client_name]).numpy()*bs
    return local_count/global_count
```

```
def scale_model_weights(weight, scalar):
    '''function for scaling a models weights'''
    weight_final = []
    steps = len(weight)
    for i in range(steps):
        weight_final.append(scalar * weight[i])
    return weight_final
```

```
def sum_scaled_weights(scaled_weight_list):
    '''Return the sum of the listed scaled weights. The is equivalent to scaled avg of the weights''
    avg_grad = list()
    #get the average grad accross all client gradients
    for grad_list_tuple in zip(*scaled_weight_list):
        layer_mean = tf.math.reduce_sum(grad_list_tuple, axis=0)
        avg_grad.append(layer_mean)
    return avg_grad
```

```
def test_model(x_test, y_test,  model):

    y_pred = model.predict(x_test, batch_size=1000)

    y_test2 = np.array(y_test).reshape(len(y_test),1) #from (n,) to (n,1)

    mse = mean_squared_error(y_test2, y_pred)


    return mse
```

Federated learning

```
#create clients
clients = clients_creation(x_train, y_train, number_of_clients=10, initial='client')

#process and batch the training data for each client
clients_batched = dict()
for (client_name, userId_movieId_list) in clients.items():
    clients_batched[client_name] = split_into_batches(userId_movieId_list)


EMBEDDING_SIZE = 50

comms_round = 5
EPOCHS = 1
STEPS = 10
LR = 0.001

loss='mse'
metrics = ['mse']


global_model = My_Recommendation_Model(number_of_users, number_of_movies, EMBEDDING_SIZE)
global_model.compile(loss=tf.keras.losses.MeanSquaredError(), optimizer=keras.optimizers.Adam(lr=LR))

mse = np.zeros(comms_round)



dt_client = 0
t0 = time.time()
for comm_round in range(comms_round):
    print(comm_round)
    # get the global model's weights - will serve as the initial weights for all local models
    global_weights = global_model.get_weights()


    #initial list to collect local model weights after scalling
    scaled_local_weight_list = list()

    #randomize client data - using keys
    names_of_clients= list(clients_batched.keys())
    random.shuffle(names_of_clients)
```

```
if comm_round == 0:
    #to initialize size of wieghts
    global_model.fit(x=x_train[0:10], y=y_train[0:10], epochs=1, verbose=0, steps_per_epoch=1, validation_split=0)


#loop through each client and create new local model
tc0 = time.time()
for client in names_of_clients:
    print(client)

    local_model = My_Recommendation_Model(mumber_of_users, mumber_of_movies, EMBEDDING_SIZE)
    local_model.compile(loss=tf.keras.losses.MeanSquaredError() , optimizer=keras.optimizers.Adam(lr=LR))


 if comm_round > 0:
     local_model.fit(x=x_train[0:10], y=y_train[0:10], epochs=1, verbose=0, steps_per_epoch=1, validation_split=0)
 #set local model weight to the weight of the global model
 local_model.set_weights(global_weights)

 #fit local model with client's data
 #validation_split = 0 for avoiding errors
 #steps_per_epoch parameter required, set to 1
 tmp_hist = local_model.fit(clients_batched[client], epochs=EPOCHS, verbose=0, batch_size=1000, steps_per_epoch=STEPS, validation_split = 0)


 #scale the model weights and add to list
 scaling_factor = weight_scalling_factor(clients_batched, client) #eager execution error see start of file
 scaled_weights = scale_model_weights(local_model.get_weights(), scaling_factor)
 scaled_local_weight_list.append(scaled_weights)


        #clear session to free memory after each communication round
        K.clear_session()
   tc1 = time.time()
   dt_client = dt_client + tc1 - tc0


   #to get the average over all the local model, we simply take the sum of the scaled weights
   average_weights = sum_scaled_weights(scaled_local_weight_list)
   average_weights2 = average_weights
   for i in range(len(average_weights2)):
       average_weights2[i] = np.array(average_weights[i])


 #update global model
 global_model.set_weights(average_weights2)

 #test global model and print out metrics after each communications round
 mse[comm_round] = test_model(x_test, y_test, global_model)
 print('comm_round: {} | MSE: {:.5}  '.format(comm_round, mse[comm_round]))



tl = time.time()

dt = tl - t0
print("Time (sec): {:.6}".format(dt))
print("Client Time (sec): {:.6}".format(dt_client))
```

```
#plot final results
plt.plot(mse)
plt.title("MSE - testing data")
plt.ylabel("MSE")
plt.xlabel("round")
plt.show()
```

Centralized model

```
EMBEDDING_SIZE = 50
global_model = My_Recommendation_Model(mumber_of_users, mumber_of_movies, EMBEDDING_SIZE)
global_model.compile(loss=tf.keras.losses.MeanSquaredError(), optimizer=keras.optimizers.Adam(lr=0.001)) #.SGD(lr=0.001))
```

```
t0 = time.time()
EPOCHS = 20
history = global_model.fit(
    x=x_train,
    y=y_train,
    batch_size=1000,
    epochs=EPOCHS,
    steps_per_epoch=10,
    verbose=0,
    validation_data = (x_test, y_test)
)
t1 = time.time()
```

```
dt_single = t1 - t0
print("Time (sec): {:.6}".format(dt_single))
```

```
mse = test_model(x_test, y_test, global_model)
print('MSE: {:.5} '.format(mse))
```

```
plt.plot(history.history["loss"])
plt.plot(history.history["val_loss"])
plt.title("Model performance")
plt.ylabel("MSE")
plt.xlabel("epoch")
plt.legend(["train", "test"], loc="upper left")
plt.show()
```