ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Χρονικές συστάσεις με βραχυπρόθεσμες και μακροπρόθεσμες προτιμήσεις

Σοφία Ι. Κυπραίου

Επιβλέποντες: **Ιωάννης Ιωαννίδης,** Καθηγητής ΕΚΠΑ
Μαριαλένα Κυριακίδη, Υποψήφια Διδάκτορας ΕΚΠΑ

ΑΘΗΝΑ

ΝΟΕΜΒΡΙΟΣ 2016

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Χρονικές συστάσεις με βραχυπρόθεσμες και μακροπρόθεσμες προτιμήσεις

Σοφία Ι. Κυπραίου
Α.Μ.: 1115201100047

ΕΠΙΒΛΕΠΟΝΤΕΣ: **Ιωάννης Ιωαννίδης,** Καθηγητής ΕΚΠΑ
Μαριαλένα Κυριακίδη, Υποψήφια Διδάκτορας ΕΚΠΑ

ΠΕΡΙΛΗΨΗ

Ο χρόνος είναι ένας σημαντικός παράγοντας όταν γίνονται προτάσεις / συστάσεις και η αποτύπωση με ακρίβεια των προτιμήσεων των χρηστών στην πάροδο του χρόνου είναι μια μεγάλη πρακτική πρόκληση για τα συστήματα συστάσεων (recommender systems). Οι αλγόριθμοι συνεργατικού φιλτραρίσματος (Collaborative Filtering algorithms), που χρησιμοποιούνται σε συστήματα προτάσεων στο διαδίκτυο, συχνά αξιολογούνται όσον αφορά την ακρίβεια των προβλέψεων για την βαθμολογία του χρήστη σε δεδομένη στιγμή και πολλές από τις σημερινές τεχνικές αξιολόγησης αγνοούν το γεγονός ότι οι χρήστες συνεχίζουν να αξιολογούν τα αντικείμενα με την πάροδο του χρόνου, και να αλλάζουν τις προτιμήσεις τους λόγω διαφορετικών εξωτερικών γεγονότων. Η συμπεριφορά των χρηστών μπορεί συχνά να προσδιοριστεί από τις μακροπρόθεσμες και βραχυπρόθεσμες προτιμήσεις.

Για την αντιμετώπιση αυτών των προκλήσεων, η πρώτη μέθοδος που ακολουθήσαμε ήταν χρονικός γράφος με βάση χρονικές περιόδους (Session-based Temporal Graph - STG), που μοντελοποιεί ταυτόχρονα τις μακροπρόθεσμες και βραχυπρόθεσμες προτιμήσεις των χρηστών με την πάροδο του χρόνου. Με βάση αυτόν τον γράφο, χρησιμοποιήσαμε τον αλγόριθμο για προτάσεις / συστάσεις Injected Preference Fusion (IPF).

Για το δεύτερο μέρος, ακολουθήσαμε μια διαφορετική προσέγγιση με το collaborative Filtering, χρησιμοποιώντας Ανάλυση Κύριων Συνιστωσών (Principal Component Analysis - PCA) και ιεραρχική ομαδοποίηση (hierarchical clustering), για ομαδοποίηση παρόμοιων χρηστών και την δημιουργία συστάσεων με βάση τον χρήστη.

Τέλος, αξιολογούμε την αποτελεσματικότητα των μεθόδων χρησιμοποιώντας την βάση δεδομένων του Yelp. Βάση των κριτικών (reviews) φτιάχνονται συστάσεις (recommendations) και αποδεικνύεται ότι η μέθοδος STG εμφανίζει πιο ακριβή αποτελέσματα από την PCA.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Συστήματα προτάσεων/συστάσεων

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: χρονικό σύστημα προτάσεων, συνεργατικό φιλτράρισμα, γράφος, ομαδοποίηση, top-n λίστα

ABSTRACT

Time is an important factor when making recommendations and accurately capturing user preferences over time is a great practical challenge in recommender systems. Collaborative Filtering (CF) algorithms, used to build web-based recommender systems, are often evaluated in terms of how accurately they predict user ratings and many current evaluation techniques disregard the fact that users continue to rate items over time and change their preferences due to different external events. User behavior can often be determined by individual's long-term and short-term preferences.

To address these challenges, the first method we used was the session-based Temporal Graph (STG) which simultaneously models users' long-term and short-term preferences over time. Based on the STG model framework, we used the recommendation algorithm Injected Preference Fusion (IPF).

For the second part we tried a different approach with the collaborative filtering, by using Principal Component Analysis (PCA) and hierarchical clustering, to group similar users and making user-based recommendations.

Finally, we evaluate the effectiveness of the methods using Yelp dataset. Based on business reviews and making recommendations, we prove that the STG method presents more accurate results than the PCA method.

SUBJECT AREA: Recommendation/ recommender systems (RSs)

KEYWORDS: temporal recommendation system, collaborative filtering, graph, clustering, top-n list

Στην Εύα και την Ιωάννα

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω τον επιβλέποντα κ. Ιωάννη Ιωαννίδη για τη συνεργασία και τη βοήθεια κατά την εκπόνηση αυτής της πτυχιακής.

Θα ήθελα επίσης να ευχαριστήσω την υποψήφια διδάκτορα Μαριαλένα Κυριακίδη για τη συνεργασία, την πολύτιμη συμβολή της και τη βοήθεια κατά την εκπόνηση αυτής της πτυχιακής.

ΠΕΡΙΕΧΟΜΕΝΑ

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

ΠΡΟΛΟΓΟΣ

Η παρούσα πτυχιακή εκπονήθηκε στην Αθήνα, το 2016, υπό την επίβλεψη του καθηγητή του Τμήματος Πληροφορικής και Τηλεπικοινωνιών του ΕΚΠΑ, κ. Ιωάννη Ιωαννίδη, και με την συνεργασία και βοήθεια της υποψήφιας διδάκτορος Μαριαλένας Κυριακίδη.

1. ΕΙΣΑΓΩΓΗ

1.1 Συστήματα Συστάσεων

Τα τελευταία χρόνια, ο αριθμός των διαδικτυακών εφαρμογών και καταστημάτων συνεχώς αυξάνεται, δίνοντας τη δυνατότητα στους χρήστες του Διαδικτύου να πλοηγηθούν και να επιλέξουν υπηρεσίες και προϊόντα από μια πληθώρα εναλλακτικών επιλογών.

Γι' αυτόν τον λόγο, τα συστήματα συστάσεων (recommender systems) έχουν γίνει απαραίτητο εργαλείο σε ένα πλήθος εφαρμογών για να διευκολύνουν την πλοήγηση των χρηστών. Τα συστήματα συστάσεων συνήθως παράγουν λίστες με προτάσεις και προσπαθούν να προβλέψουν την βαθμολογία ή την προτίμηση του χρήστη σε ένα αντικείμενο.

Μια σημαντική ώθηση στην έρευνα για τα συστήματα συστάσεων δόθηκε όταν το Netflix, μία διαδικτυακή υπηρεσία ενοικίασης DVD και βίντεο streaming, πρόσφερε το βραβείο των $1,000,000 στο πρώτο άτομο ή ομάδα που θα πρότεινε έναν αλγόριθμό συστάσεων που θα βελτίωνε αυτόν που ήδη χρησιμοποιούσαν κατά 10%. Μετά από τρία χρόνια δουλειάς, τον Σεπτέμβρη του 2009, το βραβείο δόθηκε στην ομάδα "BellKor's Pragmatic Chaos".

Πολλά από τα συστήματα χρησιμοποιούν αλγορίθμους συνεργατικού φιλτραρίσματος (collaborative filtering). Οι αλγόριθμοι του collaborative filtering βασίζονται στην συλλογή και ανάλυση μεγάλης ποσότητας πληροφοριών σχετικά με την δραστηριότητα, τη συμπεριφορά και τις προτιμήσεις των χρηστών και κάνουν προβλέψεις για τις προτιμήσεις των χρηστών με βάση την ομοιότητά τους με άλλους χρήστες.

Οι δύο πιο συνηθισμένες τεχνικές του collaborative filtering (CF) είναι το φιλτράρισμα αντικείμενο προς αντικείμενο (item-item filtering) και το φιλτράρισμα χρήστη προς χρήστη (user-user filtering).Με τη μέθοδο του item-item filtering, ο αλγόριθμος παίρνει ένα συγκεκριμένο αντικείμενο, βρίσκει χρήστες στους οποίους άρεσε αυτό το αντικείμενο και στη συνέχεια βρίσκει άλλα αντικείμενα τα οποία άρεσαν σε αυτούς τους χρήστες (ή σε χρήστες παρόμοιους με αυτούς). Αντίστοιχα, το user-user filtering παίρνει έναν συγκεκριμένο χρήστη τη φορά, βρίσκει παρόμοιους χρήστες με αυτόν με βάση παρόμοιες αξιολογήσεις που έχουν κάνει σε αντικείμενα και προτείνει αντικείμενα που άρεσαν σε παρόμοιους με αυτούς χρήστες.

Ένας απλός τρόπος για να κατηγοριοποιηθούν οι διαφορετικοί τύποι φιλτραρίσματος είναι:

- Φιλτράρισμα με βάση το περιεχόμενο: "Αν σου άρεσε αυτό το αντικείμενο, μπορεί να σου αρέσει και …"
- Item-Item collaborative filtering: "Χρήστες που τους άρεσε αυτό το αντικείμενο, τους άρεσε επίσης και …"
- User-user collaborative filtering: "Χρήστες που είναι παρόμοιοι με σένα, τους άρεσε επίσης και …"

Η τελευταία μέθοδος είναι αυτή που επιτρέπει την πρόβλεψη για το πώς ένας χρήστης θα βαθμολογήσει ένα αντικείμενο που δεν έχει βαθμολογήσει ακόμα. Ένα πλεονέκτημα

είναι πως δεν χρειάζεται πληροφορία για το περιεχόμενο ούτε των αντικειμένων, ούτε των χρηστών. Κάνει εξατομικευμένες συστάσεις γιατί βασίζεται στην εμπειρία και την ομοιότητα παρόμοιων χρηστών. Αυτή είναι και η προσέγγιση που ακολουθούμε σε αυτή την πτυχιακή εργασία.

1.2 Χρονικά Συστήματα Συστάσεων

Ο χρόνος είναι ένας σημαντικός παράγοντας για την δημιουργία συστάσεων. Η μεταβλητότητα του χρόνου σε συστήματα συστάσεων έχει μεγάλη σημασία για πολλές πραγματικές εφαρμογές, όπως πολλά e-shops (Amazon), εφαρμογές για προτάσεις ταινιών (Netflix, IMDB), για προτάσεις βίντεο (Youtube) ή εστιατορίων και υπηρεσιών (TripAdvisor, Yelp).

Οι αλγόριθμοι φιλτραρίσματος που αναφέρθηκαν και παραπάνω, συχνά αξιολογούνται μόνο με βάση την ακρίβεια των προβλέψεών τους. Πολλές από τις σημερινές τεχνικές αξιολόγησης όμως, αγνοούν το γεγονός ότι οι χρήστες συνεχίζουν να αξιολογούν τα αντικείμενα με την πάροδο του χρόνου και αλλάζουν τις προτιμήσεις τους λόγω διαφορετικών εξωτερικών γεγονότων.

Το βασικό ερώτημα που προκύπτει είναι πώς μπορούν τα συστήματα συστάσεων να εντοπίσουν με ακρίβεια τις προτιμήσεις των χρηστών με την πάροδο του χρόνου.

Για να διαχειριστούμε τον χρόνο, τον διαχωρίζουμε σε μακροπρόθεσμο και βραχυπρόθεσμο. Η συνολική συμπεριφορά ενός χρήστη μπορεί να καθοριστεί από το μακροπρόθεσμο ενδιαφέρον του (ιστορικό). Αλλά σε κάθε δεδομένη χρονική στιγμή, ο χρήστης επηρεάζεται επίσης από βραχυπρόθεσμα ενδιαφέροντα εξαιτίας παροδικών γεγονότων, όπως είναι οι νέες κυκλοφορίες προϊόντων και οι μόδες, αλλά και των ειδικών προσωπικών περιπτώσεων, όπως πχ. τα γενέθλια.

1.3 Η εργασία

Σε αυτή την πτυχιακή εργασία, μελετήσαμε τις μεθόδους για μοντελοποιήση του χρόνου και την παραγωγή προτάσεων με δύο διαφορετικούς τρόπους.

Η πρώτη μέθοδος αφορά τη χρήση γράφου για την απεικόνιση χρηστών, αντικειμένων και τη χρονική σχέση μεταξύ τους, καθώς και τη χρήση αλγορίθμων για δημιουργία συστάσεων (κορυφαίων-N) μέσω αυτού. Ο γράφος αυτός είναι ένας Χρονικός Γράφος με βάση χρονικές περιόδους (session-based Temporal Graph - STG) [1]. Η μοναδικότητα αυτού του γράφου είναι πως ο χρόνος αντιπροσωπεύεται σαν χρονική περίοδος (session). Με αυτό τον τρόπο μοντελοποιεί ταυτόχρονα και τις μακροχρόνιες και τις βραχύχρονες προτιμήσεις του χρήστη.

Η δεύτερη μέθοδος αποτελεί μία εξερεύνηση για την βελτίωση των Collaborative Filtering (CF) αλγορίθμων. Συγκεκριμένα, χρησιμοποιούμε τις αξιολογήσεις χρηστών για τα αντικείμενα ώστε να εκπαιδεύσουμε τον αλγόριθμο να παράγει συστάσεις. Ομαδοποιούμε τους χρήστες με βάση τις αξιολογήσεις τους και μέσω των όμοιων χρηστών πραγματοποιούμε

τις συστάσεις. Ένα από τα ζήτημα που προκύπτει από αυτού του είδους τους αλγορίθμους είναι τα μεγάλων διαστάσεων δεδομένα, τα οποία δυσκολεύουν αυτή την εξαγωγή των κοινών ενδιαφερόντων των χρηστών, οδηγώντας σε χαμηλής ποιότητας συστάσεις. Για την αντιμετώπιση αυτού του προβλήματος, χρησιμοποιούμε την Ανάλυση Κύριων Συνιστωσών (Principal Component Analysis) [2], όπου μειώνει και δημιουργεί νέα χαρακτηριστικά για να περιγράψει τους χρήστες.

2. ΤΡΕΧΟΥΣΕΣ ΠΡΟΣΕΓΓΙΣΕΙΣ

Ένας τρόπος για να εισαχθεί ο χρόνος στα συστήματα συστάσεων, είναι να αντιμετωπιστεί ως περιεχόμενο.

2.1 Συστήματα συστάσεων που χρησιμοποιούν περιεχόμενο

Μια από τις μεθόδους υλοποίησης είναι μέσω συστημάτων συστάσεων που χρησιμοποιούν το περιεχόμενο (context-aware recommender systems). Τα Context-Aware Recommender Systems (CARSs) διαφέρουν από τα παραδοσιακά, γιατί εκμεταλλεύονται την γνώση συναφών καταστάσεων σύμφωνα με τις οποίες έγιναν οι βαθμολογίες και της κατάστασης του χρήστη που ζητά μία σύσταση.

Για παράδειγμα, χρησιμοποιώντας χρονικό περιεχόμενο, ένα ταξιδιωτικό σύστημα συστάσεων θα πρότεινε έναν προορισμό διακοπών τον χειμώνα, που θα ήταν πολύ διαφορετικός από έναν προορισμό το καλοκαίρι. Παρόμοια, στην περίπτωση ενός εξατομικευμένου περιεχομένου σε έναν ιστότοπο, είναι σημαντικό να καθοριστεί τι περιεχόμενο χρειάζεται να παρουσιαστεί (συσταθεί) σε έναν πελάτη και πότε. Συγκεκριμένα, από Δευτέρα έως Παρασκευή ο χρήστης μπορεί να προτιμάει να διαβάζει νέα απ' όλον τον κόσμο όταν συνδέεται το πρωί και ενημερώσεις για το χρηματιστήριο το απόγευμα, ενώ τα Σαββατοκύριακα να διαβάζει κριτικές για βιβλία και για νέα εστιατόρια.

Σε σύγκριση με τα παραδοσιακά συστήματα συστάσεων δύο διαστάσεων που ασχολούνται με οντότητες δύο τύπων, χρήστες (πχ πελάτες) και αντικείμενα (πχ προϊόντα, υπηρεσίες) και προσπαθούν να εκτιμήσουν άγνωστες αξιολογήσεις μέσα από έναν πίνακα Χρηστών $\times$ Αντικειμένων, τα context-aware recommender systems (CARS) λαμβάνουν υπ' όψιν και συναφείς πληροφορίες περιεχομένου.

Στην περίπτωση του Yelp, που χρησιμοποιήσαμε σαν βάση δεδομένων σε αυτή την πτυχιακή εργασία και σχετίζεται με συστάσεις επιχειρήσεων, έστω πως ο χρήστης Χριστίνα έχει βαθμολογήσει με 4 αστέρια (σε μία κλίμακα από 1 ως 5) το εστιατόριο "Ο Μεξικάνος" που σερβίρει μεξικάνικο φαγητό. Με τα κλασσικά recommender systems (δύο διαστάσεων), αυτό αναπαρίσταται σαν $R_{\textit{επιχείρηση}}(\textit{Χριστίνα}, \text{"Ο Μ}\textit{εξικάνος}\text{"}) = 4$.

Η διαδικασία της σύστασης ξεκινάει τυπικά με τον προσδιορισμό του αρχικού πίνακα από τις βαθμολογίες. Όταν αυτές προσδιοριστούν, το σύστημα συστάσεων προσπαθεί να αξιολογήσει τη συνάρτηση βαθμολόγησης R

$$R : \textit{Χρήστης} \times \textit{Αντικείμενο} \rightarrow \textit{Βαθμολογία} \tag{1}$$

για τα ζεύγη (Χρήστης, Αντικείμενο) που δεν έχουν βαθμολογηθεί ακόμα από τους χρήστες. Μόλις η συνάρτηση R προσδιοριστεί για όλο το σύνολο Χρήστη $\times$ Αντικείμενο, το σύστημα τις ταξινομεί και μπορεί να προτείνει τα κορυφαία N αντικείμενα για κάθε χρήστη. Τα συστήματα αυτά ονομάζονται παραδοσιακά ή δύο διαστάσεων (2Δ) αφού μελετούν

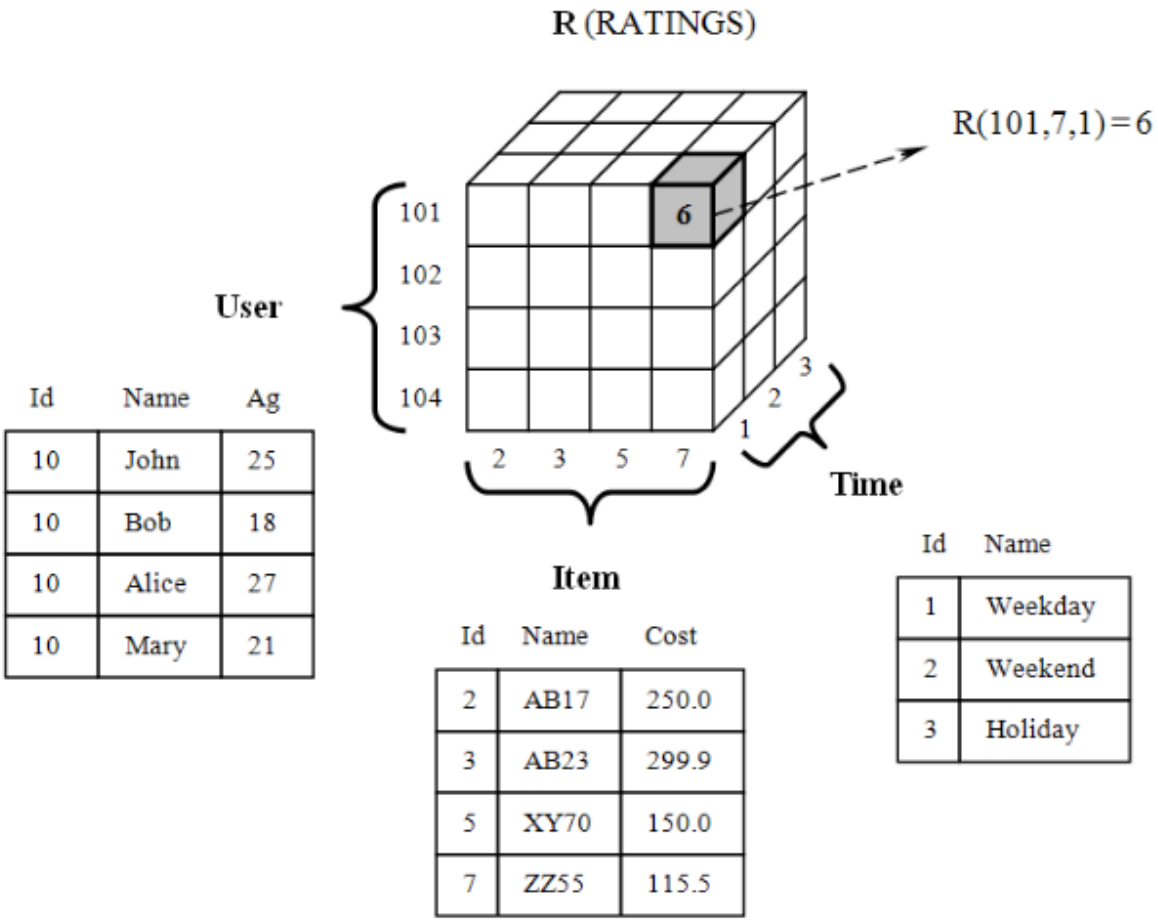
μόνο τον Χρήστη και το Αντικείμενο σαν διάσταση στην διαδικασία συστάσεων.

Όμως η Χριστίνα βαθμολόγησε την επιχείρηση επειδή της άρεσε η τοποθεσία της, ο τύπος της κουζίνας, είχε Wifi και την επισκέφτηκε το Σάββατο με την παρέα της και πέρασε καλά. Αυτές οι προτιμήσεις διαμορφώνουν την βαθμολογία και αλλάζουν την συνάρτηση της βαθμολογίας, προσθέτοντας το περιεχόμενο. Πλέον, η συνάρτηση βαθμολόγησης ορίζεται ως εξής:

$$R : \textit{Χρήστης} \times \textit{Αντικείμενο} \times \textit{Περιεχόμενο} \rightarrow \textit{Βαθμολογία} \quad (2)$$

όπου το Περιεχόμενο περιέχει την πληροφορία που σχετίζεται με την εκάστοτε εφαρμογή.

Για τις χρονικές συστάσεις, το Περιεχόμενο είναι ο χρόνος, και μοντελοποιείται ως εξής:



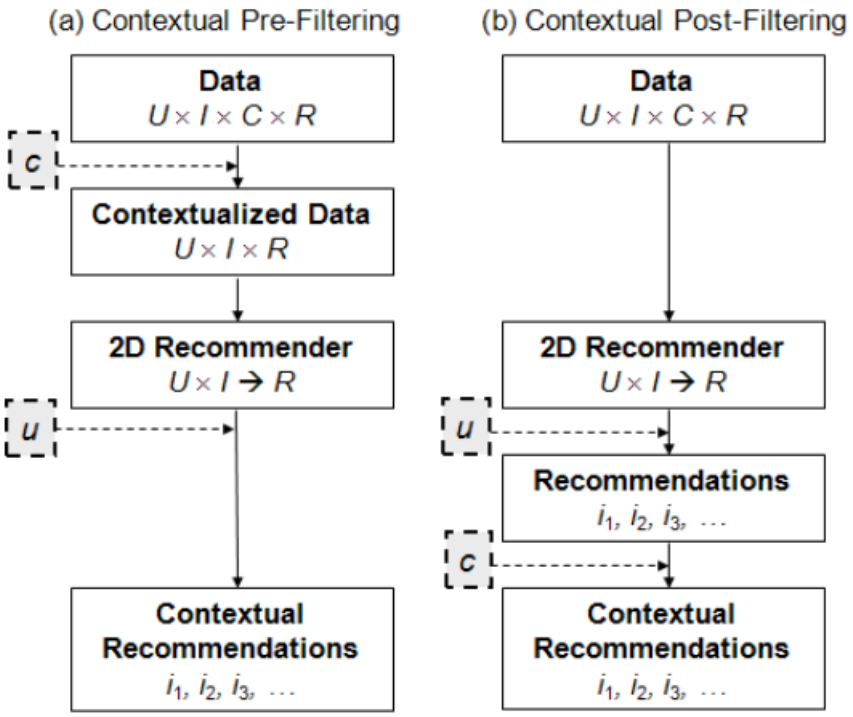
Σχήμα 1: Πολυδιάστατο μοντέλο για τον χώρο Χρήστης × Αντικείμενο × Χρόνος

Η διαδικασία συστάσεων για τα context-aware recommendation systems που βασίζεται στο περιεχόμενο, κατηγοριοποιείται ανάλογα με το στάδιο της διαδικασίας που χρησιμοποιείται το περιεχόμενο, όπως φαίνεται από το σχήμα 2.

Contextual pre-filtering Η πληροφορία για το περιεχόμενο χρησιμοποιείται για φιλτράρισμα άσχετων βαθμολογιών *πριν* χρησιμοποιηθούν για τον υπολογισμό συστάσεων

με τη χρήση κλασσικών (2Δ) μεθόδων. Στο παράδειγμα που φαίνεται και στο σχήμα 2α, η πληροφορία για ένα συγκεκριμένο περιεχόμενο C (όπως ο χρόνος), χρησιμοποιείται για να επιλεχθούν τα σχετικά σύνολα από δεδομένα (όπως οι βαθμολογίες). Μετά, οι βαθμολογίες μπορούν να προβλεφθούν χρησιμοποιώντας 2Δ συστήματα συστάσεων στα επιλεγμένα δεδομένα.

Contextual post-filtering Η πληροφορία για το περιεχόμενο χρησιμοποιείται *μετά* τη χρήση κλασσικών (2Δ) μεθόδων για φιλτράρισμα των δεδομένων που δεν περιέχουν περιεχόμενο. Στο παράδειγμα που φαίνεται και στο σχήμα 2β, η πληροφορία για το περιεχόμενο αρχικά αγνοείται, και οι βαθμολογίες προβλέπονται χρησιμοποιώντας παραδοσιακά 2Δ συστήματα συστάσεων στο σύνολο των δεδομένων. Μετά, το αποτέλεσμα στο σύνολο των συστάσεων προσαρμόζεται (φιλτράρεται με βάση το περιεχόμενο) για κάθε έναν από τους χρήστες, χρησιμοποιώντας την πληροφορία για το περιεχόμενο.



Σχήμα 2: (α) και (β) Παραδείγματα ενσωμάτωσης του περιεχομένου στο σύστημα συστάσεων

2.2 Άλλες προσεγγίσεις

2.2.1 Συστάσεις που βασίζονται σε γράφο

Οι μέθοδοι που βασίζονται σε γράφο εισάχθηκαν στις συστάσεις για να μοντελοποιήσουν την αλληλεπίδραση ανάμεσα σε χρήστες και αντικείμενα. Η ομοιότητα ανάμεσα στους κόμβους υπολογίζεται από μία καθολική προοπτική, αντί για τοπικό υπολογισμό ανάμεσα σε ζεύγη γειτόνων.

Η μέθοδος Horting είναι η γραφική προσέγγιση στο collaborative filtering. Περιλαμβάνει

έναν κατευθυνόμενο γράφο, όπου οι κορυφές αντιπροσωπεύουν τους χρήστες και οι ακμές τον βαθμό ομοιότητας ανάμεσά τους. Αν θέλουμε να προβλέψουμε την βαθμολογία του χρήστη *u* για ένα αντικείμενο *i*, χρειάζεται να βρούμε ένα κατευθυνόμενο μονοπάτι από τον χρήστη *u* προς έναν χρήστη που έχει βαθμολογήσει το *i*. Με τη χρήση γραμμικών μετασχηματισμών για την ανάθεση των ακμών κατά μήκος της διαδρομής, γίνεται η πρόβλεψη για τη βαθμολόγηση του χρήστη *u* για το αντικείμενο *i*.

2.2.2 Συστάσεις που βασίζονται σε ομαδοποίηση (clustering)

Οι τεχνικές ομαδοποίησης (clustering techniques) χρησιμοποιούνται από τα συστήματα συστάσεων για αναγνώριση ομάδων από χρήστες που έχουν παρόμοια ενδιαφέροντα. Για να υπολογίσουμε την βαθμολογία ενός χρήστη *u*, που ανήκει σε μία ομάδα (cluster), για ένα αντικείμενο *i*, υπολογίζουμε τη μέση βαθμολογία για το αντικείμενο *i* ανάμεσα στους χρήστες του cluster που ανήκει ο χρήστης.

Οι clustering techniques μπορούν επίσης να χρησιμοποιηθούν σαν τεχνικές επιλογής χρηστών που θα χρειαστούν για να μειώσουν το σύνολο των υποψηφίων γειτόνων για τον κ-Κοντινότεροι Γείτονες αλγόριθμο (k-Nearest Neighbors).

Οι τεχνικές αυτές συνήθως παράγουν λιγότερο προσωπικές συστάσεις σε σύγκριση με άλλες μεθόδους. Όμως όταν η διαδικασία clustering ολοκληρωθεί, η απόδοση μπορεί να είναι πολύ καλύτερη, αφού το σύνολο των χρηστών που πρέπει να αναλυθεί είναι πολύ μικρότερο. Έτσι, ενώ ο διαμερισμός σε clusters μπορεί να βλάψει την ακρίβεια των συστάσεων, είναι μία αξιόλογη ανταλλαγή, αν ληφθεί υπ’ όψιν η απόδοση.

3. ΜΕΘΟΔΟΙ

Οι μέθοδοι που ακολουθήσαμε ασχολούνται κυρίως με τις άλλες προσεγγίσεις. Για την αντιμετώπιση του προβλήματος χρησιμοποιήσαμε τις δύο αυτές διαφορετικές προσεγγίσεις.

3.1 IPF on STG graph

Η μέθοδος αφορά τη χρήση γράφου για την απεικόνιση χρηστών, αντικειμένων και τη χρονική σχέση μεταξύ τους, και τη χρήση αλγορίθμων για δημιουργία συστάσεων (κορυφαίων-N) μέσω αυτού. Ο γράφος αυτός είναι ένας χρονικός γράφος με βάση χρονικές περιόδους (session-based Temporal Graph - STG). Η μοναδικότητα αυτού του γράφου είναι πως ο χρόνος αντιπροσωπεύεται σαν χρονική περίοδος (session). Με αυτό τον τρόπο μοντελοποιεί ταυτόχρονα και τις μακροχρόνιες και τις βραχύχρονες προτιμήσεις του χρήστη. Για την δημιουργία των συστάσεων έχει προταθεί ο αλγόριθμος Injected Preference Fusion (IPF) [1]. Ο αλγόριθμος χρησιμοποιείται για να εξισορροπήσει τις επιπτώσεις των μακροχρόνιων και βραχύχρονων προτιμήσεων του χρήστη.

3.1.1 STG γράφος

Ο STG είναι ένας κατευθυνόμενος διμερής γράφος G(U, S, I, E, w) που περιέχει ένα σύνολο U από χρήστες, ένα σύνολο I από αντικείμενα και ένα σύνολο από S από χρονικές περιόδους (sessions). Το βάρος w : E → R συμβολίζει τη μη αρνητική συνάρτηση βάρους για τις ακμές.

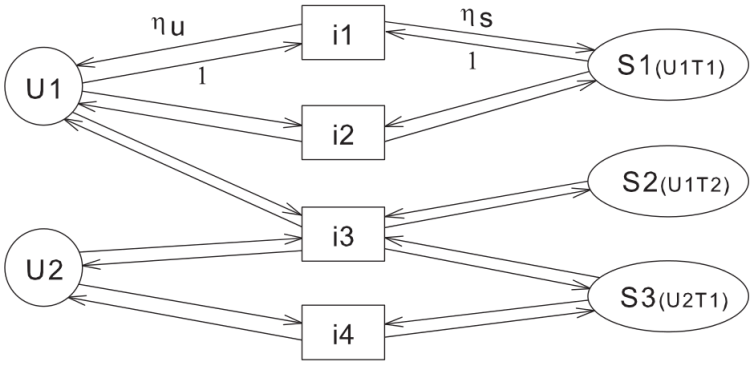
Ο συμβολισμός που χρησιμοποιείται και στη συνέχεια του κεφαλαίου, φαίνεται και στον πίνακα 1

Σύμβολα	**Περιγραφή**
G	ο διμερής γράφος STG
E	το σύνολο των ακμών του G
U, I, S	το σύνολο των κόμβων χρηστών, αντικειμένων και χρονικών περιόδων αντίστοιχα
u_u, u_i, u_{ut}	κόμβος χρήστη, αντικειμένου, χρονικής περιόδου
w	η συνάρτηση βάρους που ορίζεται στις ακμές
N(u)	το σύνολο τον αντικειμένων που έχει δει ο χρήστης *u*
N(u, t)	το σύνολο τον αντικειμένων που έχει δει ο χρήστης *u* τη χρονική στιγμή *t*
P	ένα μονοπάτι στον STG
P(u,i)	ένα σύνολο μονοπατιών από το u_u, u_{ut} στο u_i
$\psi(u_k, u_{k+1})$	η συνάρτηση προβολής από τον κόμβο u_k στον u_{k+1}
β	η παράμετρος που καθορίζει την αναλογία των προτιμήσεων που θα προβληθούν ανάμεσα στον κόμβο του χρήστη και στους χρονικούς κόμβους

Πίνακας 1: Σύμβολισμός για το STG

Το σχήμα 3 είναι ένα απλό παράδειγμα STG γράφου που περιέχει 2 κόμβους χρηστών, 3 κόμβους sessions, και 4 κόμβους αντικειμένων. Δείχνει τον χρήστη u_1 να αλληλεπιδρά

με τα αντικείμενα i_1, i_2, i_3 και τον χρήστη u_2 να αλληλεπιδρά με τα αντικείμενα i_3 και i_4. Επιπλέον, τα αντικείμενα i_1 και i_2 συνδέονται επίσης με τον χρονικό κόμβο s_1 γιατί και τα δύο τα είχε δει ο χρήστης u_1 τη χρονική στιγμή t_1. Το i_3 συνδέεται με τον χρονικό κόμβο s_2 και τα i_3, i_4 είναι επίσης συνδεδεμένα με τον κόμβο s_3.



Σχήμα 3: Παραδείγματα αναπαράστασης ενός STG γράφου

Στον STG γράφο, ο κόμβος του χρήστη u_u συνδέει όλα τα αντικείμενα που έχει δει ο χρήστης u, που συμβολίζονται ως $\mathrm{N}(u)$ και αντιπροσωπεύουν τις μακροχρόνιες προτιμήσεις του u. Ο χρονικός κόμβος u_{ut} συνδέει μόνο τα αντικείμενα που έχει δει ο χρήστης u τη χρονική στιγμή t, που συμβολίζονται ως $\mathrm{N}(u,t)$ και αντιπροσωπεύουν τις βραχύχρονες προτιμήσεις του u τη χρονική στιγμή t. Επομένως, αν ξεκινήσουμε από έναν κόμβο χρήστη u_u, θα περάσουμε μέσα από τα $\mathrm{N}(u)$ και μετά θα φτάσουμε σε άγνωστα αντικείμενα που είναι παρόμοια με τα $\mathrm{N}(u)$. Αν ξεκινήσουμε από τον χρονικό κόμβο u_{ut}, θα φτάσουμε σε άγνωστα αντικείμενα παρόμοια με αντικείμενα στο $\mathrm{N}(u,t)$. Με αυτόν τον τρόπο, ο κόμβος του χρήστη u_u είναι μία αναπαράσταση των μακροπρόθεσμων προτιμήσεων του χρήστη και ο χρονικός κόμβος u_{ut} είναι μία αναπαράσταση των βραχυπρόθεσμων προτιμήσεών του.

Αν δύο αντικείμενα συνδέονται μέσω κόμβων χρηστών, οι μακροχρόνιες προτιμήσεις τους συμβάλουν στην ομοιότητα μεταξύ τους· αν δύο αντικείμενα συνδέονται μέσω χρονικών κόμβων, οι βραχύχρονες προτιμήσεις τους συμβάλουν στην ομοιότητα μεταξύ τους.

3.1.2 IPF αλγόριθμος

Οι μακροχρόνιες και οι βραχύχρονες προτιμήσεις έχουν μοντελοποιηθεί στον STG γράφο από τους κόμβους χρηστών και τους χρονικούς κόμβους αντίστοιχα. Για να καλυφθεί η αλληλεπίδραση μεταξύ των δύο ειδών προτιμήσεων και η επίδρασή τους στις συστάσεις, έχει προταθεί ο αλγόριθμος IPF.

Η βασική ιδέα είναι να θεωρήσουμε τον κόμβο χρήστη u_u και τον αντίστοιχο χρονικό κόμβο u_{ut} σαν τη πηγή που θα εισαχθούν οι προτιμήσεις του χρήστη. Οι προτιμήσεις που προβάλλονται από τον κόμβο χρηστών, θα μεταδίδονται στα αντικείμενα $\mathrm{N}(u)$ που έχει δει ο χρήστης ανά πάσα στιγμή και μετά μεταδίδονται στα άγνωστα αντικείμενα σχετικά με τις

μακροχρόνιες προτιμήσεις του *u*, ενώ οι προτιμήσεις που προβάλλονται από τον χρονικό κόμβο θα μεταδίδονται στα αντικείμενα $N(u, t)$ που έχει δει ο χρήστης κάποια στιγμή *t*, και μετά μεταδίδονται στα άγνωστα αντικείμενα που σχετίζονται με τις βραχύχρονες προτιμήσεις του *u*. Στον STG γράφο, θα υπάρχει ένα σύνολο από μονοπάτια από τους κόμβους-πηγή σε άγνωστους κόμβους αντικειμένων. Οι προτιμήσεις του χρήστη σε έναν άγνωστο κόμβο αντικειμένου είναι το άθροισμα των βαρών όλων των εισερχόμενων μονοπατιών από τον κόμβο του χρήστη και τον αντίστοιχο χρονικό κόμβο. Ο κόμβος αντικειμένου με το μεγαλύτερο βαθμό θεωρείται ο καλύτερος για να προταθεί στον χρήστη. Οι προτιμήσεις που προβάλλονται σε αυτόν τον κόμβο είναι ο συνδυασμός των μακροχρόνιων και βραχύχρονων προτιμήσεων, ισορροπώντας τα δύο είδη προτιμήσεων.

Καθώς ο STG είναι ένας διμερής γράφος, η απόσταση από κόμβο χρήστη/χρόνου σε ένα άγνωστο αντικείμενο είναι πάντα ένας περιττός αριθμός, ίσος ή μεγαλύτερος του 3. Δεδομένου ενός χρήστη και ενός άγνωστου αντικειμένου, θα υπάρχουν άπειρα μονοπάτια (απόστασης >= 3) ανάμεσα σε αυτούς τους δύο κόμβους του STG. Αλλά τελικά υπολογίζουμε μόνο τα κοντινότερα μονοπάτια για να μετρήσουμε την προτίμηση του χρήστη στο αντικείμενο για δύο λόγους: αρχικά ένα μεγαλύτερο μονοπάτι συμβάλει λιγότερα βάρη απ' ότι τα κοντινότερα μονοπάτια στους τελικούς κόμβους αντικειμένων και μπορεί να προσθέσει ακόμα και "θόρυβο"· επιπλέον, τα κοντινότερα μονοπάτια μπορούν να βρεθούν ευκολότερα με τον αλγόριθμο Αναζήτησης Πρώτα σε Πλάτος (Bread-First-Search - BFS)

```
Data: STG G, user u, time t
Result: Recommendation for user u at time t
Queue Q;
NodeSet V;
Q.append(v_u);
Q.append(v_ut);
distance[v_u] = distance[v_ut] = 0;
rank[v_u] = β;
rank[v_ut] = 1 − β;
while Q is not empty do
    Node v = Q.top();
    if V.contains(v) then
        continue;
    if distance[v] > 3 then
        break;
    V.insert(v);
    foreach v' ∈ out(v) do
        if !V.contains(v') then
            distance[v'] = distance[v] + 1;
            Q.append(v');
        if distance[v] < distance[v'] then
            rank[v'] = rank[v'] + rank[v] ·ψ(v, v');
rank.sort();
return top-N unknown items;
```

Σχήμα 4: Ψευδοκώδικας του IPF για τη δημιουργία συστάσεων σε έναν χρήστη *u* που είναι ενεργός τον χρόνο *t*

Συνοπτικά, ξεκινώντας από έναν χρήστη *u*, υπάρχουν τέσσερα είδη από κοντινότερα μο-

νοπάτια που ξεκινάνε από τους κόμβους-πηγή u_u, u_{ut} και καταλήγουν σε έναν άγνωστο κόμβο αντικειμένου με τρία βήματα:

- *χρήστης - αντικείμενο - χρήστης - αντικείμενο* (P1): Αυτό το είδος των μονοπατιών ξεκινάνε από έναν κόμβο χρήστη u_u και μεταβαίνουν σε όλα τα αντικείμενα $\mathrm{N}(u)$ που έχει δει ο u. Η τελευταία μετάβαση είναι προς όλα τα άγνωστα αντικείμενα προς τον u, μέσω άλλων χρηστών που έχουν δει επίσης κάποια από τα αντικείμενα στο $\mathrm{N}(u)$.
- *χρήστης - αντικείμενο - χρονικό διάστημα - αντικείμενο* (P2): Αυτό το είδος των μονοπατιών είναι παρόμοιο με το P1, με τη διαφορά όμως ότι συνδέει τα αντικείμενα που έχει δει ο χρήστης με άγνωστα αντικείμενα μέσω των χρονικών κόμβων αντί των κόμβων χρηστών.
- *χρονικό διάστημα - αντικείμενο - χρήστης - αντικείμενο* (P3): Αυτό το είδος των μονοπατιών ξεκινάνε από έναν χρονικό κόμβο u_{ut} και μεταβαίνουν σε όλα τα αντικείμενα $\mathrm{N}(u,t)$ που έχει δει ο u τη χρονική στιγμή t. Η τελευταία μετάβαση είναι προς όλα τα άγνωστα αντικείμενα προς τον u, μέσω άλλων χρηστών που έχουν δει επίσης κάποια από τα αντικείμενα στο $\mathrm{N}(u)$.
- *χρονικό διάστημα - αντικείμενο - χρονικό διάστημα - αντικείμενο* (P4): Αυτό το είδος των μονοπατιών είναι παρόμοιο με το P3, αλλά συνδέει τα αντικείμενα που έχει δει ο χρήστης με άγνωστα αντικείμενα μέσω των χρονικών κόμβων αντί των κόμβων χρηστών.

Περιληπτικά, τα μονοπάτια P1 και P2 ξεκινάνε από έναν κόμβο χρήση u_u, και στο τέλος καταλήγουν σε άγνωστα αντικείμενα παρόμοια με τα $\mathrm{N}(u)$, αντανακλώντας τις μακροχρόνιες προτιμήσεις του u. Τα μονοπάτια P3 και P4 ξεκινάνε από έναν χρονικό κόμβο u_{ut}, και στο τέλος φτάνουν σε άγνωστους κόμβους αντικειμένων παρόμοια με το $\mathrm{N}(u,t)$, αντικατοπτρίζοντας τις βραχύχρονες προτιμήσεις του χρήστη. Τα P1 και P3 συνδέουν τα αντικείμενα μέσω κόμβων χρηστών, μετρώντας την ομοιότητα των αντικειμένων κυρίως από τις μακροχρόνιες προτιμήσεις των χρηστών. Αντίστοιχα, τα μονοπάτια P2 και P4 συνδέουν αντικείμενα με τους αντίστοιχους χρονικούς κόμβους, μετρώντας την ομοιότητα των αντικειμένων κυρίως από τις βραχύχρονες προτιμήσεις των χρηστών.

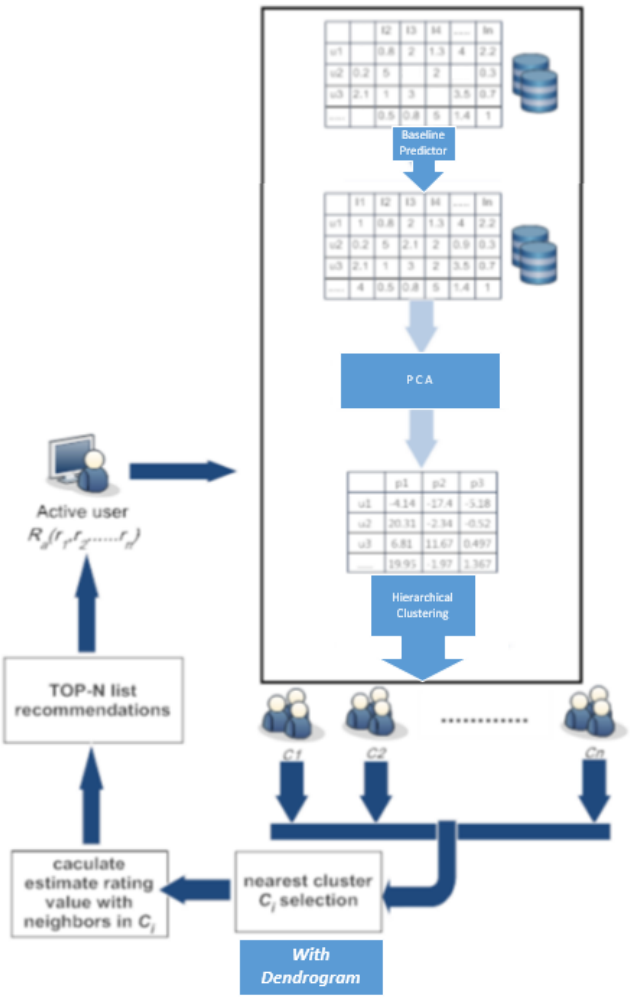
3.2 PCA with Hierarchical clustering

Για τη δεύτερη μέθοδο χρησιμοποιήσαμε την τεχνική της ομαδοποίησης (clustering). Παρόλο που το clustering δεν είναι δημοφιλής μέθοδος για συστήματα συστάσεων εξαιτίας της μειωμένης απόδοσης αναφορικά με τις στατιστικές μετρήσεις ακριβείας, επιλέξαμε να τον χρησιμοποιήσουμε γιατί προσφέρει επεκτασιμότητα που συμβάλει στην ταχύτητα στα σημερινά συστήματα συστάσεων μεγάλων διαστάσεων. Επιπλέον, καθώς είναι μία από τις πιο δημοφιλείς τεχνικές εξόρυξης δεδομένων, αξίζει να ερευνηθεί για το πώς μπορεί να επεκταθεί για να αξιοποιήσει τη χρονική πληροφορία.

Οι παραδοσιακές μέθοδοι στα collaborative filtering των συστημάτων συστάσεων ψάχνουν όλο τον χώρο για να εντοπίσουν τους κ-κοντινότερους γείτονες από τον χρήστη που ψάχνουμε. Δεδομένου όμως των μεγάλων διαστάσεων που έχουν οι πίνακες του προφίλ των χρηστών, είναι δύσκολο να υπολογίσουμε την ομοιότητα για να βρούμε παρόμοια σκεπτόμενους γείτονες, που οδηγεί σε κακά αποτελέσματα λόγω του χώρου. Για να αντιμετωπίσουμε αυτό το ζήτημα, ακολουθήσαμε την τεχνική της Ανάλυσης Κύριων Συνιστωσών (Principal Component Analysis), με στόχο να συγκεντρωθεί η πληροφορία σε έναν σχετικά χαμηλών διαστάσεων και πυκνό χώρο.

Η διαδικασία που ακολουθούμε σε αυτή την μέθοδο περιγράφεται από την εικόνα 1 και είναι η εξής:

1. Δημιουργία ενός πίνακα Χρήστες $\times$ Αντικείμενα που περιέχει τις βαθμολογίες. Για τις κενές βαθμολογίες (για όσους χρήστες δεν έχουν βαθμολογήσει τα συγκεκριμένα αντικείμενα), χρησιμοποιούμε τη συνάρτηση **βασικής πρόβλεψης (baseline predictor)**.
2. Φιλτράρισμα δεδομένων και επιλογή των σημαντικότερων χαρακτηριστικών με την χρήση του **Principal Component Analysis (PCA)**.
3. Ομαδοποίηση (clustering) των όμοιων χρηστών με τον αλγόριθμο **ιεραρχικής ομαδοποίησης (hierarchical clustering).** Από τα clusters των όμοιων χρηστών, επιλέγονται οι 30 πιο κοντινοί γείτονες, με βάση την απόσταση από την κορυφή και το πως έχουν οργανωθεί στο **δενδροδιάγραμμα (dendrogram).**
4. Επιλογή από όλους τους παρόμοιους χρήστες των αντικειμένων που έχουν δει, αλλά δεν έχει δει ο χρήστης και αξιολόγησή τους με βάση τη **συνάρτηση αξιολόγησης** που περιγράφεται εδώ [4] (και αναπτύχθηκε στην ίδια βάση δεδομένων που χρησιμοποιούμε στην πτυχιακή εργασία).
5. Ταξινόμηση των επιχειρήσεων με βάση την βαθμολογία τους και σύσταση στον χρήστη των **κορυφαίων-Ν αντικειμένων**



Εικόνα 1: Διαδικασία για τη μέθοδο PCA με hierarchical clustering

3.2.1 Baseline predictor

Οι baseline predictors χρησιμοποιούνται σε παραγοντικά μοντέλα (factor model). Τα factor models είναι μία από τις πιο πετυχημένες προσεγγίσεις στο collaborative filtering, χρησιμοποιώντας πίνακα. Οι πίνακες στα factor models αντιστοιχίζουν τους χρήστες στα αντικείμενα (πχ ταινίες, επιχειρήσεις) και περιεχόμενό τους είναι η βαθμολογία που έχουν δώσει οι χρήστες σε αυτά τα αντικείμενα.

Αυτού του είδους τα μοντέλα είναι χρήσιμα στην αποτύπωση τις αλληλεπίδρασης ανάμεσα στους χρήστες και τα αντικείμενα. Ωστόσο, μεγάλος αριθμός από τις τιμές που έχουν οι αξιολογήσεις σχετίζεται είτε με τους χρήστες είτε με τα αντικείμενα και είναι ανεξάρτητες από την αλληλεπίδρασή τους. Ένα χαρακτηριστικό παράδειγμα είναι ότι τα τυπικά δεδομένα στο CF παρουσιάζουν μεγάλες διαφορές λόγω των προκαταλήψεων και των προτιμήσεων που έχουν οι χρήστες. Ο τρόπος που βαθμολογούν οι χρήστες είναι διαφορετικός για κάθε χρήστη και επηρεάζεται από διαφορετικούς για τον καθένα παράγοντες - συστηματικά κάποιοι χρήστες δίνουν μεγαλύτερη βαθμολογία από κάποιους άλλους και κάποια αντικείμενα λαμβάνουν μεγαλύτερη βαθμολογία από κάποια άλλα.

Για να αποφύγουμε τέτοια φαινόμενα, ώστε να μην περιλαμβάνεται η αλληλεπίδραση χρή-

στη - αντικειμένου, χρησιμοποιούμε τους *baseline predictors*. Οι baseline predictors τείνουν να εντοπίζουν αρκετή από την αλληλεπίδραση που υπάρχει και συγκεκριμένα αρκετή από την χρονική δυναμική που έχουν τα δεδομένα. Ως εκ τούτου, είναι ζωτικής σημασίας για το μοντέλο να διαμορφώσει με ακρίβεια τις βαθμολογίες. Αυτό επιτρέπει την καλύτερη αναγνώριση του τμήματος που αντιπροσωπεύει την πραγματική αλληλεπίδραση χρήστη - αντικειμένου.

Ένας πρακτικός τρόπος για τη δημιουργία ενός στατικού baseline predictor είναι ο ακόλουθος: Συμβολίζουμε με μ την συνολική μέση βαθμολογία (μέση βαθμολογία που προκύπτει από όλους τους χρήστες και όλα τα αντικείμενα). Ένας baseline predictor για μία άγνωστη βαθμολογία r_{ui} ενός χρήστη u σε ένα αντικείμενο i συμβολίζεται με b_{ui} και περιλαμβάνει τις κύριες επιδράσεις των χρηστών και αντικειμένων ως εξής:

$$b_{ui} = \mu + b_u + b_i \tag{3}$$

Οι παράμετροι b_u και b_i δείχνουν τις αποκλίσεις του χρήστη u και του αντικειμένου i από τον μέσο όρο.

Επιστρέφοντας στο παράδειγμα που είχαμε χρησιμοποιήσει και στην αρχή, έστω πως θέλουμε να προβλέψουμε την βασική εκτίμηση για την βαθμολογία του εστιατορίου ”Ο Μεξικάνος” από τον χρήστη Χριστίνα. Ας υποθέσουμε πως η μέση βαθμολογία για όλες τις επιχειρήσεις είναι 3,9 αστέρια. Επιπλέον, ”Ο Μεξικάνος” είναι λίγο καλύτερος από μία μέση επιχείρηση, οπότε έχει 0,2 περισσότερα αστέρια από τον μέσο όρο. Από την άλλη μεριά, η Χριστίνα είναι λίγο αυστηρή συνήθως με τις αξιολογήσεις της και βαθμολογεί με 0,3 αστέρια χαμηλότερα από τον μέσο όρο. Επομένως, η βασική εκτίμηση για την βαθμολογία του εστιατορίου ”Ο Μεξικάνος” από την Χριστίνα είναι $3,9 - 0,3 + 0,2 = 3,8$ αστέρια.

Αφού περιγράψαμε τι είναι, ήρθε η ώρα να δούμε πως χρησιμοποιούνται οι baseline predictors στη δημιουργία του πίνακα Χρήστες $\times$ Αντικείμενα. Κατά την κατασκευή του πίνακα βάζουμε για όλους τους χρήστες τις γνωστές βαθμολογίες που έχουν δώσει στα αντικείμενα. Ένα όμως πρόβλημα που προκύπτει είναι πως δεν έχουν βαθμολογήσει όλοι οι χρήστες όλα τα αντικείμενα. Έτσι ανάλογα με την πυκνότητα των δεδομένων έχουμε περισσότερα ή λιγότερα μηδενικά για τους χρήστες που δεν έχουν βαθμολογήσει τα αντικείμενα (οι βαθμολογίες είναι στην κλίμακα 1-5 αστέρια). Αυτός όμως ο πίνακας σαν είσοδο στην επόμενη μέθοδο θα δώσει λάθος αποτελέσματα, αφού η μη βαθμολόγηση θα προσμετρηθεί σαν μηδέν.

Για να αντιμετωπίσουμε αυτό το πρόβλημα κάνουμε κανονικοποίηση στον πίνακα, με τη χρήση του baseline predictor. Συγκεκριμένα, αφαιρούμε από την βαθμολογία του χρήστη, την βαθμολογία που προκύπτει από τον baseline predictor. Στο παράδειγμα που χρησιμοποιούμε, η πραγματική τιμή που έδωσε η Χριστίνα είναι 4 αστέρια, οπότε στον πίνακα πλέον θα έχουμε την απόκλιση $4 - 3,8 = 0,2$. Με αυτό τον τρόπο, τα μηδενικά πλέον θα αντικατασταθούν από την εκτίμηση για την βαθμολογία. Τότε το μόνο που μένει από

το collaborative filtering είναι να εντοπίσει με ακρίβεια την αλληλεπίδραση ανάμεσα στους χρήστες και τα αντικείμενα.

3.2.2 Prefiltering δεδομένων με PCA

Από την παραπάνω διαδικασία προκύπτει ένας πίνακας με *Όλους τους Χρήστες* $\times$ *Όλα τα Αντικείμενα*, με μεγάλες όμως διαστάσεις. Για να μην αλλοιωθούν τα χαρακτηριστικά των χρηστών από την επιπλέον πληροφορία και οδηγηθούμε σε φτωχά αποτέλεσμα, κάνουμε μία ανάλυση των δεδομένων αυτών με την τεχνική του PCA.

Καθότι είναι μία από τις δημοφιλέστερες τεχνικές εξαγωγής χαρακτηριστικών στην ανάλυση δεδομένων, το PCA χρησιμοποιείται ευρέως στην επεξεργασία και το φιλτράρισμα χαρακτηριστικών και στην μείωση των διαστάσεων στα συστήματα του collaborative filtering.

Ας δούμε όμως πρώτα τι κάνει διαισθητικά η τεχνική του PCA πριν προχωρήσουμε σε τεχνικές λεπτομέρειες με ένα παράδειγμα. Όπως είπαμε και στην εισαγωγή, οι χρήστες αξιολογούν τις επιχειρήσεις με διάφορα κριτήρια, όπως η ποιότητα του προσφερόμενου προϊόντος, η τοποθεσία, η εξυπηρέτηση, η ατμόσφαιρα του χώρου, το ωράριο και άλλα. Μπορούμε να φτιάξουμε μία ολόκληρη λίστα από διαφορετικά χαρακτηριστικά για κάθε χρήστη, αλλά πολλά από αυτά θα μετράνε παρόμοιες ιδιότητες, επομένως θα είναι περιττά. Για να μπορούμε να συνοψίσουμε τον τρόπο που βαθμολογεί κάθε χρήστης με λιγότερα χαρακτηριστικά, χρησιμοποιούμε το PCA.

Με τη μέθοδο όμως του PCA δεν επιλέγουμε κάποια χαρακτηριστικά απορρίπτοντας κάποια άλλα. Αντί γι' αυτό, δημιουργούνται *νέα* χαρακτηριστικά που αποδεικνύεται πως συνοψίζουν τα προηγούμενα. Φυσικά, αυτά τα νέα χαρακτηριστικά δημιουργούνται από τα παλιά· ένα νέο χαρακτηριστικό για παράδειγμα, μπορεί να είναι το ωράριο συν την τοποθεσία ή κάποιος άλλος παρόμοιος συνδυασμός. Όπως βλέπουμε, δεν έχουν απαραίτητα κάποια φυσική σημασία αυτά τα νέα χαρακτηριστικά.

Στην πραγματικότητα, ο PCA βρίσκει τα καλύτερα δυνατά χαρακτηριστικά, αυτά που περιγράφουν όσο το δυνατόν καλύτερα τον τρόπο αξιολόγησης των χρηστών.

Για να δημιουργήσει τα νέα χαρακτηριστικά, απαιτείται να υπάρχει μεγάλο εύρος στις τιμές που θα έχει το συγκεκριμένο χαρακτηριστικό. Η επιλογή κάποιας ιδιότητας που έχει πολύ μικρή διακύμανση στις τιμές της ανάμεσα στους περισσότερους χρήστες, δεν θα ήταν ιδιαίτερα χρήσιμη. Οι χρήστες είναι όλοι διαφορετικοί, όμως αυτή η νέα ιδιότητα τους κάνει να φαίνονται όλοι ίδιοι. Αντί γι' αυτό, ο PCA ψάχνει για χαρακτηριστικά που έχουν όσο το δυνατόν μεγαλύτερη ποικιλία ανάμεσα στους χρήστες.

Από τεχνικής πλευράς, η βασική ιδέα του PCA είναι η μετατροπή των αρχικών δεδομένων σε έναν νέο χώρο συντεταγμένων που αναπαρίσταται από τις κύριες συνιστώσες (principal components) των δεδομένων με τις υψηλότερες ιδιοτιμές. Οι συνιστώσες ταξινομούνται με βάση την ιδιοτιμή τους από την υψηλότερη στην χαμηλότερη. Το πρώτο διάνυσμα της κύριας συνιστώσας περιέχει την πιο σημαντική πληροφορία. Οι μικρότερης

σημασίας συνιστώσες αγνοούνται για να σχηματίσουν ένα χώρο με λιγότερες διαστάσεις από το αρχικό.

Από την προηγούμενη μέθοδο έχει προκύψει ο πίνακας χρηστών $\times$ επιχειρήσεων, διαστάσεων $m \times n$ όπου $m = |U|$ και $n = |I|$ το πλήθος των χρηστών και των αντικειμένων αντίστοιχα. Το n-διαστάσεων διάνυσμα αντιπροσωπεύει το προφίλ του χρήστη. Μετά από την ανάλυση των ιδιοτιμών από τις n κύριες συνιστώσες, επιλέγουμε μόνο τις πρώτες d συνιστώσες $(d \ll n)$ για να αποκτήσουμε τον νέο χώρο δεδομένων, ο οποίος βασίζεται συσσωρευτικά στο 90% της πληροφορίας των αρχικών διαστάσεων. Συνεπώς, ο μειωμένος διανυσματικός πίνακας από το PCA είναι έτοιμος να τροφοδοτήσει τον αλγόριθμο ιεραρχικής ομαδοποίησης (hierarchical clustering).

3.2.3 Hierarchical Clustering

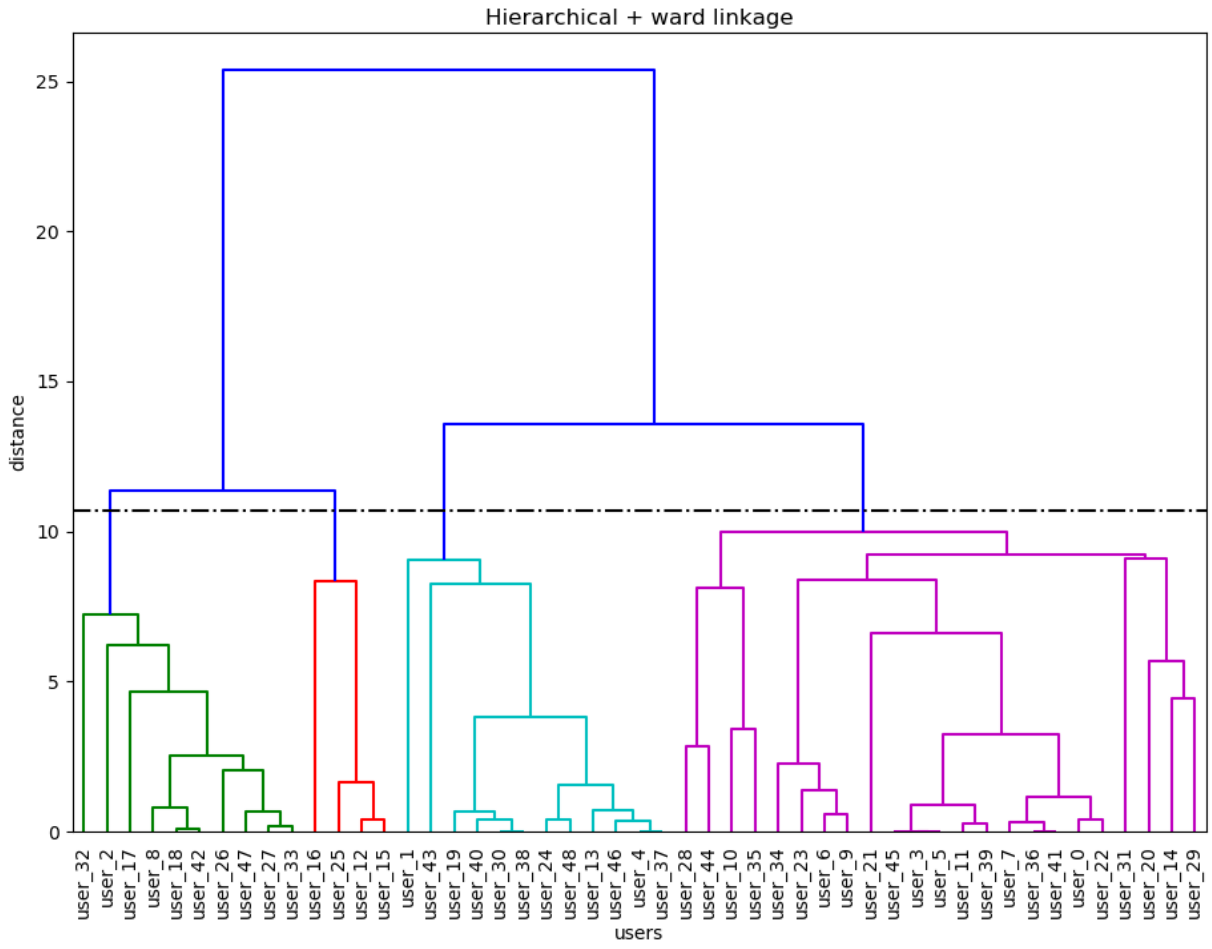
Η ομαδοποίηση (clustering) είναι ένα κρίσιμο και σημαντικό βήμα στην μέθοδό μας, γιατί οι ομάδες (clusters) που δημιουργούνται μας δίνουν την γειτονιά του ενεργού χρήστη. Όλοι οι υπολογισμοί που πραγματοποιούνται μετά από αυτό το βήμα (η επιλογή των βαρών, η δημιουργία των συστάσεων) εξαρτώνται από τα clusters. Οι μέθοδοι για τα δίκτυα των clusters χωρίζονται σε δύο κατηγορίες: Σε διαχωρισμό γράφου και σε μοντελοποίηση από blocks (οργάνωση πειραματικών μονάδων σε ομάδες που είναι παρόμοιες μεταξύ τους). Το hierarchical clustering ανήκει στην δεύτερη κατηγορία.

Εστιάσαμε στο hierarchical clustering δεδομένου ότι δεν απαιτεί παραδοχές σχετικά με τη δομή του συμπλέγματος του δικτύου. Η ιδέα είναι να ”χτίζουμε” συνεχώς, ή να διαιρούμε μία ιεραρχία από clusters (που ονομάζεται *δενδρόγραμμα*), με τα φύλλα να είναι οι αρχικοί κόμβοι του δικτύου, τη ρίζα να αντιπροσωπεύει ολόκληρη τη γραφική παράσταση και τους εσωτερικούς κόμβους να αντιστοιχούν στα clusters σε διαφορετικά βήματα του αλγορίθμου.

Μπορούμε να σταματήσουμε τον αλγόριθμο σε ένα οποιοδήποτε βήμα του, όταν η διαμέριση που έχει προκύψει είναι η βέλτιστη σύμφωνη με ορισμένα μέτρα.

Στην εικόνα 2 βλέπουμε το δενδρόγραμμα που έχει προκύψει από την ιεραρχική ομαδοποίηση 50 τυχαίων χρηστών από τη βάση του Yelp το 2007. Στον άξονα των *x* είναι τα ονόματα των χρηστών, και στον άξονα των *y* οι αποστάσεις (με τη μέθοδο ’ward’). Οι οριζόντιες γραμμές δείχνουν το σημείο που γίνεται η συγχώνευση των clusters, ενώ οι κάθετες δείχνουν ποια clusters ή ποιοι χρήστες είναι μέρος της συγχώνευσης που δημιουργεί το νέο cluster. Το ύψος των οριζόντιων γραμμών εκφράζει την απόσταση που χρειάζεται για να ενωθεί και να δημιουργηθεί ένα νέο cluster.

Όπως φαίνεται από την εικόνα 2, αν επιλέξουμε να σταματήσουμε τον αλγόριθμο σε απόσταση λίγο μεγαλύτερη από το 10, βλέπουμε πως σχηματίζονται 4 ομάδες (clusters), που φαίνονται από το διαφορετικό τους χρώμα.



Εικόνα 2: Δενδρόγραμμα που έχει δημιουργηθεί από την ομαδοποίηση 50 τυχαίων χρηστών του 2007

Από τα clusters που δημιουργούνται, επιλέγουμε τους 10 κοντινότερους γείτονες του χρήστη όπως αυτοί προκύπτουν από το δενδρόγραμμα. Αν δεν έχει τόσους χρήστες, τότε επιλέγουμε όλους τους χρήστες που είναι στο ίδιο cluster.

3.2.4 Συνάρτηση αξιολόγησης

Σε αυτό το στάδιο έχουμε τον χρήστη *u* που θέλουμε να του κάνουμε συστάσεις και τους 10 χρήστες που έχουν παρόμοιες προτιμήσεις.

Για να βρούμε και να αξιολογήσουμε τα αντικείμενα που θα προτείνουμε στον χρήστη, αρχικά παίρνουμε όλα τα αντικείμενα που έχουν αξιολογήσει οι 10 χρήστες και επιλέγουμε από αυτά όσα δεν έχει αξιολογήσει ο χρήστης. Αξιολογούμε τα αντικείμενα με βάση το γραμμικό μοντέλο που έχει προταθεί [4], και έχει δοκιμαστεί πάνω στην ίδια βάση δεδομένων.

Το μοντέλο βασίζεται σε έναν απλό γραμμικό συνδυασμό του baseline predictor και της ομοιότητας των χρηστών. Σε κάθε μία από τις συνιστώσες της συνάρτησης, αποδίδεται και ένα βάρος. Για να βρεθεί η πρόβλεψη για την ομοιότητα των χρηστών για έναν χρήστη *u* σε μία επιχείρηση *b*, το πρώτο βήμα είναι να αναγνωρίσουμε τους κορυφαίους *K* χρή-

στες που είναι παρόμοιοι στον u, έχουν βαθμολογήσει την b, και να βρούμε τον μέσο όρο βαθμολογίας τους για την b. Η συνάρτηση αξιολόγησης ορίζεται ως εξής:

$$rating(u, b) = w_1\alpha + w_2\theta + w_3\eta \tag{4}$$

Σε αυτό το μοντέλο, το α είναι ο μέσος όρος της βαθμολογίας που έχει δώσει ο χρήστης u σε όλες τις επιχειρήσεις εκτός από τη b θ είναι η μέση βαθμολογία της επιχείρησης b από όλους τους χρήστες εκτός από τον u και η είναι η μέση βαθμολογία των κορυφαίων K χρηστών παρόμοιων με τον u, για την επιχείρηση b.

Τυπικά ένας χρήστης έχει υψηλή ομοιότητα με έναν πολύ μικρό αριθμό από χρήστες, γι' αυτό και επιλέξαμε $K = 10$ χρήστες.

Τα βάρη w_1, w_2, w_3 ρυθμίζουν το ποσοστό επιρροής της κάθε παραμέτρου και το άθροισμά τους ισούται με 1. Το σύνολο που έδωσε το καλύτερο αποτέλεσμα ήταν $(w_1, w_2, w_3) = (0.01, 0.43, 0.56)$. Τα βάρη αυτά δείχνουν ότι ο ρόλος του α, δηλαδή του μέσου όρου της βαθμολογίας που έχει δώσει ο χρήστης σε όλες τις επιχειρήσεις, δεν είναι μεγάλης σημασίας εξαιτίας της υψηλής διασποράς στις βαθμολογίες που έχει δώσει ο χρήστης. Επιπλέον, βλέπουμε πως ο όρος η έχει το υψηλότερο βάρος, δείχνοντας πως όντως οι παρόμοιοι χρήστες βαθμολογούν παρόμοια.

4. ΠΕΙΡΑΜΑΤΑ

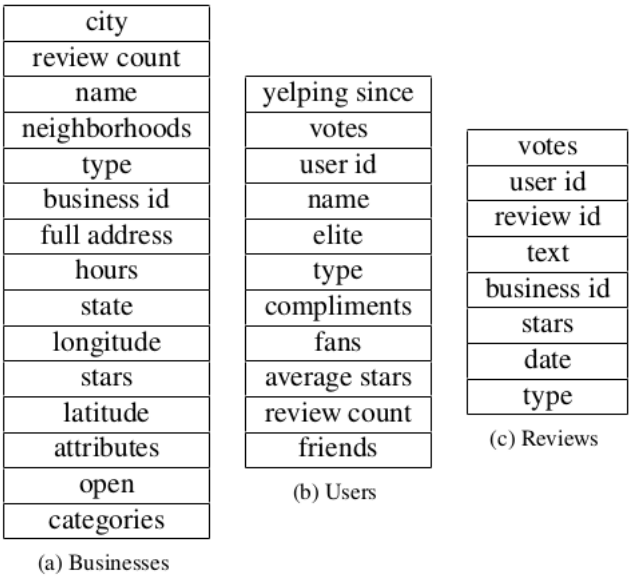
Σε αυτή την ενότητα παρουσιάζονται τα πειράματα που πραγματοποιήθηκαν και οι πληροφορίες που σχετίζονται με αυτά (δεδομένα, μέθοδοι αξιολόγησης, περιγραφή πειράματος).

4.1 Περιγραφή Δεδομένων

Στόχος μας είναι να προτείνουμε μία ακριβή λίστα με top-N προτάσεις σε πραγματικούς χρήστες. Για να πραγματοποιήσουμε τα πειράματά μας χρησιμοποιήσαμε μία ρεαλιστική βάση δεδομένων, ώστε να έχουν πραγματικό αντίκτυπο, το Yelp [3].

Το Yelp είναι ένα site όπου χρήστες αξιολογούν επιχειρήσεις και σχηματίζουν ένα κοινωνικό δίκτυο. Το ίδιο το site έχει σελίδες αφιερωμένες σε μεμονωμένες τοποθεσίες, όπως εστιατόρια ή μπαρ, όπου οι χρήστες μπορούν να γράψουν μία κριτική για τα προϊόντα ή τις υπηρεσίες, χρησιμοποιώντας ένα σύστημα αξιολόγησης από ένα έως πέντε αστέρια. Εκτός από το να γράφουν κριτικές, οι χρήστες μπορούν επίσης να αντιδρούν σε αυτές (αν είναι αστεία, χρήσιμη κλπ). Οι επιχειρήσεις μπορούν να ανανεώνουν τις πληροφορίες τους (όπως πληροφορίες επικοινωνίας, ωράριο) αλλά και να προσθέτουν tags (ετικέτες) με τα χαρακτηριστικά τους (είδος επιχείρησης, είδος κουζίνας, παροχή wifi).

Η βάση του Yelp περιέχει την δομή που φαίνεται και στην εικόνα 3. Ο πίνακας που χρησιμοποιήσαμε κυρίως είναι ο *Reviews* που αποτελείται από 2.225.213 κριτικές, 52.689 χρήστες, 77.079 επιχειρήσεις και περιλαμβάνει κριτικές από τις 12 Οκτωβρίου 2004 ως τις 24 Δεκεμβρίου 2015.



Εικόνα 3: Η μορφή της βάσης δεδομένων του Yelp

Επειδή ο όγκος των δεδομένων είναι πολύ μεγάλος για να τον διαχειριστούμε, για τα πειράματα χρησιμοποιήσαμε χρονικά υποσύνολα του ενός έτους, δημιουργώντας ένα φυσικό

όριο στα δεδομένα. Έτσι χρησιμοποιήσαμε τις χρονιές 2010, 2011, 2012.

Οι πίνακες 2 και 3 δείχνουν την σύγκριση των δεδομένων. Ο πίνακας 2 δείχνει τα πραγματικά δεδομένα όπως αντλήθηκαν από τη βάση του Yelp. Και στις τρεις χρονιές υπήρχαν πάρα πολλοί χρήστες που είχαν βαθμολογήσει μόνο μία επιχείρηση, και πολλές επιχειρήσεις που είχαν λίγες βαθμολογίες. Αυτό φαίνεται και από το ποσοστό στον πίνακα της πυκνότητας, που εκφράζει πόσο αραιά είναι τα δεδομένα. Το ποσοστό αυτό ονομάζεται sparsity (μη πυκνότητα), μετράει τα κενά που υπάρχουν στον πίνακα Χρήστες $\times$ Αντικείμενα και ορίζεται ως εξής:

$$sparsity = 1 - \frac{\text{Αριθμός Κριτικών}}{\text{Αριθμός Χρηστών} \times \text{Αριθμός Αντικειμένων}} \tag{5}$$

Χρονιά	Μέγεθος			Πυκνότητα		
	Αριθμός Χρήστη	Αριθμός Αντικειμένο	Αριθμός Κριτικών	(%)	Μ.Ο Κριτικών /Χρήστες	Μ.Ο Κριτικών /Αντικείμενα
2010	32798	24369	138075	99.98	4.20	5.66
2011	58305	31615	211252	99.98	3.62	6.68
2012	76187	38050	246993	100.0	3.24	6.49

Πίνακας 2: Στατιστική σύγκριση των συνόλων δεδομένων πριν την επεξεργασία

Η χαμηλή πυκνότητα όμως των δεδομένων οδηγεί σε κακής ποιότητας αποτελέσματα [7]. Για την πτυχιακή εργασία χρησιμοποιήσαμε ένα πιο πυκνό υποσύνολο αυτών των ετών, αφαιρώντας πρώτα τις επιχειρήσεις που είχαν λιγότερο από 5 κριτικές, και χρήστες που έχουν πραγματοποιήσει λιγότερες από 10 κριτικές. Τα νέα σύνολα δεδομένων φαίνονται στον πίνακα 3 που ακολουθεί:

Χρονιά	Μέγεθος			Πυκνότητα		
	Αριθμός Χρήστη	Αριθμός Αντικειμένο	Αριθμός Κριτικών	(%)	Μ.Ο Κριτικών /Χρήστες	Μ.Ο Κριτικών /Αντικείμενα
2010	1091	2904	25325	99.20	23.21	8.72
2011	1114	3673	30958	99.24	27.78	8.42
2012	1110	9321	41574	99.60	37.45	4.46

Πίνακας 3: Στατιστική σύγκριση των συνόλων δεδομένων μετά την επεξεργασία

4.1.1 Επιλογή χρηστών

Για το πείραμα επιλέξαμε για κάθε χρονιά 100 τυχαίους χρήστες. Τα κριτήρια που έπρεπε να πληρούν είναι τα εξής:

- Να έχουν πραγματοποιήσει τουλάχιστον 10 reviews το τελευταίο τρίμηνο της χρονιάς που εξετάζουμε (Οκτώβριο, Νοέμβριο, Δεκέμβριο).
- Να έχουν πραγματοποιήσει τουλάχιστον 10 reviews τον υπόλοιπο χρόνο.

4.2 Μέθοδοι Αξιολόγησης

Για να αξιολογήσουμε τα αποτέλεσμα των χρηστών εφαρμόσαμε μεθόδους ποσοτικής και ποιοτικής αξιολόγησης. Οι πρώτοι σχετίζονται μόνο με τις αριθμητικές διαφορές που υπάρχουν ανάμεσα στις λίστες που προκύπτουν από τις δύο μεθόδους, ενώ οι δεύτεροι εξετάζουν το περιεχόμενο των λιστών και το ποσοστό επιτυχίας στις προτάσεις που έκαναν.

4.2.1 Ποσοτική Αξιολόγηση

Αρχικά συγκρίνουμε και αξιολογούμε τις λίστες για κάθε χρήστη που προκύπτουν και από τις δύο μεθόδους. Για κάθε χρήστη συγκρίνουμε τις δύο λίστες με τους εξής τρόπους:

(1) Αριθμητικές διαφορές

(2) Μετρώντας τις αναστροφές (inversions)

Με την μέθοδο (1) βρίσκουμε πόσο διαφέρουν οι δύο λίστες, μετρώντας πόσα αντικείμενα υπάρχουν στη μία λίστα, αλλά όχι στην άλλη.

Με την μέθοδο (2) μετράμε πόσες αναστροφές υπάρχουν στις δύο λίστες για να μετρήσουμε την ομοιότητα των δύο κατατάξεων. Η μέτρηση αναστροφών για μία λίστα υποδηλώνει πόσο απέχει μία λίστα από το να είναι ταξινομημένη. Αν είναι ήδη ταξινομημένη, τότε οι αναστροφές είναι 0. Η μέγιστη τιμή προκύπτει αν η λίστα είναι ταξινομημένη με την ακριβώς αντίθετη σειρά.

Για να τις υπολογίσουμε, επιλέγουμε μία από τις δύο λίστες και αντιστοιχίζουμε τα νούμερα 1 ως N (όπου N το μέγεθος της top-N λίστας) με την σειρά που εμφανίζονται στη λίστα, και την ορίζουμε ως ταξινομημένη. Η δεύτερη λίστα, για τα αντικείμενα που είναι κοινά, θα περιέχει μία μετάθεση των αντικειμένων της πρώτης λίστας.

Για να μετρήσουμε το πλήθος των μεταθέσεων σε μία λίστα, έστω: $r_1, r_2, ..., r_N$ Στη συνέχεια μετράμε το πλήθος των ζευγαριών που βρίσκονται εκτός ταξινόμησης (που έχουν αναστραφεί). Δύο αντικείμενα r_i και r_j σχηματίζουν μία αναστροφή αν $r_i > r_j$ και $i < j$.

Παράδειγμα: Η ακολουθία $2, 4, 1, 3, 5$ έχει τρεις αναστροφές $(2, 1)$, $(4, 1)$, $(4, 3)$.

4.2.2 Ποιοτική Αξιολόγηση

Στόχος είναι να αξιολογήσουμε ποιοτικά τα αποτελέσματα της τοπ-N λίστας προτάσεων, και να βρούμε με τι ποσοστό επιτυχίας οι αλγόριθμοι εμφάνισαν στην λίστα επιχειρήσεις που έχουν δει οι χρήστες, αλλά και σε ποια θέση βρίσκονται αυτές στη λίστα.

Για αυτό τον λόγο, τα δεδομένα χωρίζονται σε δύο κατηγορίες, τα δεδομένα που χρησιμοποιούνται για την εκπαίδευση του αλγορίθμου (training set) και τα δεδομένα που χρησιμοποιούνται για την επαλήθευση του αλγορίθμου (testing set). Ένα training set είναι ένα σύνολο δεδομένων που χρησιμοποιείται για να ανακαλύψει πιθανές σχεσιακές προβλέψεις. Ένα test set είναι ένα σύνολο δεδομένων που χρησιμοποιείται για να γίνει η επαλήθευση

των προβλέψεων που έχουν γίνει χρησιμοποιώντας το training set.

Το πρωτόκολλο αξιολόγησης που χρησιμοποιήσαμε για την αξιολόγηση είναι το “όλα-εκτός-από-30%” (*“all-but-30%”*). Τα δεδομένα του κάθε χρήστη χωρίστηκαν σε training και testing set, χωρίζοντας τον χρόνο στο 30%. Έτσι, ως training set έχουμε όλες τις κριτικές που έχουν γίνει τους μήνες Ιανουάριο με Σεπτέμβριο της κάθε χρονιάς, και ως testing set το τελευταίο τρίμηνο του έτους.

Για την αξιολόγηση χρησιμοποιούμε τον λόγο ευστοχίας (Hit Rate) [5]. Όταν κάνουμε προτάσεις, φτιάχνουμε μία λίστα από N (N=10) αντικείμενα για κάθε έναν από τους χρήστες που έχουμε επιλέξει. Αν η επιχείρηση που ανήκει στο test set, υπάρχει στην λίστα, τότε το μετράμε σαν ευστοχία (hit). Υπολογίζουμε δύο τιμές για τον λόγο ευστοχίας:

1. Να έχει βρει και τις 10 επιχειρήσεις στην λίστα
2. Να έχει βρει τουλάχιστον μία επιχείρηση στην λίστα

Έστω ότι n το σύνολο των χρηστών. Ο βαθμός ευστοχίας υπολογίζεται με τον ακόλουθο τρόπο:

$$\textit{Hit Rate}\ (HR) = \frac{1}{n}\sum_{n}(T_u \in R(u)) \tag{6}$$

όπου $I()$ η συνάρτηση που δείχνει αν υπήρξε ευστοχία, $R(u)$ είναι η λίστα με τα top-N αντικείμενα που προτάθηκαν στον χρήση u, και T_u είναι το αντικείμενο που έχει δει ο χρήστης και βρίσκεται στο test set.

Αν το HR ισούται με 1.0 τότε αυτό σημαίνει πως ο αλγόριθμος ήταν πάντα σε θέση να προτείνει όλα τα αντικείμενα στη λίστα, ενώ αν ισούται με 0.0 τότε αυτό είναι δείγμα πως ο αλγόριθμος δεν κατάφερε ποτέ να βρει κανένα αντικείμενο.

Ένας περιορισμός αυτού του μέτρου είναι ότι χειρίζεται το ίδιο όλες τις ευστοχίες, ανεξαρτήτως της θέσης όπου εμφανίζονται στη λίστα με τις top-N προτάσεις. Δηλαδή, αν ένα αντικείμενο βρίσκεται στην κορυφή της λίστας αντιμετωπίζεται το ίδιο με ένα αντικείμενο που βρίσκεται στη θέση N της λίστας. Αυτός ο περιορισμός αντιμετωπίζεται από τον μέσο αμοιβαίο βαθμό ευστοχίας (Average Reciprocal Hit-Rank - ARHR) [6], που επιβραβεύει κάθε ευστοχία με βάση τη θέση της στη λίστα. Αν h είναι ο αριθμός των ευστοχιών στις θέσεις $p_1, p_2, \cdots, p_h$, μέση στην top-N λίστα (όπου $1 \leq p_i \leq N$), τότε ο μέσος αμοιβαίος βαθμός ευστοχίας υπολογίζεται ως εξής:

$$\textit{Average Reciprocal HitRank}\ (ARHR) = \frac{1}{n}\sum_{i=1} h\frac{1}{p_i}, \quad 1 \leq p_i \leq N \tag{7}$$

Με αυτόν τον τρόπο, οι ευστοχίες που βρίσκονται σε υψηλότερες θέσεις στην top-N λίστα έχουν μεγαλύτερο βάρος από αυτές που βρίσκονται σε χαμηλότερες θέσεις. Η μέγιστη τιμή του ARHR είναι ίση με τον βαθμό ευστοχίας. Συμβαίνει όταν όλες οι ευστοχίες βρί-

σκονται στην πρώτη θέση, ενώ η ελάχιστη τιμή του ισούται με τον *βαθμό ευστοχίας*/*N*, που συμβαίνει όταν όλες οι ευστοχίες βρίσκονται στην τελευταία θέση της λίστας.

4.3 Αποτελέσματα

Τα αποτελέσματα των πειραμάτων παρουσιάζονται ανά χρονιά στους πίνακες 4, 5, 6. Το *HR* (Hit Rate) αντιπροσωπεύει το ποσοστό ευστοχίας στις προβλέψεις των δύο μεθόδων και το *ARHR* (Average Reciprocal HitRank) αξιολογεί την θέση των συστάσεων στη λίστα.

Αξιολόγηση	**STG**	**PCA**
HR (και τις 10 επιχειρήσεις)	0.09	0.03
HR (τουλάχιστον 1)	0.55	0.22
AVHR	0.09	0.01

Πίνακας 4: Αποτελέσματα για το 2010

Αξιολόγηση	**STG**	**PCA**
HR (και τις 10 επιχειρήσεις)	0.09	0.03
HR (τουλάχιστον 1)	0.43	0.23
AVHR	0.09	0.03

Πίνακας 5: Αποτελέσματα για το 2011

Αξιολόγηση	**STG**	**PCA**
HR (και τις 10 επιχειρήσεις)	0.25	0.016
HR (τουλάχιστον 1)	0.53	0.12
AVHR	0.09	0.02

Πίνακας 6: Αποτελέσματα για το 2012

Οι χρονιές 2010 και 2011 παρατηρούμε ότι παρουσιάζουν αρκετές ομοιότητες. Η μέθοδος του STG παρουσιάζει μία μικρή μείωση όσον αφορά το ποσοστό ευστοχίας τουλάχιστον μίας επιχείρησης. Ωστόσο, τόσο το ποσοστό ευστοχίας και για τις 10 επιχειρήσεις αλλά και το ARHR παραμένουν σταθερά. Παρόμοια, η μέθοδος του PCA παραμένει σχεδόν σταθερή, με μία σχεδόν αμελητέα αύξηση στο ποσοστό του ARHR.

Το 2012 όμως, παρουσιάζει μεγάλη διαφορά στις δύο μεθόδους. Αρχικά, στη μέθοδο του STG, παρατηρούμε πως το ποσοστό ευστοχίας για τουλάχιστον μία επιχείριση παραμένει σχεδόν σταθερό. Υπάρχει ωστόσο μεγάλη αύξηση, της τάξης του 117 % στο συνολικό ποσοστό ευστοχίας και των 10 επιχειρήσεων. Αντίθετα, η μέθοδος του PCA παρουσιάζει αρκετά φτωχά αποτελέσματα. Συγκεκριμένα, βλέπουμε ότι ο βαθμός ευστοχίας πέφτει σχεδόν στο μισό σε όλα τα κριτήρια αξιολόγησης.

Τέλος, από όλα τα αποτελέσματα, βλέπουμε πως η μέθοδος του STG υπερτερεί σε όλες τις περιπτώσεις της μεθόδου του PCA.

5. ΣΥΜΠΕΡΑΣΜΑΤΑ

Σε αυτή την πτυχιακή εργασία χρησιμοποιήσαμε δύο διαφορετικές προσεγγίσεις για να συγκρίνουμε την επιρροή του χρόνου στα συστήματα συστάσεων.

Η πρώτη προσέγγιση αφορά στην μοντελοποίηση της επιρροής των βραχυπρόθεσμων και μακροπρόθεσμων προτιμήσεων του χρήστη. Αυτό έγινε με την χρήση του Χρονικού Γράφου (Session-based Temporal Graph - STG), όπου ο χρόνος μοντελοποιείται σε χρονικές περιόδους διαστήματος μίας εβδομάδας. Με αυτόν τον τρόπο μπορούν να απεικονιστούν ταυτόχρονα τόσο οι μακροπρόθεσμες όσο και οι βραχυπρόθεσμες προτιμήσεις του χρήστη. Με βάση αυτόν τον γράφο, χρησιμοποιήσαμε τον αλγόριθμο Injected Preference Fusion (IPF) για την πραγματοποίηση χρονικών συστάσεων. Ο αλγόριθμος αυτός εξισορροπεί την επιρροή που θα έχουν τα δύο είδη των προτιμήσεων του χρήστη (βραχυπρόθεσμες - μακροπρόθεσμες), στην λίστα συστάσεων με τις κορυφαίες-N επιχειρήσεις.

Η δεύτερη προσέγγιση αφορά ένα υβριδικό μοντέλο συνεργατικών συστάσεων (collaborative filtering), που βασίζεται στην επιλογή των κύριων χαρακτηριστικών των χρηστών με τη χρήση της μεθόδου Ανάλυσης Κύριων Συνιστωσών (Principal Component Analysis - PCA). Η προσέγγιση την αυτή συνδυάζει την τεχνική για την μείωση των διαστάσεων (με το PCA), με τον αλγόριθμο ιεραρχικής ομαδοποίησης (hierarchical clustering). Στα αραιά σύνολα δεδομένων, η επιλογή χρηστών που είναι παρόμοιοι με τον χρήστη με βάση τις κοινές αξιολογήσεις, είναι ζωτικός παράγοντας για την παραγωγή συστάσεων υψηλής ποιότητας. Στη προσέγγιση που χρησιμοποιήσαμε, η ανάλυση των χαρακτηριστικών έγινε με το PCA στο σύνολο του πίνακα Χ*ρήστες* $\times$ Α*ντικείμενα*. Στη συνέχεια οι ομάδες (clusters) παρήχθησαν από τον μετασχηματισμό διανυσματικού χώρου σε χαμηλότερη διάσταση, όπως προέκυψε από το πρώτο βήμα. Με αυτόν τον τρόπο, ο αρχικός χώρος των χρηστών, που γίνεται πιο πυκνός και αξιόπιστος, χρησιμοποιείται για την επιλογή γειτόνων, αντί να ψάχνουμε σε ολόκληρο τον αρχικό χώρο. Επιπλέον, με την χρήση του ιεραρχικού αλγορίθμου, μπορούμε να βρούμε τους κοντινότερους και ποιο όμοιους γείτονες του χρήστη που επιθυμούμε να κάνουμε την πρόταση.

Τα πειράματα που έγιναν με τη χρήση του πραγματικού συνόλου δεδομένων του Yelp, αποδεικνύουν ότι **η πιο αποτελεσματική προσέγγιση για τη δημιουργία συστάσεων είναι η πρώτη, με τη χρήση του γράφου STG.** Με βάση τα αποτελέσματα, η μέθοδος αυτή απέδειξε πως είναι ικανή να παρέχει πολύ υψηλή ακρίβεια και μεγαλύτερη αξιοπιστία κατά την δημιουργία συστάσεων, σε σχέση με τη δεύτερη μέθοδο. Έτσι επιβεβαιώνεται ο καθοριστικός ρόλος που έχει η μοντελοποίηση του χρόνου και η χρήση του κατά τη διάρκεια της εξαγωγής των συμπερασμάτων.

Αυτό που προέκυψε επιπλέον από αυτή την ανάλυση, είναι η χαμηλή αποτελεσματικότητα της δεύτερης προσέγγισης, συγκριτικά με την πρώτη. Ένας από τους βασικούς παράγοντες είναι η έλλειψη του χρόνου ως παράμετρος. Επιπλέον, κατά την ανάλυση και επεξεργασία του πίνακα Χ*ρήστες* $\times$ Α*ντικείμενα*, φαίνεται πως η μέθοδος του PCA δεν μπορεί να εντοπίσει σωστά τους γείτονες των χρηστών και τις χρονικές εξαρτήσεις. Αυτό

έχει σαν αποτέλεσμα όμοιοι γείτονες να είναι αρκετά διασκορπισμένοι, γεγονός που οδηγεί σε λανθασμένη εκτίμηση της πρόβλεψης λίστας αντικειμένων προς σύσταση.

Μελλοντικές μελέτες μπορούν να αφορούν τον συνδυασμό των δύο αυτών μεθόδων, καθώς η τεχνική της ομαδοποίησης επιφέρει μεγάλη βελτίωση στην απόδοση, όσο αφορά τον χρόνο που απαιτείται για την δημιουργία μίας λίστας συστάσεων. Επιπλέον, για μεγαλύτερη ακρίβεια και εξατομίκευση κατά τη δημιουργία συστάσεων, μπορούν να προστεθούν άλλα χαρακτηριστικά των χρηστών, όπως το περιεχόμενο (context), οι ετικέτες (tags) που έχουν τα αντικείμενα που επιλέγουν, αλλά και το δίκτυο από χρήστες που εμπιστεύονται.

ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ

συστήματα προτάσεων / συστάσεων	recommender systems
χρονικές συστάσεις	temporal recommendations
συνεργατικό φιλτράρισμα	collaborative filtering
ομαδοποίηση	clustering
ανάλυση κύριων συνιστωσών	principal component analysis
λόγος ευστοχίας	hit rate
χρονική περίοδος	session
δεδομένα που χρησιμοποιούνται για την εκπαίδευση του αλγορίθμου	training set
δεδομένα που χρησιμοποιούνται για τον έλεγχο του αλγορίθμου	testing set
συνάρτηση βασικής πρόβλεψης	baseline predictor
ιεραρχική ομαδοποίηση	hierarchical clustering

ΣΥΝΤΜΗΣΕΙΣ, ΑΡΚΤΙΚΟΛΕΞΑ ΚΑΙ ΑΚΡΩΝΥΜΙΑ

Ακολουθεί δείγμα συντμήσεων, αρκτικόλεξων και ακρωνυμίων.

IPF	Injected Preference Fusion
STG	Session-based Temporal Graph
PCA	Principal Component Analysis
CF	Collaborative Filtering
HR	Hit Rate

ΑΝΑΦΟΡΕΣ

[1] Liang Xiang, Quan Yuan, Shiwan Zhao, Li Chen, Xiatian Zhang, Qing Yang and Jimeng Sun *Temporal Recommendation on Graphs via Long- and Short-term Preference Fusion*

[2] Zan Wang, Xue Yu, Zhenhua Wang *An Improved Collaborative Movie Recommendation System using Computational Intelligence* Available: http://ksiresearchorg.ipage.com/seke/dms14paper/paper18.pdf

[3] “Yelp dataset challenge,” Availabe: https://www.yelp.com/dataset_challenge [Accessed: Mar 12, 2016].

[4] Kritika Singh, *Predicting user rating for Yelp businesses leveraging user similarity* Available: http://cseweb.ucsd.edu/~jmcauley/cse255/reports/wi15/Kritika_Singh.pdf

[5] George Karypis, *Evaluation of item-based top-n recommendation algorithms*

[6] Amukund Deshpande, George Karypis, *Item-Based Top-N Recommendation Algorithms,*. Available: http://glaros.dtc.umn.edu/gkhome/fetch/papers/itemrsTOIS04.pdf

[7] Miha Grcar, Dunja Mladenic, Blaz Fortuna, and Marko Grobelnik *Data Sparsity Issues in the Collaborative Filtering Framework*, Jozef Stefan Institute, Ljubljana, Slovenia, Available : http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.106.6112&rep=rep1&type=pdf