ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Ένα μοντέλο εξατομικευμένων ταξιδιωτικών συστάσεων με χρήση αλγορίθμων μηχανικής μάθησης

Διπλωματική Εργασία

Αντωνιάδης Γεώργιος

Επιβλέπουσα: Τουσίδου Ελένη

Βόλος 2021

UNIVERSITY OF THESSALY

SCHOOL OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

A Personalized Travel Recommender Model using machine learning algorithms

Diploma Thesis

Antoniadis Giorgos

Supervisor: Tousidou Eleni

Volos 2021

Εγκρίνεται από την Επιτροπή Εξέτασης:

Επιβλέπουσα **Τουσίδου Ελένη**

Εργαστηριακό Διδακτικό Προσωπικό, Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, Πανεπιστήμιο Θεσσαλίας

Μέλος **Βασιλακόπουλος Μιχαήλ**

Αναπληρωτής Καθηγητής, Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, Πανεπιστήμιο Θεσσαλίας

Μέλος **Τσομπανοπούλου Παναγιώτα**

Αναπληρώτρια Καθηγήτρια, Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, Πανεπιστήμιο Θεσσαλίας

Ημερομηνία έγκρισης: 06-07-2021

ΕΥΧΑΡΙΣΤΙΕΣ

Αρχικά θέλω να ευχαριστήσω την επιβλέπουσα μου, κα. Ελένη Τουσίδου για την πολύτιμη καθοδήγησή της στην ανάπτυξη της διπλωματικής εργασίας. Επιπλέον θέλω να ευχαριστήσω τον κ. Μιχαήλ Βασιλακόπουλο και την κα. Παναγιώτα Τσοπανοπούλου για την συμμετοχή τους στην εξεταστική μου επιτροπή.

ΥΠΕΥΘΥΝΗ ΔΗΛΩΣΗ ΠΕΡΙ ΑΚΑΔΗΜΑΪΚΗΣ ΔΕΟΝΤΟΛΟΓΙΑΣ ΚΑΙ ΠΝΕΥΜΑΤΙΚΩΝ ΔΙΚΑΙΩΜΑΤΩΝ

Με πλήρη επίγνωση των συνεπειών του νόμου περί πνευματικών δικαιωμάτων, δηλώνω ρητά ότι η παρούσα διπλωματική εργασία, καθώς και τα ηλεκτρονικά αρχεία και πηγαίοι κώδικες που αναπτύχθηκαν ή τροποποιήθηκαν στα πλαίσια αυτής της εργασίας, αποτελεί αποκλειστικά προϊόν προσωπικής μου εργασίας, δεν προσβάλλει κάθε μορφής δικαιώματα διανοητικής ιδιοκτησίας, προσωπικότητας και προσωπικών δεδομένων τρίτων, δεν περιέχει έργα/εισφορές τρίτων για τα οποία απαιτείται άδεια των δημιουργών/δικαιούχων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον και πληρούν τους κανόνες της επιστημονικής παράθεσης. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο, αρχεία ή/και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή. Αναλαμβάνω πλήρως, ατομικά και προσωπικά, όλες τις νομικές και διοικητικές συνέπειες που δύναται να προκύψουν στην περίπτωση κατά την οποία αποδειχθεί, διαχρονικά, ότι η εργασία αυτή ή τμήμα της δεν μου ανήκει διότι είναι προϊόν λογοκλοπής.

Ο Δηλών

Αντωνιάδης Γεώργιος
06/07/2021

ΠΕΡΙΛΗΨΗ

Η επιλογή ενός προορισμού και των επιμέρους δραστηριοτήτων που συγκροτούν μία ταξιδιωτική εμπειρία, είναι οι δύσκολες αποφάσεις που πρέπει να πάρει ο τουρίστας πριν και κατά την διάρκεια του ταξιδιού του. Αυτή η διαδικασία μπορεί να γίνει πιο εύκολη και πολλές φορές πιο αποδοτική με την χρήση των Συστημάτων Συστάσεων (ΣΣ), τα οποία έχουν μπει στην καθημερινότητα με πολλές μορφές στην ψυχαγωγία, στις αγορές κτλ. Με την χρήση τους, ο τουρίστας είναι σε θέση να συγκρίνει και να διαλέξει από μία έτοιμη λίστα επιλογών που του προσφέρεται.

Σκοπός της εργασίας είναι να αναπτύξει ένα Εξατομικευμένο ΣΣ για τον χώρο του τουρισμού στην Ελλάδα, κάνοντας χρήση τεχνικών μηχανικής μάθησης. Για την κατασκευή του έγινε μελέτη σε προηγούμενες έρευνες και μελέτες πάνω στα συστήματα συστάσεων.

Επιπλέον η έρευνα προτείνει ένα υβριδικό μοντέλο, το οποίο αντιμετωπίζει ένα από τα μεγαλύτερα προβλήματα που εμφανίζονται στα ΣΣ, το πρόβλημα της ψυχρής εκκίνησης (*cold start problem*). Χρησιμοποιείται μία λύση δύο βημάτων, η οποία κάνει χρήση ορισμένων προτιμήσεων που συλλέγονται από τον χρήστη, ώστε να μπορέσει να κάνει ακριβείς συστάσεις. Το σύστημα έχει κατασκευαστεί με πραγματικά δεδομένα, τα οποία έχουν συλλεχθεί για να επιτευχθεί το καλύτερο δυνατό αποτέλεσμα.

Τέλος παρουσιάζεται ένα γραφικό περιβάλλον, το οποίο επιτρέπει στον χρήστη να αλληλεπιδράσει με τις προτάσεις που του γίνονται. Το γραφικό περιβάλλον προσφέρει στον χρήστη πληροφορίες για τις διαθέσιμες επιλογές, βελτιώνοντας την εμπειρία του.

ABSTRACT

Choosing a travel destination, as well as a number of activities that make up for a complete traveling experience, are some of the most crucial decisions a tourist must make, before or during a trip. This decision can become easier and some times more effective with the use of recommendation systems (RS). While using them, tourists are capable of comparing and choosing from a list of suggestions provided by the system.

The purpose of this project is the development of a Personalized Recommendation System for the tourism domain in Greece, using machine learning algorithms and techniques. The development of this project is based on a number of relevant studies on the subject of recommendation systems.

The project presents a hybrid model that is capable of resolving a major problem on RS, the cold start problem. A two-step solution is used, that takes into account user preferences, in order to make accurate recommendations. The system is built with real life data that were collected in order to achieve the best possible results.

Finally a Graphical User Interface (GUI) is also presented, in order to allow user interaction with the suggestions that are available. Information about the final options are presented to make user experience friendlier.

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΛΙΣΤΑ ΕΙΚΟΝΩΝ

ΛΙΣΤΑ ΠΙΝΑΚΩΝ

ΣΥΝΤΟΜΟΓΡΑΦΙΕΣ

ΣΣ	Σύστημα Συστάσεων
ΠΒΧ	Πίνακας Βαθμολογίας Χρηστών
ΠΠΑ	Πίνακας Περιεχομένων Αντικειμένων
ΔΑΔ	Διερευνητική Ανάλυση Δεδομένων
ΤΠΕ	Τεχνολογία Πληροφοριών και Επικοινωνίας

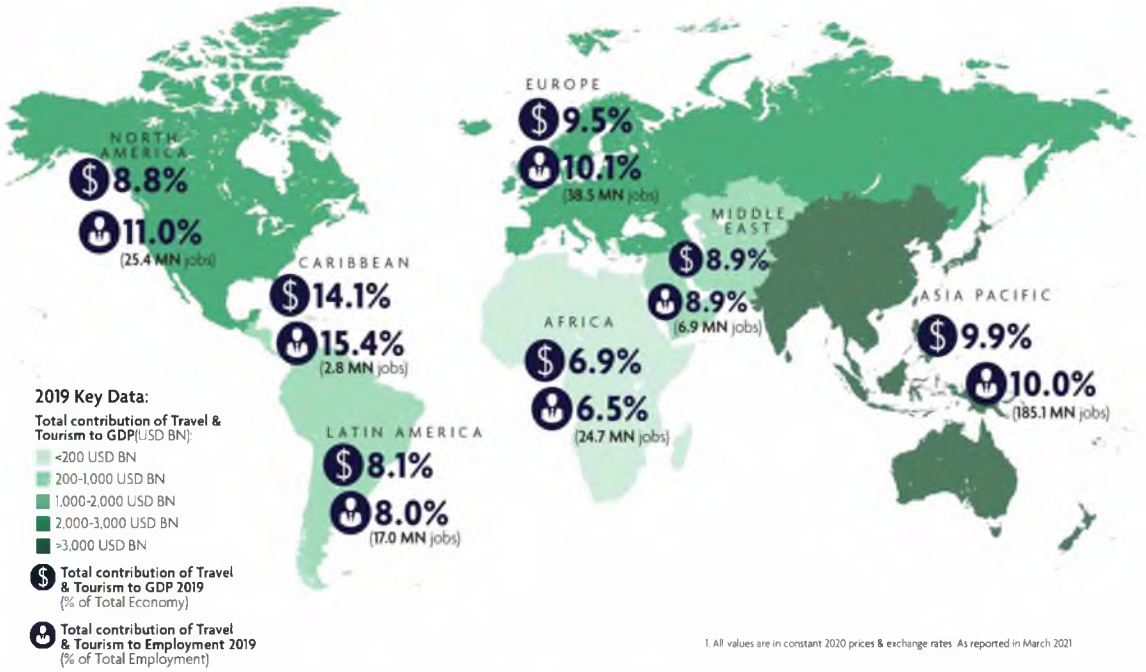
RS	Recommendation System
GUI	Graphical User Interface
RMSE	Root Mean Square Error
CBF	Content Based Filtering
CF	Collaborative Filtering
ICM	Item Content Matrix
URM	User Rating Matrix
MAE	Mean Absolute Error
MSE	Mean Squared Error
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
ROC	Receiver Operating Characteristic
AUC	Area Under the Curve
AP	Average Precision
MAP	Mean Average Precision
SVD	Singular Value Decomposition

API	Application Programming Interface
HTML	HyperText Markup Language
EDA	Exploratory Data Analysis
CSS	Cascading Style Sheets
JSON	JavaScript Object Notation
IP	Internet Protocol
RDBMS	Relational Database Management System
ACID	Atomicity, Consistency, Isolation, Durability

ΚΕΦΑΛΑΙΟ 1

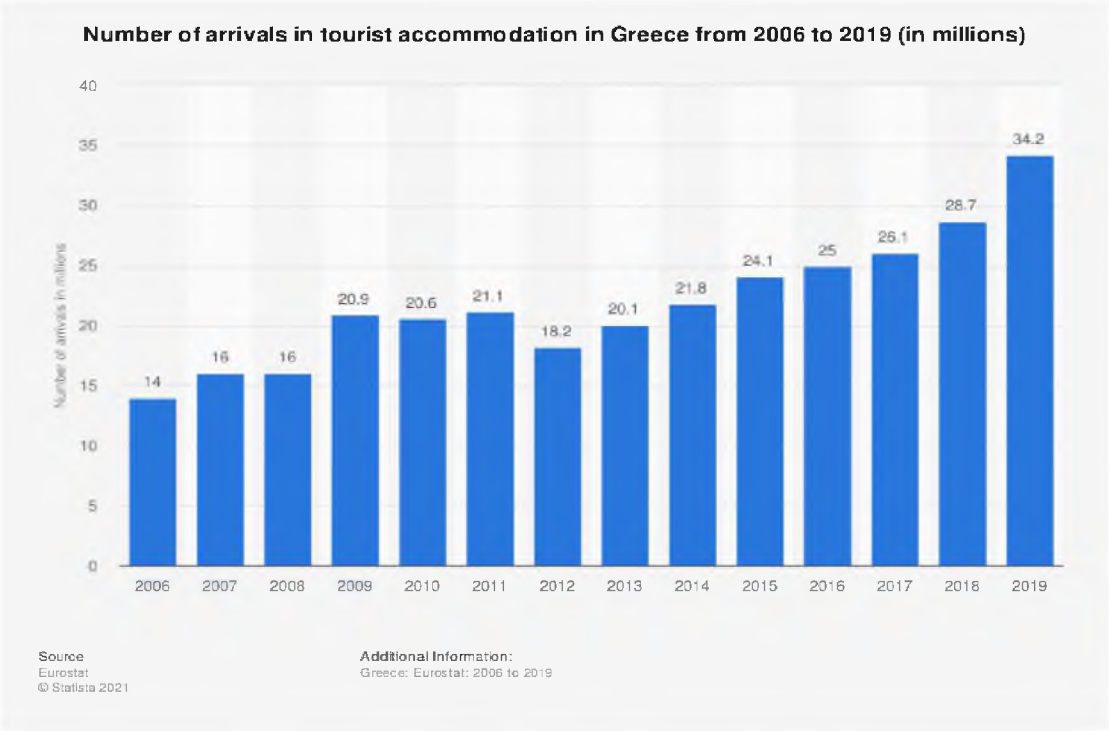
ΕΙΣΑΓΩΓΗ

Ο τουρισμός είναι πολύ σημαντικός σε παγκόσμιο επίπεδο, προσφέροντας 10% της παγκόσμιας οικονομίας (εικόνα 1) με εκτιμώμενη αύξηση 10.3% κατά μέσο όρο την επόμενη δεκαετία (World Travel and Tourism Council ,2019).

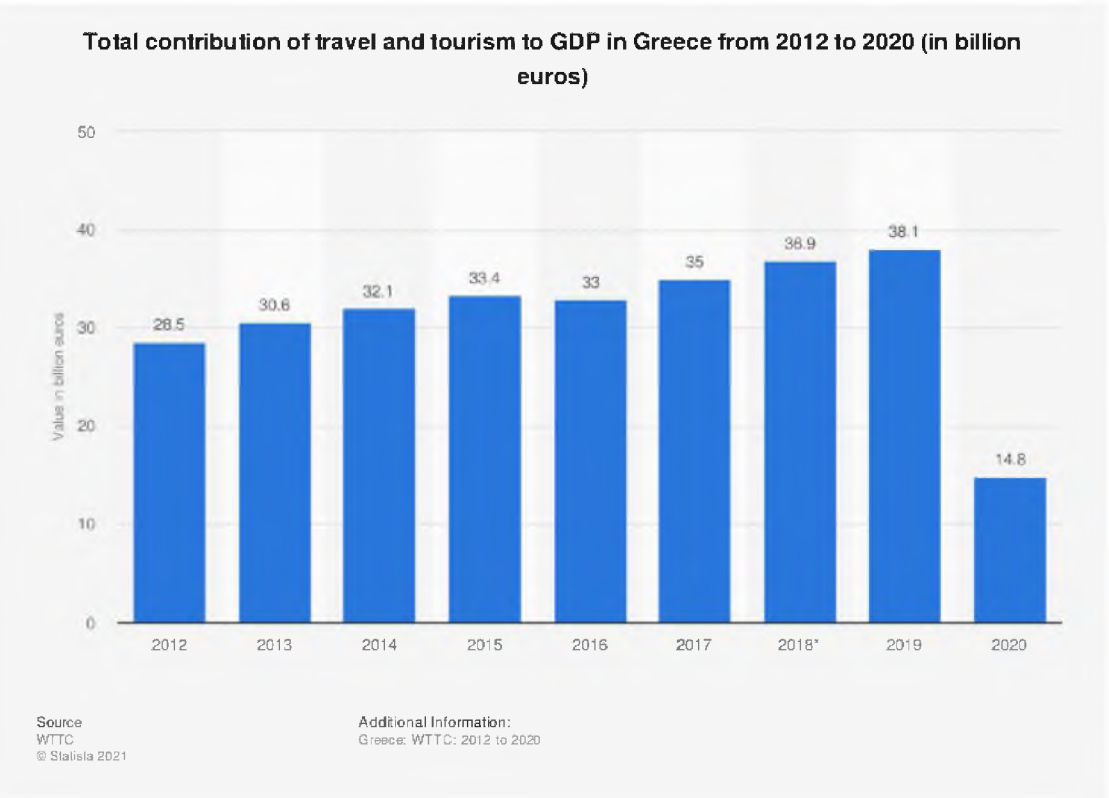


Εικόνα 1. Συνολική συμμετοχή τουρισμού στο ΑΕΠ παγκοσμίως. Πηγή: https://wttc.org/Research/Economic-Impact

Ο αριθμός των τουριστών παγκοσμίως αυξάνεται συνεχώς. Η Ελλάδα, κάθε χρόνο, αποτελεί τόπο έλξης για εκατομμύρια τουρίστες λόγω του κλίματος, της κουλτούρας και της φυσικής ομορφιάς που έχει να προσφέρει. Το 2018 απασχολήθηκαν στην βιομηχανία του τουρισμού 381.800 άτομα (OECD Tourism Trends and Policies 2020), αποτελώντας το 10% του συνολικού εργατικού δυναμικού. Το 2019 πάνω από 34 εκατομμύρια τουρίστες (εικόνα 2) επισκέφτηκαν την Ελλάδα προσφέροντας πάνω από 30 δις ευρώ έσοδα (εικόνα 3).



Εικόνα 2. Αριθμός τουριστών στην Ελλάδα από 2006-2019. Πηγή: https://www.statista.com/statistics/413222/number-of-arrivals-spent-in-short-stay-accommodation-in-greece/



Εικόνα 3. Συμμετοχή τουρισμού στο ΑΕΠ στην Ελλάδα 2012-2020. Πηγή: https://www.statista.com/statistics/644573/travel-tourism-total-gdp-contribution-greece/

Με τον τουρισμό σε άνθηση την τελευταία δεκαετία, οι πηγές πληροφοριών παίζουν μεγάλο ρόλο στην επιλογή του προορισμού, των επιμέρους δραστηριοτήτων και των παροχών που θα επιλέξουν οι ταξιδιώτες. Το Internet είναι πια η βασική πηγή πληροφορίας για προϊόντα και υπηρεσίες. Με τον όγκο της πληροφορίας που υπάρχει, η τελική επιλογή του προορισμού γίνεται πολλές φορές δύσκολη και χρονοβόρα, καθώς υπάρχουν πολλές διαθέσιμες επιλογές και διαφορετικοί παράγοντες που πρέπει να λάβει κανείς υπόψιν. Κάποιοι από τους βασικούς παράγοντες στον προγραμματισμό διακοπών είναι ο προϋπολογισμός, οι συνολικές μέρες, οι δραστηριότητες ο καιρός κτλ.

1.1 Η Ιδέα

Ο τουρισμός έχει επωφεληθεί από την τεχνολογία πληροφοριών και επικοινωνίας (ΤΠΕ) και πιο συγκεκριμένα από το Internet και τις εφαρμογές του. Εργαλεία για διευκόλυνση των αποφάσεων, όπως τα ΣΣ, ήρθαν να βοηθήσουν με την σειρά τους. Οι τουρίστες μπορούν να αναζητήσουν, να συγκρίνουν και να επιλέξουν πολύ πιο γρήγορα και αποδοτικά μέσω των επιλογών που τους προσφέρονται.

Μέσα στον τεράστιο όγκο διαφορετικών πληροφοριών που υπάρχουν διαθέσιμες, τα ΣΣ μπορούν να λειτουργήσουν σαν φίλτρα. Η επιλογή των χαρακτηριστικών και των υπηρεσιών που υπάρχουν για έναν άγνωστο προορισμό -οι οποίες πρέπει να συμφωνούν με τις προτιμήσεις του χρήστη- μπορούν να οδηγήσουν σε δύσκολες αποφάσεις. Ακόμα και με τις μηχανές αναζήτησης οι προτάσεις που μπορούν να προκύψουν είναι πολλές, δυσχεραίνοντας το έργο της επιλογής. Τα ΣΣ μέσω των αλγορίθμων τους μπορούν να μειώσουν σημαντικά τις επιλογές που θεωρούν βέλτιστες.

Τα περισσότερα ΣΣ που χρησιμοποιούνται στοχεύουν στο να προτείνουν προορισμούς, δραστηριότητες και υπηρεσίες (πχ. εστιατόρια, ξενοδοχεία, μετακίνηση) βάσει των προτιμήσεων των χρηστών, σε περιοχές που έχουν επιλέξει· από τεχνική σκοπιά αποσκοπούν μόνο στο να φιλτράρουν και να ταξινομήσουν. Για να προσφερθεί κάτι διαφορετικό στους χρήστες, πρέπει το σύστημα να είναι έξυπνο, ποιοτικό και ακριβές στις προτάσεις του. Αυτή η εργασία επικεντρώνεται στο να προτείνει προορισμούς με βάση ένα γενικότερο ολοκληρωμένο πλάνο σε τουρίστες, οι οποίοι έχουν σαν σκοπό να επισκεφτούν πολλά διαφορετικά μέρη.

Για να κατασκευαστεί ένα πετυχημένο ΣΣ, ικανοποιητικό σε πρακτικό και τεχνικό επίπεδο, πρέπει να αντιμετωπιστούν κάποιες δυσκολίες.

- Βελτίωση των επιλογών του χρήστη: Για να βελτιωθούν οι επιλογές που έχουν να κάνουν οι χρήστες, πρέπει να αναλυθεί η πηγή των προτιμήσεών τους. Αυτό επιτυγχάνεται μέσω ανάλυσης των δεδομένων που αφορούν βαθμολογίες οι οποίες έχουν δοθεί στο παρελθόν, σε σχέση με τα γνωρίσματα των αντικειμένων.
- Βελτίωση της απόδοσης του ΣΣ: Πολλά ΣΣ αξιολογούν μόνο με βάση έναν μετρητή ακρίβειας, ενώ σε πολλές περιπτώσεις δεν κάνουν καθόλου αξιολόγηση [44]. Σε αυτή την εργασία το σύστημά βελτιστοποιείται μέσω διαφορετικών μεθόδων αξιολόγησης, όπως οι μετρητές ακρίβειας επιλογών (*hit rate*) και το μέσο τετραγωνικό σφάλμα (*RMSE*), αφού τα δεδομένα έχουν καθαριστεί. Για να βελτιωθεί η ποιότητα του ΣΣ ρυθμίζονται οι υπερπαράμετροι των αλγορίθμων μέσω δοκιμών.
- Βελτίωση της εμπειρίας του χρήστη: Μία άλλη δοκιμασία που πρέπει να φέρουμε εις πέρας είναι η βελτίωση της εμπειρίας χρήστη. Για να επιτευχθεί αυτό, δημιουργήθηκε ένα φιλικό προς τον χρήστη περιβάλλον, που προσφέρει τις απαραίτητες πληροφορίες για το προτεινόμενο πρόγραμμα διακοπών.

Η εργασία προτείνει ένα ΣΣ που λαμβάνει υπόψιν τις παραπάνω δυσκολίες. Το σύστημά της μελέτης τρέχει εκτός σύνδεσης και περιλαμβάνει αρχικά συλλογή των δεδομένων, επεξεργασία και ανάλυση τους, όπως επίσης εκτίμηση και παρουσίαση των συστάσεων.

1.2 Στόχος

Στόχος της εργασίας είναι η κατασκευή ενός μοντέλου ΣΣ Διακοπών, το οποίο θα καθοδηγεί τον χρήστη πριν το ταξίδι στην Ελλάδα και θα προσφέρει ένα σύνολο προτάσεων που θα ικανοποιούν του ενδιαφέροντά του.

Προκειμένου να επιτευχθεί τέθηκαν οι ακόλουθοι στόχοι :

1. Ανάγνωση και κατανόηση της θεωρίας πίσω από τις τεχνικές και τους αλγορίθμους που χρησιμοποιούνται στα ΣΣ. (κεφ. 2)

2. Κατασκευή υποπρογράμματος για την συλλογή (scrape) των δεδομένων που χρησιμοποιούνται από τον ιστότοπο του Tripadvisor. (κεφ. 3)
3. Ανάλυση και προεπεξεργασία/καθαρισμός των δεδομένων. (κεφ. 3)
4. Εκπαίδευση του μοντέλου και κατασκευή του ΣΣ. (κεφ. 4)
5. Αξιολόγηση του μοντέλου. (κεφ. 4)
6. Κατασκευή γραφικού περιβάλλοντος, φιλικού προς τον χρήστη. (κεφ5)

Η διαδικασία που ακολουθήθηκε αναπαρίσταται στην εικόνα 4.

Εικόνα 4. Ροή διαδικασίας εργασίας

Ορισμένα από τα ΣΣ μπορούν να αντιμετωπίσουν τεχνικά ζητήματα (όπως η αξιολόγηση του συστήματος) και πρακτικά ζητήματα (όπως η ευκολία χρήσης και η αποδοχή του χρήστη όσον αφορά την απόδοση σε πραγματικές συνθήκες) που πρέπει να λάβουν υπόψιν. Για να αντιμετωπιστούν ανάλογα ζητήματα οι ακόλουθες ερωτήσεις πρέπει να απαντηθούν:

1. Πώς θα αποκτηθούν δεδομένα πραγματικών συνθηκών;
2. Πώς θα αντιμετωπιστεί το πρόβλημα που δημιουργείται για τους νέους χρήστες;
3. Ποια γνωρίσματα αντικειμένων έχουν βασικό ρόλο στην επιλογή των προτάσεων που θα παρουσιαστούν στους χρήστες;
4. Πώς θα αξιολογηθεί το σύστημά μας;
5. Πώς θα μειωθεί ο χρόνος που απαιτείται για την κατασκευή του μοντέλου;
6. Πώς θα επιτευχθεί η προβολή των αντικειμένων και η αλληλεπίδραση του χρήστη, με σκοπό μια πιο ευχάριστη εμπειρία;

1.3 Υπάρχουσες έρευνες/υλοποιήσεις για Συστήματα Συστάσεων στον τουρισμό

Έχουν μελετηθεί πολλά συστήματα συστάσεων για τουρισμό παγκοσμίως. Κάποιες από τις μελέτες αυτές αναφέρονται κάτω.

Οι Y. Huang και L. Bian [24] προσφέρουν ένα ΣΣ το οποίο προτείνει μια σειρά από δραστηριότητες στον χρήστη στην επιλεγμένη τοποθεσία. Το σύστημα λαμβάνει υπόψιν τόσο την συμπεριφορά του χρήστη, όσο και άλλων χρηστών, χρησιμοποιώντας, τεχνικές CBF και CF. Η ικανότητα να προβλέψει τις πιθανές δραστηριότητες για τον κάθε χρήστη, με χρήση μηχανικής μάθησης και *δικτύων Bayesian* ήταν καινοτόμα προσέγγιση. Επίσης η δυνατότητα να ταξινομήσει τις προτάσεις μέσω κόστους και απόστασης ήταν ενδιαφέρουσα.

Οι D. Yeh και C. Cheng [25] προτείνουν ένα βασισμένο στην γνώση ΣΣ (*Knowledge-Based*) για δραστηριότητες στην Ταϊβάν. Το σύστημα χρησιμοποιεί γνώση από ειδικούς στον τομέα του τουρισμού. Σκοπός είναι να προτείνει στον χρήστη δραστηριότητες μέσω κάποιων επιλογών που πρέπει να κάνει όπως, αγαπημένες κατηγορίες δραστηριοτήτων ταξιδιού (μουσεία, φύση κτλ.). Η πρόκληση της συγκεκριμένης έρευνας αφορά την αύξηση της απόδοσης του συστήματος μειώνοντας το ποσοστό των αραιών δεδομένων χρησιμοποιώντας καινοτόμες μεθόδους.

Οι L. Ardissono, A. Goy και G.Petrone [26] στην εργασία τους προσφέρουν μια web based και για φορητές συσκευές πλατφόρμα, για το Τορίνο της Ιταλίας. Το ΣΣ τους προτείνει στον χρήστη μέρη να επισκεφτεί όπως αξιοθέατα λαμβάνοντας υπόψιν δεδομένα με βάση την παρέα που έχει μαζί του, για παράδειγμα οικογένειες, ηλικιωμένους, παιδιά κτλ. Το ΣΣ ζητάει από τον χρήστη ως είσοδο τις μέρες που θα μείνει, μέρα αναχώρησης/επιστροφής , αρχικό και τελικό προορισμό όπως και επιθυμητή διάρκεια. Το σύστημα βασίζεται πολύ σε *τεχνικές με βάση τον χρήστη*. Τέλος δίνει την δυνατότητα προγραμματισμού του ταξιδιού τόσο στην αρχή όσο και κατά την διάρκεια του, κάτι αρκετά απαιτητικό στην υλοποίηση.

Οι L. Sebastia, I. Garcia, E.Onaindia, C.Alvarez [27] στην εργασία τους παρουσιάζουν μια εφαρμογή προγραμματισμού ταξιδιού που χρησιμοποιεί ένα υβριδικό σύστημα. Αρχικά στον χρήστη εμφανίζεται μια λίστα με πιθανούς προορισμούς η οποία εκτιμάται με βάση δημογραφικά στοιχεία του χρήστη, τις προηγούμενες επιλογές του όπως και προτιμήσεις για το συγκεκριμένο ταξίδι. Στην συνέχεια το σύστημα κατασκευάζει ένα πρόγραμμα χρησιμοποιώντας ορισμένους περιορισμούς που έχει δώσει ο χρήστης, προσφέροντας ένα αναλυτικό πλάνο με οργανωμένες προτάσεις.

Οι Z.Bahramian και R. Ali Abbaspour [28] στην εργασία τους παρουσιάζουν ένα μοντέλο CBF, που χρησιμοποιεί *οντολογική πληροφορία* (*ontological information*). Η δομή της

οντολογίας που προτείνουν αναπαριστά τόσο το προφίλ του χρήστη όπως επίσης και τα αντικείμενα. Χρησιμοποιείται για να υπολογίσει την ομοιότητα μεταξύ των προτιμήσεων του χρήστη και τα χαρακτηριστικά των αντικειμένων με σκοπό την παρουσίαση μίας λίστας εξατομικευμένων προτάσεων. Αυτή η έρευνα προτείνει ένα καινούργιο είδος CBF αλγορίθμου, υιοθετώντας την έννοια της *οντολογίας* (*ontology*), με σκοπό την βελτίωση του παραδοσιακού μοντέλου.

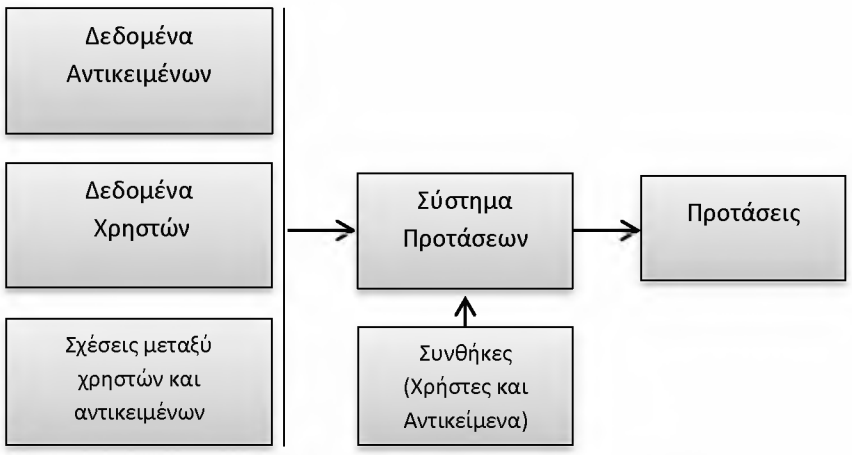
Τέλος οι H.Chiang και T. Huang [29] παρουσίασαν ένα βασισμένο σε web μορφή μοντέλο ΣΣ που προγραμματίζει ξενοδοχεία, εστιατόρια και δραστηριότητες, με βάση τις απαιτήσεις των χρηστών όπως συνολικές μέρες, προϋπολογισμό, σημείο έναρξης ταξιδιού, ώρες γευμάτων. Το σύστημα προτείνει αντικείμενα, βασισμένα απόλυτα πάνω στις προϋποθέσεις που έχουν θέσει οι χρήστες. Επίσης παρουσιάζεται ένας αλγόριθμος, για να λύσει το πρόβλημα της σειράς με την οποία προτείνονται τα αντικείμενα, με βάση την απόσταση μεταξύ τους.

ΚΕΦΑΛΑΙΟ 2

ΣΥΣΤΗΜΑΤΑ ΣΥΣΤΑΣΕΩΝ

2.1 Σχετικά με τα Συστήματα Συστάσεων

Ένα ΣΣ, είναι ένα σύστημα που αναλύει και φιλτράρει τα εισερχόμενα δεδομένα, με σκοπό να παρέχει προτάσεις στους χρήστες, σχετικά με αντικείμενα που μπορεί να τους ενδιαφέρουν (εικόνα 5).



Εικόνα 5. Βασική μορφή συστήματος συστάσεων

Πάνω σε αυτή την τεχνολογία βασίζονται πολλές εφαρμογές μεγάλων εταιριών όπως η Amazon, που με βάση το ιστορικό των αγορών που έχει πραγματοποιήσει ο χρήστης, προτείνει παρόμοια αντικείμενα και εφαρμογές ψυχαγωγίας όπως το Spotify, το YouTube και το Netflix.

Τον Σεπτέμβριο του 2009 έλαβε χώρα ένας διαγωνισμός του Netflix (*Netflix Prize*) με έπαθλο 1.000.000€, που είχε ως στόχο την υλοποίηση του καλύτερου δυνατού μοντέλου συνεργατικού φιλτραρίσματος με βάση τις βαθμολογίες των χρηστών. Εξαιτίας αυτού, η έρευνα πάνω στα ΣΣ επεκτάθηκε και οι αλγόριθμοι τους εξελίχθηκαν σημαντικά.

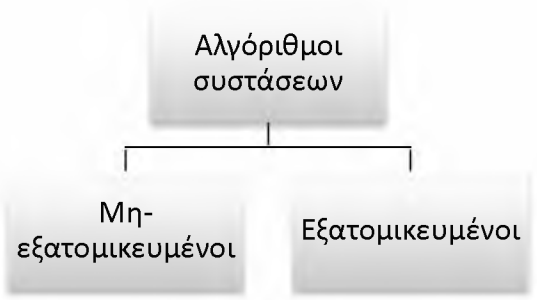
2.1.1 Εισερχόμενα Δεδομένα

Όπως σε όλα τα συστήματα, έτσι και στα ΣΣ βασικό παράγοντα συνιστούν τα εισερχόμενα δεδομένα. Οι πιο συχνές κατηγορίες είναι:

- Δεδομένα Αντικειμένων: Μια λίστα από τα διαθέσιμα αντικείμενα είναι η πρώτη βασική εισαγωγή σε κάθε αλγόριθμο προτάσεων. Η περιγραφή του κάθε αντικειμένου γίνεται με ένα σύνολο γνωρισμάτων. Για παράδειγμα σε ένα σύστημα που προτείνει ταινίες, το είδος, ο σκηνοθέτης και η χρονολογία μπορεί να είναι κάποια από αυτά. Στην προκειμένη περίπτωση, για μία πρόταση διαμονής κάποια πιθανά γνωρίσματα είναι η τοποθεσία, οι παροχές, η τιμή κτλ.
- Δεδομένα Χρηστών**:** Άλλη μια χρήσιμη πηγή πληροφορίας είναι τα γνωρίσματα των χρηστών όπως τα δημογραφικά χαρακτηριστικά (ηλικία, φύλο) και η τοποθεσία.
- Σχέσεις μεταξύ χρηστών και αντικειμένων**:** Οι σχέσεις των χρηστών με τα αντικείμενα μας επιτρέπουν να ερμηνεύσουμε την γνώμη τους για κάποια από αυτά. Για παράδειγμα, αν ένας χρήστης έχει βαθμολογήσει κάποια καταλύματα, άμεσα καταλαβαίνουμε την γνώμη του για αυτά, θετική η αρνητική. Εναλλακτικά, σε κάποιους αλγορίθμους χρησιμοποιείται έμμεση πληροφορία προκειμένου να σχηματίσουμε γνώμη, όπως για παράδειγμα σε μία αγορά από ένα ηλεκτρονικό κατάστημα. Σε αυτή την περίπτωση υποθέτουμε ότι ο χρήστης είναι πιθανό να ενδιαφέρεται για προϊόντα ανάλογα με αυτά που αγόρασε.
- Συνθήκες**:** Οι σχέσεις των χρηστών με τα αντικείμενα φέρουν επίσης κάποια χαρακτηριστικά. Η μέρα της εβδομάδας, η ώρα της ημέρας, η διάθεση του χρήστη ή ο καιρός μπορεί να είναι κάποια από αυτά. Ο ίδιος χρήστης μπορεί να έχει διαφορετική άποψη για ένα συγκεκριμένο αντικείμενο, ανάλογα με τις συνθήκες. Για παράδειγμα ένα εστιατόριο μπορεί να είναι ιδανικό για μία οικογενειακή συνάθροιση, όχι όμως για ένα ρομαντικό δείπνο. Το είδος της παρέας είναι ένα παράδειγμα συνθήκης όταν προτείνουμε ένα εστιατόριο.

2.2 Κατηγορίες ΣΣ

Οι αλγόριθμοι συστάσεων μπορούν αρχικά να χωριστούν σε δύο κατηγορίες. Τους μη-εξατομικευμένους και τους εξατομικευμένους (εικόνα 6) [1].



Εικόνα 6. Κατηγορίες συστημάτων συστάσεων

Με χρήση των μη-εξατομικευμένων αλγορίθμων, όλοι οι χρήστες θα λάβουν τις ίδιες συστάσεις άσχετα με τις προτιμήσεις τους. Παραδείγματα αποτελεσμάτων αυτής της κατηγορίας αλγορίθμων είναι: τα καλύτερα σε κριτικές καταλύματα, τα πιο δημοφιλή εστιατόρια κτλ.

Με τη χρήση εξατομικευμένων αλγορίθμων κάθε χρήστης θα λάβει διαφορετικές συστάσεις με βάση τις προτιμήσεις του [2]. Σκοπός τους είναι να κάνουν πιο ακριβείς συστάσεις σε σύγκριση με την προηγούμενη τεχνική.

2.2.1 Μη εξατομικευμένοι αλγόριθμοι συστάσεων

Υπάρχουν δύο βασικοί αλγόριθμοι προκειμένου να προτείνουμε τα ίδια αντικείμενα σε όλους τους χρήστες [3].

- Πιο δημοφιλή (*Most Popular*)
- Με την καλύτερη βαθμολογία (*Best Rated*)

Στην πρώτη περίπτωση, χρησιμοποιώντας τον Πίνακα Βαθμολογίας Χρηστών (ΠΒΧ), μετράται το σύνολο των μη-μηδενικών βαθμολογιών για κάθε αντικείμενο. Εδώ πρέπει να σημειωθεί, ότι δεν λαμβάνεται υπόψιν η γνώμη κάθε χρήστη για το αντικείμενο, αλλά

μόνο το αν έχει βαθμολογήσει ή όχι. Στη δεύτερη περίπτωση υπολογίζουμε τον μέσο όρο για κάθε αντικείμενο (1).

$$b_i = \frac{\sum_u r_{ui}}{N_i} \quad (1)$$

όπου r_{ui} : βαθμολογία του χρήστη *u* για το αντικείμενο *i,*

N_i : σύνολο χρηστών που βαθμολόγησαν το αντικείμενο *i*

Ο τύπος του μέσου όρου για το κάθε αντικείμενο *i* δεν λαμβάνει υπόψιν τον αριθμό των χρηστών που το έχουν βαθμολογήσει, κάτι το οποίο οδηγεί σε προκατειλημμένα αποτελέσματα. Ένα αντικείμενο που έχει βαθμολογηθεί θετικά από μόνο έναν χρήστη, ίσως υπερτερήσει ενός άλλου με πολύ περισσότερες εξίσου καλές κριτικές. Για να αποφύγουμε αυτό το φαινόμενο, χρησιμοποιούμε μία σταθερά συρρίκνωσης (*shrink term*) [4] με τιμή βάσει του ΠΒΧ (2). Μια τιμή που χρησιμοποιείται είναι 1.

$$b_i = \frac{\sum_u r_{ui}}{N_i + C} \quad (2)$$

όπου C : σταθερά συρρίκνωσης

2.2.2 Εξατομικευμένοι αλγόριθμοι συστάσεων

Οι εξατομικευμένοι αλγόριθμοι μπορούν να χωριστούν περαιτέρω σε υποκατηγορίες (εικόνα 7).

Εικόνα 7. Κατηγορίες εξατομικευμένων αλγορίθμων

Η ιδέα πίσω από το *Φιλτράρισμα με Βάση το Περιεχόμενο* (Content Based Filtering - CBF) είναι η πρόταση αντικειμένων ομοίων με εκείνα που άρεσαν στον χρήστη στο παρελθόν. Για παράδειγμα αν σε ένα χρήστη άρεσε μία δραστηριότητα που λάμβανε χώρα σε ένα

βουνό, το σύστημα θα πρότεινε κάποια αντίστοιχη, όπως μια πεζοπορία σε διαφορετική τοποθεσία, μία βόλτα με ποδήλατο βουνού κτλ.

Αυτή η τεχνική χρησιμοποιείται συχνά στο ηλεκτρονικό εμπόριο. Για παράδειγμα στη σελίδα ενός προϊόντος στην Amazon μπορεί να βρει κανείς προτάσεις για παρόμοια προϊόντα. Αυτή η τεχνική είναι συνήθως η πρώτη προσέγγιση στην κατασκευή ενός ΣΣ και προκειμένου αυτό να λειτουργήσει αποδοτικά, πρέπει να υπάρχει επαρκής ποσότητα γνωρισμάτων για κάθε αντικείμενο.

Στο *Συνεργατικό Φιλτράρισμα* (Collaborative Filtering - CF) δεν χρειάζονται πληροφορίες σχετικά με τα γνωρίσματα του προϊόντος καθώς αυτό βασίζεται στην γνώμη των χρηστών. Αυτή η κατηγορία μπορεί να χωριστεί περαιτέρω στις εξής υποκατηγορίες (εικόνα 8) [5].

Εικόνα 8. Κατηγορίες συνεργατικού φιλτραρίσματος

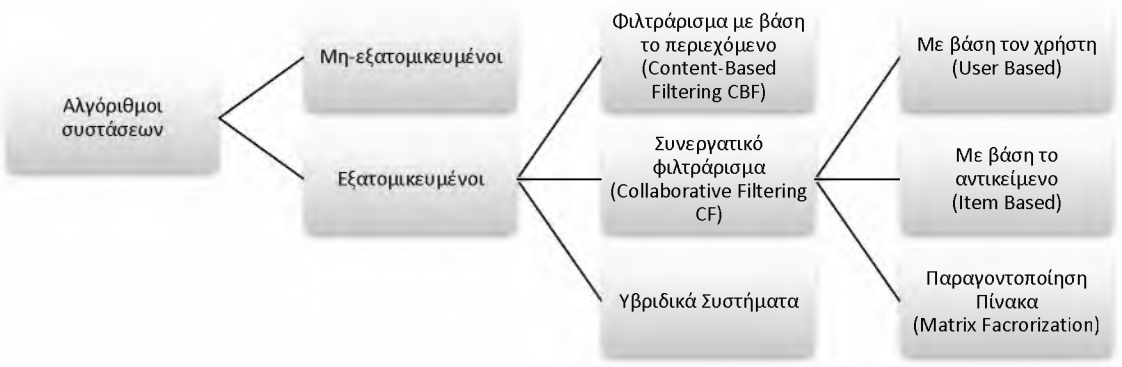
Με βάση τον χρήστη**:** Ο πρώτος τύπος CF βασίζεται στη σχέση χρήστη-χρήστη. Η βασική ιδέα είναι να ταιριάξει χρήστες με τα ίδια ενδιαφέροντα, που έχουν παρόμοια γνώμη για ένα σύνολο αντικειμένων. Αν δύο χρήστες u_1, u_2 έχουν ίδια γνώμη για ένα σύνολο αντικειμένων και στον χρήστη u_1 αρέσει το αντικείμενο i_1 τότε είναι πολύ πιθανό το ίδιο αντικείμενο να αρέσει και στον χρήστη u_2.

Με βάση το αντικείμενο**:** Ο δεύτερος τύπος CF βασίζεται στην σχέση αντικείμενο-αντικείμενο. Χρησιμοποιήθηκε πρώτη φορά από την Amazon το 1998. Η ιδέα είναι να υπολογίσει πόσο ισχυρή σχέση έχει κάθε ζευγάρι από αντικείμενα, με βάση το πόσοι χρήστες έχουν την ίδια γνώμη για αυτά. Για παράδειγμα, αν στους περισσότερους χρήστες που αρέσει το αντικείμενο i_1 αρέσει και το αντικείμενο i_2, τότε στον χρήστη u_1 που αρέσει το i_1 πολύ πιθανό να του αρέσει και το i_2.

Παραγοντοποίηση Πίνακα**:** Χάρη στον διαγωνισμό που διοργάνωσε το Netflix, μια νέα ομάδα αλγορίθμων CF δημιουργήθηκε, με βάση μια τεχνική που ονομάζεται

Παραγοντοποίηση Πινάκων. Αυτή η τεχνική είναι μέρος μιας ευρύτερης οικογένειας αλγορίθμων που βασίζεται στην μείωση διαστάσεων (*Dimensionality Reduction*) [6].

Οι μοντέρνοι αλγόριθμοι είναι σε θέση να χρησιμοποιούν ταυτόχρονα διαφορετικές πηγές εισερχόμενης πληροφορίας, ενώνοντας και βελτιώνοντας τα αποτελέσματα κάθε μίας από τις παραπάνω τεχνικές. Αυτοί οι αλγόριθμοι ονομάζονται Υβριδικοί**.** Στην γενική τους μορφή μπορούν να θεωρηθούν αλγόριθμοι CF στους οποίους τα αντικείμενα και οι χρήστες έχουν εμπλουτιστεί με επιπλέον πληροφορία [7]. Για παράδειγμα μπορούμε σε ένα ΣΣ εστιατορίων να λάβουμε υπόψιν την περιγραφή του εστιατορίου, τις παροχές του, τις κριτικές των πελατών και άλλα γνωρίσματά τους. Συνολικά ο διαχωρισμός των αλγορίθμων ενός ΣΣ σε κατηγορίες φαίνεται στην εικόνα 9.



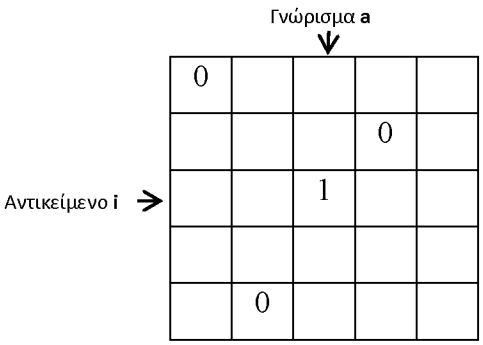
Εικόνα 9. Κατηγορίες αλγόριθμων συστήματος συστάσεων

2.3 Βασικοί πίνακες στα Συστήματα Συστάσεων

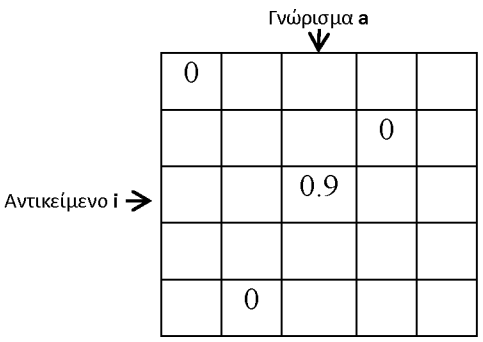
Αναφέρθηκε προηγουμένως ότι μία από τις βασικές κατηγορίες εισερχόμενων δεδομένων είναι τα δεδομένα αντικειμένων. Πρόκειται για μια λίστα με τα διαθέσιμα αντικείμενα και τα γνωρίσματά τους.

Ένας τρόπος να τα αναπαραστήσουμε είναι χρησιμοποιώντας τον Πίνακα Περιεχομένων Αντικειμένου (ΠΠΑ) [8]. Οι γραμμές του πίνακα αναπαριστούν τα αντικείμενα και οι στήλες τα γνωρίσματα. Στην πιο απλή μορφή του πίνακα τα στοιχεία είναι σε δυαδική μορφή, με το 1 να δηλώνει την ύπαρξη και το 0 τη μη ύπαρξη. Σε μια πολυπλοκότερη μορφή μπορούν να αναπαριστούν το βάρος (*weight*) κάθε γνωρίσματος για κάθε αντικείμενο.

Πίνακας 1. Δυαδικός ΠΠΑ



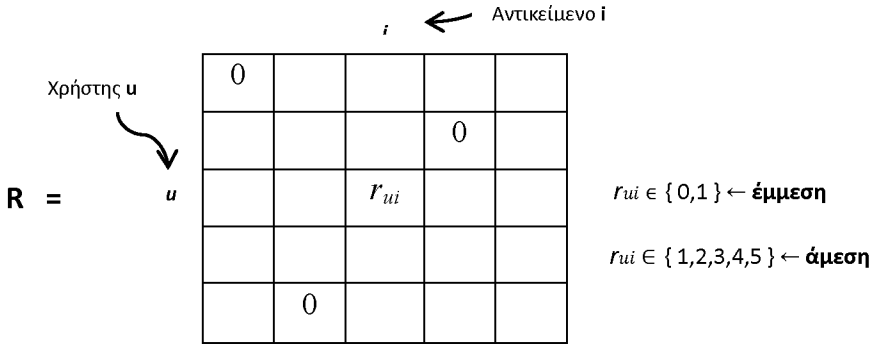
Πίνακας 2. ΠΠΑ Με Βάρη



Ως παράδειγμα, το αντικείμενο *i* αναπαριστά ένα εστιατόριο που έχει ως μία από τις βασικές κουζίνες την Ελληνική, η οποία αναπαρίσταται με το γνώρισμα a. Στον πίνακα 1, έχουμε την δυαδική μορφή, ενώ στον πίνακα 2 με χρήση βάρους.

Ένας ακόμα βασικός πίνακας είναι ο ΠΒΧ [9]. Ο συγκεκριμένος πίνακας χρησιμοποιείται για τη μαθηματική αναπαράσταση των δεδομένων χρηστών. Οι γραμμές του πίνακα αντιστοιχούν στους χρήστες και οι στήλες τα αντικείμενα (πίνακας 3). Τα περιεχόμενα του είναι βαθμολογίες που έχουν συλλεχθεί *άμεσα* ή *έμμεσα* [1]. Με τον όρο άμεσα εννοούμε ότι ο χρήστης έχει εκφέρει ρητά την άποψή του, για παράδειγμα μια βαθμολογία από ένα έως πέντε. Με τον όρο έμμεσα εννοούμε ότι ο χρήστης δεν έχει δηλώσει ρητά την γνώμη του, αλλά είχε αλληλεπίδραση με το αντικείμενο, για παράδειγμα άνοιξε μία διαφήμιση, είδε το προφίλ ενός αντικειμένου κτλ. Αν δεν έχουμε πληροφορία σχετικά με τη γνώμη ενός χρήστη για ένα αντικείμενο, θέτουμε ως τιμή το 0.

Πίνακας 3. Πίνακας Βαθμολογίας Χρηστών (ΠΒΧ)



r_{ui} = Βαθμολογία που ο χρήστης ***u*** έδωσε στο αντικείμενο ***i***

Ο στόχος κάθε αλγορίθμου συστάσεων είναι να προβλέψει τις άγνωστες τιμές στον ΠΒΧ. Ο συγκεκριμένος πίνακας είναι αρκετά αραιός όσον αφορά τις μη μηδενικές τιμές, αφού στην πραγματικότητα κάθε χρήστης μπορεί να βαθμολογήσει μόνο ένα πολύ μικρό μέρος του συνόλου των αντικειμένων [1]. Το ποσοστό των μη μηδενικών τιμών του πίνακα ονομάζονται *πυκνότητα*.

$$\text{Μέση πυκνότητα ΠΒΧ} < 0.01\ \%$$

$$\text{Πυκνότητα Πίνακα } \textit{Netflix} \approx 0.002\ \%$$

2.4 Ποιότητα ΣΣ

2.4.1 Δείκτες ποιότητας

Υπάρχουν κάποιοι δείκτες στα ΣΣ που μπορούν να μας βοηθήσουν ώστε να αναλύσουμε την ποιότητα τους [10].

Σχετικότητα (*Relevance*): Ο πιο σημαντικός δείκτης αφορά την δυνατότητα του ΣΣ να προτείνει αντικείμενα τα οποία είναι πολύ πιθανό να εκτιμηθούν από τον χρήστη.

Κάλυψη (*Coverage*): Το ποσοστό πιθανών συστάσεων που είναι ικανό να προσφέρει το σύστημα. Ένα σύστημα με υψηλή *σχετικότητα* έχει μικρή *κάλυψη*, αφού προτείνει αντικείμενα που αρέσουν στους περισσότερους χρήστες. Η *κάλυψη* είναι σημαντική από την πλευρά του παρόχου του ΣΣ, αφού μετράει την ικανότητα να προτείνει όλα τα αντικείμενα που είναι διαθέσιμα.

Καινοτομία (*Novelty*): Ικανότητα του ΣΣ να προτείνει άγνωστα αντικείμενα στον χρήστη. Ένα ΣΣ με υψηλή *σχετικότητα* και χαμηλή *καινοτομία,* θα προτείνει στον χρήστη μόνο δημοφιλή αντικείμενα, γεγονός το οποίο ενδεχομένως έχει σαν αποτέλεσμα τη μείωση του ενδιαφέροντος του χρήστη.

Ποικιλία (*Diversity*): Ικανότητα του ΣΣ να προτείνει κάτι που διαφέρει από τα γνωστά ενδιαφέροντα του χρήστη. Ως *ποικιλία* νοείται το εύρος των προτεινόμενων αντικειμένων. Για παράδειγμα, αν είναι γνωστό ότι ο χρήστης ενδιαφέρεται για την κινέζικη κουζίνα και οι προτάσεις αφορούν μόνο το συγκεκριμένο είδος, υπάρχει κίνδυνος να χαθεί το ενδιαφέρον του. Από την άλλη πλευρά, η *ποικιλία* σαν μοναδικό κριτήριο δεν έχει

ικανοποιητικά αποτελέσματα, καθώς μπορεί να παραχθεί προτείνοντας αντικείμενα με τυχαίο τρόπο, με πιθανότητα το σύστημα να οδηγηθεί σε κακές συστάσεις.

Σταθερότητα (*Consistency*): Κάποια ΣΣ είναι πολύ δυναμικά ανανεώνοντας συνεχώς το προφίλ των χρηστών τους, κάτι που έχει σαν αποτέλεσμα οι προτάσεις να αλλάζουν συνεχώς.

Αυτοπεποίθηση (*Confidence*): Ικανότητα του ΣΣ να υπολογίσει τη βεβαιότητα για μία πρόταση. Αν το σύστημα δεν είναι βέβαιο για κάποιες εξ αυτών, καλύτερο είναι να τις αποφύγει. Δυστυχώς υπάρχουν πολύ αλγόριθμοι που δεν δίνουν τη δυνατότητα μέτρησής της *αυτοπεποίθησης*.

Τυχαία Ανακάλυψη (*Serendipity*): Ικανότητα του συστήματος να προκαλέσει έκπληξη στον χρήστη, προτείνοντας του κάτι αναπάντεχο, που δεν θα έψαχνε μόνος του.

2.4.2 Τεχνικές αξιολόγησης ΣΣ

Υπάρχουν δύο κατηγορίες τεχνικών αξιολόγησης για τα ΣΣ. *Με Σύνδεση* και *Χωρίς Σύνδεση* [10].

Με Σύνδεση (*On-Line*) :

- *Άμεση Κριτική Χρήστη* (*Direct user feedback*): Ζητείται από τον χρήστη να εκφέρει την γνώμη του για το ΣΣ. Αυτό μπορεί να πραγματοποιηθεί μέσω ενός ερωτηματολογίου.
- *Έλεγχος A/B* (*A/B Testing*): Μία άλλη τεχνική υλοποιείται παρακολουθώντας την συμπεριφορά του χρήστη και εφαρμόζοντας τον *Έλεγχο A / B*. Η βασική ιδέα είναι ότι χωρίζουμε τους χρήστες σε δύο ομάδες Α, Β, προτείνοντας αντικείμενα μόνο στους χρήστες που ανήκουν στην ομάδα Α. Στην συνέχεια συγκρίνονται οι συμπεριφορές. Για παράδειγμα σε ένα πείραμα για ηλεκτρονικές αγορές, με ένα καλοσχεδιασμένο ΣΣ, αναμένεται οι χρήστες της ομάδας Α που έλαβαν συστάσεις να πραγματοποιήσουν περισσότερες αγορές από αυτούς της Β.
- *Πειράματα σε ελεγχόμενο περιβάλλον* (*Controlled Experiments*): Σε αυτή την τεχνική ένα προσχέδιο εφαρμογής (demo) διατίθεται σε μία ομάδα πιθανών χρηστών. Από τους χρήστες ζητείται να κάνουν χρήση της εφαρμογής για ένα

διάστημα και να την αξιολογήσουν, με βάση τις προτάσεις που έλαβαν. Η συγκεκριμένη τεχνική παρόλο που δεν χρειάζεται τους πόρους άλλων τεχνικών, υστερεί. Αυτό συμβαίνει διότι η εφαρμογή και οι χρήστες δεν είναι πραγματικοί, οπότε η γνώμη τους δεν είναι αξιόπιστη, αφού απουσιάζει το κίνητρο σε σχέση με ένα πραγματικό σύστημα. Για παράδειγμα αν ένας χρήστης προσφερθεί να συμμετέχει σε ένα πείραμα για ένα ΣΣ εστιατορίων και του προταθεί ένα εστιατόριο που αντιστοιχεί στις επιθυμίες του, είναι πιθανό κοιτώντας τις φωτογραφίες των πιάτων και διαβάζοντας την περιγραφή, να δώσει μία καλή κριτική. Από την άλλη πλευρά ένας πραγματικός χρήστης θα το σκεφτόταν περισσότερο να ξοδέψει χρόνο και χρήμα αν δεν ήταν σίγουρος.

- *Πληθοπορισμός* (*Crowdsourcing*): Η τελευταία τεχνική της αξιολόγησης με σύνδεση αποτελείται από χρήστες οι οποίοι με αντάλλαγμα μίας αμοιβής, κάνουν χρήση ενός προσχεδίου της εφαρμογής και απαντούν σε μία σειρά από ερωτήσεις για την γνώμη που σχημάτισαν. Αυτή η τεχνική έχει καλές προοπτικές διότι είναι εύκολο να βρεθούν χρήστες να συμμετάσχουν. Το πρόβλημα που αντιμετωπίζεται αφορά την αξιοπιστία τους, καθώς οι περισσότεροι ενδιαφέρονται καθαρά για την αμοιβή, κάτι που μπορεί να οδηγήσει σε τυχαίες απαντήσεις.

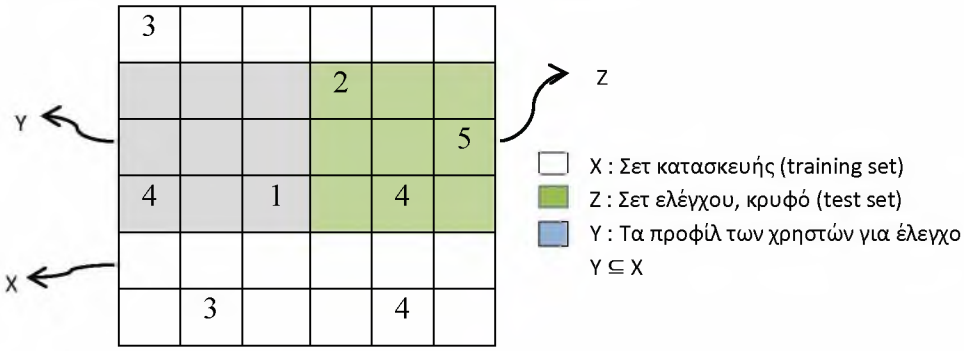
Χωρίς Σύνδεση (*Off-Line*)

Ένας αλγόριθμος ΣΣ είναι βασισμένος σε δύο συναρτήσεις f και g . Μία πιθανή είσοδος της f είναι ο ΠΒΧ. Η έξοδος είναι το μοντέλο μας, μια αναπαράσταση των προτιμήσεων των χρηστών. Η συνάρτηση g έχει ως είσοδο το μοντέλο και το προφίλ του χρήστη προκειμένου να εκτιμήσει κάποιες βαθμολογίες, οι οποίες δεν υπάρχουν εξαρχής στο σύστημα (εικόνα 10).

Εικόνα 10. Ροή αλγορίθμου ΣΣ

Προκειμένου να μετρήσουμε την ποιότητα του αλγορίθμου αυτής της κατηγορίας, πρέπει να χωρίσουμε τα δεδομένα εισόδου (πίνακας 4). Αυτό μπορούμε να το πετύχουμε αφήνοντας κάποιες από τις βαθμολογίες του ΠΒΧ εκτός της κατασκευής του μοντέλου, ώστε να τις χρησιμοποιήσουμε για την αξιολόγηση. Υλοποιείται χωρίζοντας τυχαία ένα ποσοστό του αρχικού σετ, συνήθως γύρω στο 20%, ή αφήνοντας εκτός μία βαθμολογία ανά χρήστη. Η τελευταία τεχνική ονομάζεται *Όλα πλην ενός* (*Leave One Out*) [43].

Πίνακας 4. Διαχωρισμός Δεδομένων ΠΒΧ για Έλεγχο



Χρησιμοποιείται το σετ X ως είσοδος στην f για να κατασκευαστεί το μοντέλο. Στην συνέχεια δίνεται μαζί με το σετ Y ως είσοδος στην συνάρτηση g, ώστε να υπάρξουν προτάσεις. Τέλος οι προτάσεις συγκρίνονται με το σετ Z.

Υπάρχουν τρεις τύποι μετρητών ποιότητας του ΣΣ. Μετρητής σφάλματος, μετρητής κατηγοριοποίησης και μετρητής κατάταξης.

Ο *μετρητής σφάλματος* (3) επιτρέπει την εκτίμηση της διαφοράς μίας βαθμολογίας που δίνεται από τον αλγόριθμο, σε σύγκριση με τη βαθμολογία που δίνεται από τον χρήστη.

$$e_{ui} = |r_{ui} - \hat{r}_{ui}| \tag{3}$$

όπου r_{ui} *: πραγματική βαθμολογία στο σετ ελέγχου,*

$\hat{r}_{ui}$ *: εκτίμηση βαθμολογίας απο το ΣΣ,*

e_{ui} *: σφάλμα*

Για να υπολογίσουμε την ποιότητα του ΣΣ γενικεύουμε τον προηγούμενο τύπο του *Μέσου Απόλυτου Σφάλματος* (MAE) (4):

$$MAE = \frac{\sum_{u,i \in T} |r_{ui} - \hat{r}_{ui}|}{N_T} \tag{4}$$

όπου T *: σετ ελέγχου (test set),*

N_T *: Σύνολο βαθμολογιών στο τεστ ελέγχου (μη μηδενικές τιμές)*

Μία παραλλαγή του MAE ονομάζεται *Μέσο Τετραγωνικό Σφάλμα* (MSE) και προτιμάται για δύο λόγους. Οι αλγόριθμοι ελαχιστοποίησης του MSE είναι ευκολότεροι στην υλοποίηση συγκριτικά με αυτούς του MAE. Επιπλέον με τη χρήση του MSE μια μεγάλη διαφορά μεταξύ πραγματικής και εκτιμώμενης τιμής έχει σημαντικότερη επίπτωση από ότι στο MAE (5).

$$MSE = \frac{\sum_{u,i \in T}(r_{ui} - \hat{r}_{ui})^2}{N_T} \quad (5)$$

Ο ΠΒΧ όπως έχει αναφερθεί προηγουμένως αποτελείται από βαθμολογίες, υψηλές η χαμηλές. Στόχος είναι να προταθεί στον χρήστη ένα αντικείμενο το οποίο εκτιμάται ότι θα βαθμολογηθεί θετικά. Οι βαθμολογίες μπορούν να κατηγοριοποιηθούν ως σχετικές για τις θετικές και μη-σχετικές για τις αρνητικές. Με βάση αυτές τις κατηγορίες μπορούμε να χωρίσουμε τις εξής ποσότητες [11] :

- Πλήθος Αντικειμένων που ορθά προτάθηκαν (True Positive - TP)
- Πλήθος Αντικειμένων που ορθά δεν προτάθηκαν (True Negative -TN)
- Πλήθος Αντικειμένων που λανθασμένα προτάθηκαν (False Positive - FP)
- Πλήθος Αντικειμένων που λανθασμένα δεν προτάθηκαν (False Negative - FN)

Οι *μετρητές κατηγοριοποίησης* έχουν ως σκοπό να συμπεράνουν αν στον χρήστη αρέσει ένα αντικείμενο ή όχι. Ο μετρητής *ανάκληση* (*recall*) υπολογίζει το ποσοστό των σχετικών αντικειμένων που προτάθηκαν στον χρήστη, σε σχέση με το σύνολο όλων των σχετικών αντικειμένων (6). Ένα μικρό ποσοστό σημαίνει ότι υπάρχουν πολλά αντικείμενα που δεν προτάθηκαν. Αντίθετα αν η τιμή είναι κοντά στο 1 έχουν προταθεί τα περισσότερα από τα διαθέσιμα αντικείμενα. Αναφέρεται αλλιώς και ως *hit rate*.

$$\text{Recall (K)} = \frac{TP}{FN+TP} \quad (6)$$

Ο μετρητής *ακρίβεια* (*precision*) υπολογίζει το σύνολο των σχετικών αντικειμένων που προτάθηκαν, σε σχέση με όλα τα αντικείμενα που προτάθηκαν (7).

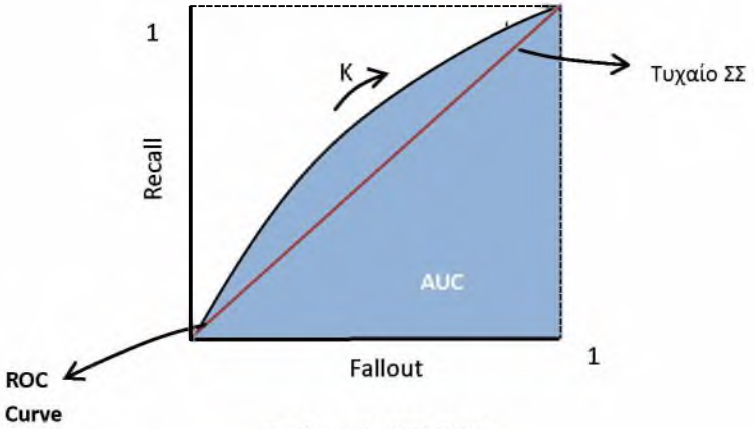
$$\text{Precision (K)} = \frac{TP}{FP+TP} \quad (7)$$

Τέλος ο μετρητής *αστοχία* (*fallout*) υπολογίζει το ποσοστό των μη-σχετικών αντικειμένων που προτάθηκαν, σε σχέση με το σύνολο των μη-σχετικών αντικειμένων (8).

$$\text{Fallout (K)} = \frac{FP}{FP+TN} \quad (8)$$

Οι *μετρητές αξιολόγησης κατάταξης* προσπαθούν να υπολογίσουν πόσο αρέσει στον χρήστη ένα αντικείμενο συγκριτικά με άλλα. Στην πράξη χρησιμοποιούνται όταν μία ταξινομημένη λίστα με προτάσεις έχει παρουσιαστεί στον χρήστη (*Top-N*). Όταν παρουσιάζεται μια λίστα αντικειμένων στον χρήστη, ιδανικά πρέπει στην κορυφή να υπάρχουν τα αντικείμενα που θα τον ικανοποιήσουν περισσότερο.

Η *καμπύλη λειτουργικών χαρακτηριστικών* (ROC) (εικόνα 11) αναπαριστά τη σχέση μεταξύ *ανάκλησης* και *αστοχίας* σε σχέση με το πλήθος των προτάσεων (K) .



Εικόνα 11. AUC ROC

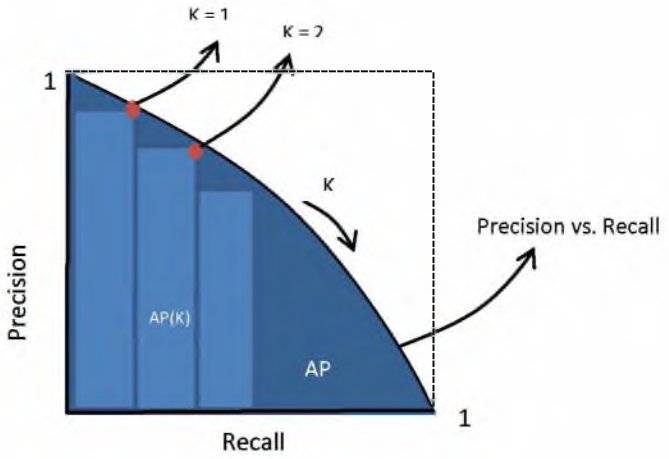
Αυξάνοντας την τιμή του K -τον αριθμό των αντικειμένων που προτάθηκαν- η τιμή της ανάκλησης αυξάνεται, όπως επίσης και της αστοχίας. Αν προτείνουμε όλα τα διαθέσιμα αντικείμενα και οι δύο τιμές θα λάβουν τη μέγιστη τιμή 1 .

Προκειμένου να αντιστοιχίσουμε την καμπύλη ROC σε μια τιμή, η *Περιοχή Κάτω Από Την Καμπύλη* (AUC) πρέπει να υπολογιστεί. Αυτή η τιμή μπορεί να υπολογιστεί μόνο στην περίπτωση που χρησιμοποιούμε άμεσες βαθμολογίες (9).

$$AUC = \frac{\sum_k Recall(k) * \Delta Fallout}{\#items} \quad (9)$$

Για ένα ιδανικό ΣΣ η τιμή AUC θα είναι 1. Για ένα τυχαίο ΣΣ , η τιμή AUC είναι 0.5 .

Ο δεύτερος μετρητής κατάταξης είναι η *Μέση Ακρίβεια* (AP) (εικόνα 12). Είναι η επιφάνεια κάτω από την καμπύλη ακρίβειας ανάκλησης. Για να υπολογιστεί η περιοχή χωρίζεται σε ορθογώνια, για κάθε τιμή του K (10).



Εικόνα 12. Μέση Ακρίβεια (AP)

$$AP(K) = \frac{\sum_k Precision(k) * \Delta Recall}{\#relevant\ items} \quad (10)$$

Η *Μέση Αριθμητική Ακρίβεια* (MAP) είναι το άθροισμα των μέσων ακριβειών για κάθε χρήστη, σε σχέση με το πλήθος τους [12] (11).

$$MAP(K) = \frac{\sum_u AP(K)_u}{N_u} \quad (11)$$

όπου N_u *: αριθμός χρηστών,*
$AP(K)_u$ *: Μέση Ακρίβεια για K για τον χρήστη u*

2.5 Φιλτράρισμα με Βάση το περιεχόμενο (CBF)

Σε αυτή την ομάδα αλγορίθμων συγκρίνεται η ομοιότητα των αντικειμένων με βάση τα γνωρίσματα τους. Η βασική υπόθεση σε αυτές τις μεθόδους, με βάση το περιεχόμενο, είναι ότι ο χρήστης που εκδήλωσε ενδιαφέρον για ένα αντικείμενο, θα ενδιαφέρεται για ανάλογα αντικείμενα [13]. Για παράδειγμα γνωρίζουμε ότι στον χρήστη αρέσει ένα ξενοδοχείο $hotel_1$ με πισίνα, εγκαταστάσεις φιλικές προς παιδιά και κουζίνα στο δωμάτιο. Στις εγκαταστάσεις του ξενοδοχείου $hotel_2$ επίσης περιλαμβάνονται οι προηγούμενες παροχές, συνεπώς θεωρούμε ότι πιθανόν να τον ενδιαφέρει.

Προκειμένου να αναπαρασταθούν τα γνωρίσματα του αντικειμένου, χρησιμοποιείται ο ΠΠΑ.

2.5.1 Τεχνικές Υπολογισμού Ομοιότητας

Μια από τις πιο συχνά χρησιμοποιούμενες τεχνικές είναι η *Ομοιότητα Συνημιτόνου* (*Cosine Similarity*) [1]. Κάθε γραμμή του ΠΠΑ αναπαριστά ένα αντικείμενο. Το κάθε αντικείμενο θεωρείται ως ένα διάνυσμα με *m* στοιχεία, όσα τα συνολικά γνωρίσματα που υπάρχουν στον πίνακα. Τα διανύσματα αυτά είναι δυαδικά, οι τιμές μπορεί να είναι 0, 1. Η ομοιότητα των δύο αντικειμένων μπορεί να υπολογιστεί ως το εσωτερικό γινόμενο τους (12).

$$s_{ij} = \vec{\imath} \cdot \vec{\jmath} = \sum_a i_a \cdot j_a = < i, j > \tag{12}$$

Στην μεταβλητή *s* έχουμε την ομοιότητα των αντικειμένων *i, j*, με βάση τον αριθμό των κοινών χαρακτηριστικών τους. Προκειμένου να υπάρχει το αποτέλεσμα σε μία μορφή, πιο εύκολη στην χρήση και την ερμηνεία, *κανονικοποιείται* (*normalize*). Αυτό επιτυγχάνεται διαιρώντας το προηγούμενο αποτέλεσμα με το μήκος των διανυσμάτων (13). Σε αυτή την περίπτωση δύο ίδια αντικείμενα θα έχουν την μέγιστη τιμή ομοιότητας 1.

$$s_{ij} = \frac{\sum_a i_a \cdot j_a}{\sqrt{\sum_a i_a^2 \cdot \sum_a j_a^2}} = \frac{\vec{\imath} \cdot \vec{\jmath}}{|\vec{\imath}|_2 \cdot |\vec{\jmath}|_2} = cos\theta \tag{13}$$

Μία σημαντική παράμετρος, όταν υπολογίζεται η ομοιότητα των αντικειμένων, είναι η *υποστήριξη* (*support*). Η παράμετρος αυτή, αναπαριστά τον αριθμό των μη-μηδενικών όρων των διανυσμάτων που συγκρίνονται και επιτρέπει να δίνεται μεγαλύτερο βάρος σε αντικείμενα με περισσότερα γνωρίσματα. Ως παράδειγμα υποθέτουμε ότι υπάρχουν δύο αντικείμενα *i, j*. Κάθε αντικείμενο έχει μόνο ένα γνώρισμα το οποίο είναι κοινό. Σε αυτή την περίπτωση έχουμε $s_{ij} = \frac{1}{\sqrt{1 \cdot 1}} = 1$. Αντίστοιχα αν στα δύο αντικείμενα, υπάρχουν στο καθένα από πέντε γνωρίσματα, με τα τέσσερα να είναι κοινά, έχουμε $s_{ij} = \frac{4}{\sqrt{5 \cdot 5}} = 0.8$. Η πρώτη περίπτωση φαίνεται να προηγείται της δεύτερης όσον αφορά το αποτέλεσμα, κάτι το οποίο δεν θα έπρεπε να ισχύει, εφόσον στην δεύτερη υπάρχει πολύ μεγαλύτερη τιμή *υποστήριξης*. Για να αποφύγουμε ανάλογες περιπτώσεις, εισάγουμε μία *σταθερά συρρίκνωσης* (14).

$$s_{ij} = \frac{\vec{\imath} \cdot \vec{\jmath}}{|\vec{\imath}| \cdot |\vec{\jmath}| + c} \tag{14}$$

Οι ομοιότητες υπολογίζονται για όλα τα διαθέσιμα ζευγάρια αντικειμένων δημιουργώντας τον *Πίνακα Ομοιότητας* (*Similarity Matrix*). Ο πίνακας είναι συμμετρικός. Κάνοντας

χρήση του πίνακα ομοιότητας μπορούμε να εκτιμήσουμε την βαθμολογία για ένα αντικείμενο *i*, που δεν έχει βαθμολογήσει ο χρήστης *u*, βασιζόμενοι σε άλλες βαθμολογίες του ίδιου χρήστη για διαφορετικά αντικείμενα *j* (15).

$$\tilde{r}_{ui} = \frac{\sum_j r_{uj} \cdot s_{ji}}{\sum_j s_{ji}} \tag{15}$$

Αν ο σκοπός είναι να προταθούν τα *καλύτερα-N* (*top-N*), η κανονικοποίηση δεν χρειάζεται, οπότε, μπορεί να παραμείνει μόνο ο αριθμητής. Γενικεύοντας τον τύπο για το σύστημα προκύπτει (16):

$$\vec{\tilde{r}_u} = \vec{r_u} \cdot S \quad \text{ή} \quad \tilde{R} = R \cdot S \tag{16}$$

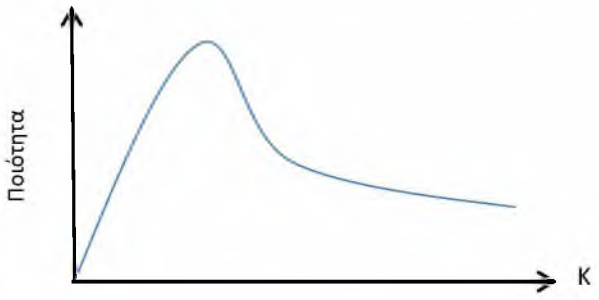
όπου R : πίνακας βαθμολογιών χρήστη URM,
S : πίνακας ομοιότητας

Ο πίνακας ομοιότητας είναι πολύ πυκνός, δηλαδή υπάρχουν λίγα κελιά χωρίς τιμή -η κύρια διαγώνιος-. Αυτό επηρεάζει τόσο την μνήμη όσο και τον χρόνο εκτέλεσης, αφού είναι πολύ απαιτητικό σε πόρους. Επίσης οι περισσότερες τιμές στον πίνακα είναι μικρές και αρκετά όμοιες μεταξύ τους, κάτι το οποίο δυσκολεύει τη διαφοροποίηση των αντικειμένων βάσει αυτών.

Μία λύση για αυτά τα προβλήματα είναι η μέθοδος των *Κ-Κοντινότερων Γειτόνων* (*K-Nearest Neighbors*) [1]. Σε αυτή την μέθοδο κρατάμε μόνο τα Κ πιο όμοια ζευγάρια στον *πίνακα ομοιότητας*, όπου *K* σταθερά. Η σχέση για την εκτίμηση της βαθμολογίας του χρήστη, σύμφωνα με την μέθοδο αυτή, κάνει χρήση των *j* πιο κοντινών γειτόνων του, για το αντικείμενο *i*.

$$\tilde{r}_{ui} = \frac{\sum_{j \in KNN(i)} r_{uj} \cdot s_{ji}}{\sum_{j \in KNN(i)} s_{ji}} \tag{17}$$

Η τιμή του *K* επηρεάζει την ποιότητα των προτάσεων, αφού με μικρή τιμή δεν υπάρχουν αρκετά δεδομένα για την εκτίμηση, ενώ με μεγάλη γίνεται χρήση μη επιθυμητών δεδομένων που οδηγούν στα προβλήματα που προαναφέρθηκαν (εικόνα 13).



Εικόνα 13. Μεταβολή ποιότητας ως προς την τιμή K

2.6 Συνεργατικό Φιλτράρισμα (CF)

Σε αυτή την ομάδα αλγορίθμων δεν ενδιαφέρουν τα γνωρίσματα των αντικειμένων αλλά η γνώμη των χρηστών.

Η βασική ιδέα είναι ότι μία βαθμολογία που δεν έχει δοθεί άμεσα από τον χρήστη, μπορεί να εκτιμηθεί, εφόσον σχετίζεται με διαφορετικούς χρήστες και αντικείμενα. Η βασική είσοδος σε αυτούς τους αλγορίθμους είναι ο ΠΒΧ που περιέχει τις παλαιότερες αλληλεπιδράσεις των χρηστών με τα αντικείμενα [14].

2.6.1 Με βάση τον χρήστη

Υπάρχουν χρήστες με παρόμοιες επιθυμίες. Η βασική ιδέα σε αυτή την υποκατηγορία αλγορίθμων είναι να βρεθούν αυτοί οι χρήστες και να τους προταθούν τα αντικείμενα που άρεσαν περισσότερο στους ομοίους τους. Το πρόβλημα έγκειται στο πώς μπορούν να βρεθούν αυτοί οι χρήστες. Η λύση είναι οι βαθμολογίες τους.

Ψάχνοντας τον ΠΒΧ και συγκρίνοντας τις βαθμολογίες μεταξύ χρηστών, είναι δυνατό να καταλάβουμε την ομοιότητά τους. Αν δύο χρήστες έχουν δώσει παρόμοια βαθμολογία σε ένα αντικείμενο, μπορούμε να υποθέσουμε ότι έχουν την ίδια γνώμη για αυτό. Αν αυτό επαναλαμβάνεται για ένα αριθμό αντικειμένων, υποθέτουμε ότι οι χρήστες είναι όμοιοι.

Σε μία απλή μορφή για έμμεσες βαθμολογίες έχουμε τον πίνακα 5:

Πίνακας 5. Παράδειγμα ΠΒΧ για 'Εμμεσες Βαθμολογίες ως Προς τον Χρήστη

Αντικείμενα

$\vec{u}$	1		1	1	1	1
$\vec{v}$	1	1	1		1	1

Χρήστες

Σε αυτή την περίπτωση έχουμε, ως ομοιότητα των χρηστών *u* και *v*, τον αριθμό των κοινών τους αλληλεπιδράσεων, δηλαδή τα αντικείμενα που από κοινού τους ενδιέφεραν (18).

$$s_{uv} = < u, v > = \sum_i r_{ui} \cdot r_{vi} = \vec{u} \cdot \vec{v} = 4 \quad (18)$$

Είναι πιο επιθυμητό να έχουμε την τιμή της ομοιότητας μεταξύ του μηδέν και ένα. Τροποποιώντας την προηγούμενη σχέση, διαιρώντας με το μήκος των διανυσμάτων, κανονικοποιείται το αποτέλεσμα στο επιθυμητό εύρος τιμών.

Όπως και στην περίπτωση του CBF, ένας τρόπος υπολογισμού της ομοιότητας είναι η *ομοιότητα συνημίτονου* (19).

$$s_{uv} = \frac{\sum_i r_{ui} \cdot r_{vi}}{\sqrt{\sum_i r_{ui}^2 \cdot \sum_i r_{vi}^2}} = \frac{\vec{r}_u \cdot \vec{r}_v}{|\vec{r}_u|_2 \cdot |\vec{r}_v|_2} \quad (19)$$

Επίσης μπορεί να προστεθεί μία σταθερά συρρίκνωσης C, ώστε να δοθεί μία έμφαση σε χρήστες με υψηλή *υποστήριξη*, δηλαδή με όσον το δυνατό περισσότερα κοινά αντικείμενα (20). Συνήθως οι τιμές του C είναι μεταξύ ένα και δέκα.

$$s_{uv} = \frac{\sum_i r_{ui} \cdot r_{vi}}{\sqrt{\sum_i r_{ui}^2 \cdot \sum_i r_{vi}^2} + C} = \frac{\vec{r}_u \cdot \vec{r}_v}{|\vec{r}_u|_2 \cdot |\vec{r}_v|_2 + C} \quad (20)$$

Όλες οι τιμές ομοιότητας μεταξύ ζευγαριών χρηστών αναπαρίστανται με έναν πίνακα ομοιότητας, ο οποίος είναι συμμετρικός, όπου το στοιχείο S_{uv} όπως και το S_{vu} αναπαριστά την ομοιότητα του χρήστη *u* με τον χρήστη *v*. Λόγω του μεγέθους του πίνακα, όπως και της πυκνότητας του, δεν είναι αποδοτικό να χρησιμοποιείται ολόκληρος προκειμένου να γίνουν συστάσεις, παρά μόνο οι πιο όμοιοι χρήστες. Ένας τρόπος για να επιτευχθεί αυτό, είναι να τεθεί *ελάχιστο όριο τιμής για την ομοιότητα* (*threshold*) ώστε να λαμβάνει υπόψιν, μόνο τους χρήστες που έχουν μεγαλύτερες η ίσες τιμές. Το πρόβλημα αυτής της υλοποίησης είναι ότι δεν είναι εύκολο να βρεθεί η ιδανική τιμή του ορίου, η οποία θα λειτουργεί αποδοτικά για όλους τους χρήστες.

Ένας πιο απλός τρόπος είναι η τεχνική των *K-κοντινότερων γειτόνων*. Βρίσκονται οι K πιο κοινοί χρήστες για κάθε χρήστη και συμμετέχουν μόνο αυτοί στους υπολογισμούς. Όπως και στην περίπτωση του CBF, έτσι και εδώ η τιμή του *K* είναι κρίσιμη ως προς την ποιότητα των προτάσεων. Η ποιότητα αυξάνεται μέχρι μία τιμή του *K* και ύστερα αρχίζει να φθίνει, όπως και στην περίπτωση του CBF.

Έχοντας υπολογιστεί ο πίνακας ομοιότητας, μπορεί να εκτιμηθεί για έναν χρήστη *u* το πόσο του αρέσει το αντικείμενο *i*. Ένας απλός αλλά αποτελεσματικός τρόπος είναι να υπολογιστεί ο *σταθμισμένος μέσος όρος* (*weighted average*) που έχουν δώσει άλλοι χρήστες σε αυτό το αντικείμενο, όπου τα *βάρη* είναι η ομοιότητα μεταξύ των χρηστών. Όσο μεγαλύτερη ομοιότητα υπάρχει μεταξύ τους, τόσο μεγαλύτερη θα είναι η επιρροή στο τελικό αποτέλεσμα.

$$r_{ui} = \frac{\sum_{v \in KNN(u)} r_{vi} \cdot s_{vu}}{\sum_{v \in KNN(u)} s_{vu}} \tag{21}$$

Η *ομοιότητα συνημιτόνου* που αναλύθηκε προηγουμένως λειτουργεί σωστά στην περίπτωση των έμμεσων βαθμολογιών (αλληλεπιδράσεων), αλλά αντιθέτως αντιμετωπίζει προβλήματα σε άμεσες βαθμολογίες.

Σε αυτή την περίπτωση πρέπει να ληφθεί υπόψιν η *προκατάληψη των χρηστών* (*user bias*) όσον αφορά τις βαθμολογίες. Δηλαδή αν είναι πιο αυστηροί η επιεικείς στον τρόπο με τον οποίο βαθμολογούν, συγκριτικά με τον μέσο χρήστη. Το ίδιο ισχύει και για τα αντικείμενα (item bias), επειδή κάποια από αυτά τείνουν να λαμβάνουν μεγαλύτερες βαθμολογίες σε σχέση με άλλα.

Η λύση είναι μία παραλλαγή του τύπου της *ομοιότητας συνημιτόνου*, η μέθοδος *ομοιότητας του Pearson* (*Pearson Correlation Coefficient*). Σε αυτή την τεχνική αφαιρούμε από την βαθμολογία των χρηστών για ένα αντικείμενο τους μέσους όρους, πετυχαίνοντας την κανονικοποίηση που θέλουμε (22). Ως αποτέλεσμα φαίνεται πόσο αρέσει στον χρήστη το αντικείμενο, με βάση τον μέσο όρο του. Μπορεί να είναι είτε θετικό σε περίπτωση που του αρέσει αρκετά, είτε αρνητικό στην αντίθετη περίπτωση [15].

$$s_{uv} = \frac{\sum_i (r_{ui} - \bar{r}_u) \cdot (r_{vi} - \bar{r}_v)}{\sqrt{\sum_i (r_{ui} - \bar{r}_u)^2 \cdot \sum_i (r_{vi} - \bar{r}_v)^2 + C}} \tag{22}$$

$$\text{όπου} \quad \bar{r}_u = \frac{\sum_i r_{ui}}{N_u + C} \ , \qquad \bar{r}_v = \frac{\sum_i r_{vi}}{N_v + C} \qquad \bar{r}_u, \bar{r}_v : user\ bias$$

Εφόσον έχουμε τις αμερόληπτες ομοιότητες μεταξύ των χρηστών από την προηγούμενη μέθοδο, μπορούμε να κάνουμε εκτίμηση της βαθμολογίας που θα έδινε ο χρήστης σε ένα αντικείμενο. Οι ομοιότητες έχουν υπολογιστεί με βάση το πόσο ικανοποιούσε τον χρήστη ένα αντικείμενο, σε σχέση με την μέση βαθμολογία του. Λόγω αυτού ο τελικός τύπος θα

είναι το άθροισμα της εκτιμώμενης γνώμης του χρήστη, σε μορφή σύγκρισης με τον μέσο όρο, συν τον μέσο όρο του (23).

$$\tilde{r}_{ui} = \frac{\sum_{v \in KNN(u)} (r_{vi} - \bar{r}_v) \cdot s_{vu}}{\sum_{v \in KNN(u)} s_{vu}} + \bar{r}_u \quad (23)$$

2.6.2 Με βάση το αντικείμενο

Στην προσέγγιση αυτή, προκειμένου να κάνουμε πρόβλεψη μιας βαθμολογίας για ένα συγκεκριμένο αντικείμενο, πρέπει να βρούμε ένα σύνολο των πιο όμοιων σε αυτό αντικειμένων. Η ιδέα διαφέρει σε συγκριτικά με το CBF, γιατί σε αυτή την περίπτωση δεν βρίσκουμε ομοιότητα με βάση τα γνωρίσματα αλλά με βάση τον αριθμό των χρηστών που έχουν βαθμολογήσει και τα δύο.

Ας υποθέσουμε ότι έχουμε έναν ΠΒΧ με έμμεσες αλληλεπιδράσεις (πίνακας 6). Οι μηδενικές τιμές παραλείπονται. Η ομοιότητα των αντικειμένων *i* και *j* είναι ο αριθμός των κοινών χρηστών που έχουν αλληλεπιδράσει μαζί τους $\boldsymbol{s_{ij} = \sum_u r_{ui} \cdot r_{uj} = \vec{i} \cdot \vec{j}}$. Στο παράδειγμα του πάνω πίνακα $\boldsymbol{s_{ij} = 3}$

Πίνακας 6. Παράδειγμα ΠΒΧ για Έμμεσες Βαθμολογίες, ως Προς το Αντικείμενο

		$\vec{i}$		$\vec{j}$	
		1		1	
		1			
		1		1	
		1		1	

Χρήστες

Αντικείμενα

Όπως στην περίπτωση με βάση τον χρήστη, έτσι και εδώ είναι επιθυμητό η τιμή της ομοιότητας να είναι μεταξύ μηδέν και ένα. Αυτό το επιτυγχάνεται κανονικοποιώντας την προηγούμενη σχέση, κάτι που οδηγεί στην *ομοιότητα συνημιτόνου* (24) (πίνακας 7).

$$s_{ij} = \frac{\sum_u r_{ui} \cdot r_{uj}}{\sqrt{\sum_u r_{ui}^2 \cdot \sum_u r_{uj}^2}} = \frac{\vec{r}_i \cdot \vec{r}_j}{|\vec{r}_i|_2 \cdot |\vec{r}_j|_2} \quad (24)$$

Πίνακας 7. Βαθμολογίες Αντικειμένων *i* και *j* του Χρήστη *u* που Χρησιμοποιούνται για την Ομοιότητα Συνημιτόνου

				i		*j*	
u				r_{ui}		r_{uj}	

Επίσης μπορούμε να προσθέσουμε μία σταθερά συρρίκνωσης C ώστε να δώσουμε έμφαση σε αντικείμενα με υψηλή υποστήριξη, δηλαδή όσον το δυνατό περισσότερους κοινούς χρήστες (25).

$$s_{ij} = \frac{\sum_u r_{ui} \cdot r_{uj}}{\sqrt{\sum_u r_{ui}^2 \cdot \sum_u r_{uj}^2} + C} = \frac{\vec{r}_i \cdot \vec{r}_j}{|\vec{r}_i|_2 \cdot |\vec{r}_j|_2 + C} \quad (25)$$

Όλες οι τιμές ομοιότητας περιγράφονται από τον πίνακα ομοιότητας, ο οποίος είναι συμμετρικός και τα στοιχεία S_{ij} και S_{ji} αναπαριστούν την ομοιότητα μεταξύ αντικειμένου *i* και *j*. Για τους λόγους που αναλύθηκαν και στην περίπτωση με βάση τον χρήστη, χρησιμοποιούμε την μέθοδο των *K-κοντινότερων γειτόνων*, για την εκτίμηση της βαθμολογίας. Για να την υπολογίζουμε χρησιμοποιούμε τον *σταθμισμένο μέσο όρο* των βαθμολογιών του χρήστη για διαφορετικά αντικείμενα και ως βάρη την ομοιότητα μεταξύ των αντικειμένων (26).

$$\tilde{r}_{ui} = \frac{\sum_{j \in KNN(i)} r_{uj} \cdot s_{ji}}{\sum_{j \in KNN(i)} s_{ji}} \quad (26)$$

Αν ο πίνακας βαθμολογίας χρηστών είναι για άμεσα δηλωμένες βαθμολογίες, ο προηγούμενος τύπος της ομοιότητας συνημιτόνου δεν δίνει τα βέλτιστα δυνατά αποτελέσματα διότι δεν λαμβάνει υπόψιν την προκατάληψη των βαθμολογιών των χρηστών, όπως και των αντικειμένων που προαναφέρθηκε. Σε αυτή την περίπτωση χρησιμοποιείται η *ομοιότητα του Pearson* (27).

$$s_{ij} = \frac{\sum_u (r_{ui} - \bar{r}_u) \cdot (r_{uj} - \bar{r}_u)}{\sqrt{\sum_u (r_{ui} - \bar{r}_u)^2 \cdot \sum_u (r_{uj} - \bar{r}_u)^2} + C} \quad (27)$$

$$\text{όπου} \quad \bar{r}_u = \frac{\sum_i r_{ui}}{N_u + C}$$

Οι τεχνικές που χρησιμοποιούνται στο CF χωρίζονται σε δύο κατηγορίες, *με βάση το μοντέλο* (*Model Based*) και με *βάση την μνήμη* (*Memory Based*). Στην περίπτωση *με βάση την μνήμη*, τεχνική την οποία χρησιμοποιούσαν οι πρώτοι αλγόριθμοι CF, ο ΠΒΧ χρησιμοποιείται απευθείας στην πρόβλεψη. Αντιθέτως στην περίπτωση με *βάση το μοντέλο* όταν υπολογίζονται οι συστάσεις, δεν χρησιμοποιείται απευθείας ολόκληρος ο πίνακας αλλά εξάγονται πληροφορίες από αυτόν, προκειμένου να κατασκευαστεί το μοντέλο [16].

Οι τεχνικές αυτές χρειάζονται δύο βήματα για τις προβλέψεις. Αρχικά πρέπει να κατασκευαστεί το μοντέλο, το οποίο είναι μια συνάρτηση f με είσοδο είτε τον ΠΒΧ, στην περίπτωση του CF, είτε τον ΠΠΑ στη περίπτωση του CBF (28).

$$model = \begin{cases} f(URM) & if\ \ CF \\ f(ICM) & if\ \ CBF \end{cases} \tag{28}$$

Το δεύτερο βήμα είναι να εκτιμηθούν οι βαθμολογίες με βάση μια συνάρτηση g που έχει σαν είσοδο το μοντέλο που κατασκευάστηκε και το προφίλ του χρήστη (29). Αν το προφίλ του χρήστη είναι υποσύνολο του ΠΒΧ, τότε χρησιμοποιείται τεχνική με *βάση την μνήμη*, διαφορετικά με *βάση το μοντέλο*.

$$estimated\ ratings = g(model, user\ profile) \tag{29}$$

$$user\ profile \in URM \rightarrow Memory\ Based$$
$$user\ profile \notin URM \rightarrow Model\ Based$$

Η βασική διαφορά των δύο τεχνικών στην πράξη είναι στην περίπτωση εισόδου νέου χρήστη στο σύστημα. Στην περίπτωση της *τεχνικής με βάση την μνήμη* πρέπει να εισαχθεί ο νέος χρήστης στον ΠΒΧ, και να υπολογιστεί ξανά η ομοιότητα του με όλους τους υπόλοιπους, κάτι που είναι απαιτητικό σε πόρους. Παρόλα αυτά, μπορεί να χρησιμοποιηθεί αποδοτικά για χρήστες που υπάρχουν ήδη στο σύστημα.

Στην τεχνική με *βάση το μοντέλο* δεν χρειάζεται να εισαχθεί ο νέος χρήστης στον ΠΒΧ, άρα δεν χρειάζεται να ανανεωθεί ο πίνακας ομοιοτήτων. Ο αλγόριθμος με *βάση το αντικείμενο*, είναι μία περίπτωση με *βάση το μοντέλο*. Μέσω του ΠΒΧ υπολογίζεται ο πίνακας ομοιότητας μεταξύ των αντικειμένων S_{II} (πίνακας 8) (30). Με την εισαγωγή νέου χρήστη ο πίνακας αυτός δεν αλλάζει σημαντικά οπότε δεν αντιμετωπίζεται πρόβλημα απόδοσης λόγω του επανα-υπολογισμού (31).

$$S_{II} = f(URM) \qquad (30)$$

$$estimated\ ratings = g(S_{II}, \vec{r}_u)$$

$$\text{όπου} \quad \vec{r}_u : user\ profile, \quad S_{II} : similarity\ matrix$$

Πίνακας 8. Παράδειγμα Πίνακα Ομοιότητας S_{II} με Βάση το Μοντέλο

item				
	-	0.2	0.17	0.3
	0.2	-	0.7	0.42
	0.17	0.7	-	0.94
	0.3	0.42	0.94	-

S_{II}

$$\tilde{r}_{ui} = \frac{\sum_{j \in KNN(i)} r_{uj} \cdot s_{ji}}{\sum_{j \in KNN(i)} s_{ji}} \quad (31)$$

Στον αλγόριθμο με *βάση τον χρήστη*, για την εκτίμηση της βαθμολογίας γίνεται χρήση του πίνακα ομοιότητας χρηστών, ο οποίος υπολογίζεται με βάση τον ΠΒΧ. Με την εισαγωγή ενός καινούργιου χρήστη, ο πίνακας URM αλλάζει, επηρεάζοντας σημαντικά τον πίνακα ομοιότητας, εφόσον σε αυτή την περίπτωση έχουμε χρήστες και όχι αντικείμενα. Ο πίνακας ομοιότητας S_{UU} (πίνακας 9) (32) πρέπει να υπολογιστεί ξανά, κάτι το οποίο είναι υπολογιστικά απαιτητικό (33). Αυτή η περίπτωση είναι με *βάση την μνήμη*.

$$S_{UU} = f(URM) \qquad (32)$$

$$estimated\ ratings = g(S_{UU}, \vec{r}_u)$$

$$\text{όπου} \quad \vec{r}_u : user\ profile, \quad S_{UU} = similarity\ matrix$$

$$\tilde{r}_{ui} = \frac{\sum_{v \in KNN(u)} r_{vi} \cdot s_{vu}}{\sum_{v \in KNN(u)} s_{vu}} \quad (33)$$

Πίνακας 9. Παράδειγμα Πίνακα Ομοιότητας S_{UU} με Βάση την Μνήμη

-	0.2	0.17	0.3	
0.2	-	0.7	0.42	
0.17	0.7	-	0.94	
0.3	0.42	0.94	-	

S_{UU}

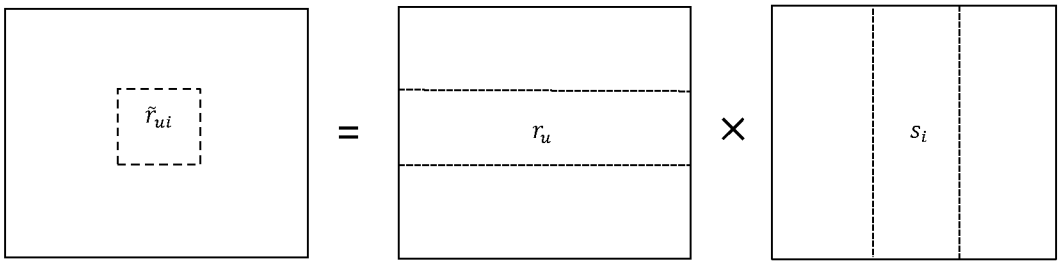
2.7 Συστήματα Συστάσεων με μηχανική μάθηση

Ένα σύστημα συστάσεων το οποίο χρησιμοποιεί αλγόριθμους CF, μπορεί να γίνει πιο αποδοτικό, δίνοντας του την δυνατότητα να βελτιωθεί από μόνο του. Για να επιτευχθεί πρέπει να μετατραπούν οι βασικοί αλγόριθμοι που χρησιμοποιεί, σε αλγορίθμους μηχανικής μάθησης.

Για να κατασκευαστεί ένα CF ΣΣ με την απλή μέθοδο, πρέπει αρχικά να αποφασιστεί αν το μοντέλο θα είναι με *βάση το αντικείμενο* ή *τον χρήστη* και στην συνέχεια ο τρόπος που θα υπολογιστούν οι ομοιότητες (*ομοιότητα συνημιτόνου*, *Pearson* κτλ.). Στην περίπτωση της μηχανικής μάθησης αποφασίζεται μόνο το είδος του μοντέλου και από εκεί και πέρα το σύστημα είναι ικανό να υπολογίσει αυτόματα τις ομοιότητες, προκειμένου να κάνει τις καλύτερες δυνατές συστάσεις.

Πρόκειται για ένα πρόβλημα βελτιστοποίησης στο οποίο υπάρχουν πολλές παράμετροι που πρέπει να υπολογιστούν. Οι παράμετροι αυτοί είναι οι ομοιότητες, για τις οποίες θέλουμε να ελαχιστοποιήσουμε μία *συνάρτηση σφάλματος* (*Loss Function*). Για παράδειγμα σ' ένα μοντέλο με *βάση το αντικείμενο*, σκοπός είναι να εκτιμηθεί η βαθμολογία που θα έδινε ένας χρήστης σ΄ ένα αντικείμενο, συνδυάζοντας τις βαθμολογίες που έχει δώσει σε άλλα αντικείμενα και τις ίδιες ομοιότητες με αυτά. Άρα οι εκτιμήσεις είναι μια συνάρτηση με βάση τον πίνακα ομοιότητας *S*.

Για την εξίσωση $\widetilde{\boldsymbol{R}}(\boldsymbol{S}) = \boldsymbol{RS}$ (εικόνα 14), ζητείται η καλύτερη δυνατή εκτίμηση του πίνακα *S*, ώστε η τιμή της συνάρτησης σφάλματος $\boldsymbol{E}(\boldsymbol{S})$ να είναι η μικρότερη δυνατή. Σε μία επιθυμητή αλλά όχι ρεαλιστική περίπτωση όπου $\boldsymbol{R} = \widetilde{\boldsymbol{R}} \Rightarrow \boldsymbol{E}(\boldsymbol{S}) = \boldsymbol{0}$.



Εικόνα 14. Εκτίμηση πίνακα βαθμολογιών

Μία περίπτωση της συνάρτησης σφάλματος είναι το μέσο τετραγωνικό σφάλμα. Σε αυτήν ισχύει $E(S) = \sum_{u,i \in R}[r_{ui} - \tilde{r}_{ui}]^2$ όπου *u* ο χρήστης και *i* το αντικείμενο. Σε μορφή πινάκων έχουμε $E(S) = \|R - RS\|_2$, δηλαδή την ευκλείδεια νόρμα του ΠΒΧ, *R*, μείον τον πίνακα των εκτιμώμενων βαθμολογιών *RS*.

Η προηγούμενη νόρμα είναι εύκολο να υπολογιστεί. Είναι το άθροισμα του σφάλματος των μη μηδενικών στοιχείων του ΠΒΧ, R [17] (34).

$$E(S) = \sum_{u,i \in R^+}[r_{ui} - \tilde{r}_{ui}]^2 \tag{34}$$

$$\text{όπου} \quad R^+ = \text{μη μηδενικές τιμές του ΠΒΧ}$$

Σκοπός είναι να βρεθούν οι καλύτερες δυνατές τιμές ομοιότητας του πίνακα *S*, ώστε να ελαχιστοποιηθεί η συνάρτηση σφάλματος *E(S)*. Πρέπει δηλαδή να βρεθούν τιμές του πίνακα *S* ώστε να έχουμε την ελάχιστη ευκλείδεια νόρμα για όλες τις πιθανές τιμές του (35).

$$S^* = \min_s \|R - RS\| = \min_s E(S) \tag{35}$$

Θεωρητικά, τη καλύτερη δυνατή λύση την δίνει ο μοναδιαίος πίνακας I, στον οποίο η κύρια διαγώνιος του ισούται με ένα, με τιμή μηδέν στα υπόλοιπα στοιχεία του. Η συγκεκριμένη λύση δίνει σφάλμα ίσον με μηδέν. Σε αυτή την περίπτωση τα στοιχεία είναι όμοια μόνο με τον εαυτό τους. Δίνοντας έτσι μηδενικό σφάλμα, αλλά και αδυναμία να προτείνουμε οποιοδήποτε αντικείμενο δεν έχει βαθμολογήσει ο χρήστης, αυτή η λύση οδηγεί σε *υπερπροσαρμογή* (*overfitting*). Για να αντιμετωπιστεί αυτό το πρόβλημα πρέπει να γίνουν δύο βήματα. Αρχικά πρέπει να τεθούν τα στοιχεία της διαγωνίου ίσον με μηδέν,

$Diag(S) = 0$ και στην συνέχεια να τροποποιηθεί η συνάρτηση σφάλματος προσθέτοντας σε αυτή όρους *κανονικοποίησης* (*regularization terms*) [18] (36).

$$E(S) = \|R - RS\|_2 + \underbrace{\lambda\|S\|_2}_{\text{Regularization term}}, \quad \text{όπου } \lambda > 0 \tag{36}$$

Σκοπός είναι, να γίνει ο πίνακας ομοιότητας S όσο το δυνατόν πιο αραιός, δηλαδή να έχει κάποιες μη μηδενικές τιμές. Προσθέτοντας την ευκλείδεια νόρμα του πίνακα S στην συνάρτηση σφάλματος, προκειμένου να ελαχιστοποιηθεί η συνάρτηση *E(S)*, εκτός του ότι γίνεται προσπάθεια να ελαχιστοποιηθεί το σφάλμα μεταξύ πραγματικών και εκτιμώμενων βαθμολογιών, πρέπει να κρατηθούν και οι τιμές στον S όσο το δυνατόν μικρότερες. Η τροποποιημένη συνάρτηση σφάλματος ονομάζεται *αμφίκλινος παλινδρόμηση* (*ridge regression*) [1].

Με βάση την τιμή του λ, ο πίνακας S αλλάζει μορφή στην προσπάθεια της συνάρτησης σφάλματος να ελαχιστοποιηθεί. Αν για παράδειγμα $\lambda = 0$ τότε επιστρέφουμε στην μορφή χωρίς κανονικοποίηση, προσπαθώντας να ελαχιστοποιήσουμε την διαφορά μεταξύ πραγματικών και εκτιμώμενων τιμών του πίνακα R. Αυτό θα οδηγήσει στον μοναδιαίο πίνακα I που ελαχιστοποιεί το σφάλμα στο μηδέν αλλά και σε *υπερπροσαρμογή* (37).

$$E(S) = \|R - RS\|_2 + \lambda\|S\|_2 \overset{\lambda=0}{\Longrightarrow} E(S) = \|R - RS\|_2 \Rightarrow S = I \tag{37}$$

Αντιθέτως αν το λ είναι μια πολύ μεγάλη τιμή ο πρώτος όρος της συνάρτησης E έχει ελάχιστη επιρροή, άρα το αποτέλεσμά της θα είναι να ελαχιστοποιήσει την ευκλείδεια νόρμα του S. Με αυτό τον τρόπο θα καταλήξουμε σε έναν μηδενικό πίνακα, όπου δεν υπάρχουν ομοιότητες (38).

$$E(S) = \|R - RS\|_2 + \lambda\|S\|_2 \overset{\lim_{\lambda\to\infty}\lambda}{\Longrightarrow} E(S) = \lambda\|S\|_2 \Rightarrow S = 0 \tag{38}$$

Πρέπει να βρεθεί η βέλτιστη τιμή του λ μεταξύ των δύο ακραίων περιπτώσεων που ελαχιστοποιεί το σφάλμα, αλλά αποφεύγει και την υπερπροσαρμογή. Το λ ονομάζεται *υπερπαράμετρος* (*hyper-parameter*) και η διαδικασία της εύρεσης της βέλτιστης τιμής *ρύθμιση υπερπαραμέτρου* (*hyper-tuning*) [19].

Μπορεί να τροποποιηθεί επιπλέον η συνάρτηση σφάλματος, προκειμένου να μειωθεί ακόμα περισσότερο το ρίσκο της υπερπροσαρμογής, προσθέτοντας την νόρμα του πίνακα

ομοιότητας S. Αυτή η κανονικοποίηση ονομάζεται *elastic net*, όπου λ_A, λ_B *υπερπαράμετροι* [20] (39).

$$E(S) = \|R - RS\|_2 + \underbrace{\lambda_A \|S\|_2 + \lambda_B \|S\|_1}_{\text{Regularization term}}, \quad \text{όπου} \quad \lambda_A, \lambda_B > 0 \tag{39}$$

2.7.1 Παραγοντοποίηση Πινάκων (Matrix Factorization)

Υπάρχουν κατηγορίες αλγορίθμων οι οποίοι στοχεύουν στην μείωση των διαστάσεων των πινάκων (*dimensionality reduction*). Μία από αυτές είναι *η παραγοντοποίηση πινάκων* (*matrix factorization*). Με χρήση αυτών, μπορούν να βελτιωθούν οι υπάρχουσες τεχνικές CF, χρησιμοποιώντας μία πιο πυκνή μορφή του ΠΒΧ [21].

Για να εξηγηθεί καλύτερα η βασική ιδέα, γίνεται χρήση ενός παραδείγματος, με την βοήθεια της τεχνικής CBF. Έστω ότι για τον χρήστη u_1, είναι γνωστό πόσο του αρέσει κάθε είδος κουζίνας σε ένα εστιατόριο. Επίσης έχουμε ένα εστιατόριο $rest_1$, το οποίο έχει παραδοσιακό Ελληνικό μενού, για το οποίο είναι γνωστή η βαρύτητα κάθε κουζίνας στο μενού του (πίνακας 10).

Πίνακας 10. Παράδειγμα Βασικής Ιδέας Παραγοντοποίηση Πίνακα

Πόσο αρέσει στον u_1 κάθε κουζίνα

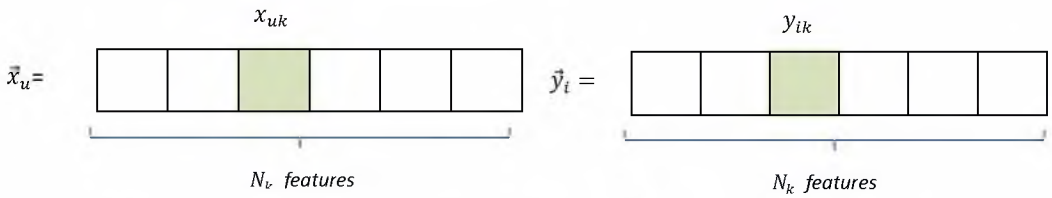
Κουζίνα	Ελληνική	Μεσογειακή	Ασιατική	Λατινική
Προτίμηση	0.9	0.7	0.1	0.2

Πόσο σημαντική είναι κάθε κουζίνα στο $rest_1$

Κουζίνα	Ελληνική	Μεσογειακή	Ασιατική	Λατινική
Βαρύτητα	1	0.8	0	0

Γενικεύοντας την ιδέα, ο χρήστης, έχει προτιμήσεις όσον αφορά τα γνωρίσματα ενός αντικειμένου και το κάθε αντικείμενο χαρακτηρίζεται σε σχέση με το κάθε γνώρισμα που το περιγράφει. Στην *παραγοντοποίηση πινάκων* τα γνωρίσματα ονομάζονται *λανθάνοντες συντελεστές* (*latent factors*) ή *χαρακτηριστικά* (*features*). Οι προτιμήσεις του χρήστη $\boldsymbol{u}$ περιγράφονται από το διάνυσμα $\vec{\boldsymbol{x}}_{\boldsymbol{u}}$, ενώ τα χαρακτηριστικά του αντικειμένου $\boldsymbol{i}$ από το διάνυσμα $\vec{\boldsymbol{y}}_{\boldsymbol{i}}$ (πίνακας 11). Το μήκος και των δύο διανυσμάτων είναι $\boldsymbol{N_k}$, που είναι το σύνολο των χαρακτηριστικών που χρησιμοποιούνται για να περιγράψουν κάθε αντικείμενο.

Πίνακας 11. Διάνυσμα Προτιμήσεων Χρήστη και Γνωρισμάτων Αντικειμένου



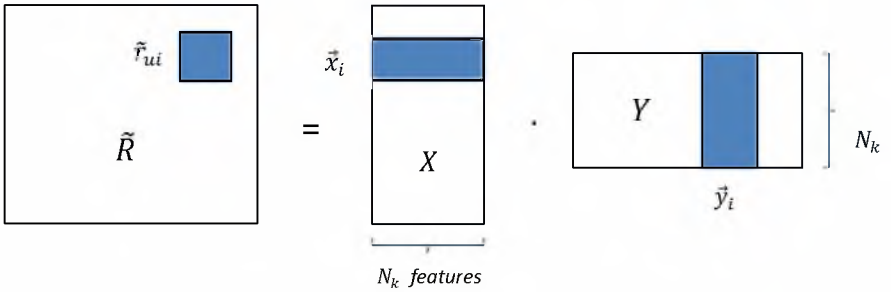
x_{uk} πόσο αρέσει στον χρήστη u το χαρακτηριστικό k , y_{ik} πόσο σημαντικό είναι το χαρακτηριστικό

Με βάση τα προηγούμενα διανύσματα, η εκτιμώμενη βαθμολογία του χρήστη u για το αντικείμενο i υπολογίζεται από την σχέση (40).

$$\tilde{r}_{ui} = \sum_k x_{uk} \cdot y_{ik} = \vec{x}_u \cdot \vec{y}_i \quad (40)$$

Αν γενικευθεί ο προηγούμενος υπολογισμός για όλους τους χρήστες και όλα τα αντικείμενα, όλες οι εκτιμώμενες βαθμολογίες είναι το γινόμενο του πίνακα χρήστη-χαρακτηριστικών X και του πίνακα αντικείμενων-χαρακτηριστικών Y (εικόνα 15).

$$\tilde{R} = X \cdot Y$$



Εικόνα 15. Εκτίμηση πίνακα $\tilde{R}$με χρήση χαρακτηριστικών

Στην μέθοδο *παραγοντοποίησης πινάκων* θεωρείται ότι δεν υπάρχουν πληροφορίες σχετικά με τις προτιμήσεις των χρηστών (πίνακας X), όπως επίσης και της σημασίας των χαρακτηριστικών για τα αντικείμενα (πίνακας Y). Οι αλγόριθμοι μηχανικής μάθησης γεμίζουν αυτούς τους ιδεώδεις πίνακες χαρακτηριστικών X^*, Y^* ελαχιστοποιώντας το σφάλμα μεταξύ του πραγματικού πίνακα χρηστών R με τον εκτιμώμενο $\tilde{R}$ (41).

$$X^*, Y^* = \min_{X,Y} \left\| R - \tilde{R} \right\|_2 \quad (41)$$

Ο πίνακας X είναι μεγέθους $N_u \times N_k$ όπου N_u το πλήθος των χρηστών και N_k το πλήθος των χαρακτηριστικών, ενώ ο πίνακας Y είναι $N_k \times N_i$, με N_i το πλήθος των αντικειμένων. Το αποτέλεσμα του γινομένου τους $\tilde{R}$ είναι $N_u \times N_i$. Τα *χαρακτηριστικά* ή αλλιώς οι

λανθάνοντες συντελεστές (*latent factors*), δεν έχουν σχέση με τα γνωρίσματα των αντικειμένων. Είναι αφηρημένες μαθηματικές έννοιες που αντιπροσωπεύουν άγνωστα γνωρίσματα προς τον χρήστη και δεν έχουν σχέση με γνωρίσματα των αντικειμένων όπως ο τύπος κουζίνας που αναφέρθηκε στο προηγούμενο παράδειγμα.

Η τιμή N_k είναι υπερ-παράμετρος που πρέπει να ρυθμιστεί με βάση τα δεδομένα. Αν υπάρχει ένας πολύ αραιός πίνακας χρηστών και επιλεχθεί μεγάλος αριθμός N_k, είναι πιθανό να υπερπροσαρμοστεί το μοντέλο μας στις λίγες διαθέσιμες κριτικές, κάτι το οποίο θα οδηγήσει το ΣΣ σε πολύ κακή επίδοση για νέες περιπτώσεις συστάσεων. Αντιθέτως αν υπάρχει ένας πυκνός πίνακας χρηστών και χρησιμοποιηθεί μικρός αριθμός *latent factors* θα περιοριστεί το σύστημα όσο αφορά την δυνατότητα του για εξατομικευμένες συστάσεις. Στην ακραία περίπτωση $N_k = 1$ το σύστημα θα προτείνει μόνο τα πιο δημοφιλή αντικείμενα, δηλαδή θα υπάρχει μη-εξατομικευμένο ΣΣ.

Όπως αναφέρθηκε και για προηγούμενες τεχνικές έτσι και στην *παραγοντοποίηση πινάκων* πρέπει να αποφευχθεί η *υπερπροσαρμογή* του μοντέλου, για αυτό προστίθενται *όροι κανονικοποίησης* στον υπολογισμό των πινάκων (42).

$$X^*, Y^* = \min_{X,Y} \left\| R - \tilde{R} \right\|_2 + \underbrace{\lambda_1 \|X\| + \lambda_2 \|Y\|}_{\text{Regularization term}} \tag{42}$$

Η χρήση των *όρων κανονικοποίησης* προτρέπουν τους πίνακες X, Y να είναι αραιοί. Οι παράμετροι λ_1, λ_2 επηρεάζουν την σημασία της *κανονικοποίησης* σε σχέση με την *συνάρτηση σφάλματος*. Μία μικρή τιμή για τα *λάμδα* θα οδηγήσει σε *υπερπροσαρμογή* αφού δεν θα επηρεάσει το μοντέλο, ενώ μία μεγάλη τιμή θα κάνει τους πίνακες να περιέχουν μόνο μηδενικά.

Για να υπολογιστούν οι βέλτιστοι πίνακες χαρακτηριστικών X, Y πρέπει να αντιμετωπιστεί ένα πρόβλημα του πίνακα χρηστών που αφορά την μορφή του. Ο πίνακας αυτός είναι στο μεγαλύτερο μέρος του γεμάτος μηδενικά, τα οποία πρέπει να αποφασιστεί πως θα αντιμετωπιστούν.

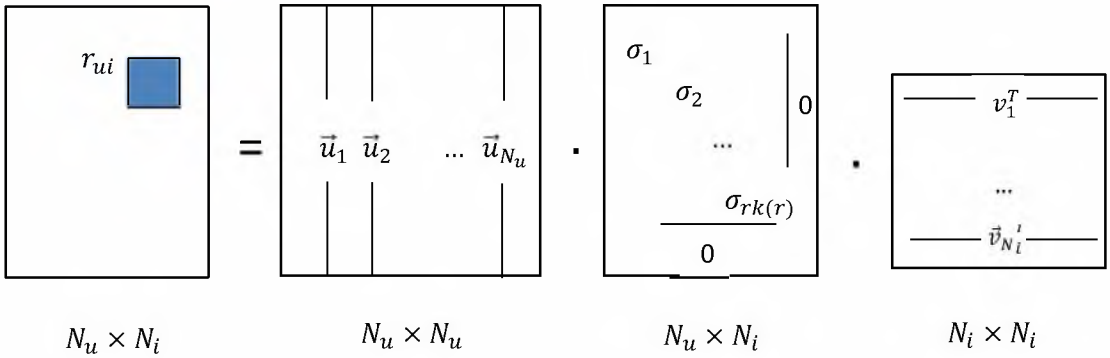
Υπάρχουν δύο υποθέσεις που μπορούν να χρησιμοποιηθούν, *απόντα ως τυχαία* (*missing as random*), στην οποία θεωρείται ότι ένα στοιχείο του ΠΒΧ που δεν έχει τιμή, είναι το ίδιο πιθανό να αρέσει ή όχι στον χρήστη. Στην δεύτερη υπόθεση, *απόντα ως αρνητικά* (*missing as negative*) θεωρούμε ότι στον χρήστη δεν αρέσει κάτι που δεν έχει βαθμολογήσει. Η

δεύτερη υπόθεση χρησιμοποιείται περισσότερο σε περιπτώσεις που βελτιστοποιείται η *ακρίβεια* και η *ανάκληση*. Η *missing as random* βασίζεται στην ευκλείδεια νόρμα ενώ η *missing as negative* βασίζεται στην νόρμα Frobenius [22].

2.7.2 Παραγοντοποίηση ιδιαζουσών τιμών (SVD)

Η *Παραγοντοποίηση ιδιαζουσών τιμών* (*SVD*) είναι μία μέθοδος στην οποία χωρίζεται ένας πίνακας σε γινόμενο τριών πινάκων U, Σ και V^T [23] (εικόνα 16) (43). Χρησιμοποιείται για να χωριστεί ο ΠΒΧ, *R*. Όπως αναφέρθηκε προηγουμένως ο πίνακας *R* είναι μεγέθους N_u (αριθμός χρηστών) επί N_i (αριθμό αντικειμένων).

$$R = U \cdot \Sigma \cdot V^T \tag{43}$$



Εικόνα 16. Παραγοντοποίηση ιδιαζουσών τιμών (SVD)

Ο πρώτος πίνακας *U* είναι τετραγωνικός και ορίζεται με *αριστερά ιδιάζοντα διανύσματα*. Οι στήλες του είναι ιεραρχικά ταξινομημένες με βάση την ικανότητα τους να περιγράψουν την απόκλιση των στηλών του πίνακα *R*. Αυτό σημαίνει ότι οι πιο σημαντικές στήλες που τον περιγράφουν, είναι τοποθετημένες αριστερά. Οι γραμμές του αναπαριστούν χρήστες, ενώ οι στήλες *λανθάνοντες συντελεστές*, οπότε κάθε κελί του πίνακα δηλώνει πόσο σημαντικό είναι για τον χρήστη της συγκεκριμένης γραμμής το συγκεκριμένο κρυφό χαρακτηριστικό.

Ο δεύτερος πίνακας *Σ*, είναι ένας διαγώνιος πίνακας με βαθμό ίσο με τον βαθμό του πίνακα R, με την διαγώνιο να περιέχει τις ιδιάζουσες τιμές του R. Οι ιδιάζουσες τιμές είναι μη μηδενικές και ταξινομημένες με φθίνουσα σειρά. Κάθε ιδιάζουσα τιμή αντιστοιχεί στην ικανότητα κάθε χαρακτηριστικού να περιγράψει αντικείμενα και χρήστες στον πίνακα R.

Η στήλη (0,0) του πίνακα αντιστοιχεί στο χαρακτηριστικό 1, η στήλη (1,1) στο χαρακτηριστικό 2 κτλ.

Ο τρίτος πίνακας V^T είναι επίσης τετραγωνικός και ορίζεται με *δεξιά ιδιάζοντα διανύσματα*. Οι γραμμές του είναι ιεραρχικά ταξινομημένες με βάση την ικανότητα τους να περιγράψουν την απόκλιση των γραμμών του πίνακα R, το οποίο σημαίνει ότι οι πιο σημαντικές γραμμές που περιγράφουν τον R είναι τοποθετημένες ψηλά στον πίνακα. Οι στήλες αντιπροσωπεύουν αντικείμενα ενώ οι γραμμές *κρυφά χαρακτηριστικά*. Η πρώτη γραμμή αντιστοιχεί στο χαρακτηριστικό 1 κτλ. Κάθε κελί καθορίζει κατά πόσο είναι σημαντικό ένα χαρακτηριστικό για να περιγράψει το αντικείμενο.

2.8 Σύνοψη κεφαλαίου

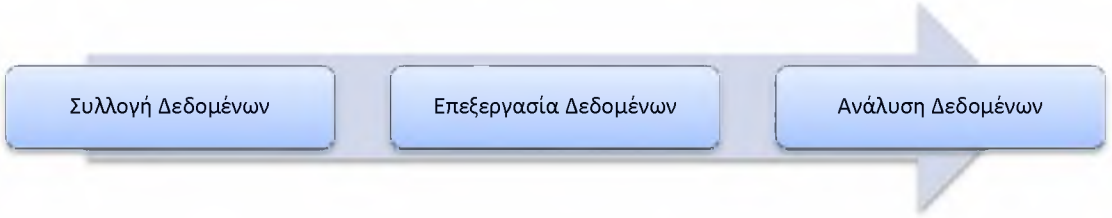
Σε αυτό το κεφάλαιο έγινε σύντομη ανάλυση στην θεωρία πίσω από τα ΣΣ. Έγινε περιγραφή τον βασικών μεθόδων και αλγορίθμων που χρησιμοποιούνται στα συστήματα αυτά, οι οποίες αναφέρονται και χρησιμοποιούνται κατά την διάρκεια των επόμενων κεφαλαίων στην παρουσίαση της υλοποίησης της εφαρμογής. Επίσης έγινε αναφορά στους τρόπους αξιολόγησης της ποιότητας ενός ΣΣ.

ΚΕΦΑΛΑΙΟ 3

ΠΡΟΣΕΓΓΙΣΗ ΤΗΣ ΕΡΓΑΣΙΑΣ ΚΑΙ ΤΩΝ ΔΕΔΟΜΕΝΩΝ

3.1 Συλλογή των δεδομένων

Τα πρώτα βήματα της εργασίας ήταν η συλλογή, η επεξεργασία και ανάλυση των δεδομένων (εικόνα 17). Σε αυτό το κεφάλαιο θα αναλύονται αυτά τα βήματα.



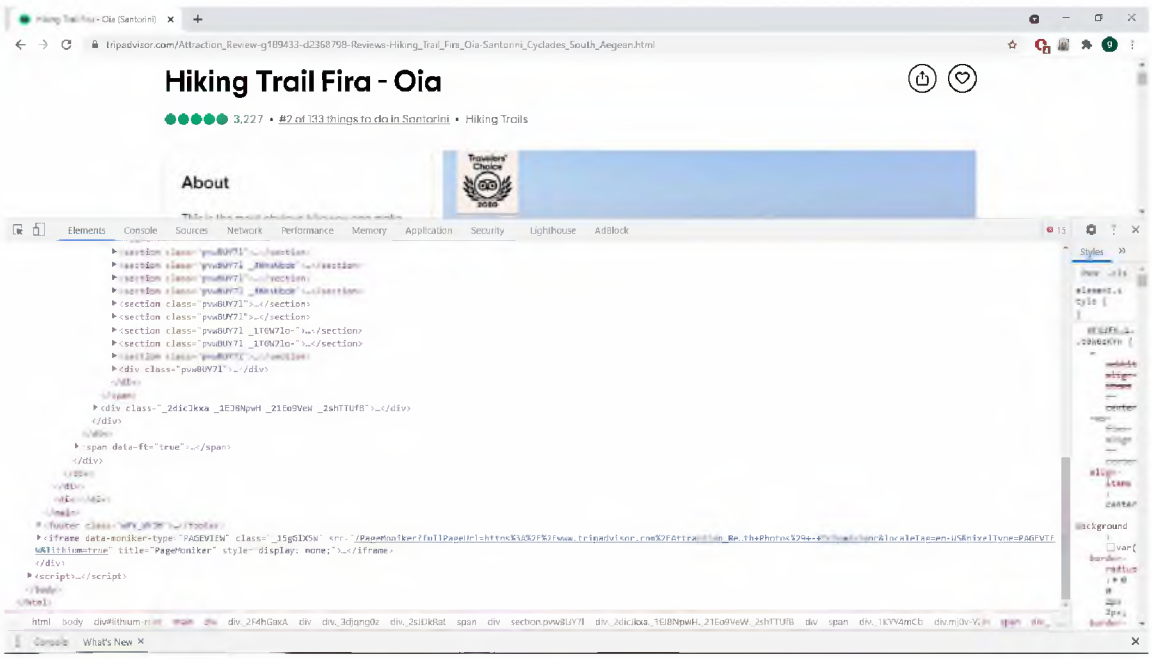
Εικόνα 17. Ροή συλλογής επεξεργασίας ανάλυσης δεδομένων

3.1.1 Σχετικά με την διαδικασία

Παρόλο τον μεγάλο όγκο που υπάρχει όσο αφορά τα δεδομένα για επιχειρήσεις και υπηρεσίες στον κλάδο του τουρισμού παγκοσμίως, διαθέσιμο για έρευνα και πειράματα παρέχεται μόνο το σετ δεδομένων του Yelp (https://www.yelp.com/dataset), που αφορά κυρίως επιχειρήσεις στις Ηνωμένες Πολιτείες και τον Καναδά, όπως και το Foursquare μέσω του API του, που δυστυχώς δεν έχει μεγάλο όγκο για την Ελλάδα.

Λόγω αυτού, η συλλογή των δεδομένων που χρησιμοποιήθηκε συλλέχθηκε (*scraped*) από το Tripadvisor. Τα δεδομένα αποκτήθηκαν επιλεκτικά με φίλτρα στο HTML αρχείο του κάθε αντικειμένου (εικόνα18), μέσω ενός προγράμματος γραμμένο σε python, με χρήση του framework ανοιχτού κώδικα της Python, Scrapy. Η συλλογή έγινε από τον ιστότοπο του Tripadvisor.com, από την παγκόσμια έκδοση λόγω του ότι η εργασία αφορά κυρίως τουρίστες του εξωτερικού.

Το Tripadvisor αποτελεί μία από τις μεγαλύτερες εταιρίες ταξιδιωτικών εταιριών παγκοσμίως, λειτουργόντας σε μορφή ιστοτόπου και εφαρμογής ασύρματων συσκευών. Προσφέρει κρατήσεις ξενοδοχείων και μεταφορών, κριτικών αντικειμένων (ξενοδοχεία, εστιατόρια, δραστηριότητες) από χρήστες όπως και μεγάλο όγκο πληροφορίας για τα αντικείμενα αυτά.



Εικόνα 18. Αρχείο κώδικα HTML αρχείου

Τα δεδομένα αφορούν τρεις κατηγορίες αντικειμένων.

- Διαμονής που περιλαμβάνει τις υποκατηγορίες ξενοδοχεία, bed & breakfast, διαμερίσματα και βίλες.
- Φαγητού, καφέ και μπαρ
- Δραστηριότητες (μουσεία, αξιοθέατα, φύση κτλ.)

Αρχικός στόχος της εργασίας ήταν να γίνει συλλογή εκτός από τον ιστότοπο του Tripadvisor, από το Google όπως επίσης και από το Facebook. Το πρόβλημα που παρουσιάστηκε ήταν στις άδειες των συγκεκριμένων ιστότοπων, οι οποίες είναι πολύ αυστηρές και απαιτούν ενοικίαση πολύ ακριβών προσωπικών proxies για την επίτευξη της συλλογής δεδομένων. Στον ιστότοπο του Tripadvisor εμφανίστηκαν επίσης κάποιες δυσκολίες, παρόλο που ήταν ελαφρώς πιο ελαστικοί στις άδειες τους, οι οποίες αντιμετωπίστηκαν χρησιμοποιώντας 5 ιδιωτικά proxies από τον ιστότοπο Limeproxies (https://www.limeproxies.com/).

3.1.2 Πληροφορίες για τα δεδομένα

Τα δεδομένα συλλέχθηκαν για κάθε κατηγορία μέσω ξεχωριστού από την εφαρμογή script, αποθηκεύοντας τα δεδομένα σε μορφή JSON. Για την χρήση τους στην εφαρμογή έγινε εισαγωγή των JSON αρχείων σε βάση SQLite.

Τα πεδία που συλλέχθηκαν είναι :

⇒ Για τα ξενοδοχεία (εικόνα 19) :

- Αναγνωριστικό (Hotel id) με αύξουσα σειρά ως κύριο κλειδί
- Κατηγορία (Category) .

  Η κατηγορία με τιμές ξενοδοχεία, bed & breakfast, διαμερίσματα και βίλες, χρησιμοποιήθηκε για την συμπλήρωση δεδομένων που έλειπαν κυρίως τιμής.
- Όνομα (Hotel name)
- Πόλη (City)
- Περιοχή (Region)
- Διεύθυνση (Address).

Στα ξενοδοχεία και τις δραστηριότητες οι συντεταγμένες δεν ήταν δυνατό να συλλεχθούν άμεσα, λόγω μασκών JavaScipt, στους ενσωματωμένους χάρτες της Google. Μέσω της διεύθυνσης, σε δεύτερο χρόνο και χρησιμοποιώντας το Geocoding API της Google, έγινε μετατροπή των διευθύνσεων σε συντεταγμένες που είναι από τα βασικά δεδομένα στην εργασία.

- Μέση Βαθμολογία (Mean Rating)
- Τιμή (Price)
- Παροχές (Amenities)
- Βαθμολογία Τοποθεσίας (Location rating)
- Βαθμολογία Καθαριότητας (Clean rating)
- Βαθμολογία Προσωπικού (Service rating)
- Βαθμολογία Κόστους (Value rating)

[14]:

	hotel_id	name	category	city	region	address	country	rating	expirience	location_rating	clean_rating	service_rating	value_rating	amenities	price	lat	lng
0	1	Apanemo	Hotels	Akrotiri	Santorini	Akrotiri 84700 Greece	Greece	5.0	Excellent	4.5	5.0	5.0	4.5	[Free Parking, Wifi Available, Pool, Free brea...	175.0	36.358986	25.396448
1	2	Pezoules	Hotels	Oia	Santorini	Thiras - Ias, Oia 84702 Greece	Greece	5.0	Excellent	5.0	5.0	5.0	5.0	[Paid Parking, Wifi Available, Pool, Yoga clas...	392.0	36.462143	25.377725
2	3	Phidias Piraeus Hotel	Hotels	Piraeus	Piraeus Region	189 Koudouriotou Passalimani, Piraeus 185 35 Gr...	Greece	4.5	Excellent	4.5	5.0	5.0	5.0	[Paid Parking, Wifi Available, Sauna, Breakfas...	61.0	37.939506	23.647036

Εικόνα 19. Σετ ξενοδοχείων

⇒ Για τις δραστηριότητες (Attractions) (εικόνα 20):

- Αναγνωριστικό (Attraction id) με αύξουσα σειρά ως κύριο κλειδί
- Όνομα (Attraction name)
- Κατηγορία (Category).

Κάθε δραστηριότητα ανήκει σε διαφορετικές κατηγορίες, για παράδειγμα μία πολύωρη ξενάγηση στα αρχαία των Αθηνών ανήκει στα αξιοθέατα, όπως επίσης και στις ημερήσιες οργανωμένες (με ξεναγό) δραστηριότητες. Στην συλλογή δεν ήταν δυνατό να συλλεχθούν απευθείας όλες τις κατηγορίες στις οποίες ανήκει το κάθε αντικείμενο λόγω μη ύπαρξης της πληροφορίας στο HTML αρχείο της κάθε σελίδας. Για αυτό τον λόγο κάθε δραστηριότητα συλλέχθηκε πολλαπλές φορές, όσες και οι κατηγορίες στις οποίες ανήκε και συνδυάστηκε σε δεύτερο χρόνο σε μία.

- Πόλη (City)
- Διεύθυνση (Address)
- Τιμή (Price)
- Μέση Βαθμολογία (Mean rating)
- Διάρκεια (Duration)
- Περιγραφή (Description)

t[6]:

	attraction_id	name	country	city	province	duration	address	price	rating	category	lat	lng
0	1	Delphi Day Trip from Athens	Greece	Athens	Attica	10.0	Leoforos Vasilisis Amalias 10, Athina 105 57,...	108.55	4.5	[Featured Tours and Tickets, Recommended Exper...	37.973953	23.735055
1	2	Athens Scenic Bike Tour	Greece	Athens	Attica	3.5	Athanasiou Diakou 16, Athina 117 42, Greece	36.92	5.0	[Recommended Experiences, Featured Tours and T...	37.968144	23.731033
2	3	Semi Private Standard	Santorini Catamaran Cr...	Greece	Fira	Santorini	5.0	Fira	111.01	5.0	[Recommended Experiences, Featured Tours and T...	41.351923	2.130679

Εικόνα 20. Σετ δραστηριοτήτων

⇒ Για τις Εστιατόρια (Restaurants) (εικόνα 21):

- Αναγνωριστικό (Restaurant id) με αύξουσα σειρά ως κύριο κλειδί
- Όνομα (Restaurant name)
- Διεύθυνση (Address)
- Μέση βαθμολογία (Mean rating)
- Τιμή (Price)

- Βαθμολογία Φαγητού (Food rating)
- Βαθμολογία Προσωπικού (Service rating)
- Βαθμολογία Τιμής (Value rating)
- Γεωγραφικό πλάτος (Latitude)
- Γεωγραφικό μήκος (Longitude)

  Στην περίπτωση των εστιατορίων το HTML αρχείο ήταν διαφορετικό, επιτρέποντας να συλλεχθούν άμεσα οι συντεταγμένες.
- Κουζίνα (Cuisine)

  Τύπος κουζίνας/παροχής του εστιατορίου για παράδειγμα Ελληνική, Μεσογειακή, Ιταλικό, Καφέ κτλ.
- Γεύματα (Meals)

  Κατηγορία γεμάτων που προσφέρει. Περιλαμβάνει τις τιμές Πρωινό (Breakfast), Δεκατιανό (Brunch), Μεσημεριανό (Lunch), Βραδινό (Dinner), Late-night, Καφέ (Café) και Ποτά (Drinks)
- Διατροφή (Diet)
- Χαρακτηριστικά (Features)

	rest_id	name	address	price	rating	food_rating	service_rating	value_rating	lat	lng	cuisine	diet	meals	features	user	review_date	review	image_urls	img_id	images
0	1.0	La Creperie Cafe & Crepes	Elfthilli Odos Portarias Zagoras 101, Portaria, ...	$	4.0	4.5	4.5	4.5	39.390470	22.998606	[Cafe, BrewPub, FastFood]	[]	[Breakfast, Lunch, LateNight, Drinks]	[ServesAlcohol, FreeWifi, TableService, Takeou...	NaN	NaN	NaN	NaN	NaN	NaN
6	2.0	Lagini Rock Cafe	__, Volou Portarias, Portaria, Volos 37011, Greece	$	5.0	5.0	5.0	5.0	39.389347	22.997894	[Cafe, Bar, Pub, WineBar]	[]	[LateNight, Drinks]	[Seating, ParkingAvailable, StreetParking, Ser...	NaN	NaN	NaN	NaN	NaN	NaN
18	3.0	Pelades Eatery & Drink	Makrinitsa, Volos 37011, Greece	$	5.0	5.0	5.0	5.0	39.401920	22.987417	[Greek]	[]	[Lunch, Dinner]	[Reservations, AcceptsCreditCards]	NaN	NaN	NaN	NaN	NaN	NaN

Εικόνα 21. Σετ εστιατορίων

Τα πάνω δεδομένα αφορούν τα ξενοδοχεία, τις δραστηριότητες και τα εστιατόρια όσο αφορά το αντικείμενο. Για κάθε ένα από αυτά έγινε συλλογή των κριτικών ώστε να δημιουργηθεί ο ΠΒΧ της κάθε κατηγορίας. Τα δεδομένα κριτικών είναι :

- Αναγνωριστικό (User id) με αύξουσα σειρά ως κύριο κλειδί
- Όνομα χρήστη (Username)
- Προφίλ χρήστη (User profile)
- Βαθμολογία (Rating)
- Κριτική (Review)
- Ημερομηνία (Review date)

Η αρχική ιδέα στην χρήση της ημερομηνίας ήταν να γίνει διαχωρισμός των προτάσεων με βάση την εποχή, όπως επίσης να χρησιμοποιηθούν κριτικές μόνο εντός 3ετίας, για να είναι όσον το δυνατό πιο έγκυρες. Στην πρώτη περίπτωση παρατηρήθηκε στην ανάλυση, ότι στην συντριπτική πλειοψηφία, οι κριτικές ήταν τους καλοκαιρινούς μήνες, κάτι απολύτως λογικό, εφόσον οι χρήστες ήταν τουρίστες του εξωτερικού, οπότε δεν υπήρχε όφελος από την χρήση της. Για την δεύτερη περίπτωση επειδή ο όγκος των δεδομένων ήταν οριακός, επίσης δεν έγινε χρήση της, αφού δεν υπήρχε η πολυτέλεια να μην ληφθούν υπόψιν δεδομένα.

Τέλος για κάθε αντικείμενο έγινε λήψη της εικόνας προφίλ του, η οποία χρησιμοποιείται στο γραφικό περιβάλλον της εφαρμογής.

3.2 Επεξεργασία των Δεδομένων

Τα πραγματικά δεδομένα είναι ελλιπή, με θόρυβο και ασύμφωνα. Στην περίπτωση της εργασίας, επειδή τα δεδομένα έχουν αποκτηθεί από τα HTML αρχεία που προαναφέρθηκαν, υπάρχουν σημεία που χρειάζονται προ-επεξεργασία και καθαρισμό λόγω δυναμικών αλλαγών στην μορφή των σελίδων. Όσο καλύτερα δεδομένα υπάρχουν τόσο καλύτερα αποτελέσματα επιτυγχάνονται [30].

Ο εντοπισμός των ελλιπών τιμών, η διαγραφή των πολύ διαφορετικών τιμών (*outliers*) και ο χειρισμός ασύμφωνων δεδομένων είναι από τα πιο σημαντικά στοιχεία της προ επεξεργασίας. Ο καθαρισμός στην συγκεκριμένη εργασία, περιλαμβάνει τα εξής βήματα. Απαλοιφή δεδομένων με αντίγραφα, χειρισμός ακραίων τιμών, χειρισμός ελλιπών δεδομένων και διακριτικοποίηση δεδομένων.

3.2.1 Απαλοιφή διπλότυπων και ακραίων τιμών

Για την απαλοιφή διπλότυπων στα σετ των αντικειμένων έγινε χρήση του συνδυασμού ονόματος και διεύθυνσης με σκοπό να εντοπιστούν. Ένα ξενοδοχείο όπως και ένα εστιατόριο, είναι πολύ πιθανό να έχει ένα κοινό όνομα, για τον λόγο αυτό χρησιμοποιείται και η διεύθυνση. Στις κριτικές μετά από έλεγχο ανακαλύφθηκε, ότι το όνομα χρήστη με το οποίο υπογράφουν τις κριτικές τους, δεν είναι μοναδικό όπως συνηθίζεται και για αυτό τον

λόγο η απαλοιφή των διπλών τιμών από το σετ των κριτικών έγινε με χρήση τόσο του ονόματος όσο και του προφίλ.

Οι ακραίες τιμές κυρίως στο πεδίο της τιμής στα ξενοδοχεία και των δραστηριοτήτων όσο και στο πεδίο της διάρκειας στις δραστηριότητες λόγω μικρού όγκου, διορθώθηκαν χειρόγραφα.

3.2.2 Ελλιπείς τιμές

Οι ελλιπείς τιμές, μπορούν να επηρεάσουν σημαντικά την ανάλυση των δεδομένων οδηγώντας σε λάθος συμπεράσματα. Οι συνηθισμένες τεχνικές που χρησιμοποιούνται στον χειρισμό των ελλιπών δεδομένων είναι οι εξής.

- Διαγραφή του γνωρίσματος αν οι εγγραφές στις οποίες λείπει είναι πολλές σε πλήθος.
- Διαγραφή της συγκεκριμένης εγγραφής στην οποία έχουμε ελλιπή δεδομένα αν το πλήθος είναι μικρό.
- Συμπλήρωση των ελλιπών τιμών. Σε αυτή την περίπτωση χρησιμοποιείται συνήθως ο μέσος όρος.

Υπήρχαν αρκετές τιμές σε διαφορετικά πεδία που έπρεπε να αντιμετωπιστούν. Αρχικά στα ξενοδοχεία όπως και στις δραστηριότητες υπήρχαν κάποιες ελλιπείς τιμές στις διευθύνσεις. Αυτό δημιουργούσε το πρόβλημα σε επόμενο στάδιο που έπρεπε να γίνει μετατροπή σε συντεταγμένες. Σε αυτές τις περιπτώσεις τέθηκε ως διεύθυνση η πόλη/χωριό που υπήρχε ως πληροφορία για το συγκεκριμένο αντικείμενο.

Άλλο ένα πεδίο ήταν η διάρκεια στις δραστηριότητες. Ως λύση σε αυτή την περίπτωση, έγινε διαχωρισμός ανά κατηγορία δραστηριότητας (μουσείο, θαλάσσια σπορ κτλ.) και ως επόμενο βήμα υπολογισμός του μέσου όρου τιμής ανά ώρα, για την κατηγορία δραστηριότητας. Σε λίγες περιπτώσεις που δεν υπήρχε ούτε πληροφορία τιμής διαγράφτηκε η εγγραφή.

Η τιμή στα ξενοδοχεία επίσης είχε ορισμένες περιπτώσεις που έπρεπε να συμπληρωθούν. Τα ξενοδοχεία είναι μία κατηγορία αντικειμένου στα οποία η τοποθεσία παίζει τον πολύ σημαντικό ρόλο σε ότι αφορά την τιμή όπως επίσης και ο τύπος τους, για παράδειγμα αν

είναι διαμέρισμα, βίλα κτλ. Αρχικά, υπολογίστηκε ο μέσος όρος από όλα τα καταλύματα και έγινε *ομαδοποίηση* (*clustering*) τους με βάση τις συντεταγμένες τους. Στην συνέχεια, με βάση τον μέσο όρο κάθε ομάδας (*cluster*), υπολογίστηκε σε ποσοστό, πόσο πιο φθηνό η ακριβό είναι το συγκεκριμένο cluster σε σχέση με τον συνολικό μέσο όρο. Έχοντας υπολογίσει τον μέσο όρο για κάθε τύπο καταλύματος και με χρήση του ποσοστού βρέθηκε η κάθε τιμή. Για παράδειγμα βρέθηκε στην διαδικασία της ανάλυσης ότι η Σαντορίνη είναι κατά 72% πιο ακριβή στα καταλύματα από το μέσο κατάλυμα της Ελλάδας και ότι η μέση τιμή του κάθε Bed and Breakfast τύπου καταλύματος είναι 60ευρώ. Σε αυτή την περίπτωση η τιμή που ψάχνουμε είναι 102ευρώ. Ως επιβεβαίωση παρατηρούμε ότι οι μέση τιμή κάθε τύπου πριν και μετά την προεπεξεργασία είναι παρόμοια (εικόνα 22).

```
print('hotel mean : ', hotels_mean , ' bbs mean :' , bbs_mean, ' condos mean : ', condos_mean, 'villas me
hotel mean :  167.23012552301256  bbs mean : 80.53846153846153  condos mean :  49.25 villas mean :  93.25
```

```
print('hotel mean : ', hotels_mean , ' bbs mean :' , bbs_mean, ' condos mean : ', condos_mean, 'villas mean : ' ,villas_mean)
hotel mean :  165.17171788319774  bbs mean : 75.62776578681355  condos mean :  40.93729475085886 villas mean :  93.1838857514
35
```

Εικόνα 22. Μέση τιμή ανά κατηγορία καταλύματος πριν και μετά την συμπλήρωση ελλεπών τιμών

3.2.3 Μετασχηματισμός και Διακριτοποίηση (Discretization) Δεδομένων

Πολύ αλγόριθμοι αποδίδουν καλύτερα με διακριτές παρά συνεχείς μεταβλητές. Οι συνεχείς μεταβλητές όπως η τιμή σε όλα τα σετ που χρησιμοποιούνται στην συγκεκριμένη εργασία περιέχουν ακραίες τιμές οι οποίες επηρεάζουν αρνητικά. Η τιμή σαν ποσότητα δεν μας απασχολεί, αλλά το εύρος της. Σκοπός είναι, να παρέχει το μοντέλο την δυνατότητα πρότασης πλάνων ταξιδιού και μεμονωμένων αντικειμένων για όλες τις οικονομικές δυνατότητες. Για την μετατροπή της τιμής, χρησιμοποιήθηκε η μέθοδος *διακριτικοποίησης ίσης συχνότητας* (*Equal-Frequency Discretization*) [41].

Για τα ξενοδοχεία :

Εύρος	Περιγραφή	Ετικέτα
<=75	Budget	0
75-160	Mid-Range	1
>=160	Luxury	2

Για τις δραστηριότητες :

Εύρος	Περιγραφή	Ετικέτα
<=50	Budget	0
50-100	Mid-Range	1
>=100	Luxury	2

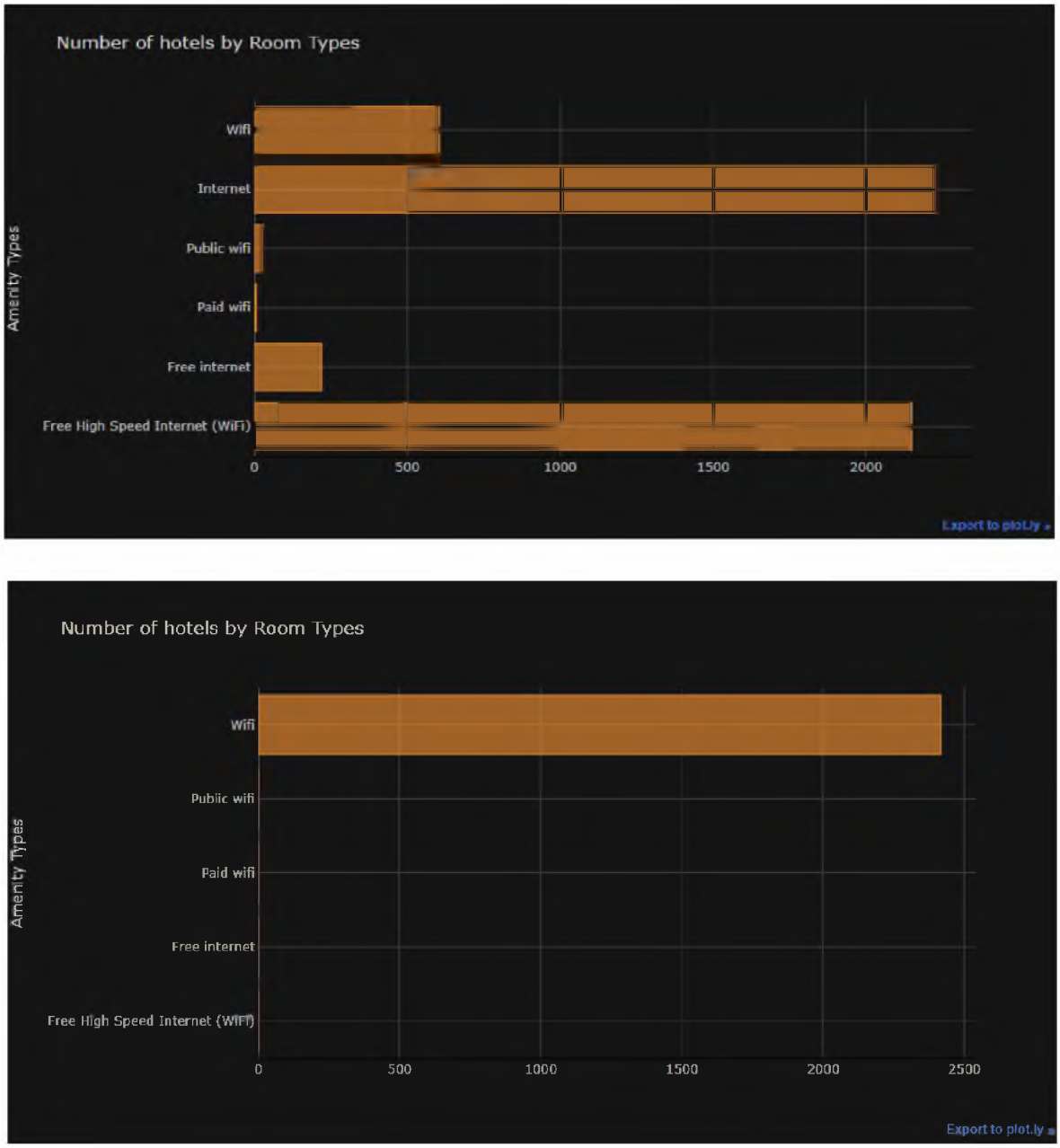
Τα εστιατόρια, συλλέχθηκαν εξαρχής με κατηγορία τιμής. Έγινε μετατροπή της κατηγορικής μεταβλητής με τιμές $, $$-$$, $$$$ σε 0, 1 ,2 αντίστοιχα. Τέλος οι κατηγορικές μεταβλητές σε μορφή λίστας συμβολοσειρών όπως οι παροχές στα ξενοδοχεία (amenities), η κατηγορία της δραστηριότητας (attraction category), η κουζίνα (cuisine) και τα γεύματα (meals) στα εστιατόρια, μετασχηματίστηκαν μέσω της τεχνικής *One Hot Encoding* [42].

3.3 Επιλογή χαρακτηριστικών (Feature selection)

Η επιλογή των σημαντικών χαρακτηριστικών είναι πολύ βασικό βήμα στην προ επεξεργασία πριν την ανάλυση. Περιλαμβάνει την επιλογή ενός υποσυνόλου από το σύνολο των χαρακτηριστικών, τα πιο σημαντικά, για την κατασκευή του μοντέλου. Η συγκεκριμένη διαδικασία έχει αρκετά οφέλη όπως η απόδοση του μοντέλου, μείωση των πόρων αποθήκευσης κτλ.

Κατηγορικά χαρακτηριστικά όπως οι παροχές του ξενοδοχείου, η κατηγορία της δραστηριότητας ή οι τύποι κουζίνας που προτιμά, έχουν σημαντικό ρόλο στην κατασκευή του προφίλ του χρήστη. Παρόλα αυτά, περιέχουν τιμές οι οποίες εμφανίζονται πολύ σπάνια στα δεδομένα, με μόνο αποτέλεσμα την αύξηση σε απαιτήσεις αποθήκευσης όσο και την μείωση της πρακτικότητας στο τελικό κομμάτι της εργασίας (το γραφικό περιβάλλον), που ζητείται από τον χρήστη να προσδιορίσει τις προτιμήσεις του.

Για παράδειγμα στις παροχές του ξενοδοχείου υπάρχουν 211 διαφορετικές τιμές από τις οποίες πολλές είναι όμοιες στην σημασία αλλά με διαφορετική ονομασία. Αρχικά έγινε ομαδοποίηση των όμοιων τιμών και στην συνέχεια χρησιμοποιήθηκαν μόνο τις πιο σημαντικές τιμές, διαγράφοντας τις υπόλοιπες (εικόνα 23).



Εικόνα 23. Παροχές ξενοδοχείου πριν και μετά την ομαδοποίηση

Η συγκεκριμένη διαδικασία εφαρμόστηκε στο σετ των δραστηριοτήτων για το γνώρισμα της κατηγορίας, όπως επίσης και στο σετ εστιατορίων στο γνώρισμα της κουζίνας. Ο

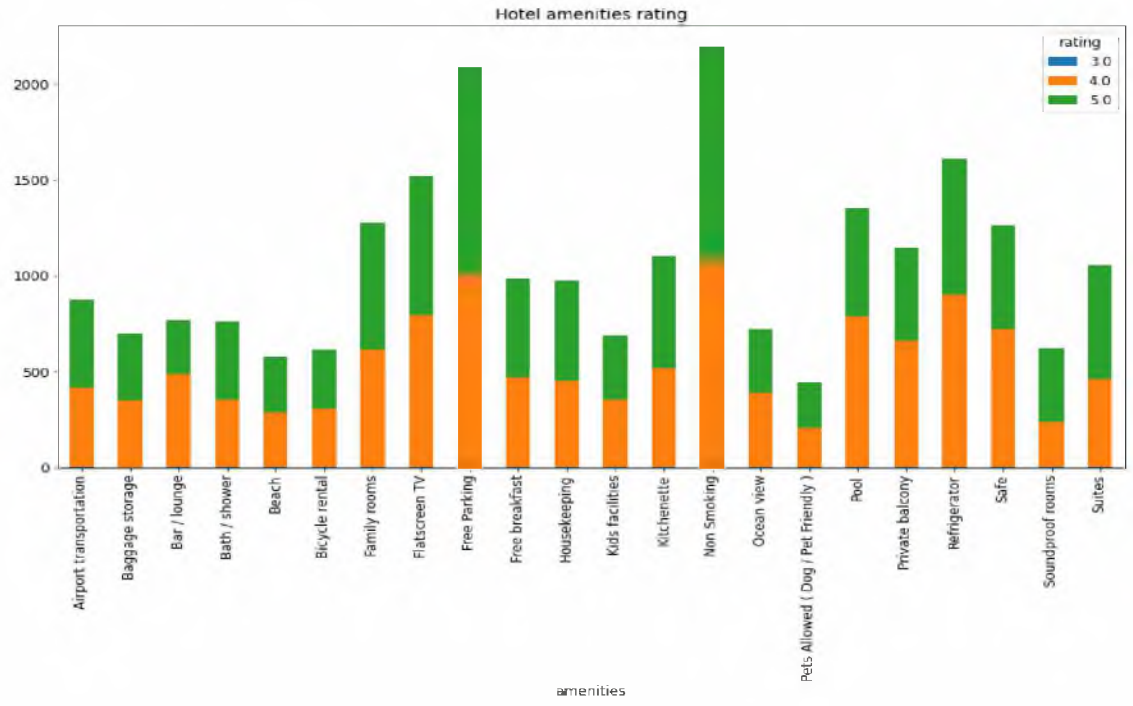
τελικός αριθμός των τιμών που διατηρήθηκαν είναι 22 παροχές ξενοδοχείου, 25 είδη κουζίνας και 15 κατηγορίες δραστηριοτήτων.

3.4 Διερευνητική Ανάλυση Δεδομένων (ΔΑΔ)

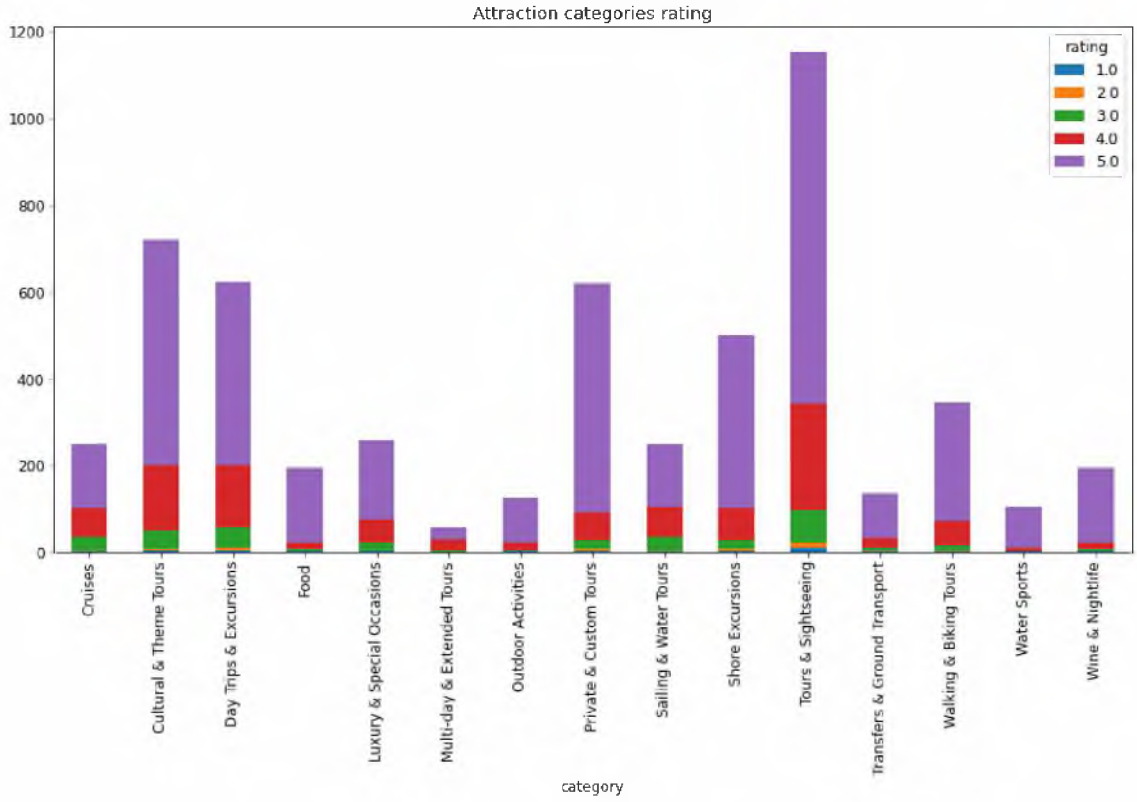
Τα δεδομένα πρέπει να αναλυθούν ώστε να οδηγήσουν σε σωστό αποτέλεσμα. Χρησιμοποιώντας την ανάλυση, μπορούμε να πάρουμε τις αποφάσεις που θέλουμε [31]. Προκειμένου να γίνει καλύτερη κατανόηση των δεδομένων που έχουν συλλεχθεί και χρησιμοποιηθεί για τις συστάσεις, εκτελείται *διερευνητική ανάλυση δεδομένων* στα γνωρίσματα του κάθε σετ ξεχωριστά.

Αυτό το βήμα είναι σημαντικό, ώστε να βοηθήσει στην κατανόηση της κατανομής της τιμής και της βαθμολογίας σε διαφορετικές κατηγορίες, όπως και των λοιπών γνωρισμάτων. Η ΔΑΔ δίνει την δυνατότητα στην κατάληξη της τελικής απόφασης, της επιλογής συγκεκριμένων γνωρισμάτων από το κάθε σετ για την δημιουργία του προφίλ του χρήστη, με βάση την απόκλιση των τιμών στα δεδομένα για την κάθε κατηγορία. Στην συνέχεια παρουσιάζονται μερικά από τα βασικά γραφήματα.

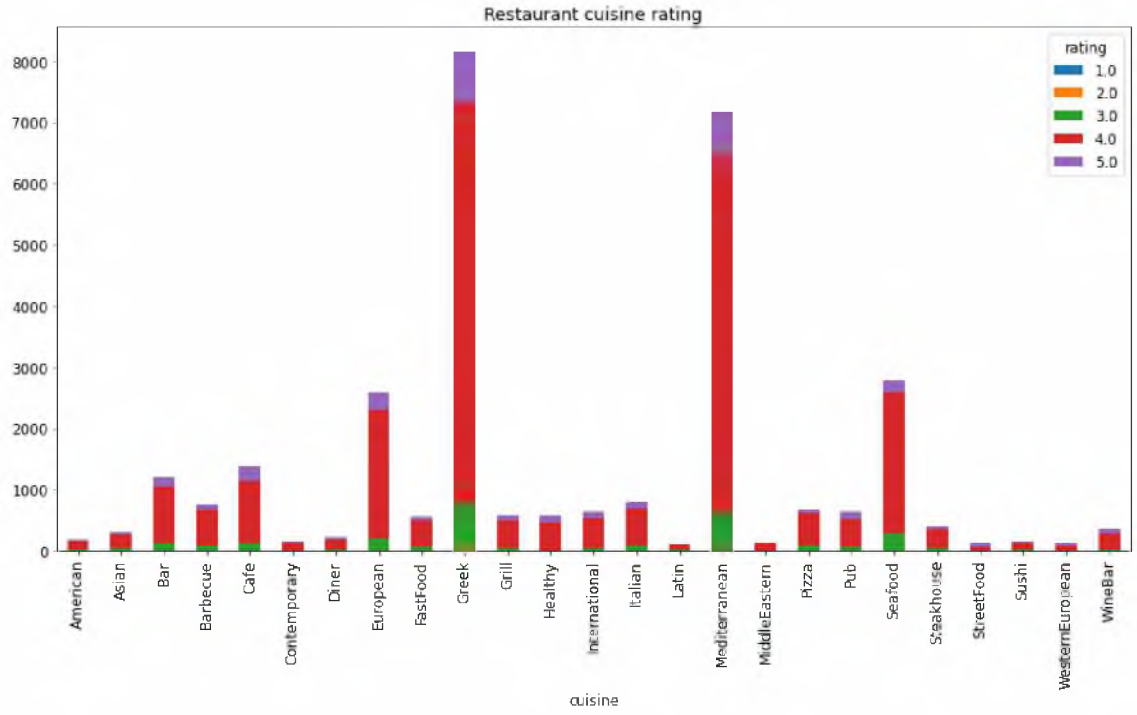
Η διακύμανση των βαθμολογιών για τα γνωρίσματα παροχές ξενοδοχείων, κατηγορία δραστηριότητας, κουζίνα και των γευμάτων (εικόνα 24-27).



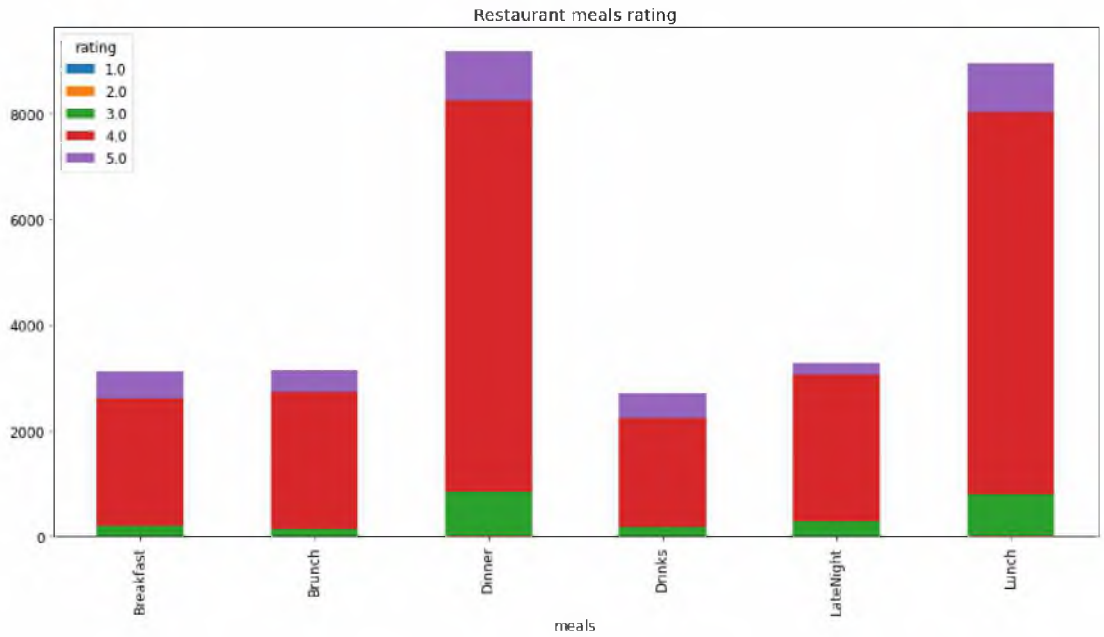
Εικόνα 24. Διακύμανση βαθμολογιών ανά παροχές



Εικόνα 25. Διακύμανση βαθμολογιών ανά κατηγορία δραστηριότητας

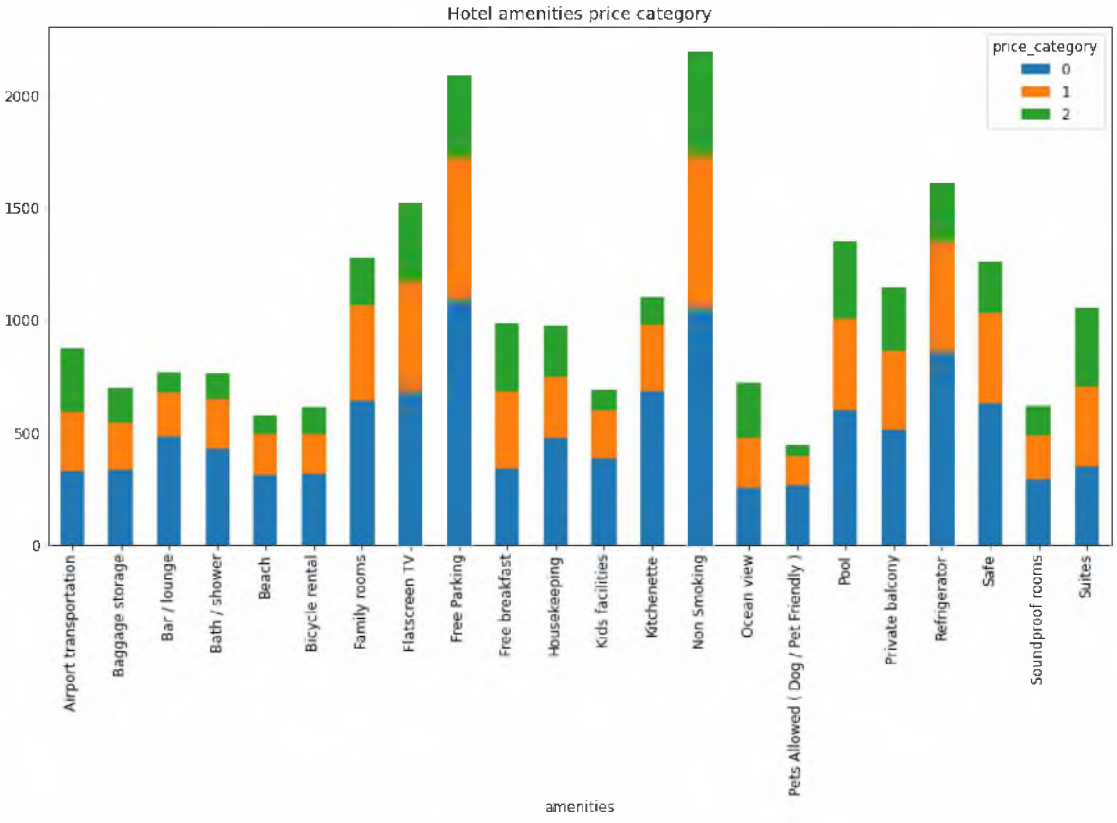


Εικόνα 26. Διακύμανση βαθμολογιών ανά τύπο κουζίνας

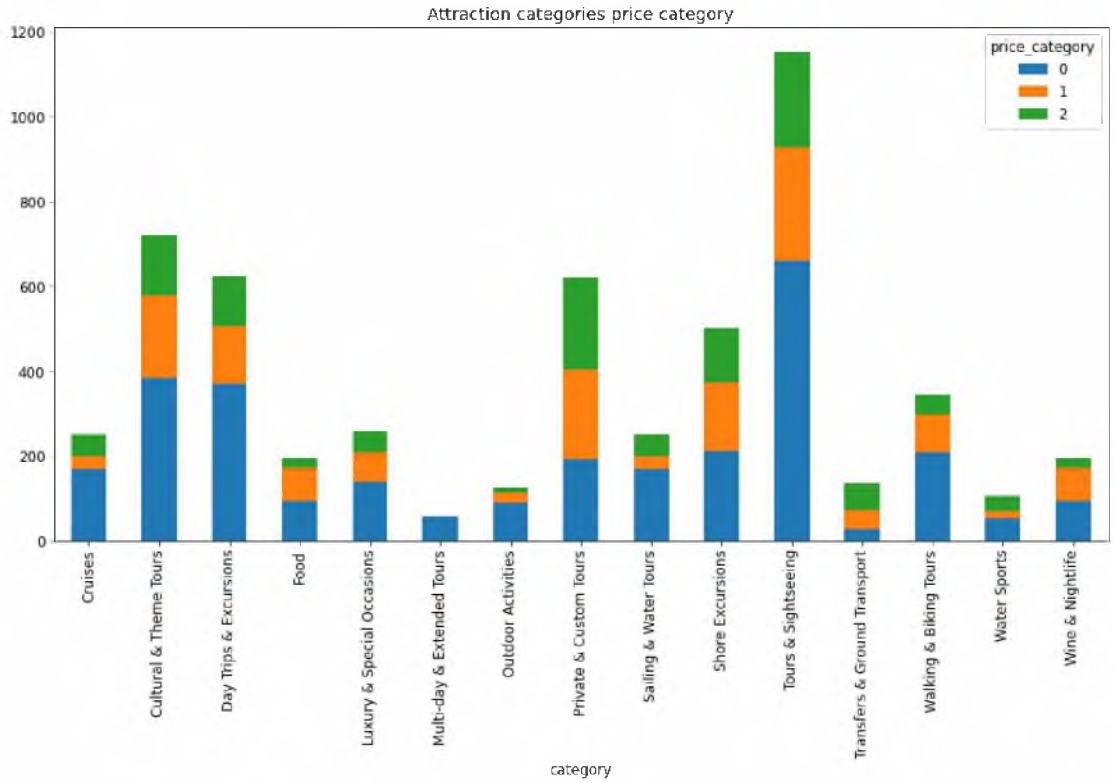


Εικόνα 27. Διακύμανση βαθμολογιών ανά τύπο γευμάτων

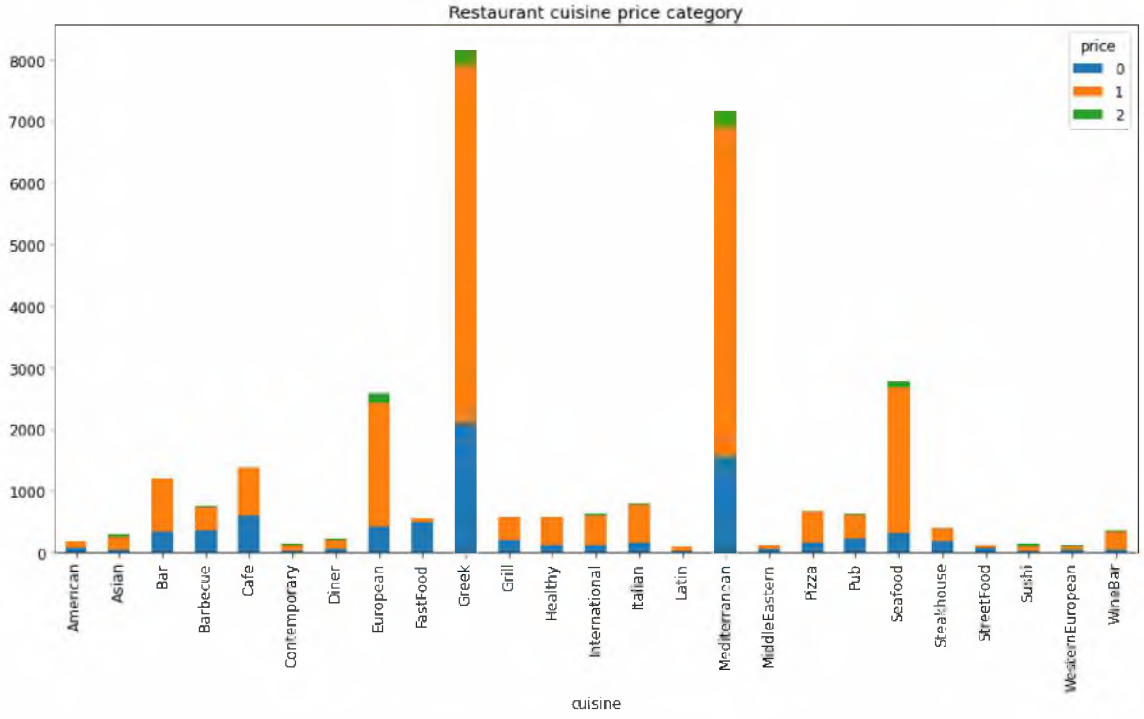
Η διακύμανση του εύρους τιμής για τα γνωρίσματα παροχές ξενοδοχείων, κατηγορία δραστηριότητας, κουζίνα και γευμάτων (εικόνα 28-31).



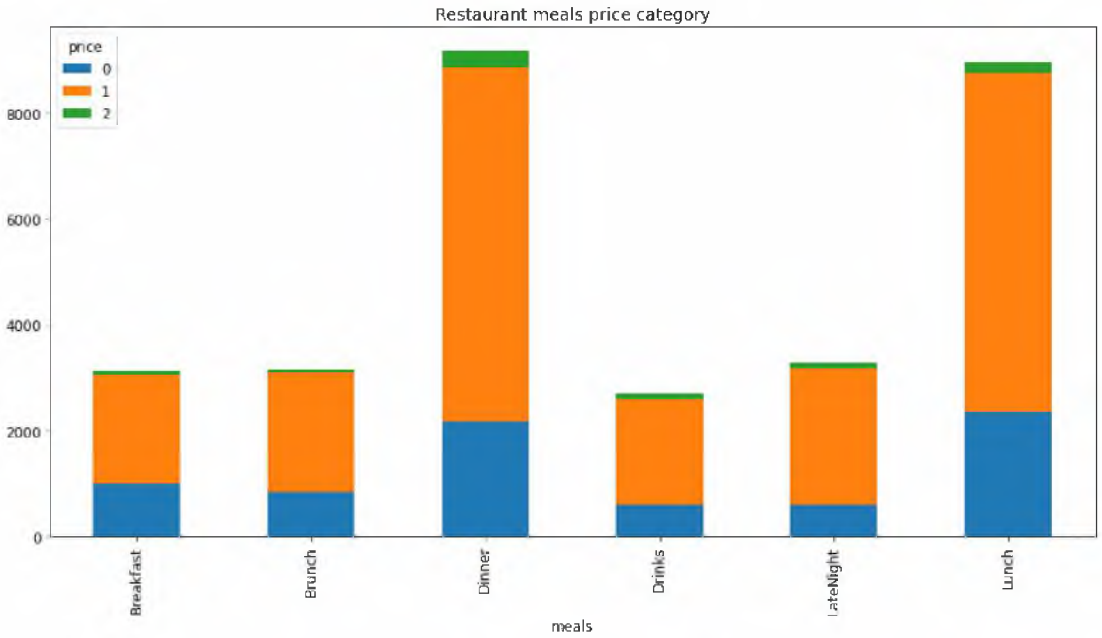
Εικόνα 28. Διακύμανση τιμής ανά παροχές



Εικόνα 29. Διακύμανση τιμής ανα κατηγορία δραστηριότητας

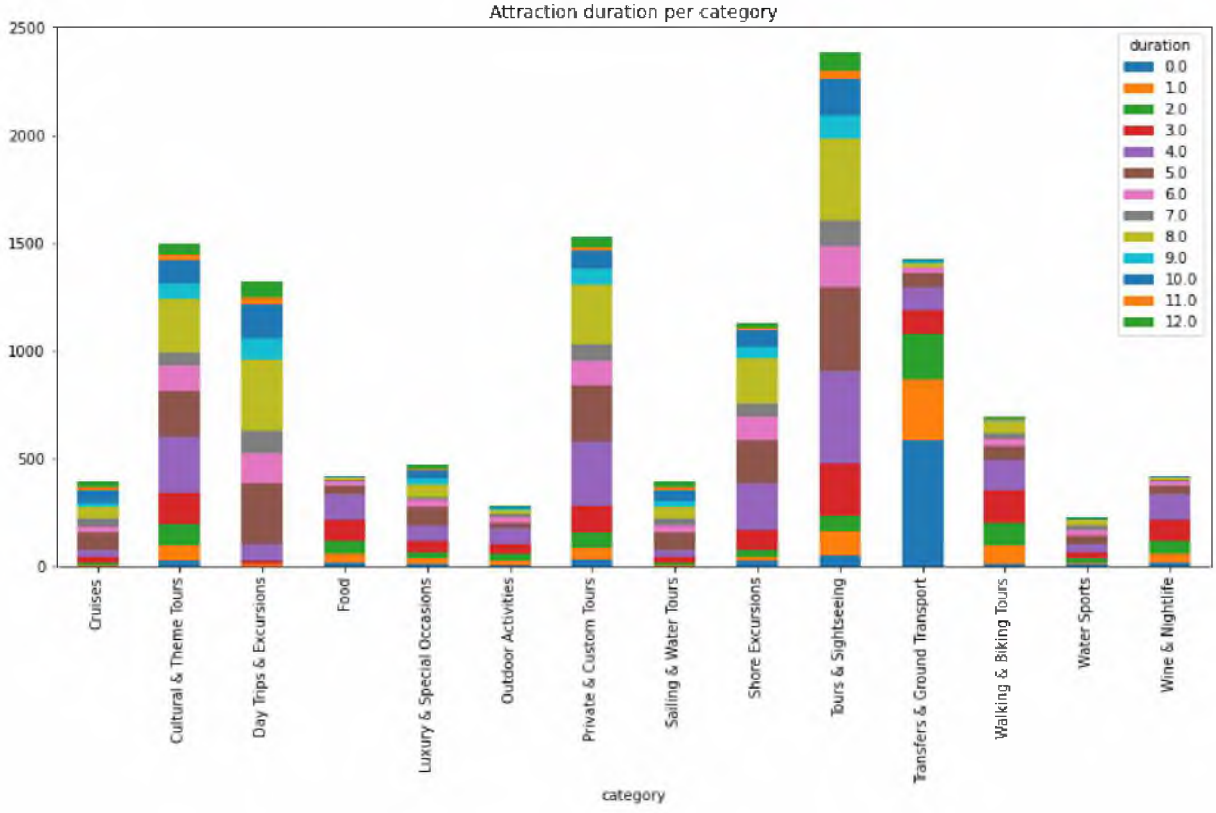


Εικόνα 30. Διακύμανση τιμής ανά τύπο κουζίνας



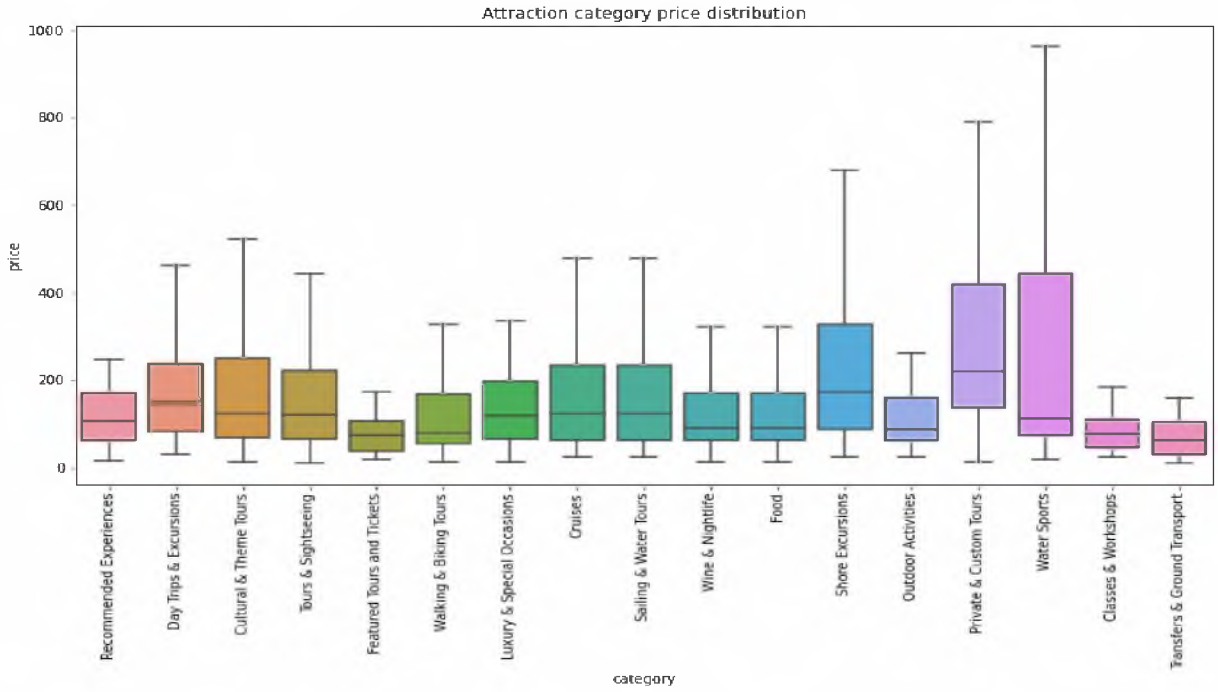
Εικόνα 31. Διακύμανση τιμής ανά είδος γεύματος

Διακύμανση της διάρκειας των δραστηριοτήτων σε ώρες (εικόνα 32).

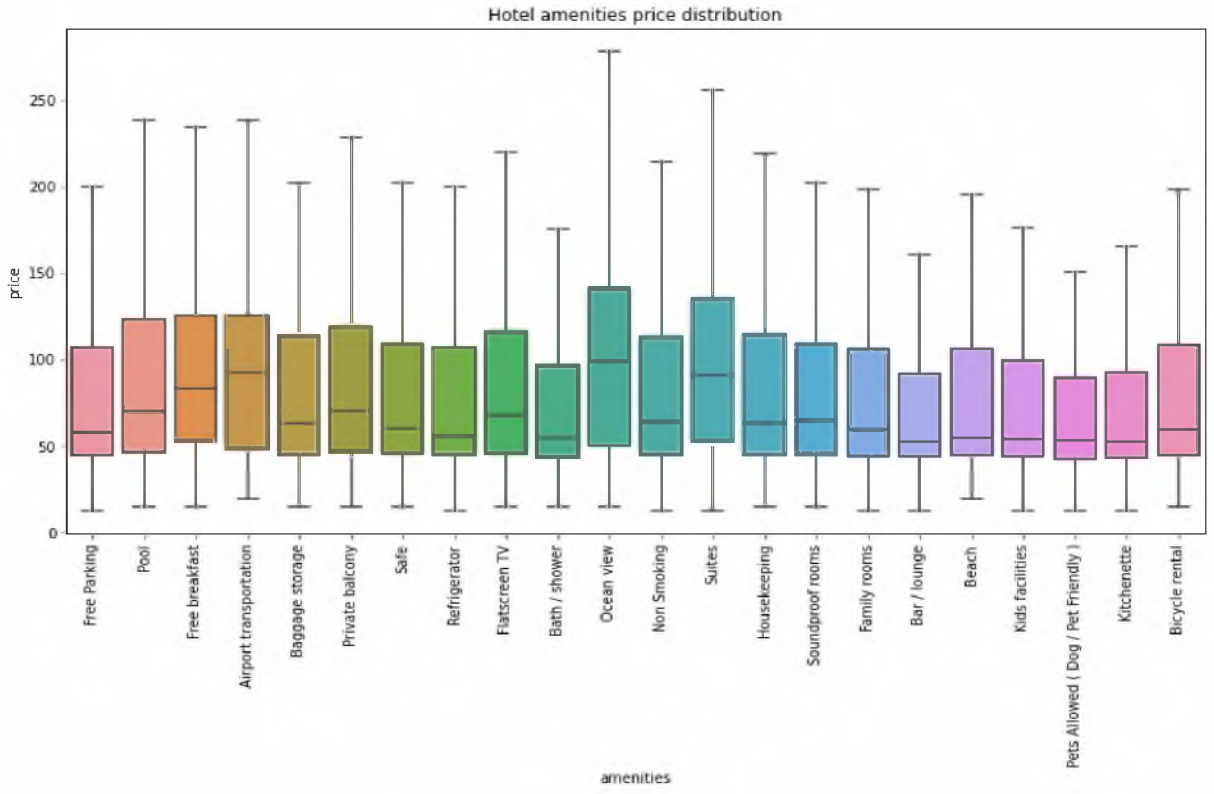


Εικόνα 32. Διακύμανση διάρκειας δραστηριοτήτων

Τα ακόλουθα *θηκογράμματα* (*boxplot*) παρουσιάζουν την κατανομή της τιμής (ελάχιστη, τεταρτημόρια, μέση, μέγιστη) για τις κατηγορίες των δραστηριοτήτων όπως επίσης και για τις παροχές των ξενοδοχείων (εικόνες 33- 34).

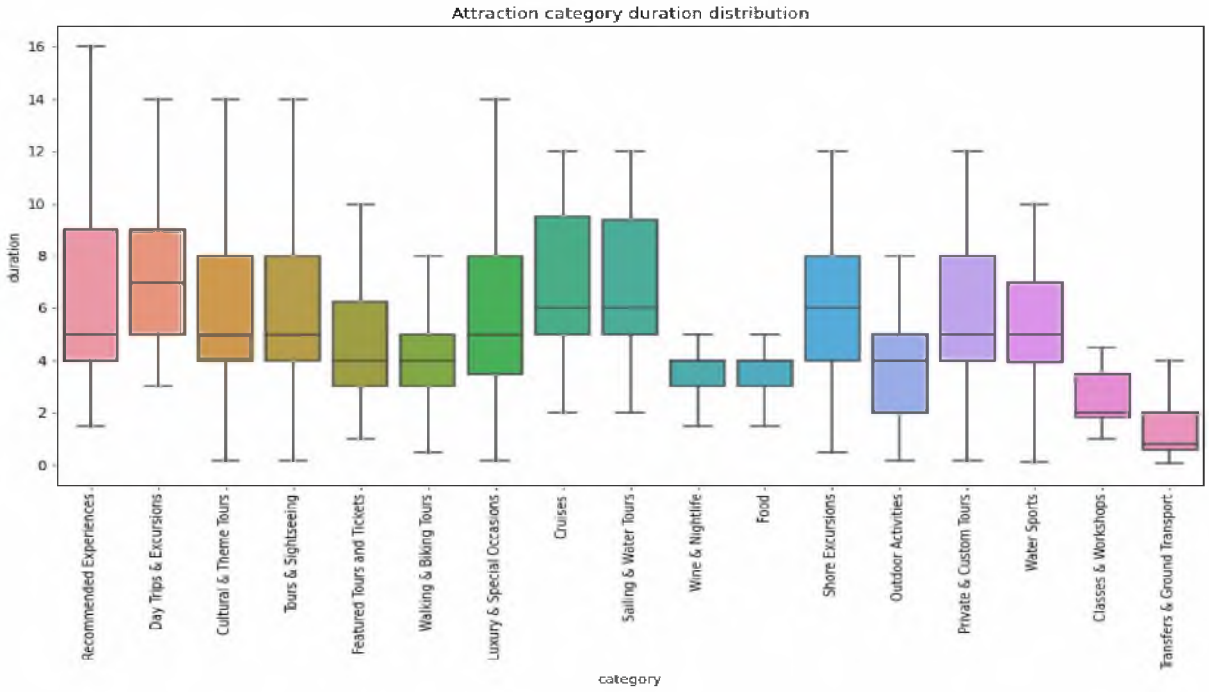


Εικόνα 33. Θηκόγραμμα τιμής ανά κατηγορία δραστηριότητας



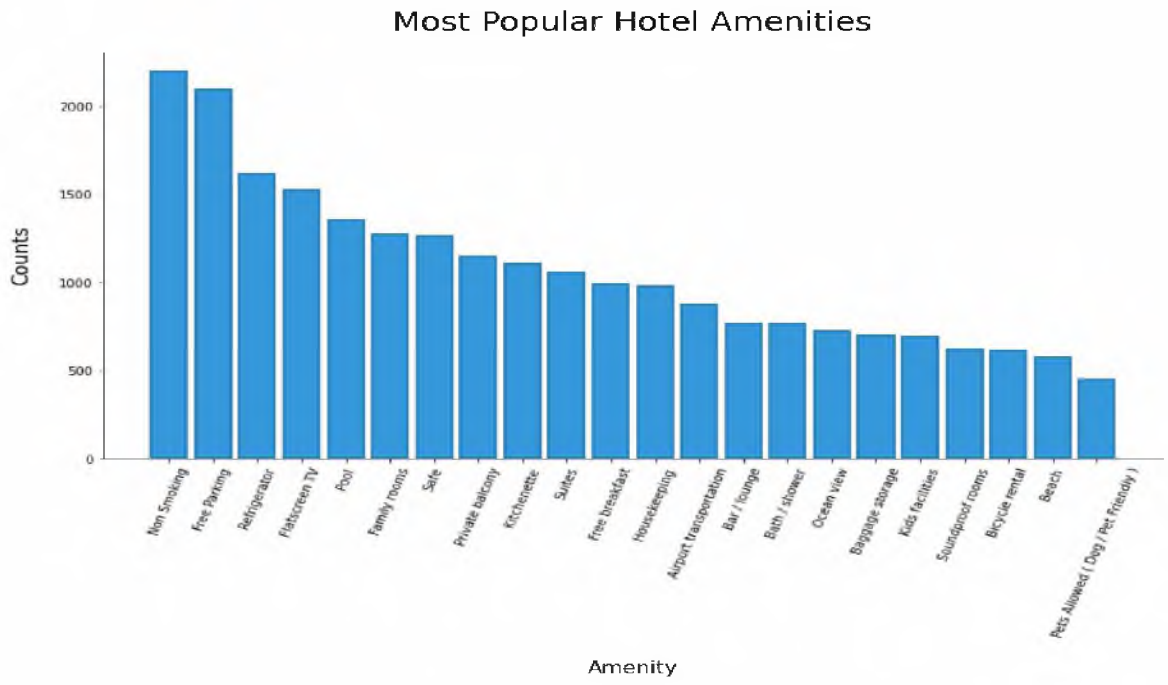
Εικόνα 34. Θηκόγραμμα τιμής ανά παροχές ξενοδοχείου

Στην εικόνα 35 φαίνεται το θηκόγραμμα για την κατανομή της διάρκειας των δραστηριοτήτων ανά κατηγορία .

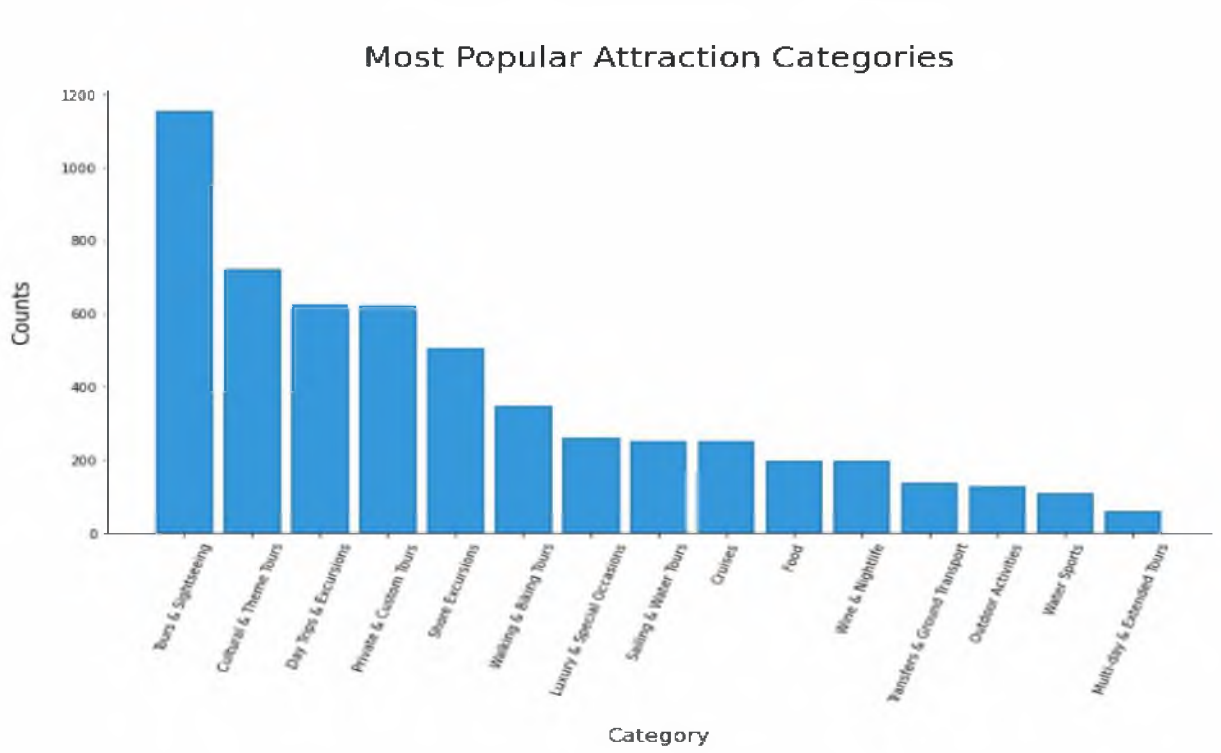


Εικόνα 35. Θηκόγραμμα διάρκειας ανά κατηγορία δραστηριοτήτων

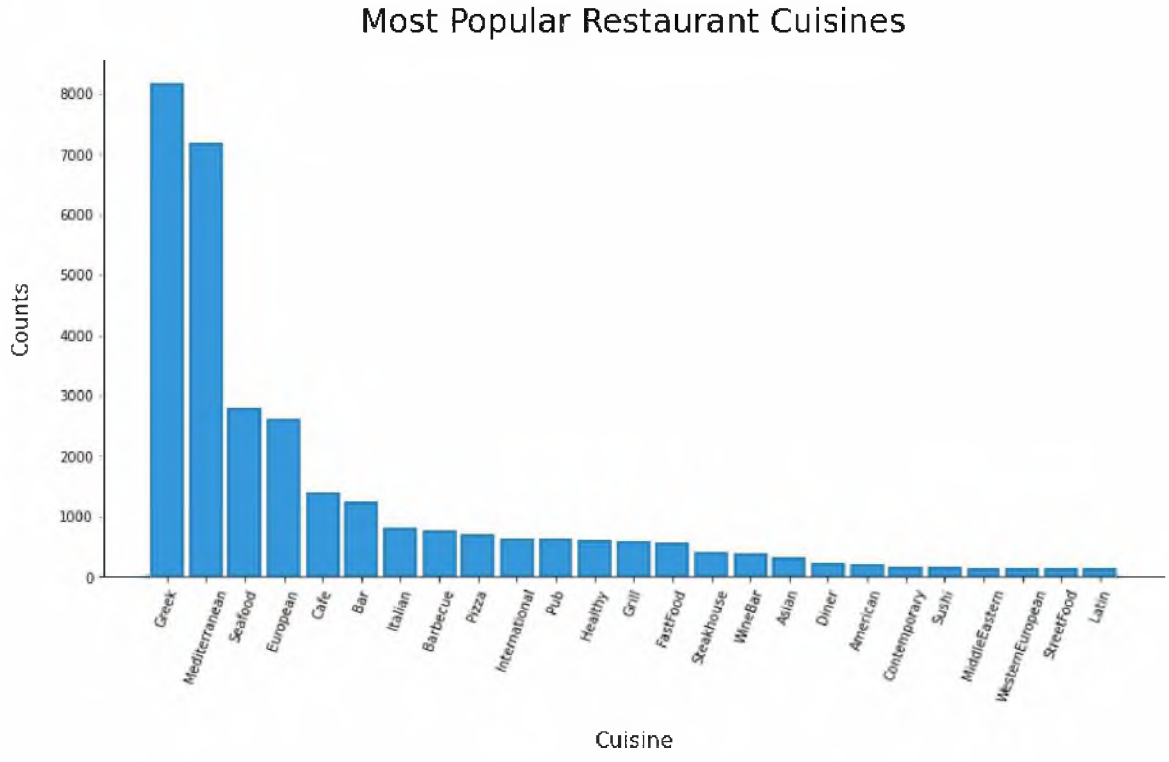
Στις εικόνες 36-39 εμφανίζονται οι πιο δημοφιλείς τιμές των παροχών, των κατηγοριών δραστηριοτήτων, των τύπων κουζινών και των γευμάτων.



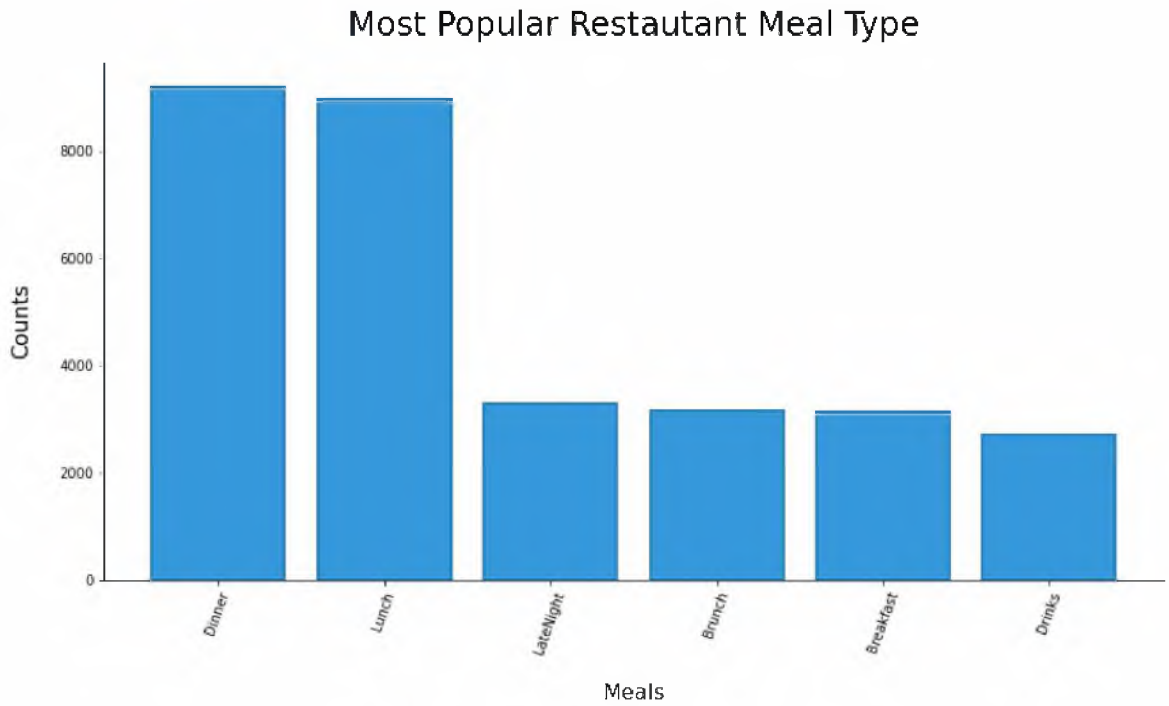
Εικόνα 36. Πιο δημοφιλής τιμές παροχών ξενοδοχείων



Εικόνα 37. Πιο δημοφιλείς τιμές του γνωρίσματος κατηγορία δραστηριότητας



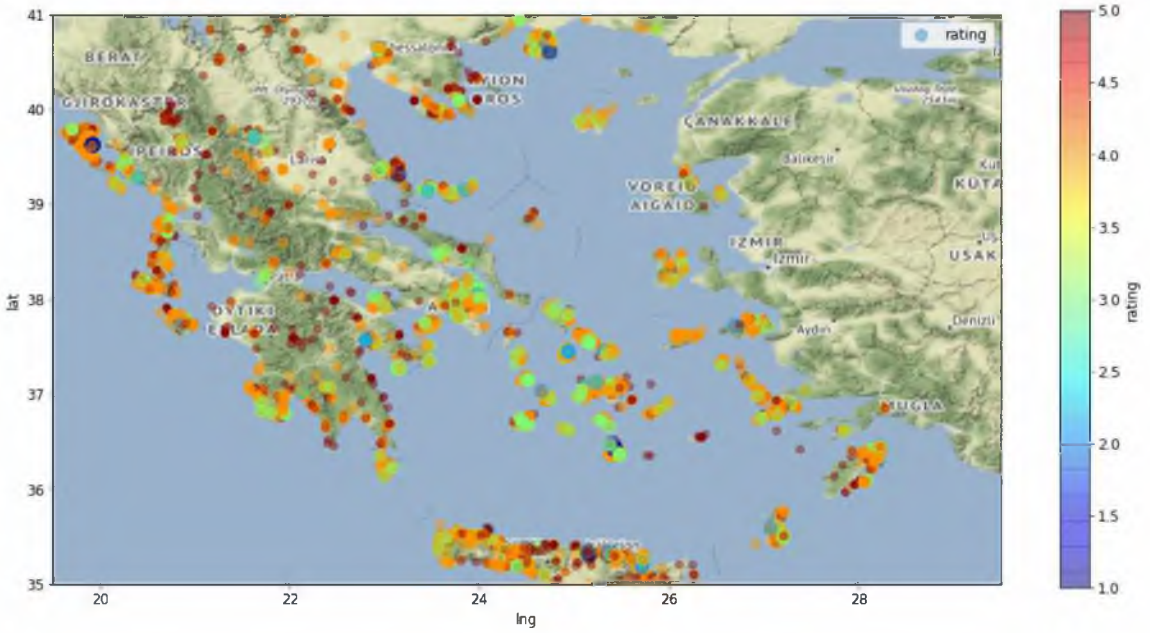
Εικόνα 38. Πιο δημοφιλής τιμές ειδών κουζίνας



Εικόνα 39. Πιο δημοφιλής τιμές κατηγοριών γευμάτων

Τέλος εμφανίζονται όλα τα αντικείμενα, ως σημεία συντεταγμένων στον χάρτη της Ελλάδας, με βάση το εύρος της τιμής και τις βαθμολογίες τους. Με αυτό τον τρόπο αντιλαμβανόμαστε με μία πρώτη ματιά την κατανομή και τις διακυμάνσεις των βαθμολογιών και της τιμής ανά περιοχές (εικόνα 40-41).

Εικόνα 40. Γεωγραφική κατανομή εύρους τιμής



Εικόνα 41. Γεωγραφική κατανομή βαθμολογιών

3.4.1 Συμπεράσματα ΔΑΔ

Άρχικά από τα προηγούμενα γραφήματα αντιλαμβανόμαστε ότι οι βαθμολογίες είναι θετικά προδιατεθειμένες. Στην συνέχεια της εργασίας, στην κατασκευή του μοντέλου, το λαμβάνουμε υπόψιν χρησιμοποιώντας τις κατάλληλες τεχνικές για να το αντιμετωπίσουμε, συγκρίνοντας την βαθμολογία του αντικειμένου με την τάση βαθμολογίας του κάθε χρήστη (μέσο όρο του χρήστη) ώστε να καταλάβουμε αν είναι στην πραγματικότητα θετική.

Όσον αφορά τα αντικείμενα ξεχωριστά, επικεντρώνουμε τα γραφήματα στις βαθμολογίες και το έυρος τιμής, δύο βασικά κριτήρια στην σχέση χρήστη-αντικειμένου. Για τα ξενοδοχεία οι διακυμάνσεις των βαθμολογιών τους, για το γνώρισμα παροχές, είναι χωρισμένες σε ίσες ποσότητες μεταξύ του 4 και του 5. Οι τιμές κάτω από 4 είναι λίγες σε πλήθος για να τις ερμηνεύσουμε στα διάγραμματα. Σχετικά με την σχέση τιμής–παροχών που έχουμε στα γράφηματα των εικόνων 28,34, παρατηρούμε ότι ορισμένες παροχές τείνουν να βρίσκονται σε πιο φθηνές επιλογές ξενοδοχείων όπως η κουζινα εντός του δωματίου, οι παιδικές εγκαταστάσεις και η δυνατότητα φιλοξενίας των κατοικιδίων. Αντιθέτως παροχές όπως μεταφορά από/στο αεροδρόμιο, σουίτες, θέα στην θάλασσα και δωρεάν πρωινό τείνουν να εμφανίζονται σε πιο ακριβές επιλογές.

Για τις δραστηριότητες εμφανίζονται στοιχεία για τα γνώρισματα κατηγορίες, διάρκεια. Αρχικά παρατηρούμε ότι οι άριστες βαθμολογίες, ειδικά σε ορισμένες επιλογές όπως το φαγητό, το ποτό και η νυχτερινή διασκέδαση, οι εξωτερικές δραστηριότητες και τα θαλλάσια σπόρ, είναι η συντριπτική πλειοψηφία. Αυτό είναι λογικό αφού οι πρώτες δύο αποτελούν δραστηριότητες πού είναι εύκολα ευχάριστες ενώ οι δύο τελευταίες είναι έντονες σαν εμπειρίες, κάτι το οποίο εντυπωσιάζει τον χρήστη. Από την τιμή τους παρατηρούμε επίσης ότι δραστηριότητες όπως εξωτερικές δραστηριότητες, βόλτες ποδηλάτου και ιστιοπλοίας έιναι κατά κύριο λόγο πιο οικονομικές από άλλες όπως ημερήσιες κρουαζιέρες, ιδωτικές ξεναγήσεις και μεταφορές με ιδιωτικά οχήματα. Η κατηγορία τιμής όπως αναφέρθηκε προηγουμένως στην εργασία αφορά την τιμή ανά ώρα δραστηριότητας. Επίσης για τις δραστηριότητες η διάρκεια ποικίλει αρκετά ανά κατηγορία. Κατηγορίες όπως μεταφορές και βραδινής διασκέδασης είναι κατά κύριο λόγο ολιγόωρες ενώ αντίθετα κατηγορίες όπως ημερήσιες εκδρομές και ιστιοπλοία είναι στην πλειοψηφία τους διάρκειας μεγαλύτερης των 6 ωρών.

Τέλος για τα εστιατόρια παρατηρούμε ότι οι άριστες βαθμολογίες είναι σαφώς λιγότερες από ότι στα προηγούμενα αντικείμενα με ίδια συχνότητα με τις μέτριες, με διαβαθμίσεις ανά τύπο κουζίνας όσο και τον τύπο γεύματος. Στην τιμή παρατηρούμε ότι κατηγορίες όπως το γρήγορο φαγητό (fastfood) είναι οικονομικές επιλογές σε σχέση με κουζίνες βασισμένες στο ψάρι και την ιαπωνική κουζίνα (sushi). Το γνώρισμα του τύπου γεύματος δεν είναι ιδιαίτερα επεξηγηματικό από μόνο του, ούτε σε συνδυασμό της τιμής-βαθμολογίας για αυτό το έχουμε σαν προαιρετική επιλογή στην δημιουργία του προφίλ χρήστη.

Σχετικά με την γεωγραφική κατανομή παρατηρούμε ότι υπάρχει ομοιογένεια στο γνώρισμα της τιμής όπως και της βαθμολογίας, με μικρές εξαιρέσεις, όπως καθαρά τουριστικά μέρη σαν την αρχαία Ολυμπία που συγκεντρώνει πολύ καλές κριτικές με υψηλές τιμές και μέρη με πιο χαμηλες τιμές, λόγω γεωγραφικής θέσης όπως η Λεσβος.

3.5 Σύνοψη κεφαλαίου

Σε αυτό το κεφάλαιο έγινε περιγραφή των δεδομένων και της διαδικασίας επεξεργασίας τους. Έγινε αναφορά στην συλλογή τους, την προεπεξεργασία τους όπως ο καθαρισμός , η

συμπλήρωση και ο μετασχηματισμός τους. Επίσης παρατέθηκαν μερικά από τα γραφήματα της διερευνητικής τους ανάλυσης.

ΚΕΦΑΛΑΙΟ 4

ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΚΑΙ ΛΕΙΤΟΥΡΓΙΑ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ ΣΥΣΤΑΣΕΩΝ

4.1 Ιδέα Αρχικού μοντέλου

Τα συστήματα συστάσεων που λειτουργούν στην λογική του CF, αντιμετωπίζουν το πρόβλημα της *ψυχρής εκκίνησης*. Αυτό συμβαίνει όταν εισέρχεται ένας καινούργιος χρήστης ή ένα καινούργιο αντικείμενο στο σύστημα. Στην περίπτωση αυτή, λόγω έλλειψης πληροφορίας (βαθμολογιών) δεν μπορεί να γίνει συσχέτιση μεταξύ του καινούργιου χρήστη και των υπαρχόντων στο σύστημα.

Μια συνηθισμένη τεχνική που χρησιμοποιείται από τα ΣΣ, είναι η χρήση μη-εξατομικευμένων αλγορίθμων, όπως τα πιο δημοφιλή αντικείμενα. Αν ο χρήστης βαθμολογήσει κάποια από τα αντικείμενα που του προτείνονται, το σύστημα αποκτά πληροφορία για να ξεκινήσει.

Προκειμένου να πραγματοποιηθούν εξατομικευμένες προτάσεις για έναν καινούργιο χρήστη, χρειάζεται να δημιουργηθεί ένα προφίλ για αυτόν. Αυτό επιτυγχάνεται μέσω της μεθόδου CBF [32][33][34][35].

Από την ανάλυση προέκυψε ότι βασικά χαρακτηριστικά για τον χρήστη είναι η τιμή και η τοποθεσία. Εκτός αυτών όσον αφορά τα ξενοδοχεία, οι παροχές είναι το βασικό χαρακτηριστικό, για τις δραστηριότητες οι κατηγορίες τους και η διάρκεια της καθεμίας και τέλος για τα εστιατόρια ο τύπος κουζίνας και οι ώρες (τύπος γεύματος) που εξυπηρετούν.

Με βάση αυτά, αρχικά ζητείται από τον χρήστη να επιλέξει για κάθε αντικείμενο τις προτιμήσεις του. Οι προτιμήσεις χρησιμοποιούνται για να βρεθούν τα πιο όμοια σε αυτόν αντικείμενα, τα οποία είναι οι προτάσεις που του γίνονται. Η συγκεκριμένη μέθοδος αν και απλή είναι από τις πιο δημοφιλής λόγω των καλών αποτελεσμάτων της και του υψηλού δείκτη *σχετικότητας*. Παρόλα αυτά υστερεί σε δείκτες, όπως η *κάλυψη*, *καινοτομία* και την *ποικιλία*, εφόσον προτείνει αντικείμενα πολύ όμοια σε αυτά που ζητάει ο χρήστης. Για παράδειγμα αν ένας χρήστης δηλώσει προτίμηση για μία κατηγορία δραστηριοτήτων, κάτι χρήσιμο για την δημιουργία προφίλ ώστε να χρησιμοποιηθεί σε επόμενα βήματα, το

σύστημα, θα του προτείνει μόνο αντικείμενα της συγκεκριμένης κατηγορίας. Μπορεί ένας χρήστης να έχει σαν χαρακτηριστικό της προσωπικότητάς του, την αγάπη για την φύση αλλά ίσως τον ενδιαφέρει να επισκεφτεί και ένα μουσείο πέρα από δραστηριότητες πεζοπορίας κτλ.

4.2 Μοντέλο με βάση το περιεχόμενο

Όπως αναφέρθηκε προηγουμένως προκειμένου να αντιμετωπιστεί ο καινούργιος χρήστης και να του παρουσιαστούν εξατομικευμένες προτάσεις πρέπει να του ζητηθεί να παρέχει ρητά τις προτιμήσεις του, για κάποια σημαντικά χαρακτηριστικά που αναφέρθηκαν προηγουμένως. Χάρις σε αυτό δίνεται η δυνατότητα να βρούμε αντικείμενα που θα τον ενδιαφέρουν δημιουργώντας το προφίλ που χρειάζεται στην συνέχεια το μοντέλο [36].

Οι χρήστες δεν είναι οι ίδιοι και στα τρία σετ που χρησιμοποιούνται οπότε η προηγούμενη μέθοδος εφαρμόζεται ξεχωριστά στα ξενοδοχεία, τα εστιατόρια και τις δραστηριότητες, δημιουργώντας ξεχωριστό προφίλ για το κάθε ένα. Αφού σκοπός της εργασίας είναι να προτείνει στον χρήστη ένα συνολικό πλάνο με προτάσεις για όλα τα αντικείμενα, ζητείται να δηλώσει με την σειρά, προτιμήσεις για το καθένα. Αρχικά για τα ξενοδοχεία επιθυμητές παροχές και εύρος τιμής, για τις δραστηριότητες κατηγορία, διάρκεια και εύρος τιμής και τέλος για τα εστιατόρια κουζίνα της προτίμησης του, είδη γευμάτων που κυρίως των ενδιαφέρουν και εύρος τιμής. Όσον αφορά το εύρος τιμής ζητείται χωριστά για κάθε κατηγορία αντικειμένου για περιπτώσεις χρηστών που είναι διατεθειμένοι να ξοδέψουν ένα μεγαλύτερο ποσό για κάποιο αντικείμενο. Για παράδειγμα ένας χρήστης μπορεί να θεωρεί πιο σημαντικό, ένα ξενοδοχείο που έχει πολύ καλή τοποθεσία και θέα, άρα συνήθως πιο ακριβό, αλλά δεν ενδιαφέρεται για ένα πολυτελές γεύμα. Επίσης δίνεται η δυνατότητα στον χρήστη να επιλέξει αν έχει προτίμηση σε ένα-πολλούς τύπους γεύματος. Μπορεί για παράδειγμα να έχει σκοπό να είναι συνεχώς απασχολημένος με κάποια δραστηριότητα αφήνοντας το βασικό του γεύμα για βράδυ ή στο ξενοδοχείο του να περιλαμβάνεται πρωινό. Αυτή η επιλογή προτίμησης είναι προαιρετική, δίνοντας ακόμα περισσότερη πληροφορία για το προφίλ του στην περίπτωση που δηλώνεται.

4.2.1 Κατασκευή προφίλ χρήστη

Αφού δηλώσει ο χρήστης της προτιμήσεις του, κατασκευάζεται ένα εικονικό αντικείμενο με τα γνωρίσματα αυτά. Όπως αναφέρθηκε προηγουμένως αυτό γίνεται τρεις φορές, μία για κάθε κατηγορία αντικειμένων. Αναφέρεται ως αντικείμενο μηδέν αφού έχει δεσμευθεί το συγκεκριμένο αναγνωριστικό.

Στην εικόνα 42 έχουμε ως παράδειγμα έναν χρήστη με προτιμήσεις υποδομών το ελεύθερο πάρκινγκ, εγκαταστάσεις φιλικές προς παιδιά και την ύπαρξη πισίνας όπως επίσης και ότι ενδιαφέρεται για χαμηλού κόστους ξενοδοχεία (0='budget'). Οι συντεταγμένες έμειναν κενές διότι στο βασικό μοντέλο της εργασίας δεν χρησιμοποιούνται σε αυτό το στάδιο.

```
In [ ]: hotel_features = {
            'hotel_id': 0,
            'hotel_idx': 0,
            'amenities': ['Free Parking','Kids Facilities' 'Pool'],
            'price_category': 0
            'lat':
            'lng':
        }
```

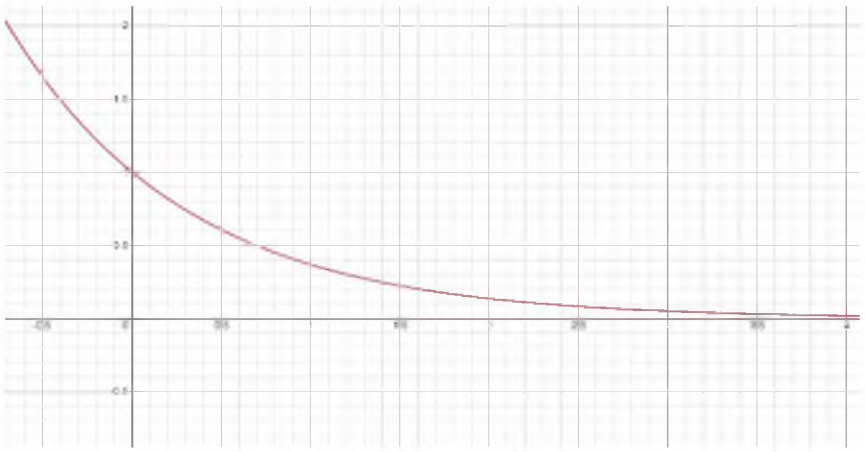
Εικόνα 42. Προτιμήσεις χρήστη

Για το κάθε γνώρισμα χρησιμοποιείται διαφορετική μέθοδος υπολογισμού της ομοιότητας, με βάση τον τύπο του όπως και της σημαντικότητας της απόκλισης της τιμής του από την επιθυμητή. Είναι διαφορετική η απόκλιση της μίας μονάδας στην διάρκεια μίας δραστηριότητας, π.χ. αν έχει δηλωθεί ως επιθυμητή η διάρκεια των δύο ωρών και εμείς συγκρίνουμε ένα αντικείμενο των τριών, σε σχέση με την μοναδιαία απόκλιση στην κατηγορία τιμής για την οποία αναφέρθηκε στο προηγούμενο κεφάλαιο ότι αποτελείται από τις τιμές εύρους 0, 1, 2. Στην περίπτωση αυτή η διαφορά είναι τόσο σημαντική, ώστε πρέπει να απορρίπτει το αντικείμενο. Η μέθοδος υπολογισμού για το κάθε ένα δεν είναι ξεκάθαρη εξαρχής και πρέπει να υπολογιστεί εμπειρικά με δοκιμές.

Η ομοιότητα μεταξύ αντικειμένων έχει εύρος τιμής [0,1]. Συνήθως ένα αντικείμενο στην πραγματική ζωή, ομοιότητα ένα έχει μόνο με τον εαυτό του. Για τον υπολογισμό της ομοιότητας τιμής χρησιμοποιείται, μία φθίνουσα εκθετική συνάρτηση, η οποία ελαττώνεται ομαλά στο διάστημα [0,1] (εικόνα 43) (44).

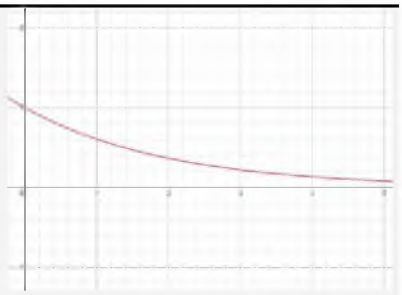
$$s = e^{-x} \tag{44}$$

όπου το x αλλάζει με βάση το πόσο σημαντική είναι η απόκλιση για το γνώρισμα.



Εικόνα 43. Εκθετική συνάρτηση

Στην συνέχεια παρουσιάζεται ένας πίνακας, με τους τύπους και τις ενδεικτικές μετρήσεις ανά γνώρισμα.

Εύρος τιμής	Ενδεικτικές τιμές	
$s = e^{-(diff*2)}$, $diff = \lvert price1 - price2 \rvert$	$diff = 0$, $s = 0$ $diff = 1$, $s = 0.135$ $diff = 2$, $s = 0.018$	
Διάρκεια δραστηριότητας	Ενδεικτικές τιμές	
$s = e^{-(\frac{diff}{2})}$, $diff = \lvert duration1 - duration2 \rvert$	$diff = 0$, $s = 0$ $diff = 1$, $s = 0.61$ $diff = 2$, $s = 0.37$ $diff = 6$, $s = 0.05$	

Τα γνωρίσματα τύπου λίστας αλφαριθμητικών όπως οι παροχές, οι κατηγορίες στις δραστηριότητες, ο τύπος κουζίνας και τα γεύματα πρέπει να χρησιμοποιηθούν διαφορετικά. Αρχικά τα συγκεκριμένα γνωρίσματα επειδή είναι κατηγορικά δεν μπορούν να χρησιμοποιηθούν από τον αλγόριθμο. Αναφέρθηκε στην προεπεξεργασία ότι

χρησιμοποιείται η μέθοδος *one hot encoding* ώστε να μετατραπούν τα γνωρίσματα σε αριθμητικές τιμές. Η λογική είναι όμοια με τον ΠΠΑ που αναφέρθηκε στην παράγραφο 2.4. Το κάθε αντικείμενο γίνεται μια εγγραφή στον πίνακα (σειρά) και οι διαφορετικές τιμές του στήλες. Για παράδειγμα στην περίπτωση των ξενοδοχείων θα δημιουργηθεί ένας πίνακας περιεχομένων αντικειμένου $(N + 1) \times 22$ με N το πλήθος των αντικειμένων και τις 22 διαφορετικές τιμές για τις παροχές, συν μία εγγραφή για το εικονικό αντικείμενο μηδέν με τις προτιμήσεις του χρήστη. Κάθε πεδίο περιγράφει αν το αντικείμενο περιέχει την συγκεκριμένη τιμή.

Στο σημείο αυτό η ομοιότητα μεταξύ τους μπορεί να υπολογιστεί με διάφορες μεθόδους, όπως η *ομοιότητα συνημιτόνου* και η *Ευκλείδειος απόσταση* (*Euclidean distance*).

Επιλέγεται η *ομοιότητα συνημιτόνου* καθώς έχουμε κανονικοποιημένο αποτέλεσμα μεταξύ [0,1], ενώ στην περίπτωση της *ευκλείδειας απόστασης* αριθμό που δεν είναι χρήσιμος στην τελικό υπολογισμό. Για παράδειγμα αν υπάρχει ένα ξενοδοχείο $hotel_1$ με παροχές ελεύθερο πάρκινγκ, κουζίνα στο δωμάτιο, φιλικό προς τα παιδιά και πισίνα και ως i_0 οι προτιμήσεις μας, ελεύθερο πάρκινγκ, φιλικό προς τα παιδιά και πισίνα, τα δύο αντικείμενα με χρήση συνημιτόνου θα έχουν ομοιότητα ίση με $s = \frac{3}{\sqrt{3 \cdot 4}} = 0.87$.

Η τελική ομοιότητα υπολογίζεται ως το άθροισμα όλων των επιμέρους ομοιοτήτων προς το πλήθος τους.

Ξενοδοχεία :	$s_{hotel} = (s_{price} + s_{amenities})/2$
Δραστηριότητες :	$s_{attractions} = (s_{price} + s_{category} + s_{duration})/3$
Εστιατόρια :	$s_{restaurants} = (s_{price} + s_{cuisine} + s_{meals})/3$

Τα αντικείμενα με ομοιότητα πάνω του 0.8 αποθηκεύονται για το επόμενο βήμα, με βαθμολογία ίση με τον μέσο όρο τους.

4.2.2 Αξιολόγηση του μοντέλου με βάση το περιεχόμενο

Το προηγούμενο βήμα της κατασκευής του προφίλ του χρήστη είναι στην πράξη ένα *top-N* μοντέλο. Τα *top-N* μοντέλα παρουσιάζουν στον χρήστη τις N καλύτερες προτάσεις για

αυτόν, ταξινομημένες με βάση την εκτίμηση. Για την κατασκευή του χρησιμοποιήθηκαν οι προτιμήσεις του χρήστη, όπως αναφέρθηκε προηγουμένως.

Μια μικρή διαφορά στην εξεταζόμενη περίπτωση, είναι το N το οποίο δεν είναι σταθερό, αλλά μεταβάλλεται με βάση πόσα αντικείμενα είναι πάνω από το όριο ομοιότητας που έχει τεθεί, 80%.

Για να αξιολογηθεί ένα *top-N* μοντέλο ένας αποδοτικός τρόπος είναι ο μετρητής *hit rat*e [37]. Η ιδέα είναι ότι για κάθε χρήστη βρίσκουμε όλα τα αντικείμενα που έχει βαθμολογήσει και αφαιρούμε τυχαία ένα από αυτά (*Leave One Out Cross Validation*) [38]. Τα υπόλοιπα αντικείμενα χρησιμοποιούνται στην εκπαίδευση του μοντέλου, ώστε να προταθούν τα *top-N* αντικείμενα. Αν το αφαιρούμενο αντικείμενο ανήκει στις προτάσεις, έχουμε *hit*. Το συνολικό *hit rate* του συστήματος είναι το σύνολο των *hits* προς το πλήθος των χρηστών του ελέγχου.

Η προηγούμενη λογική εφαρμόζεται σε συστήματα CF όπου δίνεται στον αλγόριθμο, ολόκληρος ο ΠΒΧ, ώστε να δώσει τα αποτελέσματα των προτάσεων.

Στην εξεταζόμενη περίπτωση χρησιμοποιείται η ιδέα ώστε να επαληθευθεί η απόδοση του κομματιού της κατασκευής του προφίλ, μέσω CBF. Αρχικά για 50 τυχαίους χρήστες με περισσότερες από 5 κριτικές, αφαιρείται ένα από τα βαθμολογημένα αντικείμενα τους. Στην συνέχεια σημειώνονται τα κύρια χαρακτηριστικά που εμφανίζονται συχνά στο προφίλ του χρήστη, με βάση την κατηγορία αντικειμένου, παροχές για το σετ των ξενοδοχείων, κατηγορίες για τις δραστηριότητες κτλ. όπως επίσης κατηγορία τιμής και η διάρκεια. Ως συχνό, θεωρείται ένα χαρακτηριστικό που βρίσκεται σε περισσότερα από τα μισά αντικείμενα.

Για παράδειγμα αν στα βαθμολογημένα ξενοδοχεία ενός χρήστη, υπάρχει η παροχή πισίνα σε 4 από τις 5 κριτικές ξενοδοχείων, η παροχή της κουζίνας στο δωμάτιο και του ιδιωτικού μπαλκονιού σε 3 από αυτά, έχουμε ως προτιμήσεις αυτές τις τιμές του συγκεκριμένου γνωρίσματος. Με βάση αυτά γίνονται προτάσεις στον χρήστη με N=15. Η σταθερή τιμή του N χρησιμοποιείται μόνο στην αξιολόγηση. Όπως αναφέρθηκε προηγουμένως στην λειτουργία της εφαρμογής, το N αλλάζει δυναμικά, με βάση την ομοιότητα των αντικειμένων. Το *hit rate* παρόλο που είναι μία συχνά χρησιμοποιούμενη τεχνική, δεν δίνει μεγάλα ποσοστά σαν αποτέλεσμα, αφού, είναι αρκετά δύσκολο να προτείνουμε το μοναδικό αντικείμενο που λείπει, σε σχέση με όλο το σετ. Πρέπει να υπάρχει μεγάλος

όγκος πληροφορίας. Παρόλα αυτά με γνώση του μεγέθους των σετ αυτής της εργασίας τα αποτελέσματα είναι ενθαρρυντικά (πίνακας 12).

Πίνακας 12. Τιμές Hit Rate ανά Αντικείμενο

	Hit Rate
Ξενοδοχεία	0.16
Δραστηριότητες	0.22
Εστιατόρια	0.1

Όπως παρατηρείται, οι δραστηριότητες έχουν την μεγαλύτερη τιμή λόγω ότι, είναι λιγότερα τα συνολικά αντικείμενα σε σχέση με τα άλλα δύο σετ. Σε αυτή την περίπτωση είναι πιο πιθανό να προταθεί ένα αντικείμενο που αφαιρέθηκε, εφόσον πληροί τις προτιμήσεις του χρήστη. Στην συνέχεια καλύτερα αποτελέσματα έχουν τα ξενοδοχεία και τέλος τα εστιατόρια, που είναι το μεγαλύτερο σετ.

4.3 Συνεργατικό φιλτράρισμα – SVD

Έχοντας δημιουργηθεί το προφίλ του χρήστη, εκτελώντας το πρώτο βήμα του συστήματος -το CBF- είναι σε θέση να χρησιμοποιηθεί η μέθοδος του CF ώστε να προκύψει πιο αποδοτικό μοντέλο. Το CF, επιτρέπει να γίνονται προτάσεις με μεγαλύτερο εύρος όσον αφορά τα γνωρίσματα του αντικειμένου σε σχέση με το CBF, εφόσον λαμβάνει υπόψιν επιλογές διαφορετικών χρηστών. Σε αυτή την μέθοδο συσχετίζονται οι χρήστες με τα αντικείμενα. Υπάρχουν δύο βασικές προσεγγίσεις αυτή των γειτόνων (*neighborhood approach*) και τα μοντέλα *λανθανόντων συντελεστών*.

Χρησιμοποιώντας μοντέλα *λανθανόντων συντελεστών* όπως το SVD, μπορούν να συγκριθούν χρήστες και αντικείμενα, χωρίς τον περιορισμό της σύγκρισης διαφορετικών αντικειμένων. Συνδέεται κάθε χρήστης και κάθε αντικείμενο με f διαστάσεων διανύσματα, τα οποία θεωρούνται ως διανύσματα διαφορετικών συντελεστών (*factors*) που έχουν εξαχθεί από τις βαθμολογίες των χρηστών. Χρησιμοποιείται η μέθοδος SVD για τον ΠΒΧ.

Ένα τυπικό μοντέλο συνδέει κάθε χρήστη u με ένα διάνυσμα συντελεστών $p_u \in R^f$, και κάθε αντικείμενο με ένα διάνυσμα συντελεστών $q_i \in R^f$. Ο κλασικός αλγόριθμος SVD

δεν μπορεί να εφαρμοστεί σε πίνακα βαθμολογιών με κενές τιμές. Κάνοντας χρήση μόνο των βαθμολογιών που είναι διαθέσιμες ή γεμίζοντας τον πίνακα με τυχαίες βαθμολογίες θα οδηγηθούμε σε *υπερπροσαρμογή* και ανακριβή αποτελέσματα. Για να αποφευχθεί χρησιμοποιούνται οι διαθέσιμες βαθμολογίες με εισαγωγή *παραγόντων κανονικοποίησης* για την αποφυγή της *υπερπροσαρμογής*.

4.3.1 Περιγραφή αλγορίθμου

Κάποιοι χρήστες, τείνουν να βαθμολογούν με μεγαλύτερη επιείκεια από τον μέσο όρο και κάποια αντικείμενα να λαμβάνουν μεγαλύτερες βαθμολογίες από άλλα. Προκειμένου να ληφθεί υπόψιν αυτό το γεγονός χρησιμοποιήθηκε η *βασική εκτίμηση* (*baseline estimates*) για να προσαρμοστούν οι βαθμολογίες των σετ [39]. Οι μεταβλητές b_u και b_i είναι η *τάση του χρήστη* (*user bias*) και του αντικειμένου (*item bias*) αντίστοιχα, δηλαδή η διαφορά κάθε χρήστη και αντικειμένου από την μέση τιμή.

Ως παράδειγμα υποθέτουμε ότι η μέση τιμή των εστιατορίων είναι 4.0 και το εστιατόριο i γνωστό για τα ψητά κρέατα που προσφέρει, βαθμολογείται κατά 0.5 περισσότερο από τον μέσο όρο λόγω της καλής του ποιότητας (b_i). Ο χρήστης u έχοντας προτίμηση στην χορτοφαγική διατροφή τείνει βαθμολογεί αυτού του είδους τα εστιατόρια κατά 1 βαθμό λιγότερο, λόγω έλλειψης επιλογών. Η εκτίμηση βάσης για την περίπτωση αυτή είναι $b_{ui} = 4.0 + 0.5 - 1.0 = 3.5$

r_{ui}	*Βαθμολογία του χρήστη u για το αντικείμενο i*
$\tilde{r}_{ui}$	*Εκτιμώμενη βαθμολογία του χρήστη u για το αντικείμενο i*
μ	*Μέσος όρος όλων των βαθμολογιών*
b_u	*Τάση χρήστη (User bias)*
b_i	*Τάση αντικειμένου (Item bias)*
b_{ui}	*Εκτίμηση βάσης (baseline estimation) για τον χρήστη u και το αντικέιμενο i*
K	*Σύνολο των διαθέσιμων βαθμολογιών* r_{ui}

Η εκτιμώμενη βαθμολογία υπολογίζεται από τον τύπο (45).

$$\tilde{r}_{ui} = b_{ui} + p_u^T q_i \qquad (45)$$

Προκειμένου να βρεθούν οι τιμές p_u, q_i, b_{ui} γίνεται χρήση της *συνάρτησης σφάλματος* (46).

$$\sum_{(u,i)\in K} (r_{ui} - b_{ui} - p_u^T q_i)^2 \tag{46}$$

Όπου $b_{ui} = \mu + b_u + b_i$. Προσθέτοντας τον *συντελεστή κανονικοποίησης* προκύπτει η σχέση (47).

$$\sum_{(u,i)\in K} (r_{ui} - b_{ui} - p_u^T q_i)^2 + \lambda(\|p_u\|^2 + \|q_i\|^2 + b_u^2 + b_i^2) \tag{47}$$

Η τιμή της σταθεράς κανονικοποίησης λ, βρίσκεται πειραματικά. Για να ελαχιστοποιηθεί η προηγούμενη συνάρτηση για τις τιμές $p_u, q_i, b_u και b_i$ χρησιμοποιείται η μέθοδος της *απότομης καθόδου* (*gradient descent*) στην οποία ενημερώνονται επαναληπτικά οι τιμές (48) (49) (50) (51). Κάθε ενημέρωση πραγματοποιείται, ολοκληρώνοντας την επανάληψη στο σετ για κάθε γνωστή βαθμολογία.

$$p_u \leftarrow p_u + \gamma'((r_{ui} - \tilde{r}_{ui})q_i + \lambda' p_u) \tag{48}$$

$$q_i \leftarrow q_i + \gamma'\big((r_{ui} - \tilde{r}_{ui})q_i + \lambda' q_i\big) \tag{49}$$

$$b_u \leftarrow b_u + \gamma''\big((r_{ui} - \tilde{r}_{ui})q_i + \lambda'' b_u\big) \tag{50}$$

$$b_i \leftarrow b_i + \gamma''\big((r_{ui} - \tilde{r}_{ui})q_i + \lambda'' b_i\big) \tag{51}$$

όπου γ', γ'' το βήμα και λ', λ'' οι σταθερές κανονικοποίησης.

4.3.2 Παράδειγμα σε απλοποιημένη εκδοχή πίνακα βαθμολογιών

Ως δοκιμαστική περίπτωση θεωρείται ένα απλό σενάριο με μόνο τρεις χρήστες (u_0, u_1, u_2) και τρία αντικείμενα (i_o, i_1, i_2). Ο πίνακας βαθμολογιών είναι:

T	i_0	i_1	i_2
u_0	4	5	4
u_1	1	2	2
u_2	4	5	5

Η εγγραφή $r_{uj} = k$ δηλώνει ότι ο χρήστης u βαθμολόγησε το αντικείμενο j με βαθμολογία k. Αφαιρούνται κάποια από τα δεδομένα ώστε να προβλεφθεί η τιμή.

T	i_0	i_1	i_2
u_0	4	x_1	4
u_1	1	x_2	2
u_2	4	5	5

Στο συγκεκριμένο παράδειγμα μπορεί να μαντέψει κάποιος, ενστικτωδώς τις τιμές που λείπουν, χωρίς να είναι γνωστές οι πραγματικές τιμές, εφόσον οι χρήστες u_0 και u_1 έχουν αντίθετες επιθυμίες με βάση τις βαθμολογίες που δόθηκαν, ενώ οι u_0 και u_2 ίδιες.

Αναμένεται μεγάλη τιμή για την βαθμολογία x_1 και μικρή για την x_2. Χρησιμοποιώντας το μοντέλο που περιγράφηκε προηγουμένως προκύπτουν οι τιμές

$$r_{01} = x_1 = 4.78$$

$$r_{11} = x_2 = 2.28$$

Το τετραγωνικό σφάλμα σε αυτή την περίπτωση είναι :

$$RMSE = \frac{1}{\sqrt{2}} \cdot \sqrt{(2.28 - 2)^2 + (4.78 - 5)^2} = 0.252$$

τιμή που σημαίνει ότι η πρόβλεψη είναι ακριβής.

4.3.3 Αξιολόγηση του μοντέλου SVD

Όπως αναφέρθηκε προηγουμένως χρησιμοποιείται το RMSE, για την αξιολόγηση των προβλέψεων. Τα σετ χωρίζονται σε *εκπαίδευσης (train)* και *ελέγχου (test)* με ένα ποσοστό 80% και 20% αντίστοιχα. Για να αξιολογηθεί το αποτελέσματα του αλγορίθμου SVD συγκρίνεται το σφάλμα του με πιο απλές μεθόδους, όπως η *βασική εκτίμηση* που αναφέρθηκε προηγουμένως και μία *αφελή μέθοδο* (*naïve*), στην οποία θεωρείται ότι, η πρόβλεψη για κάθε ζευγάρι χρήστη-αντικειμένου είναι σταθερή και ίση με τον μέσο όρο

στο *σετ εκπαίδευσης*. Ο μέσος όρος του *σετ εκπαίδευσης* για τις δραστηριότητες είναι $\mu = 4.148$ που σημαίνει ότι κάθε δραστηριότητα που δεν έχει βαθμολογηθεί, θα λάβει αυτή την τιμή. Το RMSE για αυτή την μέθοδο στο σετ των δραστηριοτήτων είναι 0.94618.

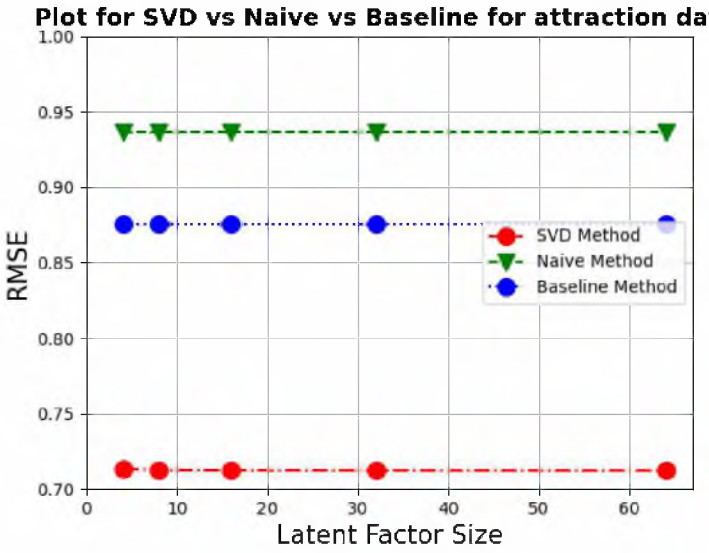
Η μέθοδος βασικής εκτίμησης που κάνει χρήση των τάσεων των χρηστών και των αντικειμένων (κεφ. 4.3.1), στον υπολογισμό της εκτίμησης b_{ui} βελτιώνει το σφάλμα RMSE σε 0.884312 (εικόνα 44).

Τέλος ο αλγόριθμος SVD βελτιώνει ακόμα περισσότερο το σφάλμα. Επιλέγονται οι βέλτιστες τιμές των *υπερπαραμέτρων*, εμπειρικά, δοκιμάζοντας αλλαγές διαδοχικά. Αυτή η διαδικασία επαναλαμβάνεται για κάθε σετ ξεχωριστά αφού επηρεάζονται από τα δεδομένα του καθενός. Οι δοκιμές έδωσαν καλύτερο σφάλμα RMSE για το σετ των δραστηριοτήτων με $\gamma' = \gamma'' = 0.09$ και $\lambda' = \lambda'' = 2.4$ όπως φαίνεται αναλυτικά στον πίνακα 13.

Πίνακας 13. Σφάλμα με Προσαρμογή Υπερπαραμέτρων για το Σετ Δραστηριοτήτων

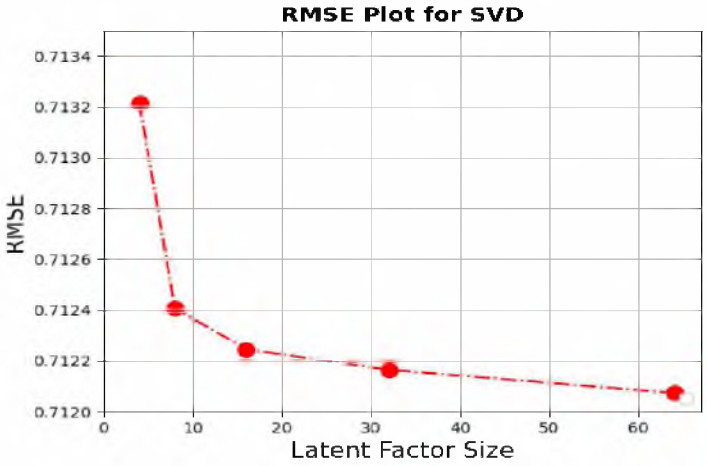
γ	RMSE
0.001	0.823870
0.005	0.797730
0.01	0.773477
0.02	0.749553
0.03	0.738860
0.04	0.732544
0.05	0.729130
0.06	0.727270
0.07	0.726235
0.08	0.725714
0.09	**0.725581**
0.1	0.725773
0.2	0.729191

λ	RMSE
0.5	0,735059
0.6	0.731911
0.7	0.728580
0.8	0.725581
0.9	0.723100
1.0	0.721102
2.0	0.712588
2.2	0.712345
2.3	0.712193
2.4	**0.712164**
2.5	0.714174
2.6	0.712219
3.0	0.712650



Εικόνα 44. Σφάλμα ανά μέθοδο , δραστηριότητες

Το σφάλμα επηρεάζεται επιπλέον και από τον αριθμό των *λανθανόντων συντελεστών* που χρησιμοποιούνται στην εκπαίδευση (εικόνα 45). Τα μοντέλα είναι εκπαιδευμένα με 64.



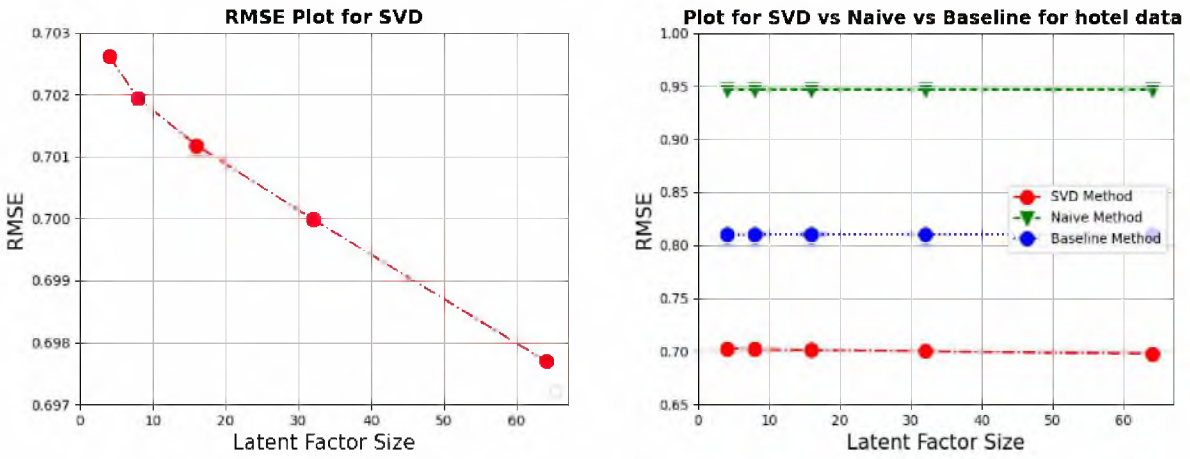
Εικόνα 45. Σφάλμα ανά αριθμό λανθανόντων συντελεστών , δραστηριότητες

Αντίστοιχα για τα ξενοδοχεία η μέθοδος *naïve* μας δίνει σφάλμα 0.94871 και η *βασική εκτίμηση* 0.82654 (εικόνα 46). Τα αποτελέσματα για τον SVD αναλύονται στον πίνακα 14. Το ελάχιστο σφάλμα επιτυγχάνεται με $\gamma' = \gamma'' = 0.08$ και $\lambda' = \lambda'' = 4.5$.

Πίνακας 14. Σφάλμα με Προσαρμογή Υπερπαραμέτρων για το Σετ Ξενοδοχείων

γ	RMSE
0.001	0.854027
0.005	0.846054
0.01	0.778961
0.02	0.760619
0.03	0.746235
0.04	0.737640
0.05	0.734897
0.06	0.732604
0.07	0.732556
0.08	**0.732413**
0.09	0.732961
0.1	0.733371
0.2	0.734272

λ	RMSE
0.5	0.753714
0.6	0.745828
0.7	0.739213
0.8	0.732413
0.9	0.728942
1.0	0.724265
2.0	0.705709
2.5	0.702798
3.0	0.701580
4.0	0.700945
4.3	0.700491
4.5	**0.697887**
4.6	0.700215
5.0	0.700636



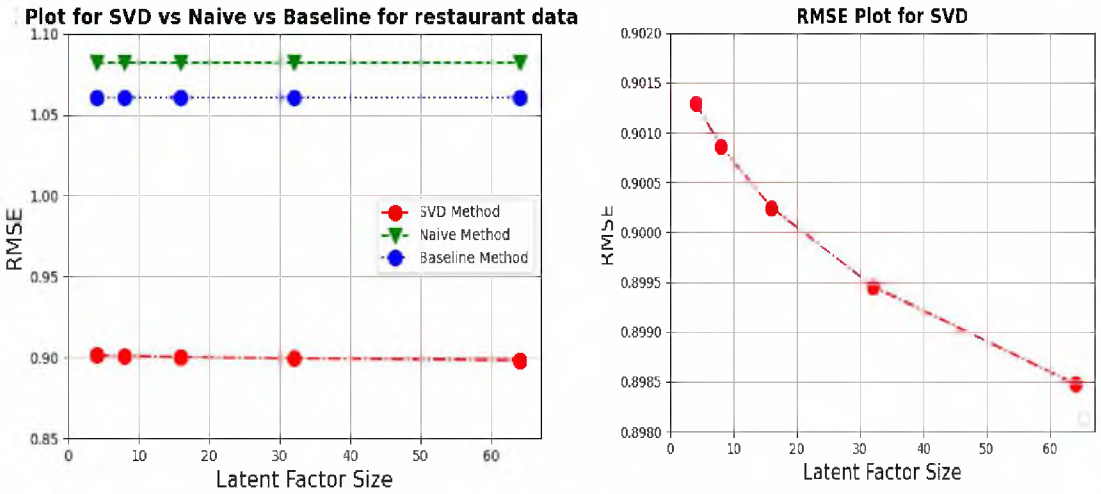
Εικόνα 46. Σφάλμα ανά μέθοδο και αριθμό λανθανόντων συντελεστών

Τέλος για τα εστιατόρια η μέθοδος *naïve* παρουσιάζει σφάλμα RMSE=1.08439 και η *βασική εκτίμηση* RMSE=1.06217 (εικόνα 47). Τα αποτελέσματα για τον SVD αναλύονται στον πίνακα 15.

Πίνακας 15. Σφάλμα με Προσαρμογή Υπερπαραμέτρων για το Σετ Εστιατορίων

γ	RMSE
0.001	0.961191
0.005	0.935783
0.01	0.922967
0.02	0.913175
0.03	0.910184
0.04	0.905179
0.05	**0.901165**
0.06	0.907182
0.07	0.912178
0.08	0.916865
0.09	0.921098
0.1	0.926071

λ	RMSE
0.5	0.909565
0.6	0.906262
0.7	0.903187
0.8	0.901165
0.9	**0.898412**
1.0	0.901127
2.0	0.910314
3.0	0.919732



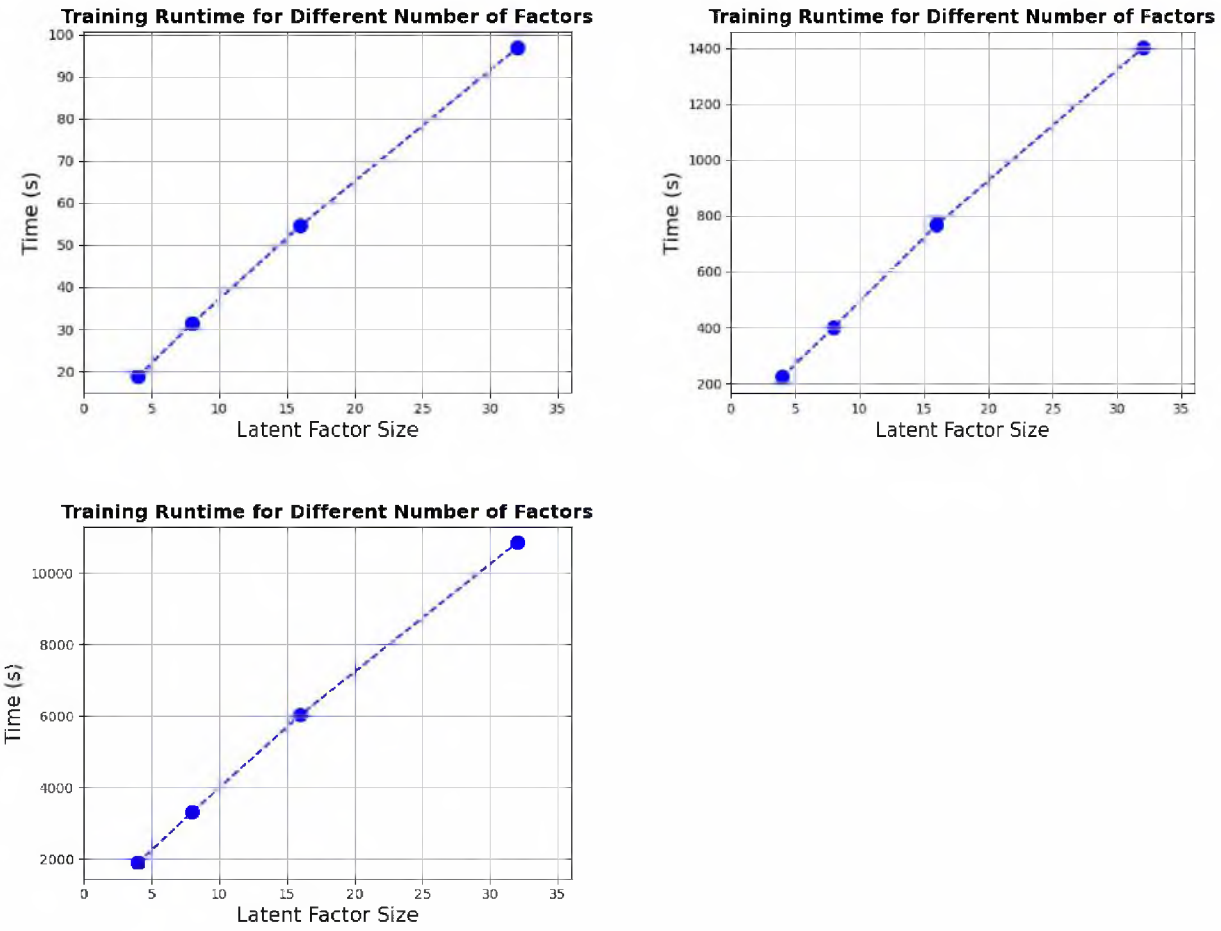
Εικόνα 47. Σφάλμα ανά μέθοδο και αριθμό λανθανόντων συντελεστών

4.3.4 Αποτελέσματα

Τα αποτελέσματα θεωρούνται ακριβή εφόσον το σφάλμα που προέκυψε από τον αλγόριθμο SVD και στα τρία σετ είναι αρκετά μικρότερο από τις πιο απλές μεθόδους που αναφέρθηκαν. Το RMSE ελαττώνεται όσο αυξάνονται οι *λανθάνοντες συντελεστές*. Επίσης

παρατηρούμε ότι αυξάνοντας των αριθμό τους, αυξάνεται γραμμικά και ο χρόνος εκτέλεσης. Οι μετρήσεις έγιναν χρησιμοποιώντας Intel®Core™ i5-8250U στα 1.8GHz.

Ο χρόνος ποικίλει ανάλογα με το σετ. Για τις δραστηριότητες έχουμε πολύ λιγότερα δεδομένα τόσο σε αντικείμενα όσο και σε κριτικές σε σχέση με τα ξενοδοχεία και τα εστιατόρια για αυτό και ο χρόνος εκτέλεσης είναι σημαντικά μικρότερος (εικόνα 48).



Εικόνα 48. Χρόνος εκπαίδευσης των μοντέλων με βάση τον αριθμό των λανθανόντων συντελεστών

4.4 Τελικό μοντέλο συστάσεων

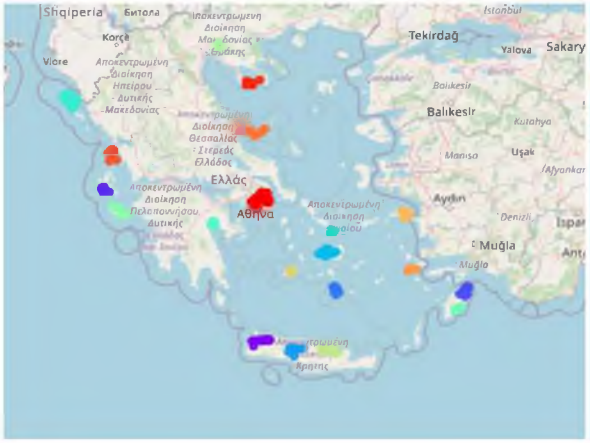
Αφού κατασκευαστεί και εκπαιδευτεί το μοντέλο είναι σε θέση να προτείνει στον χρήστη επιλογές για το κάθε αντικείμενο. Σκοπός της εργασίας είναι, να δημιουργήσει ένα ολοκληρωμένο πλάνο διακοπών για τον χρήστη, χωρίς να λάβει υπόψιν την τοποθεσία

παρά μόνο τις προτιμήσεις του. Αυτό υλοποιείται με την *ομαδοποίηση* (*clustering*) των τελικών προτάσεων.

Τα αντικείμενα κάθε κατηγορίας (δραστηριότητες, ξενοδοχεία, εστιατόρια), που θεωρεί ως κατάλληλα το μοντέλο για τον χρήστη αποθηκεύονται μαζί σε ένα καινούργιο σετ. Στην συνέχεια *ομαδοποιούνται* με βάση τις συντεταγμένες τους, με την χρήση του αλγορίθμου DBSCAN [40]. Τα δεδομένα ομαδοποιούνται τέσσερις φορές, θέτοντας διαφορετικό χιλιομετρικό όριο απόστασης μεταξύ των σημείων.

Σκοπός είναι να παρέχονται στον χρήστη πλάνα διαφορετικών ημερών, τα οποία διαφέρουν ως προς την απόσταση. Αναλύοντας τα σημεία των συντεταγμένων σε χάρτη με χρήση του πακέτου Folium της Python και κάνοντας δοκιμές καταλήξαμε στις αποστάσεις 15,25,35,50 χιλιομέτρων μεταξύ των πιο μακρινών σημείων (ορίων) της κάθε ομάδας (*cluster*). Αυτές οι τιμές αφορούν τα πλάνα των 4 ημερών, 6 ημερών, 8 ημερών και 12 ημερών αντίστοιχα

Αριστερά (εικόνα 49) ένα παράδειγμα για πλάνο των 4 ημερών και δεξιά (εικόνα 50) ένα των 12. Παρατηρούμε πόσο πιο ανοιχτές σε χιλιομετρική απόσταση είναι οι προτάσεις της δεύτερης περίπτωσης.



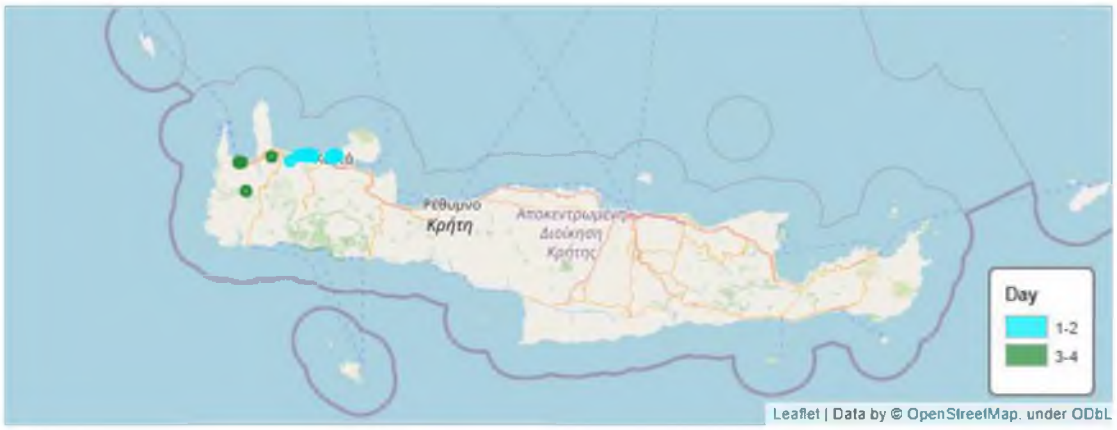
Εικόνα 49. Clustering 4 ημερών με 15km όριο

Εικόνα 50. Clustering 12 ημερών με 50km όριο

Το κάθε cluster αντιστοιχεί σε ένα πλάνο και περιέχει αντικείμενα και των τριών κατηγοριών. Προκειμένου να προταθεί στον χρήστη πρέπει να πληροί την προϋπόθεση να περιέχει τουλάχιστον ένα αντικείμενο από κάθε κατηγορία για κάθε μέρα. Εφόσον ισχύει η

προηγούμενη προϋπόθεση, το cluster ομαδοποιείται ξανά αυτή την φορά με την μέθοδο των *K-μέσων (K-means)*.

Το εσωτερικό cluster του καθενός, αντιστοιχεί σε ένα διήμερο με προτάσεις για κάθε αντικείμενο. Στην κάτω εικόνα (εικόνα 51) έχουμε ένα πλάνο 4 ημερών όπου τα γαλάζια σημεία αφορούν προτάσεις για τις 2 πρώτες ημέρες ενώ τα πράσινα για τις υπόλοιπες. Παρατηρώντας την εικόνα 49, φαίνεται ότι είναι ένα από τα αρχικά cluster που ομαδοποιήθηκε ξανά.



Εικόνα 51. Εσωτερικό cluster ενός 4ημερου πλάνου

Τέλος τα εστιατόρια που προτείνονται είναι χωρισμένα με βάση το γνώρισμα γεύμα και προτείνονται ξεχωριστά για πρωινό, μεσημεριανό, βραδινό και καφέ/μπαρ.

ΚΕΦΑΛΑΙΟ 5

ΠΕΡΙΓΡΑΦΗ ΕΦΑΡΜΟΓΗΣ

5.1 Εργαλεία εφαρμογής

Η υλοποίηση της εφαρμογής αποτελείται από έξι στάδια (εικόνα 52) κατά τα οποία χρησιμοποιήθηκαν συγκεκριμένα εργαλεία της Python.

Εικόνα 52. Στάδια εφαρμογής

5.1.1 Συλλογή Δεδομένων

Το πρώτο στάδιο της εφαρμογής αφορά την συλλογή των δεδομένων. Η συλλογή δεδομένων από το διαδίκτυο υλοποιείται μέσω προγραμμάτων λογισμικού τα οποία πραγματοποιούν την διαδικασία της συγκομιδής (scraping). Όπως αναφέρθηκε στο κεφ. 4, η συλλογή πραγματοποιήθηκε μέσω του framework ανοιχτού κώδικα της Python, Scrapy (https://scrapy.org/).

Το συγκεκριμένο εργαλείο εκτός της συγκομιδής δίνει την δυνατότητα του web crawling, δηλαδή την αυτόματη μεταφορά από μία σελίδα σε επόμενες, μέσω συνδέσμων (links) που υπάρχουν στο HTML αρχείο. Η συλλογή από το HTML αρχείο γίνεται με χρήση CSS & XPath selectors.

Το Scrapy είναι ένα πολύ δυνατό εργαλείο δίνοντας την δυνατότητα επεξεργασίας των δεδομένων πριν την αποθήκευση τους. Επίσης δίνει την δυνατότητα αποθήκευσης τους, σε διαφορετικούς τύπους αρχείων. Στην προκειμένη περίπτωση αποθηκεύτηκαν σε αρχεία

JSON. Παρόλα αυτά, ένα πρόβλημα που αντιμετωπίζει είναι τα κομμάτια JavaScript που είναι ενσωματωμένα σε κάποιες περιπτώσεις. Για να αντιμετωπιστεί έγινε χρήση, σε συνδυασμό με το Scrapy, της βιβλιοθήκης Selenium (https://www.selenium.dev/). Η λειτουργία της είναι η αυτοματοποίηση των περιηγητών ιστού (web browsers), επιτρέποντας να συλλέξουμε πληροφορία που διαφορετικά δεν θα ήταν διαθέσιμη.

Τέλος για την διαδικασία χρησιμοποιήθηκαν 5 ιδιωτικά proxies τα οποία ενοικιάστηκαν από τον ιστότοπο Limeproxies (https://www.limeproxies.com/), δίνοντας την δυνατότητα εναλλαγής των IP, με σκοπό να αποφευχθεί ένας πιθανός αποκλεισμός.

5.1.2 Επεξεργασία/Καθαρισμός Δεδομένων

Τα δεδομένα αποθηκεύτηκαν σε μορφή JSON όπως προαναφέρθηκε με σκοπό να μετατραπούν σε *πλαίσια δεδομένων* (*DataFrames*), τα οποία μπορούν να επεξεργαστούν με ευκολία. Για την διαδικασία αυτή χρησιμοποιήθηκε η βιβλιοθήκη Pandas της Python, η οποία βασίζεται σε δύο βασικές βιβλιοθήκες, την Numpy και την Matplotlib. Πρόκειται για μία από τις πιο συχνά χρησιμοποιούμενες βιβλιοθήκες χάρη στην ταχύτητα και την απλότητά της.

Επίσης για την μετατροπή των διευθύνσεων σε συντεταγμένες, έγινε χρήση του Geocoding API της Google.(https://developers.google.com/maps/documentation/geocoding/overview)

5.1.3 Διερευνητική ανάλυση δεδομένων

Βασικό κομμάτι της ανάλυσης πέρα από τα στατιστικά στοιχεία που γίνονται διαθέσιμα μέσω της βιβλιοθήκης Pandas, είναι η *απεικόνιση* (*visualization*) των δεδομένων. Για τον σκοπό αυτό χρησιμοποιήθηκαν οι βιβλιοθήκες Seaborn και Plotly.

Η βιβλιοθήκη Seaborn είναι από τις πιο γνωστές για την απεικόνιση δεδομένων και είναι χτισμένη πάνω από την βασική Matplotlib, προσφέροντας περισσότερες δυνατότητες και πιο απλή χρήση : '*If Matplotlib "tries to make easy things easy and hard things possible", seaborn tries to make a well-defined set of hard things easy too" — Michael Waskom (Creator of Seaborn)*'.

Επίσης χρησιμοποιήθηκε η βιβλιοθήκη γραφημάτων/διαγραμμάτων ανοιχτού κώδικα της Python, Plotly (https://plotly.com/) .

Τέλος για την απεικόνιση των δεδομένων συντεταγμένων σε διαδραστικούς χάρτες, έγινε χρήση της βιβλιοθήκης Folium (http://python-visualization.github.io/folium/). Πρόκειται για πλούσια σε δυνατότητες βιβλιοθήκη, απεικόνισης *γεωχωρικών δεδομένων* (*geospatial data*) η οποία είναι κατασκευασμένη πάνω στην βιβλιοθήκη της JavaScript, leaflet.js.

5.1.4 Αποθήκευση σε Βάση Δεδομένων

Αφού ολοκληρώθηκαν τα προηγούμενα βήματα, προκειμένου να χρησιμοποιηθούν τα δεδομένα για την κατασκευή του μοντέλου και στην συνέχεια από το γραφικό περιβάλλον, αποθηκεύτηκαν σε βάση δεδομένων. Χρησιμοποιήθηκε η SQLite3, μία υψηλών επιδόσεων βιβλιοθήκη που παρέχει ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS). Χρήσιμα χαρακτηριστικά της είναι η *serverless αρχιτεκτονική της*, δεν χρειάζεται διακομιστή (server) για να τρέξει επειδή η βάση είναι ενσωματωμένη στην εφαρμογή που την προσπελαύνει, *η αυτονομία της* (*self-contained*), δηλαδή η απαίτηση ελάχιστης υποστήριξης από το λειτουργικό και τις εξωτερικές βιβλιοθήκες, απαιτεί *zero-configuration* εφόσον έχει *serveless αρχιτεκτονική* δεν χρειάζεται εγκατάσταση πριν την χρήση της και τέλος είναι *transactional* είναι δηλαδή συμμορφωμένη με το σύνολο ιδιοτήτων ACID που εγγυάται αξιοπιστία στις συναλλαγές της βάσης.

5.1.5 Κατασκευή μοντέλου

Για την υλοποίηση του μοντέλου αρχικά χρησιμοποιήθηκε η βιβλιοθήκη ανοιχτού κώδικα Scipy. Πρόκειται για την πιο γνωστή βιβλιοθήκη της Python όσον αφορά επιστημονικούς και τεχνικούς υπολογισμούς. Η Scipy χρησιμοποιεί πίνακες Numpy ως βασική δομή υπολογισμών και είναι πολύ χρήσιμη για προβλήματα βελτιστοποίησης, πράξεων γραμμικής άλγεβρας κτλ.

Επιπλέον χρησιμοποιήθηκε η βιβλιοθήκη ανοιχτού κώδικα Scikit learn, η οποία παρέχει αποτελεσματικά και εύκολα στην χρήση εργαλεία για την υλοποίηση και επεξεργασία αλγορίθμων μηχανικής μάθησης.

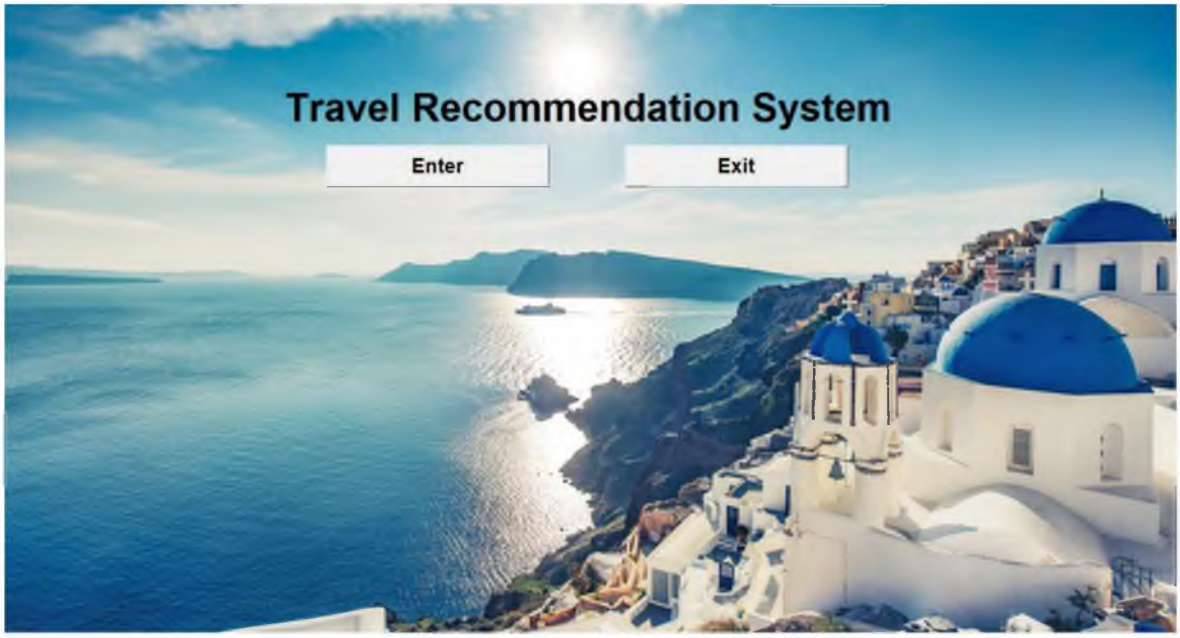
5.1.6 Γραφικό περιβάλλον

Προκειμένου να γίνει πιο ευχάριστη η εμπειρία του χρήστη, η εφαρμογή της εργασίας υλοποιήθηκε μέσω μίας *γραφικής διεπαφής χρήστη* (GUI). Για τον σκοπό αυτό χρησιμοποιήθηκε η GUI βιβλιοθήκη Tkinter της Python. (https://docs.python.org/3/library/tkinter.html).

Πρόκειται για μία διεπαφή (interface) προς το Tk GUI toolkit που είναι ενσωματωμένο στην Python. Βασίζεται στην λογική των Widgets δηλαδή γραφικών στοιχείων όπως μενού, κουμπιά, πλαίσια κειμένων κτλ. για την κατασκευή του γραφικού περιβάλλοντος.

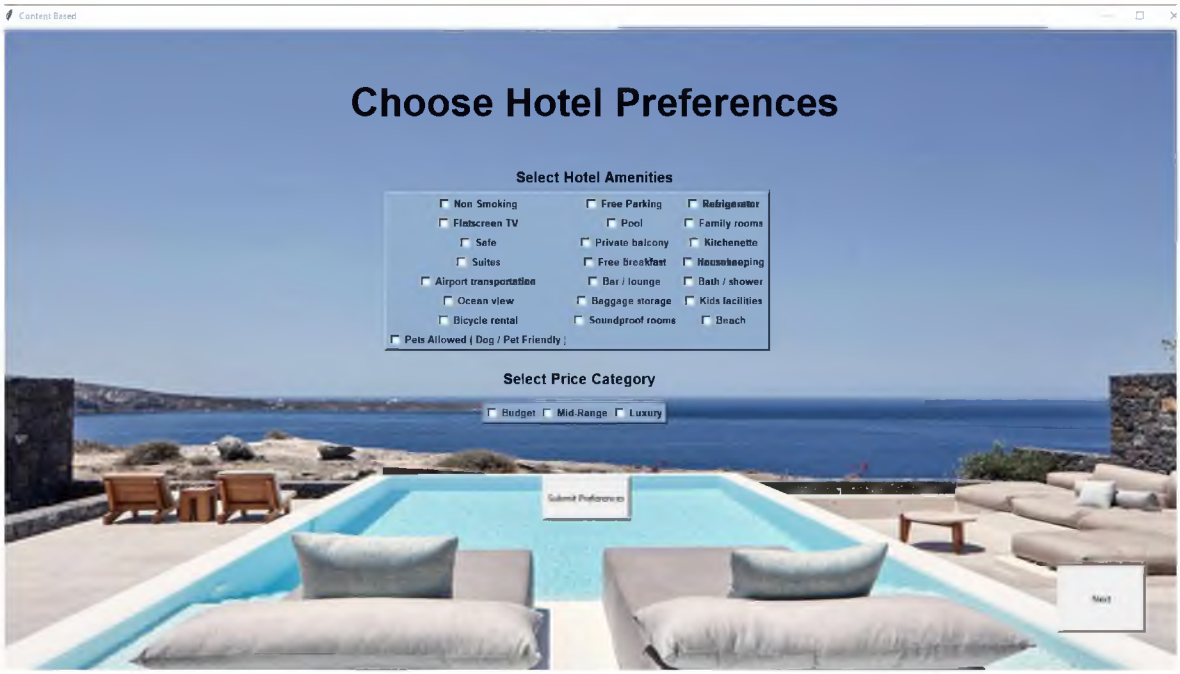
5.2 Παρουσίαση εφαρμογής

Η αρχική σελίδα της εφαρμογής περιέχει δύο κουμπιά (buttons), ένα για συνέχεια στην εφαρμογή και ένα για το κλείσιμό της (εικόνα 53).



Εικόνα 53. Αρχική σελίδα εφαρμογής

Στην συνέχεια παρουσιάζεται στον χρήστη ένα σύνολο από επιλογές, για τις οποίες πρέπει να δηλώσει την προτίμησή του, ώστε να δημιουργηθεί το προφίλ του. Η διαδικασία αυτή γίνεται για όλες τις κατηγορίες αντικειμένων δηλαδή ξενοδοχεία, δραστηριότητες και εστιατόρια (εικόνες 54-56).



Εικόνα 54. Επιλογές ξενοδοχείων

Εικόνα 55. Επιλογές Δραστηριοτήτων

Εικόνα 56. Επιλογές Εστιατορίων

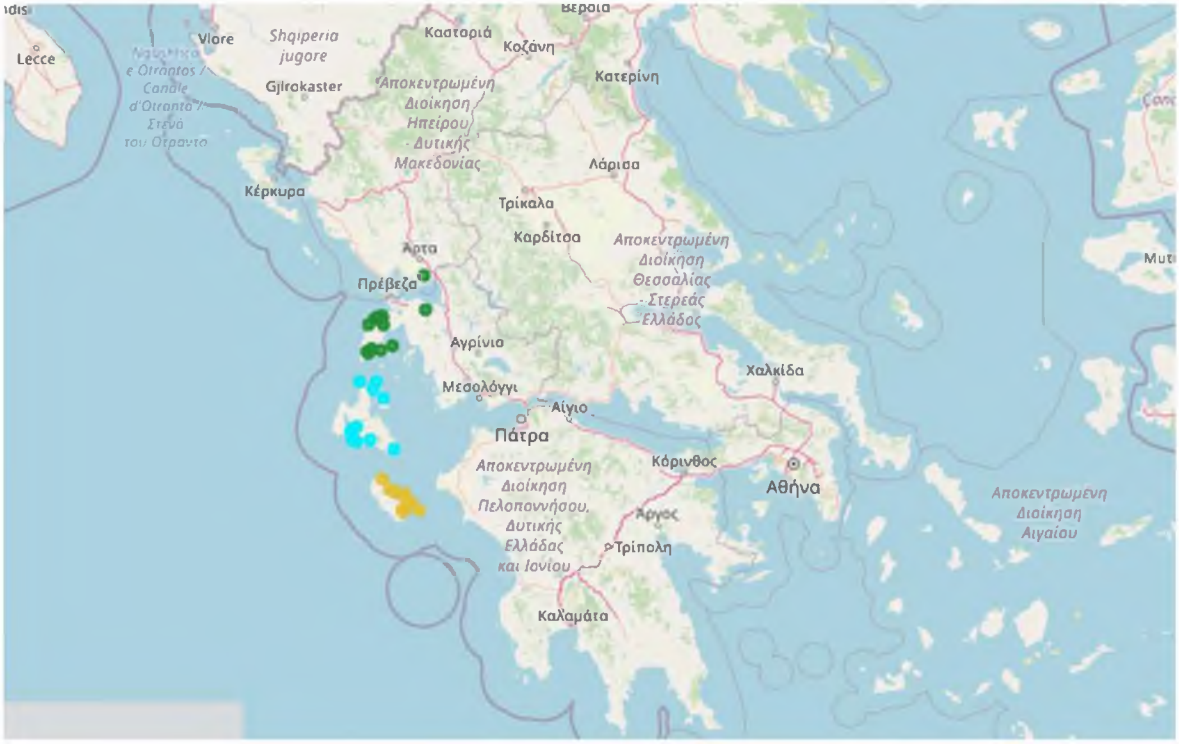
Μετά από αυτά τα βήματα το σύστημα κατασκευάζει το προφίλ του χρήστη και υπολογίζει τις προτάσεις που θα του παρουσιάσει. Όπως προαναφέρθηκε στο κεφ. 4, οι προτάσεις στην συνέχεια ομαδοποιούνται με βάση ορισμένα κριτήρια απόστασης, για να κατασκευαστούν ολοκληρωμένα πλάνα για ένα σύνολο από μέρες (εικόνα 57).

Εικόνα57. Διαθέσιμα Πλάνα

Στην συνέχεια ο χρήστης μπορεί να επιλέξει από την διαθέσιμη λίστα πλάνων που του προσφέρονται ώστε να δει αναλυτικά τις επιλογές που έχει ομαδοποιημένες ανά διήμερο (εικόνα 58).

Εικόνα 58. Εμφάνιση προτάσεων

Τέλος υπάρχει η δυνατότητα εμφάνισης του χάρτη με τις προτάσεις για κάθε πλάνο, ομαδοποιημένες ανά διήμερο (εικόνα 59).



Εικόνα 59. Χάρτης πλάνου 6 ημερών

Η εφαρμογή είναι διαθέσιμη:

https://github.com/georanton/Personalized_Recommendation_System_for_Tourism_Greece.

Το πρόγραμμα συλλογής δεδομένων είναι διαθέσιμο:

https://github.com/georanton/tripadvisor_scrape_Greece_scrapy

ΚΕΦΑΛΑΙΟ 6

ΕΠΙΛΟΓΟΣ

6.1 Σύνοψη

Σκοπός της εργασίας είναι η υλοποίηση ενός ΣΣ διακοπών για την χώρα της Ελλάδας. Οι συστάσεις αποτελούν ένα ολοκληρωμένο πλάνο διακοπών, στο οποίο περιέχονται ξενοδοχεία, δραστηριότητες και εστιατόρια για κάθε μέρα.

Πρόκειται για ένα υβριδικό μοντέλο ΣΣ το οποίο αντιμετωπίζει το πρόβλημα της *ψυχρής εκκίνησης* συλλέγοντας από τον χρήστη τις προτιμήσεις του, για το κάθε αντικείμενο δημιουργώντας ένα εικονικό προφίλ. Η διαδικασία αυτή κάνει χρήση ενός CBF μοντέλου ώστε να βρει και να διαχειριστεί τις ομοιότητες των αντικειμένων στην δημιουργία του.

Το αποτέλεσμα δίνεται ως είσοδος σε ένα μοντέλο SVD το οποίο εξάγει χαρακτηριστικά και συσχετίσεις από τον ΠΒΧ, δίνοντας τις τελικές προτάσεις για κάθε κατηγορία αντικειμένου ξεχωριστά. Οι προτάσεις ομαδοποιούνται για να δημιουργήσουν τα τελικά πλάνα διακοπών που είναι διαθέσιμα για τον χρήστη.

Οι τελικές προτάσεις όπως και η διαδικασία της συλλογής των προτιμήσεων του χρήστη, πραγματοποιούνται μέσω μίας διαδραστικής και φιλικής προς τον χρήστη εφαρμογής. Μέσω αυτής, έχει την δυνατότητα να αλληλεπιδράσει με τις προτάσεις μαθαίνοντας πληροφορίες για τα αντικείμενα, βλέποντας τις εικόνες τους όπως και τους χάρτες με τα πλάνα που του προτείνονται.

Επιπλέον γίνεται μία σύντομη ανάλυση της θεωρίας των αλγορίθμων και των μεθόδων που χρησιμοποιήθηκαν κατά την υλοποίηση, όσο και των συστημάτων συστάσεων γενικότερα. Τέλος περιγράφεται η διαδικασία της συλλογής πραγματικών δεδομένων που χρησιμοποιήθηκαν.

6.2 Μελλοντική βελτίωση

Το λειτουργικό κομμάτι της εργασίας στοχεύει στο να προτείνει ολοκληρωμένες προτάσεις διακοπών στους χρήστες με βάση τις προτιμήσεις τους. Χάρη σε διαφορετικές τεχνικές και εργαλεία της Επιστήμης Δεδομένων η εφαρμογή επιτυγχάνει στον στόχο της,

αφού στο τελικό στάδιο έχουμε τα επιθυμητά αποτελέσματα. Πρόκειται για μία εφαρμογή με ευρεία χρήση σε ένα πεδίο συνεχούς ανάπτυξης, συνεπώς υπάρχει δυνατότητα περαιτέρω βελτίωσης και αναβάθμισης. Κάποιες προτάσεις είναι οι εξής :

i. Τα δεδομένα που συλλέχθηκαν και χρησιμοποιούνται αφορούν αποκλειστικά ξενοδοχεία, δραστηριότητες και εστιατόρια στην Ελλάδα. Υπάρχει δυνατότητα να υλοποιηθεί σε πιο ευρεία κλίμακα για προορισμούς σε παγκόσμιο επίπεδο.
ii. Τα δεδομένα συλλέχθηκαν από τον ιστότοπο του Tripadvisor. Όπως αναφέρθηκε προηγουμένως στην περιγραφή στόχος ήταν να υπάρξουν περισσότερες πηγές. Η συλλογή μπορεί να γίνει από περισσότερους ιστότοπους όπως επίσης και από μέσα κοινωνικής δικτύωσης.
iii. Το σύστημα συστάσεων μπορεί να βελτιωθεί λαμβάνοντας υπόψιν τον τύπο των χρηστών, για παράδειγμα αν πρόκειται για ζευγάρι, οικογένεια όπως επίσης και τον λόγο του ταξιδιού (επαγγελματικό, ψυχαγωγίας κτλ.).
iv. Επίσης θα ήταν δυνατό αν υπήρχε μεγαλύτερος όγκος δεδομένων, οι προτάσεις να αφορούν συγκεκριμένες εποχές του χρόνου. Για παράδειγμα τα νησιά είναι συνήθως καλοκαιρινός προορισμός σε σχέση με τους χειμερινούς μήνες όπου τα θέρετρα κοντά σε χιονοδρομικά κέντρα, τείνουν να προσελκύουν περισσότερο κόσμο.
v. Επειδή στον χρήστη παρέχεται ένα ολοκληρωμένο πλάνο που περιέχει όλα τα αντικείμενα, ως παραλλαγή της υπάρχουσας υλοποίησης είναι δυνατόν να του δίνεται η δυνατότητα να επιλέξει τα αντικείμενα που τον ενδιαφέρουν. Για παράδειγμα είναι πιθανό να μην ενδιαφέρεται για ξενοδοχείο λόγω του ότι επιθυμεί να κατασκηνώσει, έχει τροχόσπιτο κτλ. Σε αυτή την περίπτωση θα πρέπει να προτείνεται πλάνο με τα υπόλοιπα αντικείμενα.
vi. Τέλος είναι δυνατόν να χρησιμοποιηθούν αλγόριθμοι και τεχνικές νευρωνικών δικτύων προκειμένου να βελτιώσουν την ακρίβεια των αποτελεσμάτων όπως και την συνολική υπολογιστική ικανότητα της εφαρμογής, όπως Restricted Boltzman Machines και Auto-Encoders.

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] Recommender Systems, The Textbook 1st Edition, 2016 , Charu C. Aggarwal

[2] P. Resnick, Hal R.Varian, 1997. Recommender systems, in: Communications of the ACM,Volume 40,Issue 3,pp 56–58,https://doi.org/10.1145/245108.245121

[3] A. Poriya, N. Patel, T.Bhagat, R. Sharma, 2014. Non-Personalized Recommender Systems and Userbased Collaborative Recommender Systems, in: International Journal of Applied Information Systems (IJAIS) – ISSN: 2249-0868

[4] Y. Deldjoo, M. Elahi, M. Quadrana, P. Cremonesi, 2015. Toward Building a Content-Based Video Recommendation System Based on Low-Level Features

[5] X. Su, T. M. Khoshgoftaar, 2009. A survey of collaborative filtering techniques, in: Advances in Artificial Intelligence, Volume 2009, Article No: 4, pp2, https://doi.org/10.1155/2009/421425

[6] S. Tsuge, M. Shishibori, S. Kuroiwa, K. Kita, 2001. Dimensionality reduction using non-negative matrix factorization for information retrieval

[7] R. Burke, 2001. Hybrid Recommender Systems: Survey and Experiments, in: User Modeling and User-Adapted Interaction 12(4), DOI: 10.1023/A: 1021240730564

[8] A. Cano, 2019. Recommender Systems and Hyper-parameter tuning, in: https://towardsdatascience.com/recommender-systems-and-hyper-parameter-tuning-25567b10e298

[9] T. Di Noia, V.Claudio Ostuni, 2015. Recommender Systems and Linked Open Data, in: Reasoning Web. Web Logic Rules, 11th International Summer School, Berlin, Germany

[10] T. Silveira, M. Zhang, X. Lin, Y. Liu, S. Ma, 2019. How good your recommender system is? A survey on evaluations in recommendation, in: International Journal of Machine Learning and Cybernetics 10:813–831 https://doi.org/10.1007/s13042-017-0762-9

[11] L. P. Phan, H. X. Huynh, 2017. User based Recommender Systems using Implicative Rating Measure, in: International Journal of Advanced Computer Science and Applications · January 2017 DOI: 10.14569/IJACSA.2017.081105

[12] M. Thiele, W.Lehner, 2011. Setting Goals and Choosing Metrics for Recommender System Evaluations. https://www.researchgate.net/publication/268381252

[13] R. van Meteren, M. van Someren, 2000. Using Content-Based Filtering for Recommendation

[14] M. A. Hammed, O. Al Jadaan, R. Sirandas, 2012. Collaborative Filtering Based Recommendation System: A survey, in: International Journal on Computer Science and Engineering

[15] S-B Sun, Z-H Zhang, X-L Dong, H-R Zhang, T-J Li, L. Zhang, 2017. Integrating Triangle and Jaccard similarities for recommendation, in: PLoS ONE 12(8): e0183570. https://doi.org/10.1371/

[16] P. H. Aditya, I. Budi, Q. Munajat, 2016. A Comparative Analysis of Memory-based and Model-based Collaborative Filtering on the Implementation of Recommender System for Ecommerce in Indonesia : A Case Study PT X

[17] G.Shani, A.Gunawardana, 2011. Evaluating Recommendation Systems, in: Recommender Systems Handbook ISBN: 978-0-387-85820-3

[18] H.Ma, D.Zhou, C.Liu, M.R.Lyu, 2011. Recommender systems with social regularization, in: Conference: Proceedings of the Forth International Conference on Web Search and Web Data Mining

[19] S. Chan, P. Treleaven, L. Capra, 2013. Continuous Hyperparameter Optimization for Large-scale Recommender Systems, in: IEEE International Conference on Big Data

[20] I. B. Mitroi, F.Frasincar, 2020. An Elastic Net Regularized Matrix Factorization Technique for Recommender Systems, in: SAC '20: The 35th ACM/SIGAPP Symposium on Applied Computing

[21] Y. Koren, R. Bell, C. Volinsky ,2009. Matrix Factorization Techniques for Recommender Systems, in: Computer, Volume: 42, Page(s):30-37, DOI: 10.1109/MC.2009.263

[22] X. Zhao, Z. Niu, K. Wang, K.Niu, Z. Liu, 2015. Improving Top-N Recommendation Performance Using Missing Data, in: Hindawi Publishing Corporation Mathematical Problems in Engineering Volume, http://dx.doi.org/10.1155/2015/380472

[23] Y. Koren, 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model, in: 14th ACM SIGKDD international conference on Knowledge discovery and data mining

[24] Y.Huang, L. Bian, 2009. A Bayesian network and analytic hierarchy process based personalized recommendations for tourist attractions over the Internet, in : Expert Systems with Applications 36(1):933-943, DOI: 10.1016/j.eswa.2007.10.019

[25] D. Yeh, C. Cheng, 2015. Recommendation system for popular tourist attractions in Taiwan using Delphi panel and repertory grid techniques, in: Tourism Management 46: 164–176, DOI: 10.1016/j.tourman.2014.07.002

[26] L. Ardissono, A. Goy, G.Petrone, M. Segnan, P. Torasso, 2002. Tailoring the Recommendation of Tourist Information to Heterogeneous User Groups, in : Hypermedia: Openness, Structural Awareness, and Adaptivity, International Workshops OHS-7, SC-3, and AH-3, Aarhus, Denmark, August 14-18, 2001

[27] L. Sebastia, I. Garcia, E.Onaindia, C.Alvarez ,2009. e-Tourism: a tourist recommendation and planning application, in: International Journal on Artificial Intelligence Tools ,Vol 18, Issue 05, Page(s) 717-738

[28] Z.Bahramian, R. Ali Abbaspour, 2015. An ontology-based tourism recommender system based on spreading activation model, in: SMPR, Kish island, Iran

[29] H.Chiang, T. Huang, 2015. User-adapted travel planning system for personalized schedule recommendation, in: Information Fusion 21(1):3–17, DOI: 10.1016/j.inffus.2013.05.011

[30] J. Hossen, S. Sayeed,K. Tawsif, Md.A. Rahman, 2018. A Survey on Cleaning Dirty Data Using Machine Learning Paradigm for Big Data Analytics , in: Indonesian Journal of Electrical Engineering and Computer Science 10(3):1234-1243 , DOI: 10.11591/ijeecs.v10.i3.pp1234-1243

[31] K Sahoo ,AK Samal ,J Pramanik ,SK Pani, 2019. Exploratory Data Analysis using Python, in: International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8, Issue-12

[32] M. Volkovs, G.W Yu, T.Poutanen, 2017. DropoutNet: Addressing Cold Start in Recommender Systems, in : Advances in Neural Information Processing Systems 30 (NIPS)

[33] J. Basini,A. Shakery,B. Moshiri,M.Z. Hayat, 2010. Addressing the New User Cold-Start Problem in Recommender Systems Using Ordered Weighted Averaging Operator

[34] Y. El Allioui, 2017. A novel approach to solve the new user cold-start problem in recommender systems using collaborative filtering , in: International Journal of Scientific & Engineering Research Volume 8, Issue 11, ISSN 2229-5518

[35] V.N. Zhao,M. Moh,T. Moh, 2016. Contextual-Aware Hybrid Recommender System for Mixed Cold-Start Problems in Privacy Protection , in: IEEE International Conference on Big Data Security on Cloud (BigDataSecurity), High Performance and Smart Computing (HPSC) and Intelligent Data and Security (IDS) , 10.1109/BigDataSecurity/HPSC37848.2016

[36] F. Wang, W. Pan,Li Chen, 2013. Recommendation for New Users with Partial Preferences by Integrating Product Reviews with Static Specifications, in: 21st Conference on User Modeling, Adaptation and Personalization, Rome, Italy

[37] M. Deschpande, G. Karypis, 2000. Evaluation of Item-Based Top-N Recommendation Algorithms , in: 10th Conference of Information and Knowledge Management (CIKM), pp. 247 - 254

[38] C. Sammut,G.I Webb, 2010. Leave-One-Out Cross-Validation. In: G.I. (eds) Encyclopedia of Machine Learning and Data Mining. Springer, Boston, MA. https://doi.org/10.1007/978-1-4899-7687-1_469

[39] Z. Tan, L. He, D. Wu, Q. Chang, B. Zhang, 2020. Personalized Standard Deviations Improve the Baseline Estimation of Collaborative Filtering Recommendation, in: Applied Sciences , DOI: 10.3390/app10144756

[40] M. Ester, Hans-Peter Kriegel, J. Sander, 1996. A density-based algorithm for discovering clusters in large spatial databases with noise, in: KDD'96: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining

[41] R. Dash,R. L. Paramguru, 2011. Comparative Analysis of Supervised and Unsupervised Discretization Techniques

[42] K. Potdar, T. S. Pardawala, C. D. Pai, 2017. A Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers, in: International Journal of Computer Applications, 10.5120/IJCA2017915495

[43] M. Stone, 1976. Cross-validatory Choice and Assessment of Statistical Predictions, in: Journal of the royal statistical society series b-methodological, 10.1111/J.2517-6161.1976.TB01573.X

[44] F. Fouss, M. Saerens, 2008. Evaluating Performance of Recommender Systems: An Experimental Comparison, in: WI-IAT: Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology - Volume 01, Pages 735–738, https://doi.org/10.1109/WIIAT.2008.252