ΠΑΝΕΠΙΣΤΗΜΙΟ ΜΑΚΕΔΟΝΙΑΣ

ΤΜΗΜΑ ΕΦΑΡΜΟΣΜΕΝΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

Όνομα: Δημήτρης

Επίθετο: Μανωλάκης

Α.Μ.: it1423

Ανάκτηση Πληροφορίας και Μηχανές Αναζήτησης

ΕΡΓΑΣΙΑ 5: Ανάπτυξη λογισμικού για Item-Item collaborative filtering

Περιεχόμενα

Εισαγωγή 3

Επεξήγηση Κώδικα 4

1. Γλώσσα Προγραμματισμού και Βιβλιοθήκες 4

2. Παραγωγή του τεχνητού συνόλου δεδομένων και τυχαία επιλογή των προς πρόβλεψη βαθμολογίων. 5

3. Παράμετροι 6

4. Adjusted Cosine & Cosine Similarity 7

Adjusted Cosine Similarity 7

Εικόνα 1: Symmetric Adjusted Cosine Matrix 8

Εικόνα 2: Max Cosine Matrix 9

Εικόνα 3: Max Cosine Pointers Matrix 9

Cosine Similarity 10

5. Jaccard & Dice similarity 10

Εικόνα 4: Jaccard & Dice Similarities Algorithm 11

6. Predictions & Mean Absolute Error Calculations 11

Πειράματα σχετικά με τα metrics & τις παραμέτρους. 13

1. Επιρροή του K (πλήθος κοντινότερων γειτόνων) στην απόδοση του συστήματος 13

Πίνακας 1: Τιμές του MAE για διαφορά K (Μέρος 1ο) 13

Πίνακας 2: Τιμές του MAE για διαφορά K (Μέρος 2ο) 14

Διάγραμμα 1: Adjusted Cosine Similarity, Weighted vs Normal για διάφορες τιμές του K 15

Διάγραμμα 2: Cosine Similarity, Weighted vs Normal για διάφορες τιμές του K 16

Διάγραμμα 3: Jaccard Similarity, Weighted vs Normal για διάφορες τιμές του K 17

Διάγραμμα 4: Dice Similarity, Weighted vs Normal για διάφορες τιμές του K 18

2. Επιρροή του X (ποσοστό των αγνώστων βαθμολογίων) στην απόδοση του συστήματος 19

Πίνακας 3: Τιμές του MAE για διαφορά X 20

Διάγραμμα 5: Adjusted Cosine Similarity, Weighted vs Normal για διάφορες τιμές του X 21

Διάγραμμα 6: Cosine Similarity, Weighted vs Normal για διάφορες τιμές του X 21

Διάγραμμα 7: Jaccard Similarity, Weighted vs Normal για διάφορες τιμές του X..... 22
Διάγραμμα 8: Dice Similarity, Weighted vs Normal για διάφορες τιμές του X..... 22
Συμπεράσματα..... 24
Επιπλεον Μεθοδος Προβλεψης (Bonus) 26
Πίνακας 4: Τιμές του MAE για διαφορά K (Adjusted Cosine) 28
Διάγραμμα 9: Adjusted Cosine, Average vs Custom vs Harmonic για διάφορες τιμές του K..... 29
Πίνακας 5: Τιμές του MAE για διαφορά K (Cosine)..... 30
Διάγραμμα 10: Cosine, Average vs Custom vs Harmonic για διάφορες τιμές του K..... 30
Πίνακας 6: Τιμές του MAE για διαφορά K (Jaccard) 31
Πίνακας 7: Τιμές του MAE για διαφορά K (Dice) 31
Διάγραμμα 11: Jaccard & Dice, Average vs Custom vs Harmonic για διάφορες τιμές του K..... 31

Εισαγωγή

Recommender (ή recommendation) system είναι ένας μηχανισμός ο οποίος προσπαθεί να προβλέψει την βαθμολογία που θα έδινε ένας χρήστης σε κάποιο αντικείμενο. Αποτελούν εδώ και χρονιά προϊόν μελέτης και ερευνάς ως προς την ανάπτυξη και την βελτίωση τους (Netflix Prize) και έχουν εφαρμοστεί με διάφορους τρόπους από ‘κολοσσούς’ της βιομηχανίας όπως Amazon και Netflix.

Στα πλαίσια της εργασίας αυτής, αναπτύσσεται λογισμικό το οποίο προσομοιώνει την λειτουργία των Recommendation systems και προσπαθεί βασιζόμενο σε τεχνητά δεδομένα, να προβλέψει την βαθμολογία που θα έδινε ένας χρήστης σε ένα αντικείμενο. Η τεχνική που χρησιμοποιείται ονομάζεται **Item-Item**

collaborative filtering, κατά την οποία οι προβλέψεις για ένα αντικείμενο βασίζονται σε βαθμολογίες του χρήστη σε αλλά ‘όμοια’ αντικείμενα.

Στόχος της εργασίας αυτής είναι να γίνει ανάλυση μέσω πολλαπλών πειραμάτων στα τεχνητά δεδομένα, της βέλτιστης τεχνικής υπολογισμού της ομοιότητας των αντικειμένων καθώς και τον εντοπισμό των ιδανικών σχετικών παραμέτρων όπως για παράδειγμα το πλήθος των γειτόνων που χρησιμοποιούνται. Είναι προφανές πως η τεχνητή φύση των δεδομένων αποτελεί εμπόδιο σε έμπρακτα συμπεράσματα που ανταποκρίνονται πλήρως στην πραγματικότητα, καθώς τα σχετικά μοτίβα και η συσχέτιση των ‘βαθμολογίων’ είναι τυχαία και προέρχονται από ομοιόμορφη κατανομή. Παρ’ όλ’ αυτά η υλοποίηση ενός τέτοιου λογισμικού είναι σίγουρο πως θα βοηθήσει στην βαθύτερη κατανόηση των μηχανισμών πίσω από ένα Recommendation system και πιθανώς κάποια από τα συμπεράσματα να αναπαράγονται και σε πειράματα με πραγματικά δεδομένα.

Επεξήγηση Κώδικα

1. Γλώσσα Προγραμματισμού και Βιβλιοθήκες

Η υλοποίηση του προγράμματος έγινε κάνοντας χρήση της **Python**. Πιο συγκεκριμένα έγινε χρήση της

βιβλιοθήκης **NumPy** για την αποθήκευση των δεδομένων σε πίνακες, την παράγωγη του τεχνητού συνόλου δεδομένων καθώς και για την τυχαία επιλογή των δεδομένων που το προγράμματα καλείται να προβλέψει. Παρόλο που η πλειοψηφία των μαθηματικών συναρτήσεων και διαδικασιών (π.χ. κανονικοποίηση πίνακα) που χρειάστηκαν υλοποιήθηκαν αυτούσιες, κάποιες πράξεις μεταξύ και επιώ πινάκων όπως dot product στηλών, ένωση πινάκων κατά γραμμή χρησιμοποιήθηκαν 'έτοιμες' μέσω της βιβλιοθήκης αυτής. Η βιβλιοθήκη **statistics** χρησιμοποιήθηκε για τον υπολογισμό αρμονικού μέσου (ως μια από τις εναλλακτικές μεθόδους πρόβλεψης).

Τέλος, υστέρα από την διεξαγωγή των πειραμάτων χρειάστηκε η δημιουργία διαγραμμάτων για την καλύτερη κατανόηση και ανάλυση των αποτελεσμάτων. Για την εργασία αυτή χρησιμοποιήθηκε η σχετική βιβλιοθήκη της Python, **Matplotlib**. Το περιβάλλον ανάπτυξης λογισμικού που χρησιμοποιήθηκε κατά την υλοποίηση του προγράμματος είναι το **Eclipse**.

2. Παραγωγή του τεχνητού συνόλου δεδομένων και τυχαία επιλογή των προς πρόβλεψη βαθμολογίων.

Βασική απαίτηση για το τεχνητό σύνολο δεδομένων ήταν να προέρχεται από **συνεχής ομοιόμορφη κατανομή**. Για τον λόγο αυτό, επιλέχθηκε το function random.uniform της βιβλιοθήκης NumPy. Αποτελεί μια

γεννήτρια πραγματικών τυχαίων αριθμών (δειγμάτων) από ομοιόμορφη κατανομή επιτρέποντας την επιλογή του επιθυμητού ορίου για το μέγεθος των αριθμών, το οποίο για τις ανάγκες της άσκησης επιλέχθηκε το [1,10]. Το τεχνητό αυτό σύνολο δεδομένων αποθηκεύεται σε πίνακα και παραμένει σταθερό κατά την διεξαγωγή των πειραμάτων.

Η συνάρτηση πυκνότητας πιθανότητας που χρησιμοποιείται από την παραπάνω γεννήτρια είναι η εξής:

$$p(x) = \frac{1}{b - a}$$

Σχετικά με την επιλογή των βαθμολογίων τις οποίες το πρόγραμμα καλείται να προβλέψει, έγινε χρήση του function random.choice της βιβλιοθήκης NumPy. Η συνάρτηση αυτή δέχεται ως εισαγωγή μια λίστα και παράγει μέσω και πάλι, ομοιόμορφης κατανομής, μια σειρά δειγμάτων προεπιλεγμένου μεγέθους (X). Η τυχαία αυτή επιλογή επαναλαμβάνεται σε κάθε πείραμα.

3. Παράμετροι

Οι σχετικές παράμετροι του προγράμματος έχουν οριστεί όπως δοθήκαν από τη εκφώνηση της εργασίας με μονή διαφορά το γεγονός ότι ως X ορίζεται το

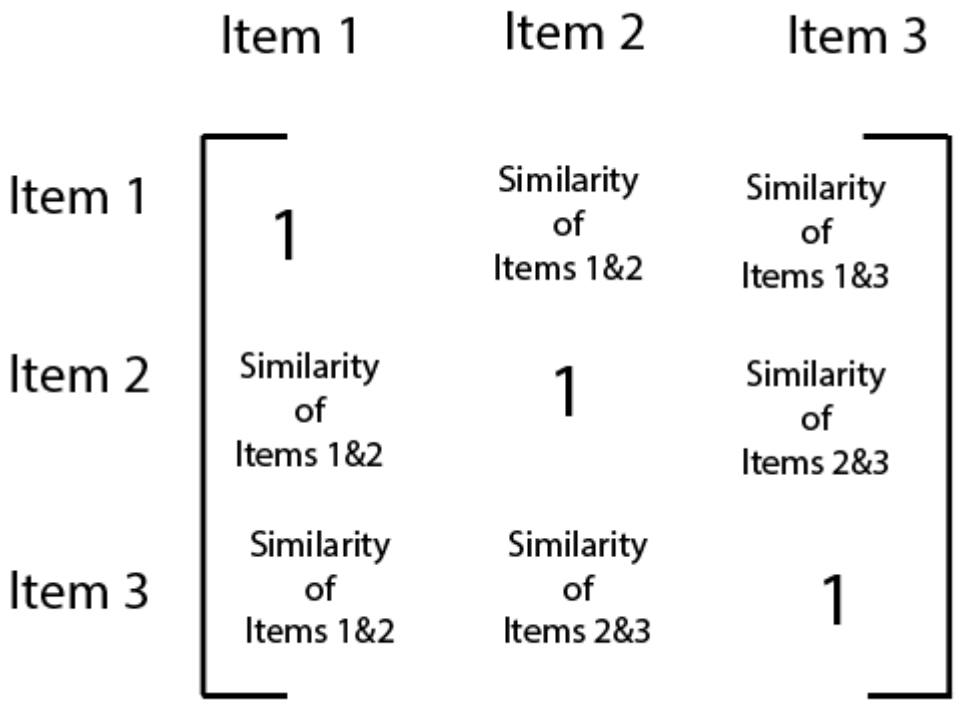
ποσοστό των βαθμολογίων που το προγράμματα καλείται να προβλέψει, ενώ 100-Χ το ποσοστό των γνωστών βαθμολογίων.

4. Adjusted Cosine & Cosine Similarity

Adjusted Cosine Similarity

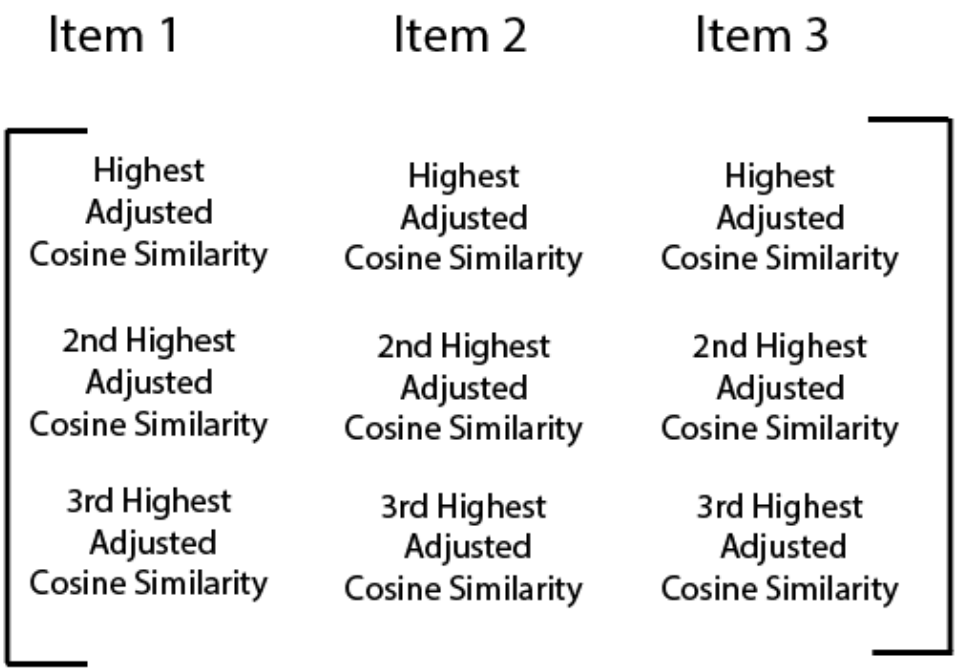
Αρχικά, υπολογίζεται και αποθηκεύεται σε μονοδιάστατο πίνακα Ν μεγέθους ο μέσος ορός της γνωστής βαθμολογίας κάθε χρήστη. Στην συνέχεια, από κάθε στοιχείο του Basic Matrix που περιέχει τις γνωστές βαθμολογίες του τεχνητού σύνολο δεδομένων αφαιρείται ο αντίστοιχος μέσος ορός της γραμμής (χρήστη) στην οποία ανήκει. Στο επόμενο βήμα, κάνοντας χρήση του μήκους (L2 Norm) κάθε στήλης δημιουργείται ο κανονικοποιημένος πίνακας Adjusted Normalized Matrix. Κάνοντας χρήση του πίνακα αυτού, δημιουργείται ο πίνακας Adjusted Cosine ο οποίος σε κάθε θέση περιέχει το dot product κάθε στήλης του προηγούμενου πίνακα. Η παρακάτω εικόνα ορίζει την μορφή του πίνακα αυτού για 3 Items:

Εικόνα 1: Symmetric Adjusted Cosine Matrix

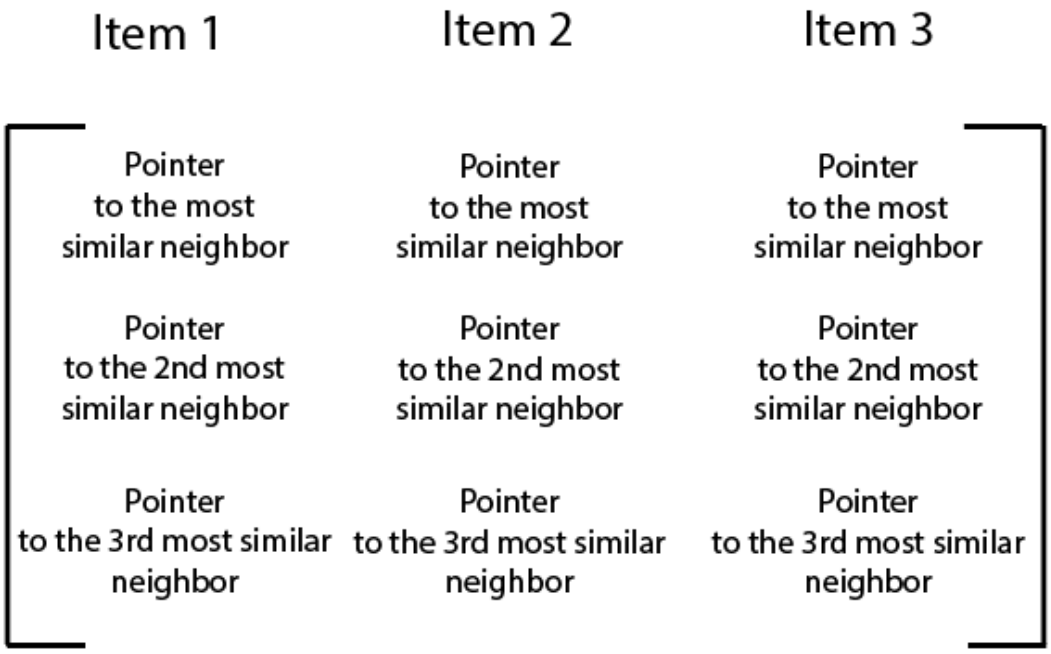


Κάνοντας χρήση του παραπάνω πίνακα υπολογίζονται δυο βασικοί πίνακες προκειμένου να εντοπίζονται σε κάθε πείραμα οι ‘γείτονες’ του κάθε αντικειμένου. Οι πίνακες αυτοί είναι ο Max Cosine Matrix και ο Max Cosine Pointers Matrix. Ο πρώτος πίνακας περιέχει σε κάθε στήλη τα Adjusted Cosine Similarities του αντίστοιχου αντικειμένου σε φθίνουσα διάταξη. Ο δεύτερος περιέχει έναν ακέραιο αριθμό που δείχνει στα όμοια items του αντίστοιχου item, σε φθίνουσα διάταξη. Οι παρακάτω εικόνες ορίζουν την μορφή κάθε πίνακα για ένα παράδειγμα με items:

Εικόνα 2: Max Cosine Matrix



Εικόνα 3: Max Cosine Pointers Matrix



Οι δυο αυτοί πίνακες περιέχουν ουσιαστικά την πληροφορία που χρειάζεται για την λειτουργία του προγράμματος. Ο πρώτος καθοδηγεί το πρόγραμμα

σχετικά με τους γείτονες διότι η επιλογή του K, εξυπηρετείται από τις K[1] πρώτες γραμμές του πίνακα. Επιπλέον, για την περίπτωση του σταθμισμένου μέσου ορού, ο πίνακας αυτός παρέχει τους Adjusted Cosine Coefficients. Ο δεύτερος πίνακας, μέσω των pointers, προσφέρει πρόσβαση στις βαθμολογίες του χρήστη στα όμοια items, οι οποίες είναι απαραίτητες για την πρόβλεψη της βαθμολογίας του υπό εξέταση item.

Cosine Similarity

Με παρόμοιο τρόπο υλοποιείται και ο υπολογισμός του Cosine Similarity για κάθε αντικείμενο. Η μονή διαφορά με το Adjusted Cosine Similarity είναι πως ο πίνακας των γνωστών τεχνητών δεδομένων δεν γίνεται adjusted κάνοντας χρήση του μέσου ορού της βαθμολογίας.

5. Jaccard & Dice similarity

Πρόκειται για δυο παρόμοια metrics οπού η εύρεση του ενός οδηγεί άμεσα στην εύρεση του αλλού. Ο αλγόριθμος υπολογίζει το Jaccard Similarity και υστέρα υπολογίζει το Dice Similarity μέσω του τύπου:

$$Dice = 2 * \frac{Jaccard}{1 + Jaccard}$$

[1] Πιθανώς να χρειαστούν περισσότερες από K γραμμές. Για παράδειγμα, στην περίπτωση οπού η βαθμολογία στου χρήστη για κάποιον από τους K γείτονες/αντικείμενα του χρήστη είναι 0 (άγνωστη).

Ο αλγόριθμος μαζί με τα σχετικά σχόλια παρουσιάζονται στην παρακάτω εικόνα:

Εικόνα 4: Jaccard & Dice Similarities Algorithm

```
#Jaccard & Dice
jaccard = np.zeros((N,M))
dice = np.zeros((N,M))

#Calculate Jaccard and Dice into symmetric Matrix.
for k in range(0,M):
    for i in range(0,M):
        intersection = 0
        union = 0
        for j in range(0,N):
            #If there is a rating for each of 2 items..
            if matrix[j][k] != 0 and matrix[j][i] != 0:
                #..increase both the union and the intersection.
                intersection = intersection + 1
                union = union + 1
            #If there is a rating only for one of the items..
            elif matrix[j][k] == 0 and matrix[j][i] != 0:
                #..increase only the union.
                union = union + 1
            #If there is a rating only for one of the items..
            elif matrix[j][k] != 0 and matrix[j][i] == 0:
                #..increase only the union.
                union = union + 1
            #If the last rating is reached calculate both the Jaccard & Dice similarities.
            if j == N-1:
                jaccard[k][i] = intersection/union
                dice[k][i] = jaccard[k][i]*2/(1+jaccard[k][i])
```

6. Predictions & Mean Absolute Error Calculations

Οι διαδικασίες αυτές γίνονται με τον ίδιο τρόπο και για τα 4 metrics. Η πρόβλεψη των βαθμολογίων γίνεται εντός τριπλού for-loop το οποίο για κάθε άγνωστη βαθμολογία που εντοπίζει εντός του τεχνητού συνόλου δεδομένων, εντοπίζει τους K ομοιότερους γείτονες του υπό εξέταση αντικειμένου. Ένα τέταρτο επαναληπτικό while loop αναλαμβάνει την περίπτωση οπού κάποια

από τα γειτονικά αντικείμενα έχουν και αυτά άγνωστη βαθμολογία και αναζητά τους αμέσως επομένους, πιο όμοιους γείτονες.

Σχετικά με την περίπτωση οπού για το αντικείμενο δεν υπάρχουν οι Κ βαθμολογημένοι γείτονες που απαιτούνται, τότε η βαθμολογία του αντικειμένου αυτού δεν υπολογίζεται (τίθεται -100). Ένας άλλος τρόπος θα ήταν σε τέτοιες περιπτώσεις, να χαλαρώνουμε τον περιορισμό σχετικά με τους Κ γείτονες και να επιτρέπεται η πρόβλεψη βαθμολογίας με λιγότερους γείτονες (Κ-1, Κ-2 κ.ο.κ.) ή να θέσουμε ένα όριο οπού επιτρέπεται για αντικείμενα που έχουν βαθμολογημένους γείτονες ίσους στο πλήθος με 2/3 του Κ να συμμετέχουν στην πρόβλεψη. Τέτοιου είδους τεχνικές πολύ πιθανώς να οδηγήσουν όμως σε καταστάσεις οπού αντικείμενα αγνώστου rating, βαθμολογούνται με βάση διαφορά πλήθη γειτόνων Κ το καθένα. Προκειμένου να διατηρηθεί η ακεραιότητα και η αξιοπιστία πειραμάτων σχετικά με την τιμή του Κ επιλέχθηκε η αυστηρή τακτική[2] οπού όλα τα αντικείμενα θα βαθμολογούνται ανάλογα με τους, ιδίου πλήθους, Κ γείτονες τους.

Οι άγνωστες βαθμολογίες αποθηκεύονται σε άλλο πίνακα (test matrix) στις ιδίες όμως ακριβώς θέσεις με αυτές που είχαν στο αρχικό σύνολο δεδομένων.

[2] Σε περίπτωση διεξαγωγής πειραμάτων με το πρόγραμμα ο χρήστης οφείλει να γνωρίζει ότι λόγω της τακτικής αυτής, όταν το σύστημα πλησιάζει τα ‘όρια’ του (π.χ. θελουμε να προβλέψουμε το 70% των δεδομένων με 40 γείτονες) τότε η πιθανότητα να έχουμε Runtime Error και συγκεκριμένα διαίρεση με το 0, είναι αρκετά υψηλή, καθώς το σύστημα δεν έχει την δυνατότητα να βρει το κατάλληλο πλήθος γειτόνων.

Παρόμοια οι βαθμολογίες που προβλέπονται αποθηκεύονται σε πίνακες (prediction matrix) στις κατάλληλες θέσεις, έτσι ώστε σωστά και ευκολά να υπολογίζεται το Mean Absolute Error.

Πειράματα σχετικά με τα metrics & τις παραμέτρους. [3]

1. Επιρροή του K (πλήθος κοντινότερων γειτόνων) στην απόδοση του συστήματος

Προκειμένου να εξεταστεί η επιρροή του K στην απόδοση του συστήματος διατηρούνται σταθερές όλες οι άλλες μεταβλητές του συστήματος, και με το 30% των βαθμολογίων του συνόλου δεδομένων να είναι άγνωστο, δοκιμάζονται διάφορες τιμές για τη παράμετρο K σε αύξουσα διάταξη ξεκινώντας από το 2 μέχρι το 70. Οι παρακάτω πίνακες περιέχουν αναλυτικά τις διάφορες τιμές του Mean Absolute Error που υπολογίστηκαν για κάθε ένα από τα Metrics και για διάφορες τιμές του K:

Πίνακας 1: Τιμές του MAE για διαφορά K (Μέρος 1ο)

	K=2	K=6	K=10	K=20	K=30

[3] Όλα τα πειράματα εκτελέστηκαν από 10 φορές, το σύνολο δεδομένων παραμένει σταθερό αλλά οι άγνωστες βαθμολογίες μεταβάλλονται, προκειμένου να εγγυάται η στατιστική σημαντικότητα αλλά και η ορθότητα των αποτελεσμάτων. Το σύνολο δεδομένων περιέχει 100 χρήστες και 100 αντικείμενα.

Adjusted Cosine Weighted	2.7473	2.5821	2.5045	2.455	2.4212
Adjusted Cosine	2.7446	2.5776	2.4994	2.4479	2.4111
Cosine Weighted	2.751	2.6154	2.5456	2.4941	2.4567
Cosine	2.7312	2.5761	2.4524	2.3937	2.4165
Jaccard Weighted	2.774	2.5912	2.5105	2.4588	2.4209
Jaccard	2.774	2.5913	2.5107	2.4588	2.4211
Dice Weighted	2.7738	2.5912	2.5105	2.4588	2.421
Dice	2.774	2.5913	2.5107	2.4588	2.4211

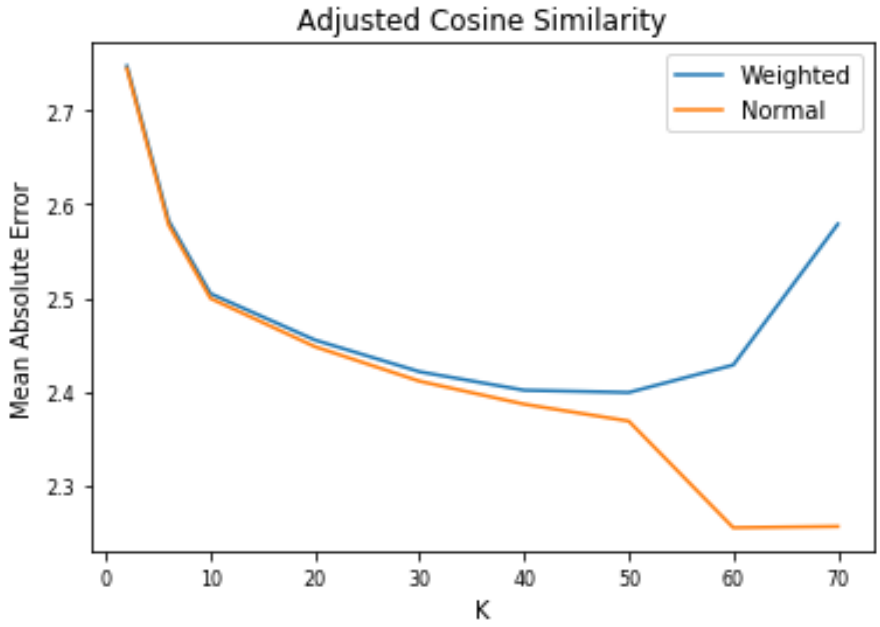
Πίνακας 2: Τιμές του MAE για διαφορά K (Μέρος 2ο)

	K=40	K=50	K=60	K=70
Adjusted Cosine Weighted	2.4015	2.3991	2.4284	2.5787
Adjusted Cosine	2.3866	2.3686	2.2549	2.2562
Cosine Weighted	2.4311	2.4127	2.2947	2.2981
Cosine	2.3924	2.3747	2.2598	2.258
Jaccard Weighted	2.3960	2.3779	2.2606	2.2613
Jaccard	2.3961	2.3778	2.2603	2.2611
Dice Weighted	2.3961	2.3778	2.2603	2.2611
Dice	2.3961	2.3778	2.2603	2.2611

Αν και οι πίνακες είναι περιεκτικοί σε πληροφορία, δεν βοηθούν στην οπτικοποίηση των αποτελεσμάτων και επομένως θα χρησιμοποιηθούν διαγράμματα αντίστοιχα για κάθε περίπτωση για καλύτερη ανάλυση.

Παράλληλα, για κάθε metric αναφέρονται και οι παράμετροι με τους οποίους επιτεύχθηκε η καλύτερη απόδοση.

Διάγραμμα 1: Adjusted Cosine Similarity, Weighted vs Normal για διάφορες τιμές του K

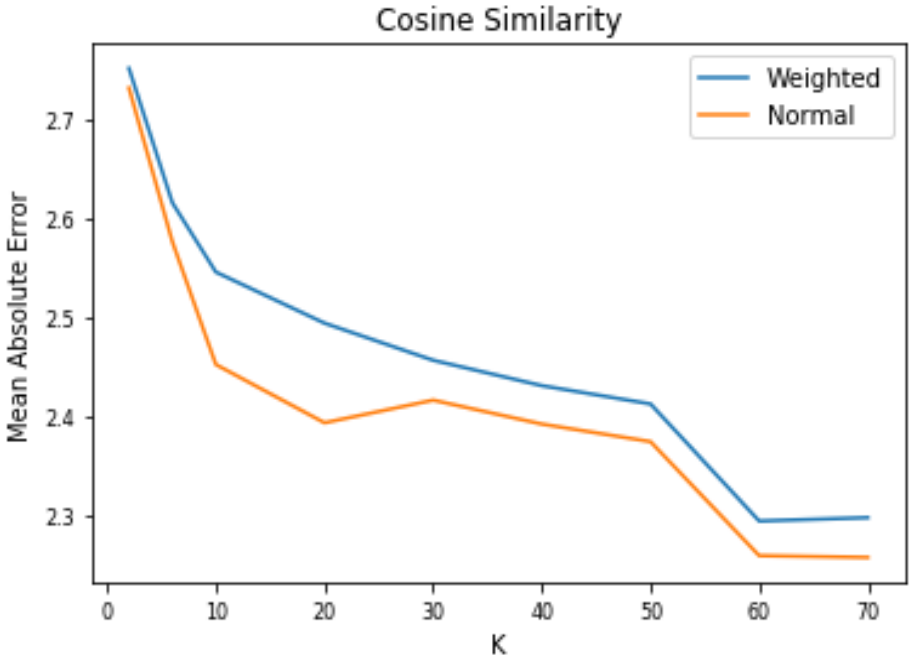


Minimum MAE:

Weighted Average → 2.3991 για K = 50

Normal Average → 2.2549 για K = 60

Διάγραμμα 2: Cosine Similarity, Weighted vs Normal για διάφορες τιμές του K

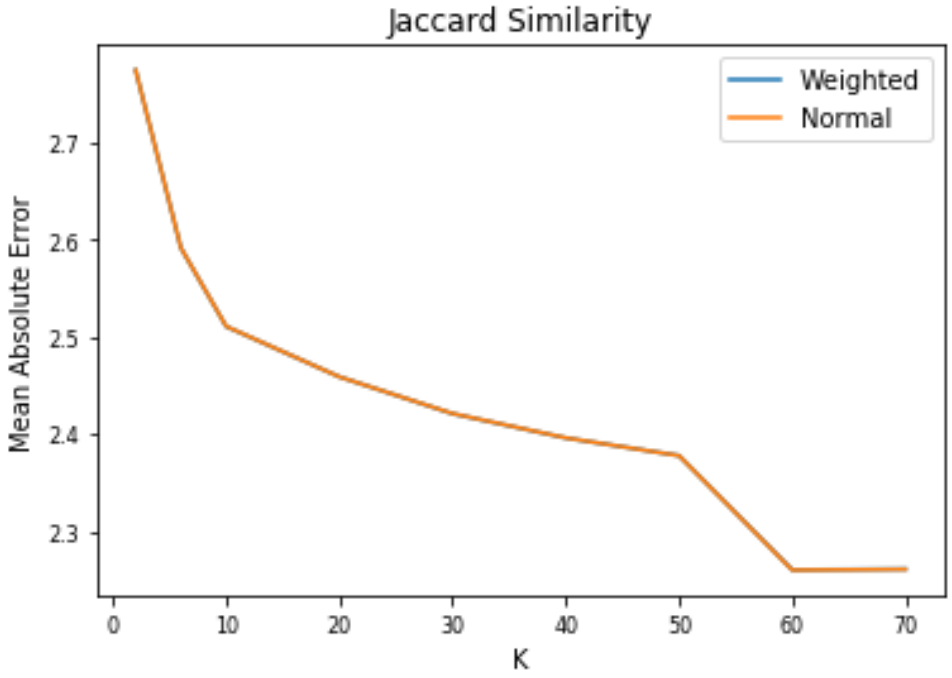


Minimum MAE:

Weighted Average → 2.2947 για K = 60

Normal Average → 2.258 για K = 70

Διάγραμμα 3: Jaccard Similarity, Weighted vs Normal για διάφορες τιμές του K

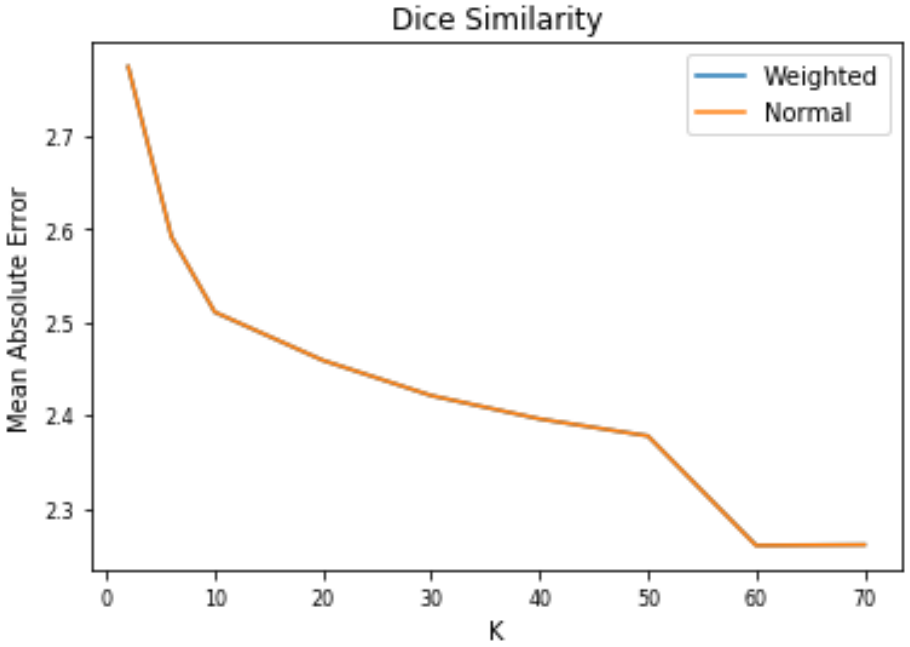


Minimum MAE:

Weighted Average → 2.2603 για K = 60

Normal Average → 2.2603 για K = 60

Διάγραμμα 4: Dice Similarity, Weighted vs Normal για διάφορες τιμές του K



Minimum MAE:

Weighted Average → 2.2603 για K = 60

Normal Average → 2.2603 για K = 60

Σχολιασμός Αποτελεσμάτων

Σχετικά με την παράμετρο K που ορίζει το πλήθος των κοντινότερων γειτόνων που εξετάζουμε προκειμένου να αποκτήσουμε μια πρόβλεψη για το εκάστοτε αντικείμενο, βλέπουμε ότι σε γενικούς βαθμούς η αύξηση των γειτόνων οδηγεί σε μείωση του Μέσου Απολυτού Λάθους. Ειδικότερα, όταν οι γείτονες αυξάνονται από 2 σε 10 η μείωση του λάθους είναι σε όλες τις περιπτώσεις αρκετά μεγάλη (από 2.75 σε 2.5). Η μείωση του λάθους αυτή συνεχίζεται μέχρι τους 60 με 70 γείτονες για τα περισσότερα metrics.

Εξαίρεση αποτελεί το Adjusted Cosine Similarity, καθώς στην περίπτωση του Weighted Average το λάθος από τους 50 γείτονες και μετά αρχίζει να αυξάνεται. Παρόμοια συμβαίνει και με τα Jaccard & Dice Similarities οπού μετά τις 60 γείτονες αρχίζουν να αυξάνονται. Το φαινόμενο αυτό πιθανώς να οφείλεται στο γεγονός ότι όσο οι απαιτούμενοι γείτονες αυξάνονται, τόσο πιο συχνά επιλέγονται λιγότερο όμοια αντικείμενα ως γείτονες λόγω βαθμολογίων που λείπουν (από τους πιο όμοιους γείτονες).

Επιπροσθέτως, παρατηρούμε ότι προβλέψεις με τη χρήση του Normal Average έχουν μικρότερο Μέσο Απολυτό Λάθος κυρίως στην περίπτωση του Adjusted/Normal Cosine ενώ για τα Jaccard/Dice η απόδοση είναι σχεδόν πανομοιότυπη. Τέλος, την καλύτερη απόδοση δείχνει ως τώρα να την έχει το Adjusted Cosine με χρήση του απλού μέσου ορού για προβλέψεις αν και οι διάφορες στις αποδόσεις αναμεσά στα metrics είναι ελάχιστα.

2. Επιρροή του X (ποσοστό των αγνώστων βαθμολογίων) στην απόδοση του συστήματος

Προκειμένου να εξεταστεί η επιρροή του X στην απόδοση του συστήματος διατηρούνται σταθερές όλες οι άλλες μεταβλητές του συστήματος, και με το πλήθος των απαιτούμενων κοντινότερων γειτόνων ίσο με 20, δοκιμάζονται διάφορες τιμές για τη παράμετρο X σε

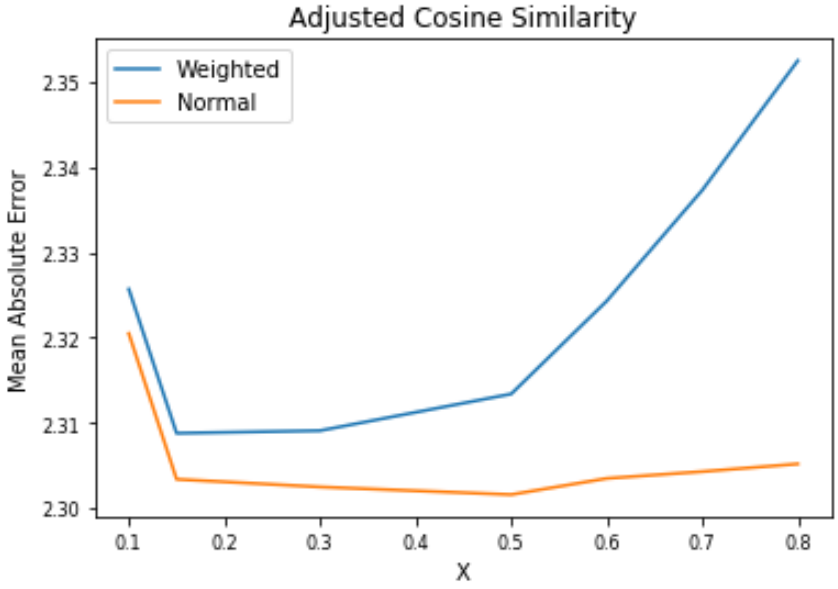
αύξουσα διάταξη ξεκινώντας από το 0.1 μέχρι το 0.8. Οι παρακάτω πίνακες περιέχουν αναλυτικά τις διάφορες τιμές του Mean Absolute Error που υπολογίστηκαν για κάθε ένα από τα Metrics και για διάφορες τιμές του X:

Πίνακας 3: Τιμές του MAE για διαφορά X

	X=0.1	X=0.15	X=0.3	X=0.5	X=0.6	X=0.7	X=0.8
Adjusted Cosine Weighted	2.3257	2.3088	2.3091	2.3134	2.3243	2.3373	2.3525
Adjusted Cosine	2.3205	2.3034	2.3025	2.3016	2.3035	2.3043	2.3052
Cosine Weighted	2.356	2.3432	2.3410	2.3375	2.3369	2.3357	2.3367
Cosine	2.31	2.2977	2.2997	2.3017	2.3032	2.304	2.3058
Jaccard Weighted	2.3164	2.2998	2.3	2.3	2.301	2.302	2.3037
Jaccard	2.3163	2.2997	2.3	2.3	2.301	2.302	2.3023
Dice Weighted	2.3164	2.2998	2.3	2.3	2.301	2.302	2.3056
Dice	2.3163	2.2997	2.3	2.3	2.301	2.302	2.3044

Και πάλι, για την καλύτερη ανάλυση των αποτελεσμάτων παρουσιάζονται τα αντίστοιχα διαγράμματα για κάθε metric μαζί με τις παραμέτρους που πέτυχαν την καλύτερη απόδοση:

Διάγραμμα 5: Adjusted Cosine Similarity, Weighted vs Normal για διάφορες τιμές του X

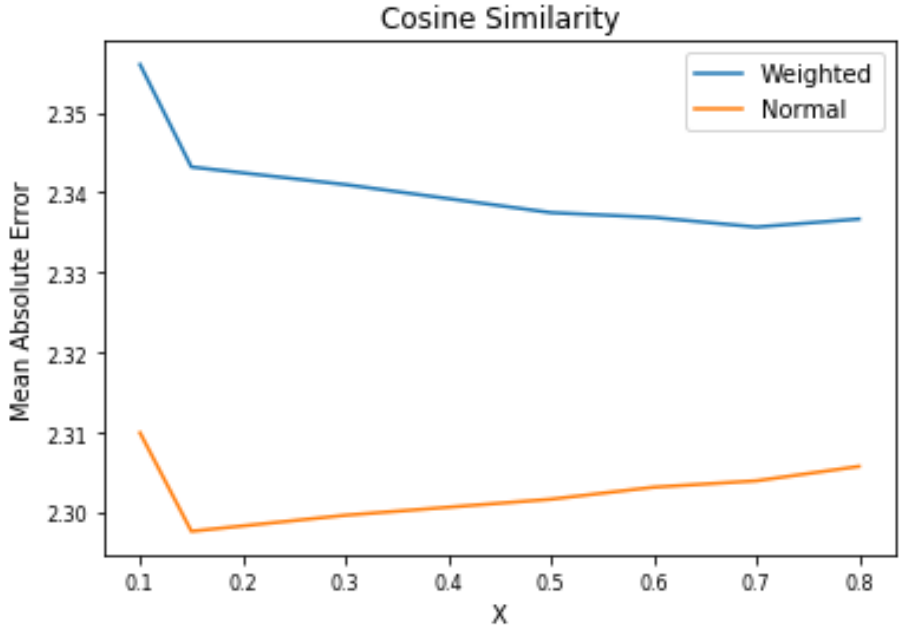


Minimum MAE:

Weighted Average → 2.3088 για X = 0.15

Normal Average → 2.3016 για X = 0.5

Διάγραμμα 6: Cosine Similarity, Weighted vs Normal για διάφορες τιμές του X

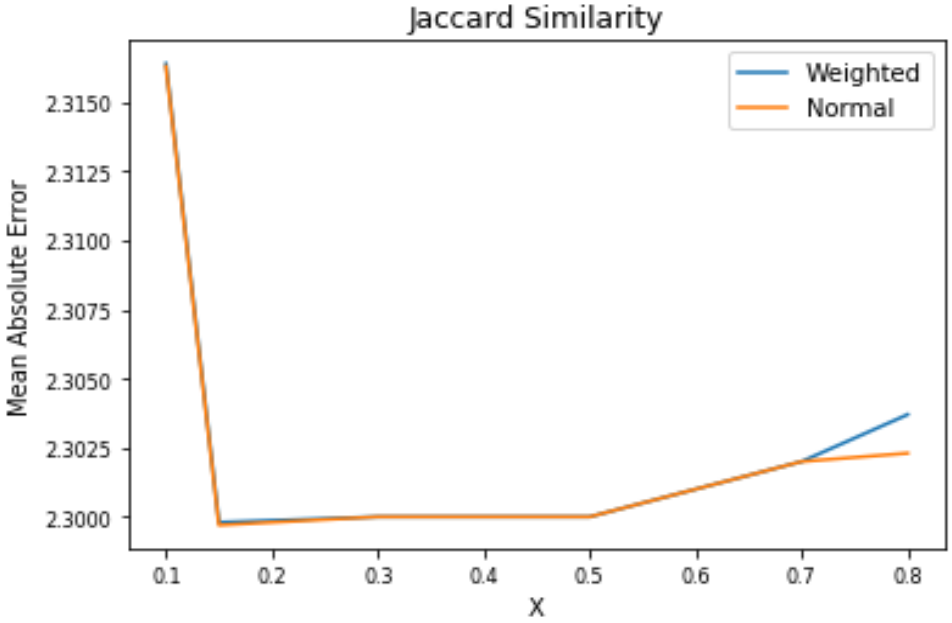


Minimum MAE:

Weighted Average → 2.3357 για X = 0.7

Normal Average → 2.2977 για X = 0.15

Διάγραμμα 7: Jaccard Similarity, Weighted vs Normal για διάφορες τιμές του X

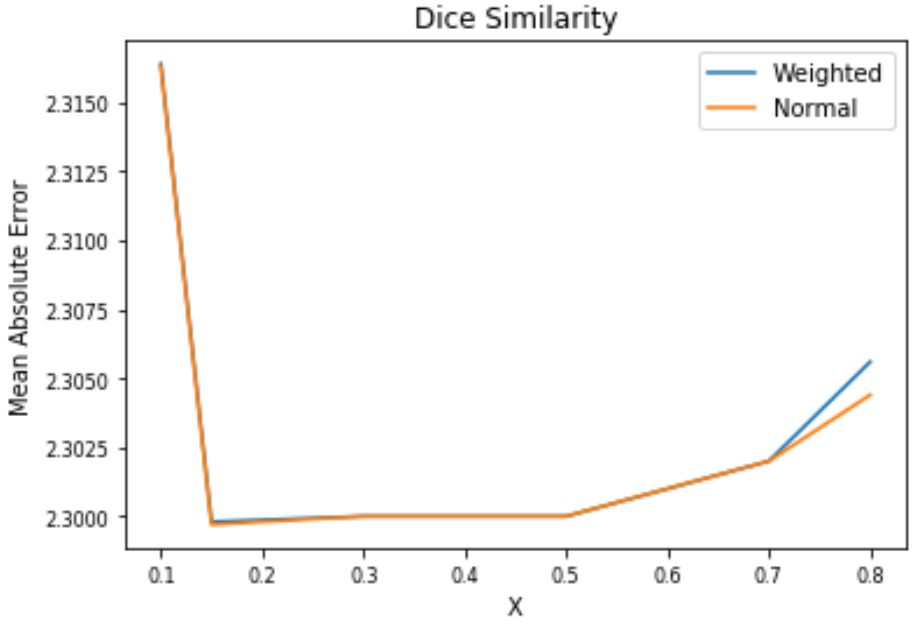


Minimum MAE:

Weighted Average → 2.2998 για X = 0.15

Normal Average → 2.997 για X = 0.15

Διάγραμμα 8: Dice Similarity, Weighted vs Normal για διάφορες τιμές του X



Minimum MAE:

Weighted Average → 2.2998 για X = 0.15

Normal Average → 2.997 για X = 0.15

Σχολιασμός Αποτελεσμάτων

Είναι φανερό από τον πίνακα αλλά και τα διαγράμματα ότι στις ακραίες τιμές του X, είτε πολύ μεγάλες είτε πολύ μικρές, η πιθανότητα το Μέσο Απολυτό Λάθος να είναι μεγάλο είναι αρκετά μεγαλύτερη. Για τα περισσότερα metrics, άσχετα από το είδος του μέσου ορού που χρησιμοποιείται για την πρόβλεψη, ένα ποσοστό αγνώστων βαθμολογίων ίσο με περίπου 15% των συνολικών δεδομένων είναι το ιδανικό για την μέγιστη απόδοση. Βλέπουμε επίσης ότι στις περισσότερες περιπτώσεις, όταν πάνω από το 50% των βαθμολογίων είναι άγνωστο, αρχίζει μια σταδιακή μείωση στην απόδοση του συστήματος. Το φαινόμενο αυτό είναι αναμενόμενο καθώς όσο περισσότερες βαθμολογίες είναι άγνωστες, αυξάνεται η πιθανότητα το σύστημα να απευθυνθεί σε λιγότερο όμοιους γείτονες προκειμένου να υπολογίσει τις σχετικές προβλέψεις.

Επιπλέον, παρατηρούμε και πάλι την ανωτερότητα του απλού μέσου ορού έναντι του σταθμισμένου σχετικά με Μέσο Απολυτό Λάθος. Η διαφορά αυτή είναι ιδιαίτερα έντονη στην περίπτωση του Adjusted και του απλού

Cosine. Είναι επίσης ξεκάθαρη στα διαγράμματα η ομοιότητα των Jaccard και Dice Similarities σχετικά με την απόδοση και το γεγονός ότι για τα δυο αυτά metrics είτε χρησιμοποιείται ο απλός μέσος ορός είτε ο σταθμισμένος η απόδοση είναι παρόμοια.

Τέλος, παρατηρώντας τις καλύτερες αποδόσεις για κάθε metric βλέπουμε ότι το Cosine Similarity πετυχαίνει την καλύτερη απόδοση σε συνδυασμό με τον απλό μέσο ορό σε σχέση με τα υπόλοιπα metrics. Η διαφορά βέβαια είναι μικρή (ειδικά με Jaccard και Dice) αλλά και σε αυτό το είδος πειραμάτων το Cosine αναδεικνύεται να έχει την μέγιστη απόδοση για το σύστημα.

Συμπεράσματα

Το βασικότερο συμπέρασμα από τα παραπάνω πειράματα είναι πως όλα τα Metrics έχουν παρόμοια απόδοση χωρίς να υπάρχει κάποιο το οποίο ξεκάθαρα υπερτερεί. Όπως είδαμε αν και με μικρή διαφορά, οι προβλέψεις που γίνονται έχοντας ως βάση το Adjusted Cosine Similarity έχουν την καλύτερη απόδοση. Σε αρκετά πειράματα, το Cosine Similarity ήταν το metric μέσω του οποίου το σύστημα έφτανε το πιο χαμηλό επίπεδο MAE για το σχετικό πείραμα, αλλά σε γενικό βαθμό το Adjusted Cosine Similarity είχε ανώτερη απόδοση σταθερά στις περισσότερες παραμετροποιήσεις του συστήματος. Επιπλέον η

διαφορά αναμεσά σε σταθμισμένο και απλό μέσο ορό είναι σχετικά πιο έντονη, από την αντίστοιχη διαφορά στα Metrics, με το δεύτερο να υπερτερεί σε όλες σχεδόν περιπτώσεις του πρώτου.

Τα συμπεράσματα σχετικά με το K, το πλήθος δηλαδή των κοντινότερων γειτόνων που απαιτούνται είναι πως βασική προϋπόθεση για την μέγιστη απόδοση του συστήματος είναι η επιλογή της μέγιστης επιτρεπόμενης τιμής για το K. Το X είναι αυτό που ορίζει ποιες τιμές του K επιτρέπονται ή όχι. Για παράδειγμα, παραπάνω στα σχετικά με το K πειράματα καταλήξαμε ότι η ιδανική τιμή θα είναι γύρω στους 70 γείτονες. Αυτό συνέβη καθώς σε όλα τα πειράματα αυτά είχαμε θέσει το ποσοστό των κρυφών τιμών να είναι ίσο με 30%. Επειδή έχουμε 100 χρήστες και 100 γείτονες, οι 70 κοντινότεροι γείτονες αποτελούν ουσιαστικά το 70% του συνόλου των δεδομένων (σχεδόν όλες δηλαδή τις γνωστές βαθμολογίες). Επομένως αν ξεπεράσουμε το όριο αυτό, το σύστημα θα αρχίσει να απευθύνεται σε όλες και λιγότερο όμοιους κοντινότερους γείτονες, ελαττώνοντας κατά αυτό το τρόπο την απόδοση των προβλέψεων. Επομένως με βάση το X θα πρέπει να κρίνουμε την μέγιστη δυνατή τιμή για το K.

Τα συμπεράσματα σχετικά με το X απαιτούν λιγότερη ανάλυση καθώς όπως είναι αναμενόμενο όσο πιο πολλές βαθμολογίες κρύβουμε από το σύστημα τόσο δυσχεραίνει η απόδοση του. Καταλήξαμε όμως ότι ακόμη και σε συνθήκες οπού το σύστημα αγνοεί μέχρι

και το 60-70% του συνόλου των δεδομένων έχει μια σταθερή απόδοση σχετικά με το Μέσο Απολυτό Λάθος προσφέροντας συνάμα μεγάλη ποσότητα πληροφορίας υπό την μορφή προβλέψεων.

Τέλος, όσο αναφορά την περίπτωση οπού πρέπει να επιλέξουμε μια συγκεκριμένη παραμετροποίηση, συνάρτηση ομοιότητας και μέθοδο πρόβλεψης, είναι προφανές από τα αποτελέσματα ότι το Adjusted Cosine Similarity σε συνδυασμό με απλό μέσο ορό έχει την καλύτερη απόδοση. Συμπεράνουμε επιπλέον ότι για τεχνητά σύνολα δεδομένων όπως αυτό που εξετάστηκε από την παρούσα εργασία, αν στόχος μας είναι να αποκτήσουμε την μέγιστη δυνατή πληροφορία (τις περισσότερες προβλέψεις) με όσο το δυνατόν καλύτερη απόδοση, τότε θα ήταν λογικό να επιλέξουμε να κρατήσουμε το μισό ουσιαστικά σύνολο δεδομένων (Χ=50%).Επιπλέον, θέτουμε τους απαιτούμενους κοντινότερους γείτονες να έχουν πλήθος στο πεδίο τιμών [40,50] προκειμένου το σύστημα να εκμεταλλευτεί τη μέγιστη δυνατή πληροφορία που έχει στην διάθεση του, χωρίς να χρειάζεται να απευθύνεται σε λιγότερο όμοιους κοντινότερους γείτονες για το εκάστοτε αντικείμενο που εξετάζει.

Επιπλέον Μέθοδος Πρόβλεψης (Bonus)

Κάνοντας χρήση του απλού μέσου ορού ουσιαστικά εξετάζουμε την βαθμολογία κάθε γειτονικού

αντικειμένου. Ως μια εναλλακτική λύση, μπορούμε πρώτα να εξετάζουμε τον μέσο ορό της βαθμολογίας που έχει δώσει ο εν λόγω χρήστης στους γείτονες του υπό εξέταση αντικειμένου. Αν ο μέσος ορός αυτός είναι από 5.5 και κάτω τότε το αντικείμενο θεωρείται πως υπάρχει μεγάλη πιθανότητα να μην αρέσει στο χρήστη και υπολογίζεται η βαθμολογία που θα έδινε μέσω του μέσου ορού μόνο των μικρότερων ή ίσων με το 5,5 βαθμολογίων. Παρόμοια αντιμετωπίζονται και περιπτώσεις οπού ο μέσος ορός της βαθμολογίας των γειτονικών αντικειμένων είναι μεγαλύτερος του 5,5.

Η τεχνική αυτή προσπαθεί να λειτουργήσει ως ένας δυαδικός κατηγοριοποιητής ο οποίος μέσω του μέσου ορού που υπολογίζει για τα γειτονικά αντικείμενα, αποφασίζει αν θα αρέσει ή όχι στον χρήστη τον υπό εξέταση αντικείμενο. Αξιολογεί, επιπλέον, την ύπαρξη προτίμησης ή την απουσία της λαμβάνοντας υπόψη μόνο τις θετικές ή τις αρνητικές, αντίστοιχα, βαθμολογίες.

Τέλος, αντί της μέτρησης του απλού μέσου ορού των βαθμολογίων που εντοπίστηκαν ως συναφής σύμφωνα με το παραπάνω κριτήριο, εξετάστηκε και η μέθοδος πρόβλεψης που υπολογίζει τον αρμονικό μέσο των βαθμολογίων αυτών. Το metric αυτό παίζει τον ρολό ενός πιο ‘απαισιόδοξου’ ή ‘ρεαλιστικού’ μέσου ορού καθώς συνήθως τείνει να βρίσκεται πιο κοντά στις μικρότερες τιμές.

Γενικότερα οι δυο νέες αυτές μέθοδοι πρόβλεψης δεν αναμένεται να έχουν ανώτερη απόδοση καθώς εισάγουν bias, ευνοούν κατά κάποιο τρόπο τις πιο ακραίες τιμές στις βαθμολογίες των γειτονικών αντικειμένων. Σε ένα τεχνητό σύνολο δεδομένων, το οποίο προέρχεται από ομοιόμορφη κατανομή, είναι φυσικό και επόμενο αυτό το bias να οδηγεί σε χειροτέρα αποτελέσματα από ένα metric όπως ο απλός μέσος ορός ο οποίος υπολογίζεται 'ομοιόμορφα', κάνοντας χρήση όλων των βαθμολογίων.

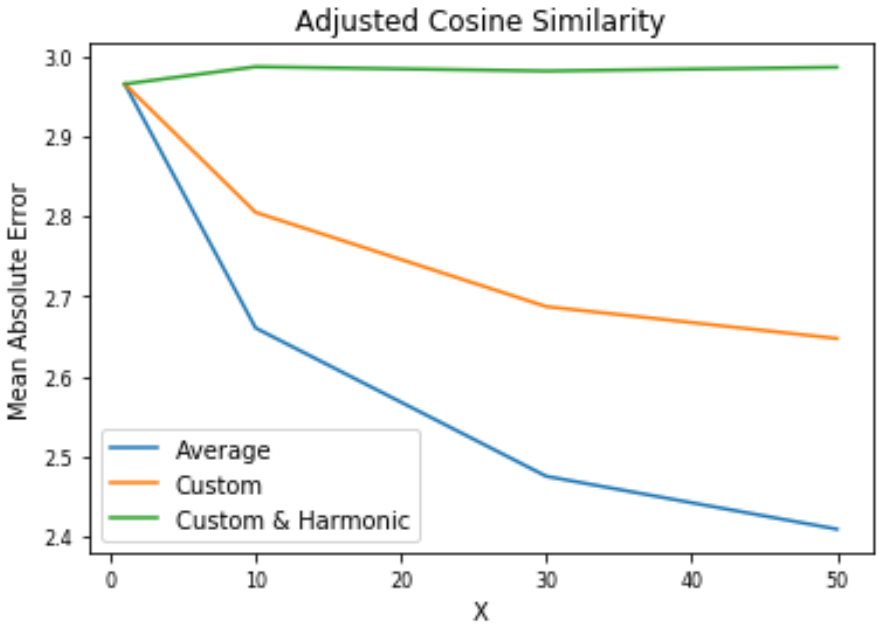
Οι παρακάτω πίνακες περιέχουν τα αποτελέσματα από πειράματα που διεξάχθηκαν για κάθε συνάρτηση ομοιότητας, κρατώντας το X σταθερό και ίσο με 0.6 και μεταβάλλοντας κάθε φορά το πλήθος των γειτόνων K. Επιπλέον, για κάθε πίνακα έχει δημιουργηθεί το αντίστοιχο διάγραμμα.

Πίνακας 4: Τιμές του MAE για διαφορά K (Adjusted Cosine)[4]

	K=1	**K=10**	**K=30**	**K=50**
Adjusted Cosine (Average)	2.9649	2.6604	2.4753	2.4093
Adjusted Cosine (Custom)	2.9649	2.8047	2.6871	2.6473
Adjusted Cosine (Custom & Harmonic)	2.9649	2.9865	2.9812	2.9859

[4] Οπού Average, ο απλός μέσος ορός όπως προηγουμένως, Custom η νέα μέθοδος πρόβλεψης που περιγράφεται παραπάνω και Custom & Harmonic η περίπτωση οπού αντί για μέσου ορού στις επιλεγμένες βαθμολογίες, υπολογίζεται ο αρμονικός μέσος.

Διάγραμμα 9: Adjusted Cosine, Average vs Custom vs Harmonic για διάφορες τιμές του K



Minimum MAE:

Average → 2.4093 για K=50

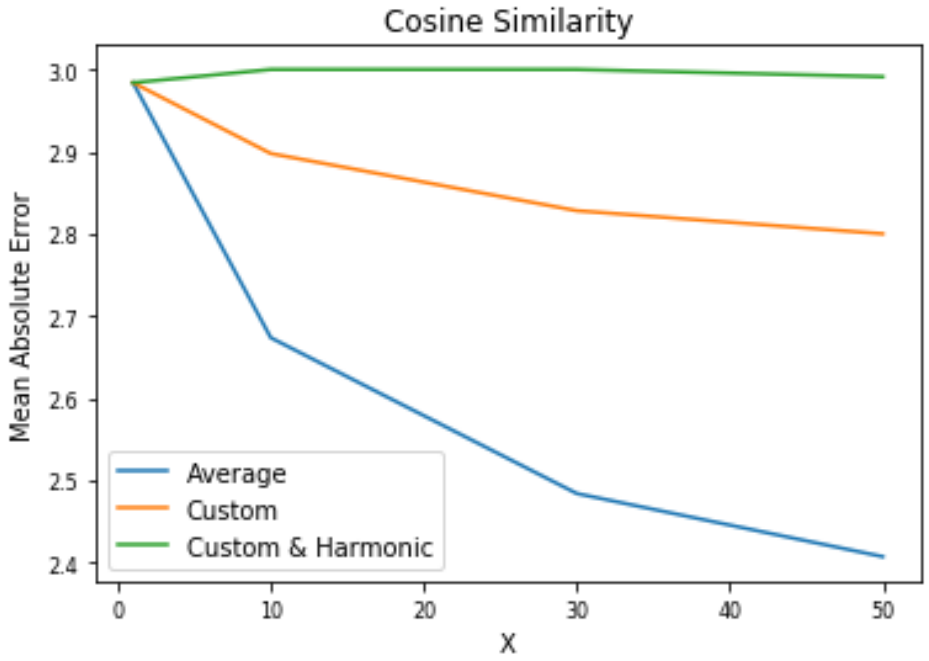
Custom → 2.6473 για K=50

Custom & Harmonic → 2.9649 για K=1

Πίνακας 5: Τιμές του MAE για διαφορά K (Cosine)

	K=1	K=10	K=30	K=50
Cosine (Average)	2.9839	2.6735	2.4839	2.4071
Cosine (Custom)	2.9839	2.8977	2.8281	2.8
Cosine (Custom & Harmonic)	2.9839	3	3	2.9912

Διάγραμμα 10: Cosine, Average vs Custom vs Harmonic για διάφορες τιμές του K



Minimum MAE:

Average → 2.4071 για K=50

Custom → 2.8 για K=50

Custom & Harmonic → 2.9839 για K=1

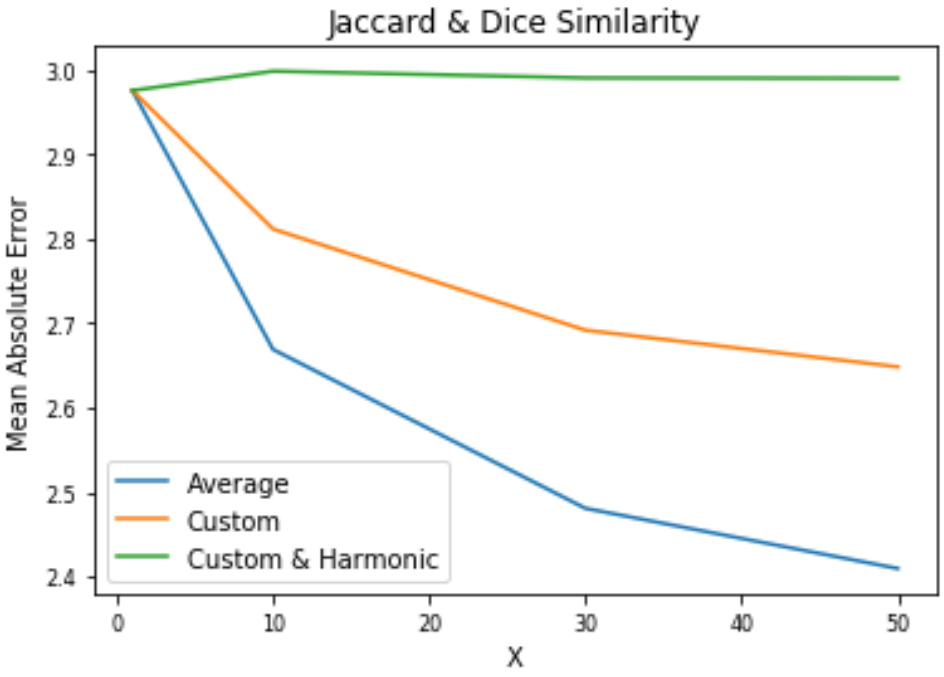
Πίνακας 6: Τιμές του MAE για διαφορά K (Jaccard)

	K=1	K=10	K=30	K=50
Jaccard (Average)	2.9759	2.6691	2.4808	2.4095
Jaccard (Custom)	2.9759	2.8118	2.6919	2.6485
Jaccard (Custom & Harmonic)	2.9759	2.9988	2.9908	2.9904

Πίνακας 7: Τιμές του MAE για διαφορά K (Dice)

	K=1	K=10	K=30	K=50
Dice (Average)	2.9759	2.6691	2.4808	2.4095
Dice (Custom)	2.9759	2.8118	2.6919	2.6485
Dice (Custom & Harmonic)	2.9759	2.9988	2.9908	2.9904

Διάγραμμα 11: Jaccard & Dice, Average vs Custom vs Harmonic για διάφορες τιμές του K



Minimum MAE:

Average → 2.4095 για K=50

Custom → 2.6485 για K=50

Custom & Harmonic → 2.9759 για K=1

Τα αποτελέσματα στα οποία καταλήγουμε είναι πως σε κάθε περίπτωση, οποία συνάρτηση ομοιότητας και αν χρησιμοποιείται η νέα μέθοδος πρόβλεψης πάντα αποδίδει ελαφρώς χειροτέρα από τον απλό μέσο ορό. Μια τέτοια απόδοση είναι όπως εξηγήθηκε παραπάνω αναμενομένη. Βλέπουμε επίσης ότι η απόδοση της νέας μεθόδου κυμαίνεται όμοια με τον απλό μέσο ορό όσο το K μεταβάλλεται.

Τέλος, παρατηρείται ότι η περίπτωση του αρμονικού μέσου ορού έχει μια σταθερή και χειρότερη απόδοση από όλες τις μεθόδους καθώς ίσως εισάγει υπερβολικά πολύ bias στις προβλέψεις. Ωστόσο, θα ήταν ενδιαφέρον να εξεταστεί η απόδοση των νέων αυτών ‘μεροληπτικών’ μεθόδων σε πραγματικά δεδομένα.