3. Αλγόριθμοι Φιλτραρίσματος βάσει Περιεχομένου

***Σύνοψη**: Στο κεφάλαιο αυτό θα μελετήσουμε τους **αλγορίθμους φιλτραρίσματος βάσει περιεχομένου (Content-based filtering)**. Θα περιγράψουμε το **μοντέλο διανυσματικού χώρου (vector space model)** καθώς και την τεχνική **Term Frequency Inverse Document Frequency (TF-IDF)** που σταθμίζει τη σημαντικότητα ενός χαρακτηριστικού στο προφίλ ενός χρήστη. Επίσης, θα γίνει αναφορά στα συστήματα συστάσεων με τη χρήση **δέντρων απόφασης (decision trees)** και του **κατηγοριοποιητή Bayes**, ο οποίος αποτελεί ένα πιθανοκρατικό μοντέλο πρόβλεψης. Στη συνέχεια, θα δοθεί μια αναλυτική περιγραφή των βασικών μεθόδων (**Gini index, Entropy,** και χ^2**-statistic**) κατάταξης των χαρακτηριστικών του προφίλ ενός χρήστη βάσει της σημαντικότητάς τους για τον αλγόριθμο **δέντρου απόφασης**.*

***Προαπαιτούμενη γνώση**: Ο αναγνώστης καλό θα ήταν να γνωρίζει βασικές τεχνικές **Εξόρυξης Κειμένου**.*

3.1 Εισαγωγή

Οι τεχνικές *Συνεργατικού Φιλτραρίσματος (ΣΦ)* που αναπτύξαμε στο προηγούμενο κεφάλαιο και, ειδικότερα, οι *αλγόριθμοι βάσει κοντινότερου γείτονα* αποτελούν επιτυχημένες τεχνικές στα συστήματα συστάσεων. Οι *αλγόριθμοι ΣΦ* για την παροχή συστάσεων βασίζονται σε παρελθούσες αξιολογήσεις χρηστών με παρόμοιες προτιμήσεις και αξιολογήσεις προϊόντων. Ωστόσο, οι *αλγόριθμοι ΣΦ* έχουν συγκεκριμένα μειονεκτήματα. Για παράδειγμα, εάν ένα νέο στοιχείο εμφανίζεται στη βάση δεδομένων για πρώτη φορά, τότε δεν υπάρχει τρόπος να συσταθεί εφόσον προηγουμένως δεν έχει ήδη αξιολογηθεί από κάποιον χρήστη. Ένα δεύτερο μειονέκτημα αφορά στην περίπτωση που οι προτιμήσεις ενός χρήστη είναι ασυνήθιστες, γεγονός που δεν επιτρέπει να εντοπιστούν εύκολα γειτονικοί χρήστες με συμπεριφορά παρόμοια μ' αυτήν του υπό εξέταση χρήστη, με αποτέλεσμα να του προτείνονται ανακριβείς συστάσεις στοιχείων.

Αντιθέτως, οι *αλγόριθμοι Φιλτραρίσματος Βάσει Περιεχομένου* (*ΦΒΠ*) υποθέτουν ότι κάθε χρήστης ενεργεί ανεξάρτητα. Συνακόλουθα, ο *αλγόριθμος ΦΒΠ* αξιοποιεί πληροφορίες που αφορούν μόνο σε χαρακτηριστικά προϊόντων (π.χ. τιμή, κατηγορία προϊόντος κ.λπ.) και προσπαθεί να φιλτράρει τα προϊόντα που θα προταθούν στον υπό εξέταση χρήστη βάσει των κοινών χαρακτηριστικών που έχουν με τα προϊόντα με τα οποία έχει αλληλεπιδράσει αυτός στο παρελθόν. Από την άλλη μεριά όμως, ένα αμιγές *σύστημα συστάσεων ΦΒΠ* παρουσιάζει το πρόβλημα της *υπερ-εξειδίκευσης*, όταν δηλαδή στον χρήστη προτείνονται μόνο στοιχεία παρόμοια με αυτά που ήδη αξιολόγησε/βαθμολόγησε στο παρελθόν. Συνεπώς, η προτεινόμενη λίστα στοιχείων προς τον υπό εξέταση χρήστη δεν θα έχει ποικιλομορφία και ούτε θα προσφέρει κάτι καινούργιο ως προς τις προηγούμενες εμπειρίες του χρήστη, δεδομένου ότι τα προϊόντα που θα εμπεριέχει θα έχουν παρόμοια χαρακτηριστικά με αυτά με τα οποία θα έχει ήδη αλληλεπιδράσει ο χρήστης στο παρελθόν.

Αναλυτικότερα λοιπόν, ο *αλγόριθμος ΦΒΠ* αρχικά κατασκευάζει ένα προφίλ χρήστη που συγκροτείται από τα χαρακτηριστικά που έχουν τα προϊόντα με τα οποία έχει αλληλεπιδράσει ο χρήστης στο παρελθόν. Σ' ένα σύστημα συστάσεων ταινιών λόγου χάρη, ένας χρήστης προτιμά μια ταινία για διάφορους λόγους, ενδεικτικά, για τους ηθοποιούς, τον σκηνοθέτη ή το είδος της ταινίας. Όλα αυτά τα χαρακτηριστικά επιδρούν διαφορετικά στην επιλογή του κάθε χρήστη και αποτελούν μέρος του προφίλ του. Τονίζεται ότι με τον όρο **"προφίλ χρήστη"** ορίζεται η καταγραφή οποιασδήποτε αλληλεπίδρασης του χρήστη με το σύστημα (αγορά προϊόντος, αξιολόγηση προϊόντος κ.λπ.).

3.2 Σχετική Βιβλιογραφία

Ο *αλγόριθμος Φιλτραρίσματος Βάσει Περιεχομένου* μελετήθηκε αρχικά από την επιστημονική κοινότητα της *Ανάκτησης Πληροφορίας*. Προϋποθέτει λοιπόν ότι κάθε χρήστης λειτουργεί ανεξάρτητα και αξιοποιεί πληροφορίες που αφορούν μόνο στα χαρακτηριστικά των προϊόντων (π.χ. τιμή, χρώμα κ.λπ.). Υπάρχουν δύο βασικά προβλήματα στον σχεδιασμό ενός *συστήματος φιλτραρίσματος βάσει περιεχομένου*: (i) η κατασκευή του *προφίλ ενός χρήστη*, και (ii) *η αναπαράσταση* ενός εγγράφου ή προϊόντος. Για παράδειγμα, το 1994 οι ερευνητές Yan και Molina [Yan and Molina, 1995] εφάρμοσαν ένα *απλό σύστημα φιλτραρίσματος περιεχομένου* για ειδησεογραφικά άρθρα του Διαδικτύου.

Κατά το παρελθόν έγιναν διάφορες προσπάθειες συνδυασμού του *αλγορίθμου ΦΒΠ* και του *αλγορίθμου ΣΦ*. Το *σύστημα Fab* [Balabanovic and Y., 1997], λόγου χάρη, υπολογίζει την *ομοιότητα μεταξύ χρηστών βάσει ΣΦ* αφού πρώτα προβεί σε κατασκευή του *προφίλ* κάθε χρήστη βάσει *ΦΒΠ*. Αυτή η διεργασία λειτουργεί αντίστροφα στο *σύστημα CinemaScreen* [Salter and Antonopoulos, 2006], το οποίο τρέχει *ΦΒΠ* στα αποτελέσματα του *ΣΦ*. Ο Melville και οι συνεργάτες του [Melville et al., 2002], από την άλλη, υλοποίησαν ένα *μοντέλο πρόβλεψης βάσει φιλτραρίσματος περιεχομένου* για την ενίσχυση των υπαρχόντων δεδομένων χρήστη, και, κατόπιν, για την παροχή εξατομικευμένων προτάσεων βασίστηκαν στο *ΣΦ*. Τέλος, ο Jin και οι συνεργάτες του [Jin et al., 2005] πρότειναν

ένα σύστημα συστάσεων όπου ενσωματώθηκαν στοιχεία συνεργατικά και περιεχομένου υπό την αρχή της απομείωσης της *εντροπίας/αβεβαιότητας* πρόβλεψης.

Όλες οι προαναφερθείσες προσεγγίσεις είναι *υβριδικές*: τρέχουν είτε τον *αλγόριθμο ΣΦ* στα αποτελέσματα του *ΦΒΠ* είτε αντίστροφα. Το *ΣΦ* εξετάζει την εξάρτηση μεταξύ των αξιολογήσεων χρηστών που αφορούν σε προϊόντα, αλλά του διαφεύγει η πιθανή εξάρτηση μεταξύ των χαρακτηριστικών (τιμή, βάρος κ.λπ.) των προιόντων. Το *ΦΒΠ* εξετάζει το τελευταίο, αλλά όχι το πρώτο.

3.3 Το Μοντέλο Διανυσματικού Χώρου (Vector Space Model)

Το *μοντέλο διανυσματικού χώρου* εφαρμόστηκε αρχικά στην *Ανάκτηση Πληροφορίας* προκειμένου να αναπαριστώνται τα έγγραφα και η συχνότητα εμφάνισης των λέξεων που αυτά εμπεριέχουν, και στη συνέχεια να υπολογίζονται οι ομοιότητες μεταξύ των εγγράφων και του ερωτήματος του υπό εξέταση χρήστη. Το μοντέλο διανυσματικού χώρου βασίζεται σ' έναν δισδιάστατο πίνακα εγγράφων και λέξεων όπου οι γραμμές είναι τα *έγγραφα* και οι στήλες είναι οι *λέξεις*. Ο πίνακας εγγράφων-λέξεων διατηρεί στα κελιά του τον αριθμό εμφάνισης κάθε λέξης για κάθε έγγραφο ξεχωριστά. Η *συχνότητα του όρου i* (*term frequency - tf*) σ' ένα έγγραφο d ορίζεται ως $tf_{i,d}$ και κρατά το πλήθος των εμφανίσεων του όρου i στο έγγραφο d. Εδώ θα πρέπει να τονίσουμε ότι η συχνότητα εμφάνισης μιας λέξης σ' ένα έγγραφο δεν καθορίζει απαραιτήτως το υπό εξέταση έγγραφο, δεδομένου ότι αυτή η λέξη μπορεί απλά να συναντάται συχνά σε πολλά έγγραφα, και γι' αυτό είναι στατιστικά λιγότερο σημαντική στο να εκφράσει ένα έγγραφο. Επομένως, τέτοιες λέξεις συχνά θα πρέπει να υποστούν μια απομείωση της σημαντικότητάς τους ως προς το κατά πόσο μπορούν να εκφράσουν ένα έγγραφο. Η διαδικασία αυτή είναι γνωστή ως *στάθμιση όρων* (*term weighting*). Ειδικότερα, η *στάθμιση* μιας λέξης ή όρου επιτυγχάνεται χρησιμοποιώντας την έννοια της *αντίστροφης συχνότητας εγγράφου* (*inverse document frequency - idf*), όπου το αντίστροφο idf_i της *συχνότητας εγγράφου* του i-στού όρου μειώνει τη σημασία του όρου i βάσει του αριθμού των εγγράφων n_i στα οποία εμφανίζεται ο i ως ακολούθως:

$$idf_i = \log(n/n_i)$$

όπου n ο συνολικός αριθμός των εγγράφων της συλλογής.

Συνδυάζοντας τη *συχνότητα όρου (tf)* και την *αντίστροφη συχνότητα εγγράφου (idf)* -τα οποία ορίσαμε προηγουμένως- μπορούμε να υπολογίσουμε τη *σταθμισμένη συχνότητα* του i-στού όρου ως ακολούθως:

$$\text{normalized } tf_i = tf_i * idf_i$$

Αυτό το μοντέλο *στάθμισης όρων* αναφέρεται ευρέως ως *μοντέλο tf-idf*, όπου το tf αντιπροσωπεύει τη *συχνότητα όρου* και το idf την *αντίστροφη συχνότητα εγγράφου*.

3.4 Συστάσεις βάσει Δεδομένων Περιεχομένου

Στην ενότητα αυτή παρουσιάζουμε τα απαραίτητα βήματα προσαρμογής του *μοντέλου στάθμισης tf-idf* -που αναπτύχθηκε προηγουμένως- από το επιστημονικό πεδίο της *ανάκτησης πληροφορίας* σε αυτό των συστημάτων συστάσεων. Συγκεκριμένα, περιγράφουμε μ' ένα παράδειγμα ένα *υβριδικό* μοντέλο που επιτρέπει τον συνδυασμό *συνεργατικών δεδομένων* με *δεδομένα περιεχομένου*. Ακολουθεί το περίγραμμα αυτής της προσαρμογής σε τέσσερα βήματα:

1. **Κατασκευή προφίλ χρήστη**: Δημιουργείται ένα *προφίλ χρήστη* (i) από τις βαθμολογίες των χρηστών για τα στοιχεία και (ii) από τα χαρακτηριστικά των στοιχείων για τα οποία αυτοί έχουν δείξει ενδιαφέρον.

2. **Στάθμιση χαρακτηριστικού**: Ποσοτικοποιείται η *σημαντικότητα* του κάθε χαρακτηριστικού στο επίπεδο του *προφίλ χρήστη* και σ' αυτό των χρηστών της συλλογής (εύρεση, δηλαδή, των σημαντικών μόνο χαρακτηριστικών για τις δύο παραπάνω ομάδες).

3. **Δημιουργία γειτονιάς του υπό εξέταση χρήστη**: Για την παροχή συστάσεων δημιουργούμε τη *γειτονιά του υπό εξέταση χρήστη* υπολογίζοντας την *ομοιότητά* του με κάθε έναν από τους άλλους χρήστες της συλλογής.

4. **Δημιουργία λίστας προτεινόμενων στοιχείων**: Παρέχουμε σε κάθε χρήστη μια *λίστα προτεινόμενων στοιχείων* βάσει της συχνότητας εμφάνισης των χαρακτηριστικών των υποψήφιων προς σύσταση στοιχείων.

Στη συνέχεια θα αναλύσουμε λεπτομερώς κάθε βήμα με την χρήση ενός παραδείγματος.

Παράδειγμα 3.1 Να προτείνετε στον χρήστη U_2 στοιχεία, βάσει του πίνακα βαθμολογιών χρηστών στοιχείων του Σχήματος 3.1(α), όπου I_1 - I_6 τα στοιχεία και U_1 - U_4 οι χρήστες. Τα μηδενικά (χωρίς βαθμολογία δηλαδή) κελιά εμφανίζονται με παύλα. Επιπλέον, στο Σχήμα 3.1(β) για κάθε στοιχείο διαθέτουμε τέσσερα χαρακτηριστικά, F_1 - F_4. Τα χαρακτηριστικά ενός στοιχείου θα μπορούσαν να είναι η τιμή ενός προϊόντος, το βάρος του κ.λπ.

3.4.1 Βήμα Κατασκευής Προφίλ Χρήστη

Αρχικά, κατασκευάζουμε *το προφίλ του υπό εξέταση χρήστη* ώστε να λαμβάνει υπόψη του τις βαθμολογίες και τα χαρακτηριστικά των στοιχείων που αυτός βαθμολόγησε. Ειδικότερα, για έναν χρήστη u που αξιολόγησε θετικά (επάνω από ένα *κατώφλι* P_τ) μερικά στοιχεία κατασκευάζουμε ένα προφίλ χαρακτηριστικών για να εντοπίσουμε τα αγαπημένα εξ αυτών.

	I_1	I_2	I_3	I_4	I_5	I_6
U_1	-	4	-	-	5	-
U_2	-	3	-	4	-	-
U_3	-	-	-	-	-	4
U_4	5	-	3	-	-	-

(α)

	F_1	F_2	F_3	F_4
I_1	0	1	0	0
I_2	1	1	0	0
I_3	0	1	1	0
I_4	0	1	0	0
I_5	1	1	1	0
I_6	0	0	0	1

(β)

	F_1	F_2	F_3	F_4
U_1	2	2	1	0
U_2	1	2	0	0
U_3	0	0	0	1
U_4	0	2	1	0

(γ)

Σχήμα 3.1: Τρέχον παράδειγμα: (α) Πίνακας *Χρήστη-Στοιχείου* R, (β) Πίνακας *Στοιχείου-Χαρακτηριστικού* F, και (γ) Πίνακας *Χρήστη-Χαρακτηριστικού* P.

Έστω ότι ο πίνακας $R(u,i)$ κρατά τη βαθμολογία του χρήστη u για κάθε στοιχείο i. Χρησιμοποιούμε έναν δυαδικό πίνακα F όπου το στοιχείο $F(i,f)$ ισούται με 1 εάν το στοιχείο i περιέχει το χαρακτηριστικό f, ενώ διαφορετικά ισούται με 0. Στο τρέχον παράδειγμα οι πίνακες R και F παρουσιάζονται στα Σχήματα 3.1(α) και 3.1(β) αντίστοιχα.

Ορισμός 3.1 (Προφίλ χρήστη) Ως *προφίλ αξιολόγησης* $R(U_k)$ του χρήστη U_k ορίζουμε το σύνολο των βαθμολογημένων (δηλαδή μη μηδενικών) στοιχείων στην k-οστή γραμμή του πίνακα R. Για παράδειγμα, $R(U_1) = \{I_2, I_5\}$.

Ορισμός 3.2 (Προφίλ στοιχείου) Ως *προφίλ στοιχείου* $F(I_k)$ του στοιχείου I_k ορίζουμε το σύνολο των χαρακτηριστικών f για τα οποία το $F(I_k, f)$ έχει θετική (δηλαδή μεγαλύτερη του μηδενός) τιμή. Για παράδειγμα, $F(I_2) = \{F_1, F_2\}$.

Ορισμός 3.3 (Προφίλ χαρακτηριστικών χρήστη) Ως *προφίλ χαρακτηριστικών* $P(U_k)$ του χρήστη U_k ορίζουμε το σύνολο των $P(u,f)$ στοιχείων στην k-οστή γραμμή του πίνακα P με τιμή μεγαλύτερη του μηδενός. Για παράδειγμα, $P(U_4) = \{(F_2, 2), (F_3, 1)\}$.

Για έναν χρήστη u λοιπόν, το *προφίλ* του ορίζεται από τον πίνακα $P(u,f)$ με τα ανάλογα στοιχεία ως εξής:

$$P(u,f) = \sum_{\forall R(u,i) > P_\tau} F(i,f) \tag{3.1}$$

Συνεπώς το $P(u,f)$ δηλώνει τη συσχέτιση μεταξύ χρήστη u και χαρακτηριστικού f. Σημειώνεται ότι χρησιμοποιούμε μόνο τα θετικώς αξιολογηθέντα από τον χρήστη u στοιχεία i (π.χ. $R(u,i) > P_\tau$) .

Στο συγκεκριμένο παράδειγμα (με P_τ=2), λοιπόν, κατασκευάζουμε τον πίνακα P συνδυάζοντας πληροφορίες από τους πίνακες R και F. Στο Σχήμα 3.1(γ) παρατηρούμε ότι ο νέος πίνα-

κας P αποκαλύπτει μια ισχυρή *ομοιότητα* (όμοιες προτιμήσεις χαρακτηριστικών) μεταξύ των χρηστών U_1 και U_4. Αυτή η *ομοιότητα* δεν θα μπορούσε να προκύψει από τις αντίστοιχες βαθμολογίες χρήστη του πίνακα R.

3.4.2 Βήμα Σταθμισμένου Χαρακτηριστικού

Έστω ότι $\mathcal{U}$ το σύνολο όλων των χρηστών και $\mathcal{F}_u$ το σύνολο των χαρακτηριστικών που συσχετίζονται με τον χρήστη u, π.χ. $\mathcal{F}_u = \{f \in \mathcal{F} | P(u, f) > 0\}$. Εφεξής, ο χρήστης u και το χαρακτηριστικό f συσχετίζονται όταν $P(u, f) > 0$.
Θα *σταθμίσουμε*, λοιπόν, τα χαρακτηριστικά του πίνακα P ώστε να προσδιορίσουμε εκείνα που: (i) περιγράφουν και χαρακτηρίζουν καλύτερα τον χρήστη u, και (ii) τον διαφοροποιούν με ευκρινή τρόπο από άλλους χρήστες.
Στο μοντέλο μας, με βάση τη *στάθμιση TF-IDF* -γνωστή από το επιστημονικό πεδίο της *Ανάκτησης Πληροφορίας* [Baeza-Yates and Ribeiro-Neto, 1999]- η ύπαρξη χαρακτηριστικών εντός του προφίλ χρήστη ποσοτικοποιείται υπολογίζοντας τη συχνότητα κάθε χαρακτηριστικού f για έναν χρήστη u. Στο εξής, αυτός ο παράγοντας θα αναφέρεται ως παράγοντας *Συχνότητας Χαρακτηριστικού* (FF). Επιπρόσθετα, η στάθμιση ενός χαρακτηριστικού f ποσοτικοποιείται υπολογίζοντας την αντιστροφή συχνότητας του f μεταξύ όλων των χρηστών. Εν συνεχεία, αυτός ο παράγοντας θα αναφέρεται ως παράγοντας *Αντίστροφης Συχνότητας Χρήστη* (IUF). Ο λόγος χρήσης του παράγοντα IUF είναι ότι τα χαρακτηριστικά που εμφανίζονται στα προφίλ πολλών χρηστών δεν μπορούν να αντιπροσωπεύσουν επαρκώς το προφίλ τους.
Επομένως, *Συχνότητα Χαρακτηριστικού* $FF(u, f)$ είναι η πυκνότητα εμφάνισης του χαρακτηριστικού f στο προφίλ του χρήστη u. Στην περίπτωσή μας ισχύει ότι $FF(u, f) = P(u, f)$. Η *Συχνότητα Χρήστη* $UF(f)$ είναι ο αριθμός των χρηστών στους οποίους το χαρακτηριστικό f εμφανίζεται τουλάχιστον μία φορά. Τέλος, η *Αντίστροφη Συχνότητα Χρήστη* $IUF(f)$ είναι δυνατόν να υπολογισθεί από το $UF(f)$ ως εξής:

$$IUF(f) = \log \frac{|\mathcal{U}|}{UF(f)} \tag{3.2}$$

όπου $|\mathcal{U}|$ ο συνολικός αριθμός των χρηστών. Η *Αντίστροφη Συχνότητα Χρήστη* ενός χαρακτηριστικού f είναι χαμηλή εάν αυτό εμφανίζεται σε προφίλ πολλών χρηστών, ενώ είναι υψηλή εάν το f εμφανίζεται σε λίγα προφίλ. Τέλος, η *σταθμισμένη τιμή* του χαρακτηριστικού f για τον χρήστη u υπολογίζεται ως ακολούθως:

$$W(u, f) = FF(u, f) \times IUF(f) \tag{3.3}$$

Η παραπάνω εξίσωση συνδυάζει τη *συχνότητα χαρακτηριστικού FF(u, f)* και την *αντίστροφη συχνότητα χρήστη IUF(f).* Με τον όρο *FF(u, f)* λαμβάνουμε υπόψη το γεγονός ότι ένα χαρακτηριστικό f είναι σημαντικό για τον χαρακτηρισμό του προφίλ του χρήστη u εφόσον εμφανίζεται συχνά σ' αυτό. Από την άλλη, με τη μεταβλητή *IUF(f)* απομειώνουμε τη *σημαντικότητα*

του χαρακτηριστικού όταν αυτό εμφανίζεται σε πολλά προφίλ χρηστών, δεδομένου ότι δεν θα μπορεί να χαρακτηρίσει με ικανοποιητικό τρόπο το προφίλ του υπό εξέταση χρήστη.
Στο τρέχον παράδειγμα ο πίνακας P του Σχήματος 3.1(γ) μετατρέπεται στον πίνακα W του Σχήματος 3.2(α). Παρατηρούμε λοιπόν ότι στον πίνακα P τα χαρακτηριστικά F_1 και F_2 για τον χρήστη U_1 έχουν την ίδια τιμή, δηλαδή ίση με δύο (2). Αντίθετα, στον πίνακα W τα ίδια χαρακτηριστικά σταθμίζονται διαφορετικά (0.60 και 0.24). Είναι προφανές, άρα, ότι το χαρακτηριστικό F_1 περιγράφει πιο αποτελεσματικά τον χρήστη U_1 από ό,τι το F_2, κάτι που δεν θα ήταν εύκολο να εντοπιστεί στον πίνακα P του Σχήματος 3.1.

	F_1	F_2	F_3	F_4
U_1	0.60	0.24	0.30	-
U_2	0.30	0.24	0	-
U_3	-	-	-	0.60
U_4	0	0.24	0.30	-

(α)

	U_1	U_2	U_3	U_4
U_1	-	0.96	0	1
U_2	0.96	-	0	1
U_3	0	0	-	0
U_4	1	1	0	-

(β)

Σχήμα 3.2: (α) *Σταθμισμένος* πίνακας *χρήστη-χαρακτηριστικού* W, (β) Πίνακας *ομοιότητας χρήστη-χρήστη*.

3.4.3 Βήμα Σχηματισμού Γειτονιάς Χρήστη

Για την παροχή συστάσεων στον υπό εξέταση χρήστη πρέπει να βρούμε χρήστες παρόμοιους με αυτόν προκειμένου να αντλήσουμε υποψήφια προς σύσταση στοιχεία από τη γειτονιά τους. Στη συνέχεια, εφαρμόζουμε *Ομοιότητα Συνημιτόνου* στον *σταθμισμένο πίνακα χρήστη-χαρακτηριστικού* W σύμφωνα με την παρακάτω Εξίσωση 3.4:

$$\mathrm{sim}(u,v) = \frac{\sum_{\forall F \in X} W(u,f)W(v,f)}{\sqrt{\sum_{\forall f \in X} W(u,f)^2}\sqrt{\sum_{\forall f \in X} W(v,f)^2}}, \quad X = \mathcal{F}_u \cap \mathcal{F}_v \tag{3.4}$$

Παρατηρήστε ότι εφαρμόζουμε την *Ομοιότητα Συνημίτονου* για εκείνα μόνο τα χαρακτηριστικά που υπάρχουν ταυτόχρονα στα προφίλ και των δύο υπό εξέταση χρηστών, u και v. Στο τρέχον παράδειγμα δημιουργούμε έναν πίνακα *ομοιότητας χρήστη-χρήστη* σύμφωνα με την Εξίσωση 3.4 όπου μπορούμε να ανακαλύψουμε τους *κοντινότερους γείτονες κάθε υπό εξέταση χρήστη* (οι *κοντινότεροι γείτονες* έχουν την υψηλότερη τιμή). Στο Σχήμα 3.2(β) παρατηρούμε ότι ο *κοντινότερος γείτονας* του χρήστη U_2 είναι ο U_4, με *ομοιότητα* 1, ενώ ο U_1 ακολουθεί με 0.96.

3.4.4 Βήμα Δημιουργίας Λίστας Προτεινόμενων Στοιχείων

Μια ευρέως χρησιμοποιούμενη τεχνική για τη δημιουργία της λίστας προτεινόμενων στοιχείων είναι αυτή που υπολογίζει τη *συχνότητα* κάθε θετικώς αξιολογημένου στοιχείου εντός της *γειτονιάς των κοντινότερων χρηστών τού υπό εξέταση χρήστη*, και προτείνει τα κορυφαία N στοιχεία με την υψηλότερη συχνότητα εμφάνισης. Ακολούθως, θα προσαρμόσουμε την παραπάνω διαδικασία προκειμένου να λάβουμε υπόψη και τη συχνότητα εμφάνισης των χαρακτηριστικών των υποψήφιων προς σύσταση στοιχείων. Ειδικότερα, για κάθε χαρακτηριστικό f εντός της γειτονιάς των *παρόμοιων χρηστών του υπό εξέταση χρήστη* μετράμε τη συχνότητα εμφάνισης του πρώτου. Στη συνέχεια, βασιζόμενοι στα χαρακτηριστικά ενός στοιχείου υπολογίζουμε τη βαρύτητά του στη *γειτονιά των παρόμοιων χρηστών του υπό εξέταση χρήστη*. Κατά αυτόν τον τρόπο λαμβάνουμε υπόψη το γεγονός ότι κάθε χρήστης έχει τους δικούς του λόγους (διαφορετικά χαρακτηριστικά) για να αξιολογήσει θετικά ένα στοιχείο/προϊόν. Για να παράξουμε επομένως μια λίστα συστάσεων που εμπεριέχει μόνο το κορυφαίο προτεινόμενο στοιχείο στον U_2 (με k=2 κοντινότερους γείτονες), εργαζόμαστε ως ακολούθως:

1. Λαμβάνουμε τους *εγγύτερους γείτονες* του $U_2 : \{U_4, U_1\}$.
2. Λαμβάνουμε τα *στοιχεία στη γειτονιά*: $\{I_1, I_3, I_5\}$.
3. Λαμβάνουμε τα *χαρακτηριστικά* κάθε στοιχείου: $I_1 : \{F_2\}, I_3 : \{F_2, F_3\}, I_5 : \{F_1, F_2, F_3\}$.
4. Βρίσκουμε τη *συχνότητά* τους στη γειτονιά: $fr(F_1) = 1, fr(F_2) = 3, fr(F_3) = 2$.
5. Για κάθε στοιχείο προσθέτουμε τη *συχνότητα* των χαρακτηριστικών του βρίσκοντας τη *βαρύτητά* του στη γειτονιά: $w(I_1) = 3, w(I_3) = 5, w(I_5) = 6$.

Έτσι, προτείνεται το I_5 επειδή αποτελείται από χαρακτηριστικά που εμφανίζονται συχνότερα στα προφίλ των γειτόνων του U_2.

3.5 Κατηγοριοποιητής Δέντρων Απόφασης

Στην ενότητα αυτή θα περιγράψουμε τις κατάλληλες προσαρμογές που πρέπει να γίνουν στον *κατηγοριοποιητή δέντρων απόφασης* προκειμένου να μπορούμε να αποφανθούμε για το εάν θα συσταθεί ένα προϊόν σ' έναν υπό εξέταση χρήστη ή όχι.

Ένα *Δέντρο Απόφασης (Decision Tree)* είναι μια συλλογή *κόμβων* και *φύλλων* ενός *γράφου*, όπου τα *φύλλα* προσδιορίζουν αποφάσεις που μπορούμε να λάβουμε. Στην περίπτωση των συστημάτων συστάσεων η απόφαση που θα πάρουμε μπορεί να είναι μία από τις ακόλουθες: (i) εάν "θα αρέσει ένα προϊόν στον χρήστη" ή (ii) εάν "δεν θα αρέσει ένα προϊόν στον χρήστη", λαμβάνοντας πάντα υπόψη τα χαρακτηριστικά των προϊόντων για τα οποία αυτός ενδιαφέρεται. Κάθε εσωτερικός

κόμβος του *δέντρου απόφασης* αποτελεί μια συνθήκη ελέγχου για τα δεδομένα που κατηγοριοποιεί. Στην περίπτωσή μας η συνθήκη ελέγχου θα είναι ένα χαρακτηριστικό ενός προϊόντος (π.χ. όταν αναφερόμαστε σε ταινίες το χαρακτηριστικό θα μπορούσε να είναι το εάν πρόκειται για πολεμική ταινία ή όχι).

Για να κατηγοριοποιήσουμε ένα προϊόν με βάση το εάν θα αρέσει σ' έναν χρήστη ή όχι, ξεκινάμε από τη "ρίζα" του δέντρου (ο πρώτος κόμβος που βρίσκεται στην κορυφή του *δέντρου απόφασης* και φαίνεται στο Σχήμα 3.3). Στη "ρίζα" επομένως βρίσκεται το πρώτο χαρακτηριστικό διαχωρισμού των δεδομένων μας. Όταν λοιπόν ένα προϊόν ικανοποιεί τη συνθήκη ελέγχου του χαρακτηριστικού διαχωρισμού της ρίζας, τότε το κατηγοριοποιούμε στον αριστερό κόμβο-παιδί, ενώ όταν δεν την ικανοποιεί, τότε πηγαίνουμε στον δεξιό κόμβο-παιδί. Στη συνέχεια επαναλαμβάνουμε την ίδια διαδικασία στον κόμβο που επισκεπτόμαστε μέχρι να φτάσουμε σ' ένα φύλλο. Τέλος, αυτό το φύλλο κατατάσσει το προϊόν ως αρεστό ή μη στον υπό εξέταση χρήστη.

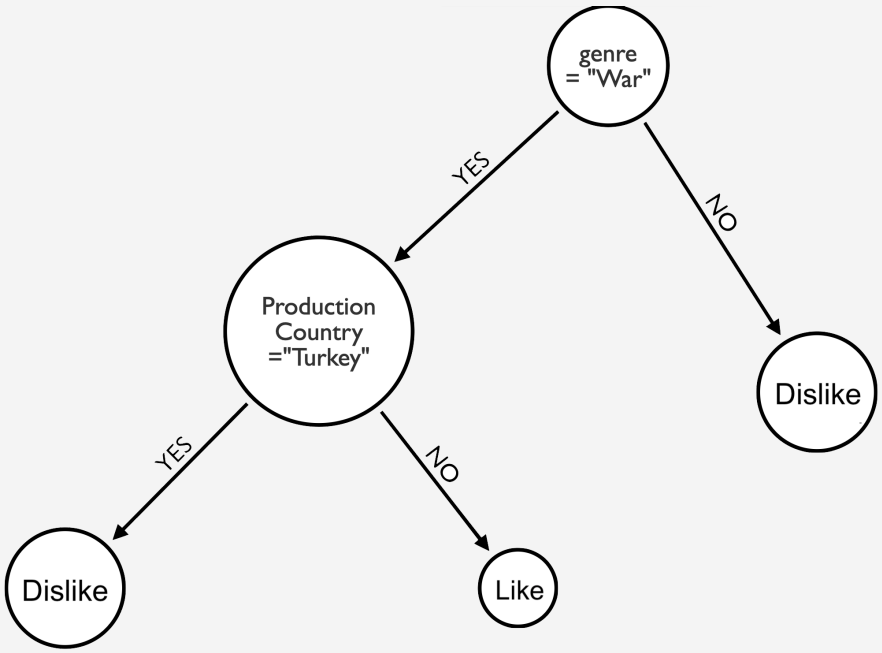
Επιπλέον, η κατασκευή ενός *δέντρου απόφασης* απαιτεί την επιλογή ενός χαρακτηριστικού διαχωρισμού για κάθε εσωτερικό κόμβο. Υπάρχουν πολλοί τρόποι για την επιλογή του καλύτερου χαρακτηριστικού διαχωρισμού των δεδομένων (π.χ. *δείκτης Gini, Εντροπία* ή X^2*-statistic* - τους οποίους τρόπους θα εξετάσουμε παρακάτω), όμως όλοι επιδιώκουν να ανατίθενται στα παιδιά-κόμβους όλα ή τα περισσότερα στιγμιότυπα δεδομένων που ανήκουν στην ίδια κλάση, προκειμένου να υπάρχει ομοιογένεια στα δεδομένα του κόμβου-παιδιού (ή αλλιώς θυγατρικού κόμβου), η οποία και θα οδηγήσει στη συνέχεια και σε μείωση της αβεβαιότητας του μοντέλου πρόβλεψής μας. Συμπερασματικά, ο ένας κόμβος-παιδί παίρνει όλα τα προϊόντα που αρέσουν στον συγκεκριμένο χρήστη, ενώ ο άλλος κόμβος-παιδί παίρνει όλα ή τα περισσότερα αρνητικά στιγμιότυπα (τα προϊόντα δηλαδή που δεν του αρέσουν).

Έπειτα, μόλις επιλέξουμε ένα χαρακτηριστικό διαχωρισμού (π.χ. είδος ταινιών: πολεμικές) για έναν κόμβο N, χωρίζουμε τις ταινίες σε δύο ομάδες: αυτές που ικανοποιούν το χαρακτηριστικό διαχωρισμού (όντας πολεμικές ταινίες) και αυτές που δεν το ικανοποιούν. Για κάθε ομάδα δεδομένων βρίσκουμε και πάλι το επόμενο χαρακτηριστικό διαχωρισμού (π.χ. χώρα παραγωγής: Τουρκία) με το οποίο να διακρίνονται καλύτερα τα θετικά από τα αρνητικά στιγμιότυπα των δεδομένων του κόμβου-πατέρα N, όπως φαίνεται στο Σχήμα 3.3. Τονίζεται ότι θετικά στιγμιότυπα δεδομένων θεωρούνται εκείνες οι ταινίες που αρέσουν στον υπό εξέταση χρήστη, ενώ αρνητικά στιγμιότυπα δεδομένων είναι αυτές που εικάζουμε ότι δεν θα αρέσουν στον χρήστη. Αυτή η διαδικασία διαχωρισμού δεδομένων και δημιουργίας κόμβων-παιδιών μπορεί να προχωρήσει σε οποιονδήποτε αριθμό επιπέδων κατασκευάζοντας έτσι ένα πολύπλοκο *δέντρο*, που ίσως όμως εξαιτίας αυτού να είναι δυσκολότερο να εξηγήσει τις προβλέψεις του. Ωστόσο, εάν η ομάδα των δεδομένων για έναν κόμβο είναι *ομοιογενής*, δηλαδή είναι όλα θετικά ή όλα αρνητικά στιγμιότυπα δεδομένων, τότε ο κόμβος μετατρέπεται αυτομάτως σε φύλλο και το *δέντρο απόφασης* δεν επεκτείνεται άλλο.

Παράδειγμα 3.2 Έστω ότι έχουμε μια ιστοσελίδα ενοικίασης ταινιών σε χρήστες (υπηρεσία παρόμοια με *Netflix, Cosmote TV, Apple TV* κ.ά.). Ο σκοπός του συστήματος συστάσεων είναι να προτείνει στους χρήστες ταινίες που είναι πιθανόν να τους αρέσουν, διαθέτοντας, επιπλέον, πληροφορίες για τα χαρακτηριστικά των ταινιών, όπως το είδος (πολεμική, αισθηματική, κωμωδία κ.λπ.), τη χώρα παραγωγής (*ΗΠΑ, Γερμανία, Ελλάδα* κ.λπ.) και την υπόθεση της ταινίας. Έστω ακόμη ότι υπάρχει ένας χρήστης *u* που του αρέσουν οι πολεμικές ταινίες εξαιρουμένων μόνο αυτών που έχουν ως χώρα παραγωγής την *Τουρκία*. Σ' αυτήν την περίπτωση, ο δισδιάστατος πίνακας που καταγράφει τις αλληλεπιδράσεις των χρηστών με τις ταινίες θα εμφανίζει μονάδα (1) για τον *u* εάν αυτός έχει αλληλεπιδράσει με μια ταινία, ενώ θα έχει κενή τιμή (*null value*) εάν δεν το έχει κάνει. Για τις ανάγκες του παραδείγματός μας θα θεωρήσουμε ότι οι μονάδες (1) συνεπάγονται "Like" (αρέσει στον χρήστη), ενώ οι κενές τιμές "Dislike" (δεν αρέσει στον χρήστη). Το μοντέλο προβλέψεών μας θα βασιστεί σε λογικές πράξεις της άλγεβρας *BOOLE* (*OR, AND* κ.λπ.) μεταξύ των χαρακτηριστικών (είδος, χώρα παραγωγής, υπόθεση) των υποψηφίων προς ενοικίαση ταινιών.

Όπως ειπώθηκε προηγουμένως, κάθε εσωτερικός κόμβος του *δέντρου απόφασης* αποτελεί μια συνθήκη ελέγχου για τα στιγμιότυπα δεδομένων που κατηγοριοποιεί. Σημειώνουμε ότι ως συνθήκη ελέγχου μπορεί να ληφθεί οποιαδήποτε *λογική έκφραση* που αποτελείται από μία ή περισσότερες προτάσεις συνδυασμένες με *λογικό ΚΑΙ (and)* ή *λογικό Η΄ (or)*, καθώς και από τις σχετικές παρενθέσεις, οι οποίες ορίζουν την προτεραιότητα μεταξύ των τελεστών. Δεδομένου λοιπόν ότι στον χρήστη *u* αρέσουν γενικώς οι πολεμικές ταινίες, μπορεί να διαπιστώσουμε ότι το καλύτερο χαρακτηριστικό για τη ρίζα του *δυαδικού δέντρου* είναι το *genre= War*. Οι ταινίες που αρέσουν στον χρήστη (όπως οι πολεμικές) θα ανήκουν στη θετική κλάση (ταινίες με τιμή 1 στη γραμμή για τον χρήστη *u* στον πίνακα χρηστών-ταινιών), ενώ οι ταινίες που δεν του αρέσουν και δεν ικανοποιούν τη συνθήκη *genre= War* θα ανήκουν στην αρνητική κλάση, αυτές δηλαδή με τις οποίες δεν έχει αλληλεπιδράσει ο χρήστης (παρουσιάζουν κενές τιμές στη γραμμή για τον χρήστη *u* στον πίνακα χρηστών-ταινιών). Το Σχήμα 3.3 δείχνει τη ρίζα καθώς και το υπόλοιπο *δέντρο απόφασης* του παραδείγματός μας. Ας υποθέσουμε τώρα ότι η ομάδα των ταινιών που δεν ικανοποιούν τη συνθήκη ελέγχου του χαρακτηριστικού της ρίζας περιλαμβάνει λίγα στιγμιότυπα δεδομένων, και άρα μπορούμε να συμπεράνουμε ότι όλες αυτές οι ταινίες ανήκουν στην κατηγορία "Dislike". Μπορούμε τότε να βάλουμε ένα φύλλο με απόφαση "Dislike" ως δεξιό παιδί της ρίζας, όπως φαίνεται στο Σχήμα 3.3. Ωστόσο κάποιες από τις ταινίες που ικανοποιούν τη λογική συνθήκη της ρίζας δεν αρέσουν στον χρήστη *u*, επειδή έχουν χώρα παραγωγής την *Τουρκία*. Επομένως, στο αριστερό παιδί της ρίζας κατασκευάζουμε μια δεύτερη *λογική συνθήκη (Production Country= Turkey)* όπως φαίνεται στο Σχήμα 3.3. Έτσι, βλέπουμε στο Σχήμα 3.3 το αριστερό παιδί της ρίζας, το οποίο εφαρμόζει αυτήν τη *λογική έκφραση*. Και τα δύο παιδιά αυτού του κόμβου είναι φύλλα, αφού

μπορούμε να υποθέσουμε ότι οι ταινίες που ικανοποιούν αυτήν τη *λογική έκφραση* είναι κατά κύριο λόγο αρνητικά στιγμιότυπα δεδομένων, ενώ εκείνα που δεν την ικανοποιούν είναι κατά κύριο λόγο θετικά.



Σχήμα 3.3: Δυαδικό *δέντρο απόφασης* του Παραδείγματος 3.1.

3.5.1 Δημιουργία Δέντρων Απόφασης βάσει Συνεργατικού Φιλτραρίσματος

Εκτός από τα *δέντρα απόφασης* που αναπτύχθηκαν στην προηγούμενη ενότητα και βασίστηκαν στα χαρακτηριστικά των στοιχείων, μπορούμε εναλλακτικά να κατασκευάσουμε *δέντρα απόφασης* απευθείας από τον πίνακα βαθμολογίας χρηστών-στοιχείων που χρησιμοποιείται στο *Συνεργατικό Φιλτράρισμα.*

Μια πρώτη, βασική, πρόκληση που έχουμε να αντιμετωπίσουμε στα *δέντρα απόφασης* που κατασκευάζονται βάσει των δεδομένων του *συνεργατικού φιλτραρίσματος* είναι ότι τα πεδία τιμών του πίνακα βαθμολογίας χρηστών-στοιχείων δεν διαχωρίζονται σαφώς κατά στήλες ως μεταβλητές χαρακτηριστικών και κλάσεων. Ένα σημαντικό πρόβλημα στο *συνεργατικό φιλτράρισμα* είναι ότι δεν γνωρίζουμε ποιο από τα στοιχεία θα πρέπει να προβλεφθεί από το *δέντρο απόφασης*, δεδομένου ότι οι εξαρτημένες και οι ανεξάρτητες μεταβλητές (στοιχεία) δεν είναι σαφώς οριοθετημένες. Το πρόβλημα αυτό είναι σχετικά εύκολο να αντιμετωπιστεί με την κατασκευή ενός ξεχωριστού *δέντρου απόφασης* για την πρόβλεψη της βαθμολογίας για καθένα από τα στοιχεία ξεχωριστά. Εάν η βαθμολογία που θα προβλέψουμε είναι υψηλότερη από ένα κατώφλι που θα ορίσουμε, τότε το στοιχείο μπορεί να προταθεί στον υπό εξέταση χρήστη, ενώ εάν δεν περάσει η βαθμολογία το κατώφλι, τότε αυτό το στοιχείο δεν θα προταθεί.

Θεωρήστε, λοιπόν, έναν πίνακα βαθμολογίας χρηστών-στοιχείων R διαστάσεων $m \times n$, όπου m οι χρήστες και n τα στοιχεία. Πρέπει να κατασκευαστεί ένα ξεχωριστό *δέντρο απόφασης* καθορίζοντας κάθε στοιχείο ως την εξαρτημένη μεταβλητή που θέλουμε να προβλέψουμε, και τα υπόλοιπα στοιχεία ως ανεξάρτητες μεταβλητές. Επομένως, ο αριθμός των *δέντρων απόφασης* που θα κατασκευαστούν είναι ακριβώς ίσος με τον αριθμό n των στοιχείων. Για την πρόβλεψη της βαθμολογίας ενός συγκεκριμένου στοιχείου για έναν υπό εξέταση χρήστη θα χρησιμοποιήσουμε το *δέντρο απόφασης* που κατασκευάσαμε για το συγκεκριμένο στοιχείο. Μ' άλλα λόγια, χρειαζόμαστε ένα *δέντρο* ανά χρήστη και ανά στοιχείο.

Μια δεύτερη σημαντική πρόκληση που θα πρέπει να αντιμετωπιστεί είναι το γεγονός ότι ο πίνακας βαθμολογίας χρηστών-στοιχείων είναι πολύ "αραιός", δεδομένου ότι απουσιάζουν οι περισσότερες τιμές. Αυτό δημιουργεί προβλήματα στην *ιεραρχική κατηγοριοποίηση* των δεδομένων εκπαίδευσης κατά τη φάση δημιουργίας του *δέντρου απόφασης*. Σκεφτείτε, λόγου χάρη, την περίπτωση όπου ένα συγκεκριμένο στοιχείο (π.χ. μια συγκεκριμένη ταινία) χρησιμοποιείται ως χαρακτηριστικό διαχωρισμού. Όλοι οι χρήστες των οποίων η βαθμολογία είναι μικρότερη από ένα κατώφλι αντιστοιχίζονται στον δεξιό κόμβο-παιδί του κόμβου-πατέρα, ενώ αυτοί των οποίων οι βαθμολογία είναι μεγαλύτερη από το κατώφλι αντιστοιχίζονται στον αριστερό κόμβο-παιδί του κόμβου-πατέρα. Δεδομένου ότι οι πίνακες βαθμολογίας χρηστών-στοιχείων είναι "αραιοί", οι περισσότεροι χρήστες δεν θα έχουν βαθμολογήσει το συγκεκριμένο στοιχείο. Σε ποιον κόμβο-παιδί (αριστερό ή δεξιό) θα πρέπει να ανατεθούν αυτοί οι χρήστες λοιπόν; Η λογική υπαγορεύει ότι οι χρήστες αυτοί θα πρέπει να αντιστοιχιστούν και στους δύο κόμβους-παιδιά (αριστερό και δεξιό). Ωστόσο, σε μια τέτοια περίπτωση, το *δέντρο απόφασης* δεν ακολουθεί πλέον μια αυστηρή κατηγοριοποίηση των δεδομένων εκπαίδευσης. Επιπλέον, σύμφωνα μ' αυτήν την προσέγγιση τα στιγμιότυπα των δεδομένων ελέγχου θα αντιστοιχούν σε πολλαπλά μονοπάτια στο *δέντρο απόφασης* και άρα οι πιθανώς αντικρουόμενες προβλέψεις από τα διάφορα μονοπάτια θα πρέπει να συνδυαστούν σε μια ενιαία τελική πρόβλεψη.

Μια δεύτερη λύση στο παραπάνω πρόβλημα της "αραιότητας" του πίνακα βαθμολογίας χρηστών-ταινιών είναι η δημιουργία μιας αναπαράστασης των δεδομένων του πίνακα σε χαμηλότερη διάσταση χρησιμοποιώντας την ανάλυση και παραγοντοποίηση πινάκων. Αυτή η αναπαράσταση των δεδομένων σε μικρότερες διαστάσεις χρησιμοποιείται για την κατασκευή ενός *δέντρου αποφάσεων* που θα βασίζεται σ' έναν δισδιάστατο πίνακα βαθμολογίας χρηστών-στοιχείων με συμπληρωμένες τιμές και άρα χωρίς το πρόβλημα της έλλειψής τους - θα αναλύσουμε την περίπτωση αυτή στο 5ο κεφάλαιο.

3.6 Κατηγοροποιητής Bayes

Κάποιες από τις σημαντικότερες μεθόδους που αναπτύχθηκαν στο επιστημονικό πεδίο της *κατηγοριοποίησης κειμένου* ήταν οι *πιθανοκρατικές* (*probabilistic*). Αυτές οι προσεγγίσεις βασίζονται

στη μέθοδο *Naive Bayes*, η οποία κάνει την "αφελή" (απλουστευτική) παραδοχή ότι υπάρχει *ανεξαρτησία μεταξύ των όρων/λέξεων ενός εγγράφου.*

Αναλυτικότερα, τα έγγραφα αναπαρίστανται με διανύσματα χαρακτηριστικών τύπου *Boolean* (0 ή 1) που περιγράφουν το εάν ένας συγκεκριμένος όρος εμφανίζεται σ' ένα έγγραφο ή όχι. Έτσι, στα πιθανοκρατικά μοντέλα το $P(t_i|C = c)$ εκφράζει τη *δεσμευμένη πιθανότητα* (την εξαρτώμενη δηλαδή από μια άλλη πιθανότητα) εμφάνισης του όρου t_i σ' ένα έγγραφο που ανήκει στην κλάση c. Τονίζουμε ότι οι *δεσμευμένες πιθανότητες* υπολογίζονται βάσει των παρατηρήσεων/στιγμιοτύπων των δεδομένων εκπαίδευσης.

Την τελευταία, λοιπόν, εικοσαετία έχει χρησιμοποιηθεί πολλές φορές ο *Αφελής Κατηγοροποιητής κατά Bayes* στο επιστημονικό πεδίο των *συστημάτων συστάσεων βάσει περιεχομένου.* Παρακάτω, θα περιγράψουμε με τη βοήθεια ενός παραδείγματος τις κατάλληλες προσαρμογές που πρέπει να γίνουν στον *Κατηγοριοποιητή Naive Bayes*, προκειμένου να μπορούμε να αποφανθούμε για το εάν θα πρέπει να προταθεί ένα στοιχείο σε έναν υπό εξέταση χρήστη ή όχι.

Παράδειγμα 3.3 Ο Πίνακας 3.1 απεικονίζει ένα απλό παράδειγμα ταινιών και κατηγοριών στις οποίες αυτές ανήκουν. Τα δεδομένα εκπαίδευσης αποτελούνται από πέντε ταινίες που έχει δει ο υπό εξέταση χρήστης καθώς και από τα είδη ταινιών (*movie genres*) στις οποίες ανήκουν αυτές οι ταινίες. Το ζητούμενο είναι να προβλεφθεί το εάν ο τρέχων χρήστης θα μπορούσε να ενδιαφερθεί για την ταινία test-6 και άρα εάν θα πρέπει να του την προτείνουμε.

Genre ⇒ Movie-Id ⇓	War	Adventure	Romance	Animation	**Class**
1	1	1	1	0	**like**
2	0	0	1	1	**dislike**
3	1	1	0	0	**like**
4	1	0	1	1	**like**
5	0	0	0	1	**dislike**
test-6	1	1	0	1	?

Πίνακας 3.1: Κατηγοριοποίηση βάσει διανυσμάτων με χαρακτηριστικά τύπου Boolean.

Στη μέθοδό μας θα υπολογίσουμε για την ταινία test-6 τη *δεσμευμένη πιθανότητα* για καθεμία από τις δύο πιθανές *κλάσεις* (*Θετική= like, Αρνητική= dislike*) στις οποίες μπορεί να ανήκει βάσει των χαρακτηριστικών που έχουν οι υπόλοιπες ταινίες, και στη συνέχεια θα επιλέξουμε την *κλάση* που θα έχει τη μεγαλύτερη πιθανότητα.

Για να προβλέψουμε την πιθανότητα η ταινία test-6 να ανήκει στη *θετική κλάση (Class= like)*, πρέπει να υπολογίσουμε τη *δεσμευμένη πιθανότητα* $P(test-6 = like|X)$, όπου X οι κατηγορίες στις οποίες ανήκει η παραπάνω ταινία: $X = (War = 1, Adventure = 1, Romance = 0, Animation = 1)$.

Για τον προσδιορισμό αυτής της πιθανότητας χρησιμοποιείται το *θεώρημα Bayes*, το οποίο μας επιτρέπει να υπολογίσουμε τη *δεσμευμένη πιθανότητα* $P(Y|X)$ ως ακολούθως:

$$P(Y \mid X) = \frac{\prod_{i=1}^{d} P(X_i \mid Y) \times P(Y)}{P(X)} \tag{3.5}$$

όπου Y η *κλάση (like, dislike)*, X_i το *χαρακτηριστικό* i ενός στοιχείου, και d ο αριθμός των χαρακτηριστικών του στοιχείου.
Για να προσδιορίσουμε τη σωστή *κλάση*, μπορούμε να υπολογίσουμε την πιθανότητα το διάνυσμα των χαρακτηριστικών X της ταινίας test-6 να ανήκει ή όχι σε μια *κλάση (like/dislike)* ως εξής:

$$\begin{aligned}
\mathrm{P(X \mid Class} = like) =& \mathrm{P(war} = 1 \mid \mathrm{Class} = like)\times \\
& \mathrm{P(adventure} = 1 \mid \mathrm{Class} = like)\times \\
& \mathrm{P(romance} = 0 \mid \mathrm{Class} = like) \times \mathrm{P(animation} = 1 \mid \mathrm{Class} = like) \\
=& 3/3 \times 2/3 \times 1/3 \times 1/3 \\
\approx& 0.072 \\
\mathrm{P(X \mid Class} = dislike) =& \mathrm{P(war} = 1 \mid \mathrm{Class} = dislike)\times \\
& \mathrm{P(adventure} = 1 \mid \mathrm{Class} = dislike)\times \\
& \mathrm{P(romance} = 0 \mid \mathrm{Class} = dislike) \times \mathrm{P(animation} = 1 \mid \mathrm{Class} = dislike) \\
=& 0/3 \times 0/3 \times 1/3 \times 2/3 \\
\approx& 0
\end{aligned}$$

Συνεπώς, η πιθανότητα να αρέσει στον υπό εξέταση χρήστη η ταινία test-6 είναι μεγαλύτερη της πιθανότητας να μην του αρέσει. Στο σημείο αυτό τονίζουμε ότι για τους παραπάνω υπολογισμούς δεν λαμβάνεται υπόψη ο παρονομαστής της Εξίσωσης 3.5 επειδή δεν εξαρτάται από το Y, με αποτέλεσμα να αρκεί να συγκρίνουμε τους αριθμητές μεταξύ τους και να μη χρειάζεται να βρούμε ακριβή τιμή πιθανότητας. Επομένως, το P(Y) (*prior probability*) μπορεί να θεωρηθεί ότι είναι ίδιο για όλα τα Y.

3.7 Επιλογή Χαρακτηριστικών (Feature Selection)

3.7.1 Δείκτης Gini

Ο *δείκτης Gini* είναι ένα μέτρο *αποτίμησης της ποιότητας των χαρακτηριστικών*, ώστε να γίνει κατάταξη αυτών ως προς την ικανότητά τους να διαχωρίζουν τα δεδομένα δημιουργώντας κατά το δυνατόν περισσότερο ομοιογενείς ομάδες δεδομένων. Μεγαλύτερη ομοιότητα μέσα στις δημιουργηθείσες ομάδες δεδομένων συνεπάγεται και υψηλότερη βεβαιότητα στις εκτιμήσεις του μοντέλου

πρόβλεψης. Ο *δείκτης Gini* είναι ένα απλό και εύκολα κατανοητό μέτρο αξιολόγησης. Επιπρόσθετα, είναι κατάλληλος για δυαδικές βαθμολογίες στοιχείων (π.χ. μου αρέσει/δεν μου αρέσει) ή βαθμολογίες που κατανέμονται σε μικρό διάστημα ακέραιων τιμών (π.χ. 1 έως 5).

Έστω, λοιπόν, c ο συνολικός αριθμός των πιθανών τιμών της βαθμολογικής κλίμακας ή αλλιώς ο διαφορετικός αριθμός των υπό εξέταση *κλάσεων*. Και έστω $p_1(f). \ldots p_c(f)$ το ποσοστό των στοιχείων -μεταξύ αυτών που περιέχουν ένα συγκεκριμένο χαρακτηριστικό f- που βαθμολογούνται σε καθεμία από αυτές τις c πιθανές τιμές (ή αλλιώς, *κλάσεις*) της βαθμολογικής κλίμακας. Τότε, ο *δείκτης Gini* του χαρακτηριστικού f ορίζεται ως εξής:

$$\text{Gini}(f) = 1 - \sum_{i=1}^{c} p_i(f)^2$$

Η τιμή του *Gini*(f) βρίσκεται πάντα στο συγκεκριμένο εύρος τιμών $(0, 1 - 1/c)$, όπου μικρότερες τιμές συνεπάγονται χαρακτηριστικά με αυξημένη ικανότητα διάκρισης, δεδομένου ότι επιφέρουν μικρότερη αβεβαιότητα στο μοντέλο πρόβλεψης. Για παράδειγμα, όταν η παρουσία του χαρακτηριστικού f έχει πάντα ως αποτέλεσμα το προϊόν να αξιολογείται με την j-οστή υψηλότερη τιμή βαθμολογίας (π.χ. στη δυαδική βαθμολογική κλίμακα 0 και 1 έχουμε $p_j(f) = 1$), τότε ένα τέτοιο χαρακτηριστικό έχει σημαντική διακριτική ικανότητα ως προς τις προβλέψεις της βαθμολογίας των στοιχείων. Αντίστοιχα, η τιμή του *δείκτη Gini* σε μια τέτοια περίπτωση είναι $1 - 1^2 = 0$. Όταν οι διαφορετικές τιμές της βαθμολογικής κλίμακας ενός χαρακτηριστικού $p_j(f)$ έχουν το ίδιο ποσοστό 1/c, τότε ο *δείκτης Gini* παίρνει τη μέγιστη τιμή τού $1 - \sum_{i=1}^{c} \left(1/c^2\right) = 1 - 1/c$, και η οποία σχετίζεται με την υψηλή αβεβαιότητα που προσδίδει το υπό εξέταση χαρακτηριστικό στο μοντέλο πρόβλεψης. Γι' αυτόν τον λόγο δεν θα πρέπει να επιλεχθεί, αλλά να προτιμηθεί κάποιο άλλο χαρακτηριστικό αντ' αυτού.

3.7.2 Εντροπία

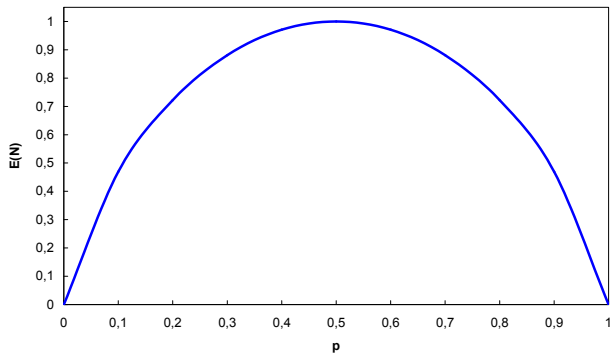
Η *εντροπία* (*entropy*) είναι ένα μέτρο *αποτίμησης της ανομοιογένειας των στιγμιοτύπων των δεδομένων ενός κόμβου*. Μοιάζει πολύ με τον *δείκτη Gini*, με τη διαφορά ότι για τον σχεδιασμό του μέτρου χρησιμοποιούνται αρχές του επιστημονικού πεδίου της *Θεωρίας της Πληροφορίας*. Όπως και στην προηγούμενη περίπτωση λοιπόν, έστω t ο συνολικός αριθμός των πιθανών τιμών της βαθμολογικής κλίμακας, και $p_1(f) \ldots p_t(f)$ το ποσοστό των στοιχείων που περιέχουν ένα συγκεκριμένο χαρακτηριστικό f και τα οποία (στοιχεία) βαθμολογούνται με καθεμία από αυτές τις c πιθανές τιμές (π.χ. διάστημα τιμών 1 έως 5). Με n συμβολίζουμε τον αριθμό των στιγμιοτύπων δεδομένων που αφορούν σ' ένα χαρακτηριστικό f, με c τον αριθμό των *κλάσεων*, ενώ n_i είναι ο αριθμός των στιγμιοτύπων δεδομένων του χαρακτηριστικού f που ανήκουν στην κλάση i (για $1 \leq i \leq c$). Άρα, εάν $p_i(f) = n_i/n$, τότε ορίζουμε την **Εντροπία** $E(f)$ του χαρακτηριστικού f ως ακολούθως:

$$E(f) = -\sum_{i=1}^{c} p_i(f) \cdot \log_2 p_i(f) \tag{3.6}$$

Για την περίπτωση δύο κλάσεων (δηλαδή $c = 2$), θέτοντας $p_i(f) = p$ έχουμε:

$$E(f) = -p \cdot \log_2 p - (1-p) \cdot \log_2(1-p)$$

Σ' αυτήν την περίπτωση, όπως φαίνεται και από τη γραφική παράσταση του Σχήματος 3.4, η ελάχιστη τιμή της *εντροπίας* $E(f)$ ενός χαρακτηριστικού f είναι ίση με 0. Αυτή η τιμή προκύπτει για $p = 0$ ή για $p = 1$, επειδή τότε μόνο το χαρακτηριστικό f οδηγεί σ' έναν τέτοιο διαχωρισμό δεδομένων ώστε να είναι εντελώς *ομοιογενή*, αφού ο κάθε κόμβος-παιδί εμπεριέχει μόνο στιγμιότυπα της ίδιας κλάσης. Η μέγιστη επομένως τιμή της *εντροπίας* $E(f)$ είναι 1 όταν το χαρακτηριστικό f σχετίζεται με πολύ *ανομοιογενή δεδομένα* (δηλαδή ίσο αριθμό στιγμιοτύπων και από τις 2 κλάσεις).

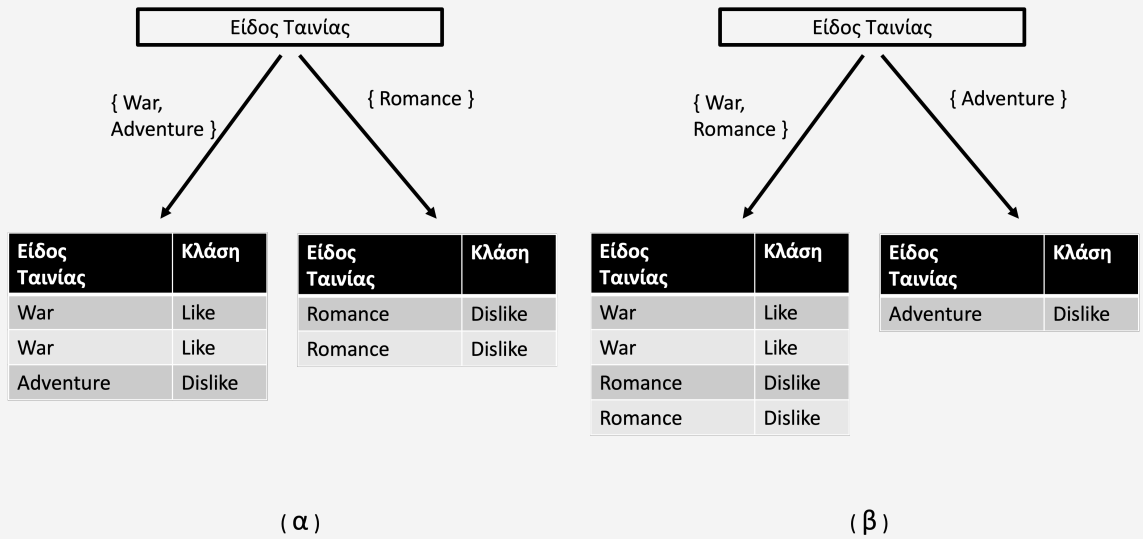


Σχήμα 3.4: Γραφική παράσταση *εντροπίας* $E(f)$ βάσει του χαρακτηριστικού f.

Για ένα χαρακτηριστικό f, η τιμή της *εντροπίας* του $E(f)$ βρίσκεται πάντα στο εύρος τιμών $(0, 1)$. Μικρότερες τιμές να σημαίνουν καλύτερη διακριτική ικανότητα για το υπό εξέταση χαρακτηριστικό, δεδομένου ότι μπορεί να διαχωρίζει τα δεδομένα κατάλληλα ώστε να δημιουργούνται περισσότερο ομοιογενείς ομάδες δεδομένων.

Συμπερασματικά, είναι εύκολο να διαπιστώσουμε ότι η *εντροπία* έχει παρόμοια χαρακτηριστικά με τον *δείκτη Gini*. Στην πραγματικότητα, αυτά τα δύο μέτρα συχνά δίνουν πολύ παρόμοια αποτελέσματα, αν και στηρίζονται σε διαφορετικές ερμηνείες των πιθανοτήτων. Ο *δείκτης Gini* γίνεται ευκολότερα κατανοητός, ενώ το *μέτρο της εντροπίας* στηρίζεται περισσότερο σε μαθηματικές αρχές που προέρχονται από τη *Θεωρία της Πληροφορίας*.

Παράδειγμα 3.4 Έστω ότι θέλουμε να προβλέψουμε εάν σ' έναν χρήστη θα αρέσει ή όχι μια ταινία βάσει του είδους (*war, adventure* κ.λπ.) στο οποίο αυτή ανήκει. Ακόμη, έστω ότι η κατασκευή του *δέντρου απόφασης* στηρίζεται στο χαρακτηριστικό "είδος ταινιών" και ότι έχει δύο διαφορετικούς τρόπους διαχωρισμού των κόμβων-παιδιών, οι οποίοι παρουσιάζονται στο Σχήμα 3.5. Βάσει των παραπάνω, λοιπόν, καλούμαστε να αξιολογήσουμε ποιος από τους διαχωρισμούς των στιγμιοτύπων των κόμβων-παιδιών είναι ο καταλληλότερος.



Σχήμα 3.5: Δύο διαφορετικές επιλογές σχηματισμού των κόμβων-παιδιών βάσει του είδους της ταινίας.

Ο αλγόριθμος κατασκευής του *δέντρου απόφασης* επιδιώκει να φθάσει σε ομοιογενή φύλλα, που ως εκ τούτου θα περιέχουν στιγμιότυπα δεδομένων από μία μόνο *κλάση*. Συγκρίνοντας τις περιπτώσεις των Σχημάτων 3.5(α) και 3.5(β), με μια πρώτη ματιά σχηματίζουμε την εντύπωση ότι οι κόμβοι-παιδιά που προκύπτουν στο πρώτο σχήμα φαίνεται να είναι περισσότερο ομοιογενείς από αυτούς του δεύτερου. Επομένως, χρειαζόμαστε ένα ποσοτικό μέτρο για τον προσδιορισμό του βαθμού ομοιογένειας των κόμβων-παιδιών που παράγονται από μία διάσπαση, ώστε να επιλέξουμε αυτήν που επιφέρει τη μεγαλύτερη ομοιογένεια. Και ο λόγος είναι ότι η κατασκευή του *δέντρου απόφασης* θα τερματιστεί συντομότερα, παράγοντας έτσι ένα κατά το δυνατόν μικρότερο (και άρα απλούστερο) *δέντρο απόφασης*, το οποίο επομένως θα μπορούμε να ερμηνεύσουμε και ευκολότερα.

Παράδειγμα 3.5 Στη συνέχεια θα υπολογίσουμε την *εντροπία* του αριστερού και του δεξιού κόμβου-παιδιού που παράγεται από τις δύο εκδοχές που παρουσιάζονται στο Σχήμα 3.5. Το σύμβολο p της Εξίσωσης 3.6 της *εντροπίας* αντιστοιχεί στην αναλογία των στιγμιοτύπων δεδομένων της *κλάσης Like*. Στο Σχήμα 3.5(α) η *εντροπία* του αριστερού κόμβου-παιδιού είναι ίση με $-2/3 \cdot \log_2 \cdot 2/3 - 1/3 \cdot \log_2 1/3 = 0.9183$, ενώ του δεξιού είναι $-0 \cdot \log_2 0 - 1 \cdot \log_2 1 = 0$. Και στο Σχήμα 3.5(β) η *εντροπία* του αριστερού κόμβου-παιδιού ισούται με $-2/4 \cdot \log_2 2/4 - 2/4 \cdot \log_2 2/4 = 1$, ενώ του δεξιού με $0 \cdot \log_2 0 - 1 \cdot \log_2 1 = 0$.

Έχοντας πλέον ορίσει ένα ποσοτικό μέτρο για την αποτίμηση της *ομοιογένειας* των δεδομένων ενός κόμβου N, εν συνεχεία θα χρειαστούμε όμως και ένα συνολικό μέτρο αποτύπωσης της *ομοιογένειας* που προκύπτει με τη διάσπαση ενός κόμβου N με n στιγμιότυπα δεδομένων σε δύο κόμβους-παιδιά.

Έστω, επομένως, ότι μετά τη διάσπαση N_l θα είναι ο αριστερός κόμβος με n_l στιγμιότυπα δεδομένων, ενώ N_r θα είναι ο δεξιός κόμβος με n_r στιγμιότυπα δεδομένων. Το **Πληροφοριακό Κέρδος** (*Information Gain*) IG που θα προκύψει από την παραπάνω περίπτωση είναι ένα ποσοτικό μέτρο το οποίο ορίζεται ως εξής:

$$IG = E(N) - \frac{n_l}{n} \cdot E(N_l) - \frac{n_r}{n} \cdot E(N_r)$$

όπου $E(N)$ η *εντροπία* του κόμβου-γονέα πριν τη διάσπασή του, $E(N_l)$ η *εντροπία* του αριστερού κόμβου-παιδιού, και $E(N_r)$ η *εντροπία* του δεξιού κόμβου-παιδιού. Συμπερασματικά, για κάθε διαφορετικό τρόπο διαχωρισμού των δεδομένων μας μετρούμε το πληροφοριακό κέρδος που θα προκύψει και άρα επιλέγουμε αυτόν που θα το μεγιστοποιήσει.

Παράδειγμα 3.6 Στη συνέχεια, υπολογίζουμε το πληροφοριακό κέρδος που παράγεται από κάθε περίπτωση του Σχήματος 3.5. Πριν τη διάσπαση ο κόμβος έχει 5 στιγμιότυπα εκ των οποίων τα 2 ανήκουν στην *κλάση Like* και τα 3 στην *κλάση Dislike*. Συνάγεται επομένως ότι η *εντροπία* του είναι ίση με $-2/5 \cdot \log_2 2/5 - 3/5 \cdot \log_2 3/5 = 0.971$. Στο Σχήμα 3.5(α) το πληροφοριακό κέρδος είναι ίσο με $0.971 - 3/5 \cdot 0.9183 - 2/5 \cdot 0 = 0.42$, ενώ στο 3.5(β) είναι $0.971 - 4/5 \cdot 1 - 1/5 \cdot 0 = 0.171$. Συνεπώς, το μέγιστο πληροφοριακό κέρδος προκύπτει από τη διάσπαση της πρώτης εκδοχής, Σχήμα 3.5(α), κάτι που επιβεβαιώνει και την ορθότητα της αρχικής μας εκτίμησης ως προς την επιλογή αυτού του τρόπου διαχωρισμού των δεδομένων, δηλαδή, αφενός *Είδος Ταινίας* $\in$ {*War, Adventure*}, και αφετέρου *Είδος Ταινίας* $\in$ {*Romance*}.

3.7.3 X^2-statistic

Το χ^2-*statistic* είναι ένα μέτρο αποτίμησης του βαθμού συσχέτισης ενός χαρακτηριστού f με μια *κλάση* c (π.χ. η *κλάση* "Αγοράζω το προϊόν" ή η *κλάση* "Δεν αγοράζω το προϊόν"). Πιο συγκεκριμένα, το χ^2-*statistic* μετρά τη *σταθμισμένη απόκλιση* μεταξύ των παρατηρούμενων και των εκτιμώμενων τιμών για τις διαφορετικές περιπτώσεις ενός *πίνακα συνάφειας* (*contigency table*). Στη στατιστική ο *πίνακας συνάφειας* εμφανίζει την κατανομή συχνοτήτων των μεταβλητών.

Έστω, λοιπόν, O_i η παρατηρούμενη και E_i η εκτιμώμενη τιμή του i-ου κελιού του *πίνακα συνάφειας*. Τότε, το χ^2-*statistic* υπολογίζεται ως εξής:

$$\chi^2 = \sum_{i=1}^{p} \frac{(O_i - E_i)^2}{E_i}$$

όπου p ο αριθμός των κελιών του *πίνακα συνάφειας*.

Παράδειγμα 3.7 Έστω ότι μας ζητείται να προβλέψουμε εάν ένα συγκεκριμένο χαρακτηριστικό f είναι *συναφές* με τα αγοραστικά ενδιαφέροντα ενός χρήστη. Ας υποθέσουμε, τώρα, ότι ο χρήστης έχει αγοράσει περίπου το 20% των προϊόντων του καταστήματος και το χαρακτηριστικό f εμπεριέχεται στο 40% των προϊόντων του καταστήματος. Επίσης,

έστω ότι ο συνολικός αριθμός των προϊόντων στο κατάστημα είναι 100. Υπό τις παραπάνω συνθήκες, οι **εκτιμώμενες τιμές** (**expected values**) εμφάνισης κάθε πιθανού συνδυασμού μεταξύ της ύπαρξης ή μη ενός χαρακτηριστικού f σ' ένα προϊόν και των δύο ενδεχομένων της κλάσης c είναι οι παρακάτω:

	Το προϊόν έχει το f	Το προϊόν δεν έχει το f
Ο χρήστης αγόρασε το προϊόν	$E_1 = 100 * 0.2 * 0.4 = 8$	$E_2 = 100 * 0.2 * 0.6 = 12$
Ο χρήστης δεν αγόρασε το προϊόν	$E_3 = 100 * 0.8 * 0.4 = 32$	$E_4 = 100 * 0.8 * 0.6 = 48$

Προσέξτε ότι, εάν αθροίσουμε τις 4 *εκτιμώμενες τιμές* (που έχουμε δηλαδή προβλέψει) του παραπάνω πίνακα, τότε παίρνουμε το 100, που είναι και ο συνολικός αριθμός των προϊόντων του καταστήματός μας. Επίσης, οι *εκτιμώμενες τιμές* υπολογίζονται υπό την παραδοχή ότι η εμφάνιση του χαρακτηριστικού f και το ενδιαφέρον του χρήστη να αγοράσει το υπό εξέταση προϊόν είναι ανεξάρτητα μεταξύ τους. Στην πράξη ωστόσο, το χαρακτηριστικό f μπορεί να έχει υψηλή *συνάφεια* με το συγκεκριμένο προϊόν. Για παράδειγμα, λάβετε υπόψη σενάριο όπου οι *εκτιμώμενες τιμές*, οι οποίες εμφανίζονται στον προηγούμενο *πίνακα συνάφειας*, αποκλίνουν από τις **παρατηρούμενες τιμές** (**observed values**), οι οποίες εμφανίζονται στον παρακάτω πίνακα:

	Το προϊόν έχει το f	Το προϊόν δεν έχει το f
Ο χρήστης αγόρασε το προϊόν	$O_1 = 6$	$O_2 = 4$
Ο χρήστης δεν αγόρασε το προϊόν	$O_3 = 14$	$O_4 = 76$

Το παραπάνω γεγονός υποδεικνύει ότι ο χρήστης είναι πολύ πιθανόν να αγοράσει το προϊόν που εμπεριέχει το χαρακτηριστικό f. Επομένως, με βάση τα δεδομένα των δύο παραπάνω *πινάκων συνάφειας* το χ^2*-statistic* υπολογίζεται ως εξής:

$$\begin{aligned}\chi^2 &= \frac{(6-8)^2}{8} + \frac{(4-12)^2}{12} + \frac{(14-32)^2}{32} + \frac{(76-48)^2}{48} \\ &= 0.5 + 5.33 + 10.125 + 16.33 \\ &= 32.288\end{aligned}$$

Τέλος, τονίζουμε ότι υψηλές τιμές χ^2*-statistic* δείχνουν ότι ένα συγκεκριμένο χαρακτηριστικό f και η αγορά ενός προϊόντος σχετίζονται σε μεγάλο βαθμό. Από την άλλη μεριά, εάν οι παρατηρούμενες τιμές είναι ακριβώς ίσες με τις εκτιμώμενες τιμές, τότε αυτό σημαίνει ότι το αντίστοιχο χαρακτηριστικό f είναι άσχετο με την αγορά του συγκεκριμένου προϊόντος. Σε μια τέτοια περίπτωση το χ^2*-statistic* θα πάρει τη μικρότερη δυνατή τιμή του, δηλαδή ίση με 0. Επομένως, στο τέλος της διαδικασίας διατηρούμε τα κορυφαία k χαρακτηριστικά των προϊόντων, που έχουν και τη μεγαλύτερη τιμή χ^2*-statistic*, προκειμένου να τα χρησιμοποιήσουμε στις προβλέψεις μας.

3.8 Ερωτήσεις Κεφαλαίου

1. Αναφέρετε τις κύριες διαφορές μεταξύ των μεθόδων *Συνεργατικού Φιλτραρίσματος* και *Φιλτραρίσματος βάσει Περιεχομένου*. Ποια τα πλεονεκτήματα και τα μειονεκτήματα της κάθε μεθόδου έναντι της άλλης;

2. Τί γνωρίζετε για το *μοντέλο διανυσματικού χώρου*; Να ορίσετε τη *συχνότητα όρου (tf)* και την *αντίστροφη συχνότητα εγγράφου (idf)*, και βάσει αυτών και την *τεχνική στάθμισης όρων tf-idf*.

3. Πώς μπορούν να συνδυαστούν τα δεδομένα *Συνεργατικού Φιλτραρίσματος* με τα *Δεδομένα Περιεχομένου*; Να καταγράψετε συνοπτικά τα τέσσερα βήματα για την υλοποίηση ενός *υβριδικού* μοντέλου που θα αξιοποιεί και τα δύο.

4. Αναφέρετε τους βασικούς *κατηγοριοποιητές* και συγκρίνετέ τους ως προς τα πλεονεκτήματα και μειονεκτήματά τους.

5. Περιγράψτε συνοπτικά τον *κατηγοριοποιητή Δέντρων Απόφασης* για τα συστήματα συστάσεων.

6. Σχεδιάστε ένα *δυαδικό Δέντρο Απόφασης* για ένα σύστημα συστάσεων.

7. Περιγράψτε συνοπτικά τον *κατηγοριοποιητή Naive Bayes* για τα συστήματα συστάσεων.

8. Να ορίσετε με τη βοήθεια ενός μαθηματικού τύπου την *εντροπία* ενός χαρακτηριστικού f για την περίπτωση δύο κλάσεων και να περιγράψετε κάθε μεταβλητή του μαθηματικού τύπου που ορίσατε.

9. Να ορίσετε μαθηματικά τον *Gini index* ενός χαρακτηριστικού f και να περιγράψετε κάθε μεταβλητή του προαναφερθέντος μαθηματικού τύπου.

10. Να ορίσετε μαθηματικά το *πληροφοριακό κέρδος (information gain)* G από τη διάσπαση ενός κόμβου N, και να περιγράψετε κάθε μεταβλητή του παραπάνω μαθηματικού τύπου.

11. Ποια είναι κατά τη γνώμη σας η πιο αποτελεσματική *μέθοδος επιλογής χαρακτηριστικών* (*feature selection*) και γιατί;

12. Έστω ότι ένας χρήστης παρέχει αξιολογήσεις του τύπου "μου αρέσει/δεν μου αρέσει" για ένα σύνολο 20 ταινιών από τις οποίες αξιολογεί 9 ως "αρεστές" και τις υπόλοιπες ως "μη αρεστές". Ας υποθέσουμε, επίσης, ότι 7 περιγραφές ταινιών περιέχουν τη λέξη "θρίλερ" και ότι ο χρήστης αντιπαθεί 5 από αυτές. Να υπολογίσετε τον *δείκτη Gini* σε σχέση με την αρχική κατανομή των δεδομένων και σε σχέση με το υποσύνολο των ταινιών που περιέχουν τη λέξη "θρίλερ". Θα πρέπει οι *αλγόριθμοι επιλογής χαρακτηριστικών* να λαμβάνουν υπόψη τη λέξη "θρίλερ" στις περιγραφές των ταινιών ή όχι, και γιατί;

13. Έστω ένα σύστημα συστάσεων ταινιών στο οποίο οι ταινίες ανήκουν σ' ένα ή περισσότερα είδη ταινιών και εμφανίζονται στον παρακάτω πίνακα. Επιπρόσθετα, έστω ότι ο υπό εξέταση χρήστης μας παρέχει τις ακόλουθες αξιολογήσεις των ταινιών. Να υπολογίσετε την πιθανότητα να αρέσει ή μη η ταινία Test-1 και Test-2 στον υπό εξέταση χρήστη βάσει του *κατηγοριοποιητή Naive Bayes*.

Genre ⇒ Movie-Id ⇓	War	Adventure	Romance	Animation	Like or Dislike
1	1	0	1	0	Dislike
2	1	1	1	0	Dislike
3	0	0	0	0	Dislike
4	0	1	1	0	Like
5	0	1	1	1	Like
6	0	0	0	1	Like
Test-1	0	1	0	1	?
Test-2	0	0	0	0	?

3.9 Ασκήσεις Προγραμματισμού

***Σύσταση υποψηφίων φίλων βάσει του αλγορίθμου φιλτραρίσματος περιεχομένου*, ο οποίος εφαρμόζεται σε δεδομένα που αφορούν στα είδη ταινιών που αρέσουν στους κινηματογραφόφιλους (*σινεφίλ*).**

Μας δίνεται μια λίστα με τους παρακάτω 4 θεατές (*viewers*) με το id (*αριθμό ταυτοποίησης*) τους:

```
viewers = [
    { "viewer_id": 0 },
    { "viewer_id": 1 },
    { "viewer_id": 2 },
    { "viewer_id": 3 }
]
```

Επίσης, μας δίνεται μια λίστα με τα είδη ταινιών (π.χ. *πολεμική*, *ρομαντική* κ.λπ.) που αρέσουν στον κάθε θεατή:

```
genres = [
(0, "Action"), (0, "Mystery"), (0, "Animation"), (0, "Biography
    "), (0, "History"), (0, "Comedy"), (0, "Concert"),
```

```
(1, "Crime"), (1, "Documentary"), (1, "History"), (1, "Animation"), (1, "Drama"),
(2, "Educational"), (2, "Horror"), (2, "Fantasy"), (2, "Film Noir"), (2, "Game show"), (2, "History"),
(3, "Horror"), (3, "Educational"), (3, "History"), (3, "Musical"), (3, "Mystery")]
```

Για παράδειγμα, στους θεατές με id 0 και 1 αρέσουν από κοινού δύο είδη ταινιών *History* και *Animation*, αντίστοιχα. Στη συνέχεια, θα υλοποιήσουμε μια συνάρτηση η οποία θα προτείνει ως υποψηφίους φίλους στον υπό εξέταση θεατή άλλους θεατές που τους αρέσουν παρόμοια είδη ταινιών (π.χ. *ρομαντικές*). Αρχικά, θα κατασκευάσουμε ένα *λεξικό (dictionary)* στο οποίο, όταν θα δίνεται το *id* ενός θεατή, θα παράγεται μια λίστα με όλα τα είδη ταινίας που του αρέσουν βάσει του παρακάτω κώδικα:

```
# Δημιούργησε ένα λεξικό όπου τα κλειδιά του είναι τα ids των θεατών και οι τιμές του λεξικού είναι λίστες με τα είδη ταινιών που αρέσουν στον καθένα από αυτούς
viewerid_and_a_list_of_genres_he_likes = {viewer["viewer_id"]: [] for viewer in viewers}
for viewer_id, genre in genres:
    viewerid_and_a_list_of_genres_he_likes[viewer_id].append(genre)
viewerid_and_a_list_of_genres_he_likes
```

Ο προηγούμενος κώδικας δημιουργεί το παρακάτω *dictionary*:

{0: ['Action', 'Mystery', 'Animation', 'Biography', 'History', 'Comedy', 'Concert'],
1: ['Crime', 'Documentary', 'History', 'Animation', 'Drama'],
2: ['Educational', 'Horror', 'Fantasy', 'Film Noir', 'Game-show', 'History'],
3: ['Horror', 'Educational', 'History', 'Musical', 'Mystery']}

Για παράδειγμα, ο θεατής με id 1 ενδιαφέρεται για τα είδη ταινίας *Crime, Documentary, History, Animation,* και *Drama.*

Επίσης, θα κατασκευάσουμε ένα δεύτερο *λεξικό (dictionary)* όπου, όταν θα δίνεται ένα είδος ταινίας, θα παράγεται μια λίστα με τους χρήστες που ενδιαφέρονται γι' αυτό.

```
# Δημιούργησε ένα λεξικό όπου τα κλειδιά του είναι τα είδη ταινίας και οι τιμές του λεξικού είναι λίστες με τα ids των θεατών στους οποίους αρέσουν αυτά τα είδη ταινίας

genreName_and_a_list_of_viewerids_they_like_it = {}
for viewer_id, genre in genres:
    if genre not in genreName_and_a_list_of_viewerids_they_like_it:
        genreName_and_a_list_of_viewerids_they_like_it[genre] = []
    if viewer_id not in genreName_and_a_list_of_viewerids_they_like_it:
        genreName_and_a_list_of_viewerids_they_like_it[genre].append(viewer_id)

genreName_and_a_list_of_viewerids_they_like_it
```

Ο προηγούμενος κώδικας δημιουργεί το ακόλουθο *dictionary*:

{'Action': [0],
'Mystery': [0, 3],
'Animation': [0, 1],
'Biography': [0],
'History': [0, 1, 2, 3],
'Comedy': [0],
'Concert': [0],
'Crime': [1],
'Documentary': [1],
'Drama': [1],
'Educational': [2, 3],
'Horror': [2, 3],
'Fantasy': [2],
'Film Noir': [2],
'Game-show': [2],
'Musical': [3]}

Για παράδειγμα, το είδος ταινίας *Animation* αρέσει στον θεατή με id 0, καθώς και στον θεατή με id 1. Βάσει των δύο *λεξικών* που δημιουργήσαμε, μπορούμε να φτιάξουμε μια συνάρτηση που θα εντοπίζει τον θεατή που θα έχει τα περισσότερα κοινά είδη ταινίας με τον υπό εξέταση θεατή. Η παρακάτω συνάρτηση προτείνει στον υπό εξέταση θεατή μια *διατεταγμένη λίστα προτεινόμενων υποψηφίων φίλων*, οι οποίοι κατατάσσονται βάσει των κοινών ενδιαφερόντων που έχουν με τον υπό εξέταση χρήστη:

```
def most_common_genres(viewer):

    dictionary = {}
    for ids in viewers:
        counter = 0
        for genre in viewerid_and_a_list_of_genres_he_likes[viewer["viewer_id"]]:
            for my_viewer_id in genreName_and_a_list_of_viewerids_they_like_it[genre]:
                if ids['viewer_id'] == my_viewer_id:
                    if my_viewer_id != viewer["viewer_id"]:
                        counter +=1
                        dictionary[my_viewer_id] = counter

    return dict(sorted(dictionary.items(), key=lambda x: x[1], reverse=True))

most_common_genres(viewers[3])
```

Το αποτέλεσμα του παραπάνω κώδικα είναι: {2: 3, 0: 2, 1: 1}.

Συνεπώς, για τον υπό εξέταση θεατή με id 3 ο *κορυφαίος υποψήφιος προς σύσταση φίλος* είναι ο θεατής με id 2, επειδή έχουν κοινά 3 είδη ταινίας (*Horror, Educational* και *History*). Ακολουθεί ο θεατής με id 0 με δύο (*Mystery, History*), και ο θεατής με id 1 μ' ένα μόλις κοινό είδος ταινίας (*History*).

Βιβλιογραφία

R. Baeza-Yates and B. Ribeiro-Neto. *Modern information retrieval*, volume 463. ACM press New York, 1999.

M. Balabanovic and S. Y. Fab: Content-based, collaborative recommendation. *ACM Communications*, 40(3):66--72, 1997.

X. Jin, Y. Zhou, and B. Mobasher. A maximum entropy web recommendation system: Combining collaborative and content features. In *Proc. ACM SIGKDD Conf.*, pages 612--617, 2005.

P. Melville, R. J. Mooney, and N. R. Content-boosted collaborative filtering for improved recommendations. In *Proceedings AAAI Conference*, pages 187--192, 2002.

J. Salter and N. Antonopoulos. Cinemascreen recommender agent: Combining collaborative and content-based filtering. *Intelligent Systems Magazine*, 21(1):35--41, 2006.

W. T. Yan and H. G. Molina. Sift: a tool for wide-area information dissemination. In *Proceedings UNSENIX Conference*, pages 177--186, 1995.