ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

”Μείωση Δεδομένων με διαχωρισμό του χώρου για σύνολα δεδομένων πολλαπλών ετικετών”

Της φοιτητήτριας
Φωτιάδου Γεωργία
Αρ. Μητρώου: 175047

Επιβλέπων
Ουγιάρογλου Στέφανος
Βαθμίδα ΕΔΙΠ

7 Ιουλίου 2021

Τίτλος Π.Ε.: Μείωση Δεδομένων με διαχωρισμό του χώρου για σύνολα δεδομένων πολλαπλών ετικετών
Κωδικός Π.Ε. 21199
Ονοματεπώνυμο φοιτητή/ών: Φωτιάδου Γεωργία
Ονοματεπώνυμο εισηγητή: Ουγιάρογλου Στέφανος
Ημερομηνία ανάληψης Δ.Ε.: 19-03-2021
Ημερομηνία περάτωσης Δ.Ε.: 19-06-2021

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως διπλωματική εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Πρόλογος

Η συγκεκριμένη διπλωματική εργασία αφορά τον τομέα της Εξόρυξης Δεδομένων. Στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνές Πανεπιστημίου διδάσκεται το μάθημα "Οργάνωση Δεδομένων και Εξόρυξη Πληροφορίας", το οποίο δίνει τη βάση για την εισαγωγή εννοιών και την κατανόηση αγλορίθμων και τεχνικών που σχετίζονται με την εξόρυξη δεδομένων. Αυτό το μάθημα προσελκύει αρκετά το ενδιαφέρον του φοιτητή, καθώς το αντικείμενο ανάλυσης του, που είναι αλγόριθμοι κατηγοριοποίησης, τεχνικές μείωσης δεδομένων, συσταδοποίηση κ.α. εφαρμόζονται σε παραδείγματα της καθημερινότητας και έτσι μπορεί κανείς να τα προσεγγίσει και να τα καταλάβει. Προσωπικά, επειδή το μάθημα αυτό διδάσκεται προς τα τελευταία εξάμηνα του προγράμματος σπουδών και είναι μάθημα κατεύθυνσης, με βοήθησε να καταλάβω ότι είναι το αντικείμενο που με ενδιαφέρει αρκετά και μου δίνει το κίνητρο για να ασχοληθώ με έρευνα που αφορά τα δεδομένα. Αναπτύσσονται πολλές έρευνες σχετικά με την εξόρυξη των δεδομένων, που βοηθούν διάφορους τομείς όπως την ιατρική, την χημεία, τη βιολογία κ.α., καθώς προσφέρονται ποικίλοι τρόποι για τη διαχείριση ενός συνόλου δεδομένων και για την εξαγωγή συμπερασμάτων έπειτα από επεξεργασία. Τέλος, είναι σημαντικό για μένα ότι γίνονται προσπάθειες επέκτασης των ήδη υπάρχοντων αλγορίθμων και τεχνικών, επομένως με αφορμή την διπλωματική αυτή, ίσως μπορέσω κι εγώ να συνεισφέρω στο έργο αυτής της επιστήμης.

Περίληψη

Η παρούσα εργασία εστιάζει στις μεθόδους κατηγοριοποίησης (Classification) δεδομένων πολλαπλών ετικέτων και στους αλγόριθμους μείωσης των δεδομένων (Data Reduction Algorithms). Στο παρελθόν έχουν προταθεί πολυάριθμοι αλγόριθμοι μείωσης δεδομένων. Οι αλγόριθμοι αυτοί μπορεί αν είναι είτε επιλογής (prototype selection) είτε παραγωγής (prototype generation) πρωτύπων και είναι κατάλληλοι για τον κατηγοριοποίητη κ εγγύτερων γειτόνων και για προβλήματα μόνης ετικέτας (single label classification). Οι αλγόριθμοι αυτοί δημιουργούν ένα μικρό σύνολο προτύπων που αντιπροσωπεύει όσο το δυνατόν περισσότερο τα αρχικά δεδομένα εκπαίδευσης. Αυτό το σετ ονομάζεται σετ συμπύκνωσης (condensing set) και έχει το πλεονέκτημα του χαμηλού υπολογιστικού κόστους, διατηρώντας παράλληλα την ακρίβεια σε υψηλά επίπεδα. Ωστόσο, οι αλγόριθμοι επιλογής και παραγωγής δεν είναι κατάλληλοι για προβλήματα με πολλές ετικέτες όπου ένα στιγμιότυπο μπορεί να ανήκει σε περισσότερες από μία κλάσεις. Η δημοφιλής μέθοδος μετατροπής Binary Relevance είναι ανεπαρκής για να συνδυαστεί με έναν αλγόριθμο επιλογής ή παραγωγής προτύπων λόγω των πολλαπλών δυαδικών συνόλων συμπύκνωσης που δημιουργεί. Η μείωση δεδομένων με διαμερισμό του χώρου είναι μια δηλοφιλής τεχνική παραγωγής προτύπων. Ο πιο γνωστός αλγόριθμος που ακολουθεί αυτή την τεχνική είναι ο αλγόριθμος RSP3. Ο αλγόριθμος αυτός είναι απλός και δεν απαιτεί από τον χρήστη ορίσει παραμέτρους. Η παρούσα εργασία προτείνει μια παραλλαγή του RSP3 για σύνολα δεδομένων εκπαίδευσης πολλαπλών ετικετών. Ο προτεινόμενος αλγόριθμος ονομάζεται MRSP3, κληρονομεί όλα τα χαρακτηριστικά του RSP3 και παράγει πρότυπα πολλαπλών ετικετών. Η πειραματική μελέτη που βασίστηκε σε εννέα σύνολα δεδομένων πολλαπλών ετικετών δείχνει ότι ο MRSP3 επιτυγχάνει υψηλούς ρυθμούς μείωσης χωρίς να επηρεάζει αρνητικά τις τιμές της hamming-loss. Στην πειραματική μελέτη, χρησιμοποιείται η μετρική Hamming-Loss, η οποία είναι ικανή να αξιολογήσει αλγορίθμους που εφαρμόζονται πάνω σε δεδομένα πολλαπλών ετικετών. Σύμφωνα με αυτήν, ο συνδυασμός του προτεινόμενου αλγόριθμου μείωσης δεδομένων MRSP3 με τον κατηγοριοποιητή BRk-NN αποδίδει θετικά, καθώς οι τιμές της μετρικής hamming-loss δεν επηρεάζονται ή μειώνονται ελάχιστα. Επιπλέον, ο MRSP3 καταφέρνει να μειώσει αρκετά τον όγκο των δεδομένων σε αρκετά σύνολα δεδομένων, αν όχι σε όλα, περισσότερο από το 50%.

«Data Reduction by Space Partitioning for multi-label data»

«Georgia Fotiadou»

Abstract

This paper focuses on multi-label data classification methods and data reduction algorithms. Numerous data reduction algorithms have been proposed in the past. These algorithms can be either prototype selection or prototype generation and are suitable for k nearest neighbors classifier and single label classification problems. These algorithms create a small set of prototypes that represent as much as possible the initial training data. This set is called condensing set and has the advantage of low computational cost, while maintaining high levels of accuracy. However, selection and generation algorithms are not suitable for multi-label problems where an item may belong to more than one class. The popular Binary Relevance transformation method is insufficient to combine with selection or generation algorithm due to the multiple binary condensing sets it creates. Data reduction by partitioning space is a popular generation prototypes technique. The best known algorithm that follows this technique is the RSP3 algorithm. This algorithm is simple and does not require the user to set parameters. This paper proposes a variant of RSP3 for multi-label training datasets. The proposed algorithm is called MRSP3, it inherits all the features of RSP3 and generates multi-labrl prototypes. The experimental study based on nine multi-label datasets shows that MRSP3 achieves high reduction rates without affecting hamming-loss negatively. In the experimental study, the Hamming-Loss metric is used, which is able to evaluate algorithms applied to multi-label data. According to that, the combination of the proposed MRSP3 data reduction algorithm with BRk-NN classifier performs positively, as the values of the hamming-loss metric are not affected or reduced slightly. In addition, MRSP3 manages to significantly reduce data volume in several, if not all, datasets more than 50%.

Ευχαριστίες

Αρχικά, ευχαριστώ θερμά τον κύριο Ουγιάρογλου Στέφανο, μέλος ΕΔΙΠ του Τμήματος Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου Ελλάδος, ο οποίος είναι ο επιβλέπων καθηγητής αυτής της πτυχιακής εργασίας, για την υπέροχη συνεργασία και την απόλυτη καθοδήγηση που έδωσε, ώστε να εκπονηθεί αυτή η εργασία. Εκτιμώ πολύ τον χρόνο που αφιέρωσε στην επίλυση προβλημάτων που αντιμετώπισα κατά τη διάρκεια συγγραφής και στην ολοκληρωμένη επεξήγηση της έρευνας μας, για να είναι πλήρως κατανοητή. Έπειτα, θέλω να ευχαριστήσω την οικογένεια μου και τα άτομα που με στήριξαν καθ' όλη τη διάρκεια της διεξαγωγής της έρευνας, καθώς οι μήνες αυτοί ήταν αρκετά πιεστικοί ψυχολογικά λόγω της πανδημίας που έχει ξεσπάσει εδώ και ένα χρόνο παγκοσμίως. Είναι σημαντικό να είμαστε ψυχολογικά πλήρεις και έτοιμοι να αφοσιωθούμε σε κάτι που μας ενδιαφέρει, ώστε να το κάνουμε με πλήρης επιτυχία και με δύναμη για γνώση και εφευρετικότητα.

Περιεχόμενα

Κατάλογος Σχημάτων

Κατάλογος Πινάκων

Κεφάλαιο 1ο: Εισαγωγή

1.1 Κατηγοριοποίηση

Η εξόρυξη δεδομένων είναι η επιστήμη που επικεντρώνεται στην ανάλυση μεγάλου όγκου δεδομένων για την εξαγωγή συμπερασμάτων, έπειτα από λεπτομερή επεξεργασία τους. Οι τεχνικες της εφαρμόζονται σε πολλούς τομείς της καθημερινότητας, όπως η ιατρική για την πρόβλεψη και διάγωνση διάφορων ασθενειών και παθήσεων, στην οικονομία για την ορθότερη ληψη αποφάσεων σχετικά με οικονομικά δεδομένα που ενισχύουν τις αγορές, την τηλεπικοινωνία για την βελτιστοποίηση της ποιότητας των υπηρεσιών κ.α. Γενικότερα, η εξόρυξη δεδομένων είναι μία επιστήμη, η οποία εξελίσσεται συνέχεια, καθώς ο όγκος των δεδομένων αυξάνεται καθημερινά με ταχείς ρυθμούς.

Κατηγοριοποίηση ενός στιγμιοτύπου είναι η κατάταξη του σε διάφορες κατηγορίες με βάση κάποια χαρακτηριστικά. Η συγκεκριμένη διαδικασία λαμβάνει χώρα σε πολλούς τομείς της καθημερινότητας όπως για παράδειγμα, στην ιατρική με το διαχωρισμό των καρκινικών κυττάρων σε καλοήθη ή κακοήθη, στην οικονομία με το διαχωρισμό των πολιτών ανάλογα με το εισόδημα τους, για να λάβουν επιδόματα κ.α. Όσον αφορά τον τομέα της Επιστήμης των Δεδομένων, η κατηγοριοποίηση αποτελεί σημαντικό κομμάτι, καθώς οι αλγόριθμοί της φέρουν μεγάλα ποσοστά απόδοσης και αποτελεσματικότητας στην κατηγοριοποίηση νέων αταξινόμητων δεδομένων.

Πιο συγκεκριμένα, ένας κατηγοριοποιητής διέρχεται από δύο φάσεις. Στην πρώτη φάση, τα στιγμιότυπα του συνόλου δεδομένων εκπαίδευσης(training set) φέρουν μια γνωστή ετικέτα κλάσης και κατηγοριοποιούνται σε προκαθορισμένες κλάσεις με βάση το χαρακτηριστικό της ετικέτας αυτής. Έτσι δημιουργείται ένα μοντέλο, το οποίο κατα τη δεύτερη φάση μπορεί να κατατάσσει οποιοδήποτε νέο στιγμιότυπο στην κατάλληλη κλάση [1]. Ένα απλό και κατανοητό παράδειγμα είναι η εκμάθηση αναγνώρισης της ελληνικής σημαίας σε ένα παιδί. Ο υπολογιστής βρίκεται στο ρόλο του παιδιού κι εμείς στο ρόλο του γονέα. Για να μάθουμε στο παιδί ποιά είναι η σημαία της Ελλάδας, του δείχνουμε αρκετές φωτογραφίες, που η καθεμία απεικονίζει μια διαφορετική σημαία. Όταν εμφανίζεται η ελληνική σημαία στην φωτογραφία του λέμε ”Ναι”, διαφορετικά του λέμε ”Όχι”. Αυτή η διαδικασία αποτελεί την πρώτη φάση της κατηγοριοποίησης η οποία ονομάζεται εκπαίδευση. Στη συνέχεια, αφού εκτελέσουμε την παραπάνω τεχνική πολλές φορές, φέρνουμε μία νέα φωτογραφία στο παιδί και το ρωτάμε αν είναι η ελληνική σημαία, κι αυτό απαντά με ”Ναι” ή ”Όχι”. Αυτό το βήμα αποτελεί την δεύτερη φάση και μαζί με αυτήν ολοκληρώνεται κι η διαδικασία της κατηγοριοποίησης.

Επιπλέον, η κατηγοριοποίηση ανήκει στην κατηγορία της εποπτεύομενης μάθησης, όπου οι ετικέτες κλάσης του συνόλου δεδομένων εκπαίδευσης είναι γνωστές και τα νέα στιγμιότυπα που εισέρχονται στο σύστημα χρησιμοποιούνται μαζί με την επιθυμητή ετικέτα. Υπάρχουν δύο τύποι κατηγοριοποιητών αυτής της κατηγορίας μάθησης, οι πρόθυμοι (eager) και οι οκνηροί (lazy), οι οποίοι και οι δύο στοχεύουν στην εξακρίβωση της ετικέτας του νέου στιγμιοτύπου, όμως λειτουργούν διαφορετικά. Οι οκνηροί κατηγοριοποιητές αποθηκεύουν απλά τα σύνολα δεδομένων εκπαίδευσης και περιμένουν μέχρι να εμφανιστούν νέα στιγμιότυπα χωρίς να δημιουργήσουν κάποιο μοντέλο. Όταν συμβαίνει αυτό, η κατηγοριοποίηση πραγματοποιείται με βάση τα πιο σχετικά δεδομένα στα αποθηκευμένα σύνολα δεδομένων εκπαίδευσης. Έτσι, οι οκνηροί κατηγοριοποιητές αφιερώνουν λιγότερο χρόνο στην εκπαίδευση αλλά περισσότερο χρόνο στις προβλέψεις. Σε αυτή την κατηγορία ταξινομητών ανήκουν οι k-nearest

neighbor, Case-based reasoning, Lazy Naive Bayes rules κ.α. Αντίθετα, οι πρόθυμοι κατηγοριοποιητές κατασκευάζουν ένα μοντέλο κατηγοριοποίησης με βάση το σύνολο δεδομένων εκπαίδευσης πριν λάβουν νέα στιγμιότυπα για κατηγοριοποίηση. Λόγω της κατασκευής του μοντέλου, αφιερώνουν περισσότερο χρόνο για την εκπαίδευση και λιγότερο χρόνο για πρόβλεψη. Μερικά παραδείγματα αυτού του τύπου κατηγοριοποιητή αποτελούν τα Δέντρα Αποφάσεων, Naive Bayes, Τεχνητά νευρωνικά δίκτυα κ.α.

Αναλυτικότερα, εάν συγκρίνουμε πρόθυμους με οκνηρούς κατηγοριοποιητές προκύπτουν κάποια χρήσιμα συμπεράσματα. Ένα από αυτά είναι πως οι πρόθυμοι κατηγοριοποιητές υπερισχύουν σε χρόνο και σε απαιτήσεις αποθηκευτικού χώρου, καθώς δημιουργούν μοντέλο κατηγοριοποίησης πριν την είσοδο νέου στιγμιοτύπου και δεν χρειάζονται τα δεδομένα εκπαίδευσης για την κατηγοριοποίηση αυτού. Αντίθετα, οι οκνηροί κατηγοριοποιητές υπερισχύουν σε ευρύ φάσμα υποθέσεων που μπορούν να καλυψουν, γιατί έχουν πάντα διαθέσιμο ολόκληρο το σύνολο δεδομένων εκπαίδευσης.

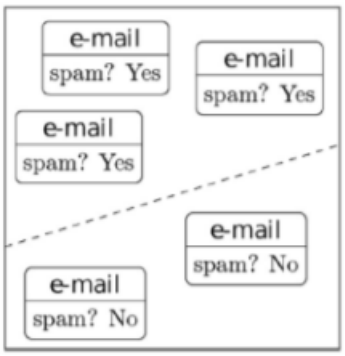
Τα τελευταία χρόνια, το φαινόμενο της κατηγοριοποίησης έχει απασχολήσει πολλές ερευνητικές ομάδες από διάφορους τομείς της επιστήμης. Γι' αυτό, έχουν αναπτυχθέι υλοποιήσεις ποικίλων πρόθυμων και οκνηρών κατηγοριοποιητών που έχουν γίνει ευρέως γνωστοί κι είναι αποκτήσιμοι.

Μια από τις πιο αναγνωρισμένες υποκατηγορίες πρόθυμων κατηγοριοποιητών αποτελούν τα δέντρα απόφασης. Ένα δέντρο απόφασης [2] δημιουργεί μοντέλα κατηγοριοποίησης με τη μορφή μιας δομής δέντρου. Η ιδέα που υιοθετείται σε αυτό τον τύπο κατηγοριοποίησηης είναι η κατάτμηση του χώρου σε ορθογώνιες περιοχές συνεχώς μέχρις ότου γίνουν ομοιογενείς είτε μέχρι να φτάσουμε σε ένα μέγιστο βάθος κατάτμησης. Ομοιογενής περιοχή θεωρείται μια περιοχή, όπου όλα τα δείγματα προέρχονται από την ίδια κλάση. Άλλη μία υποκατηγορία αυτών των κατηγοριοποιητών είναι αυτοί που βασίζονται στα τεχνητά νευρωνικά δίκτυα. Τα τεχνητά νευρωνικά δίκτυα [3] είναι μια προσπάθεια αποτύπωσης του ανθρώπινου εγκεφάλου σε υπολογιστική μορφή, όπου οι υπολογιστικοί νευρώνες αποτελούνται από δεδομένα εισόδου με τις συνάψεις, έναν αθροιστή και μια συνάρτηση ενεργοποίησης. Αυτή η τεχνική έχει εφαρμοστεί σε πολλούς τομείς όπως οικονομία, ιατρική, βιομηχανία κ.α. Τέλος, οι πιθανολογικοί κατηγοριοποιητές ανήκουν στους πρόθυμους κατηγοριοποιητές κι ένας από τους πιο γνωστούς είναι ο Naive Bayes [4]. Ειδικότερα, όλοι οι αλγόριθμοι που εκπαιδεύουν αυτόν τον κατηγοριοποιητή υιοθετούν μια κοινή αρχή: ότι η τιμή ενός συγκεκριμένου χαρακτηριστικού είναι ανεξάρτητη από την τιμή οποιουδήποτε άλλου χαρακτηριστικού, δεδομένης της μεταβλητής κλάσης. Για παράδειγμα, ένα φρούτο μπορεί να θεωρηθεί μήλο εάν είναι κόκκινο, στρογγυλό και περίπου 10 cm σε διάμετρο. Ένας τέτοιος κατηγοριοποιητής θεωρεί ότι κάθε ένα από αυτά τα χαρακτηριστικά συμβάλλει ανεξάρτητα στην πιθανότητα ότι αυτό το φρούτο είναι ένα μήλο.

Όσον αφορά τους οκνηρούς κατηγοριοποιητές, ένας από τους πιο γνωστούς είναι ο k-εγγύτερων γειτόνων (k-NN). Περισσότερες λεπτομέρειες για τον συγκεκριμένο κατηγοριοποιητή θα δούμε στα επόμενα κεφάλαια, καθώς αποτελεί κύριο κομμάτι αυτής της εργασίας.

1.2 Κατηγορίες προβλημάτων κατηγοριοποίησης

Στη συγκεκριμένη ενότητα, αναλύονται οι τρεις κατηγορίες προβλημάτων κατηγοριοποίησης που έχουν προταθεί μέχρι σήμερα κι αποτελούν αντικείμενα ανάλυσης πολλών ερευνητών. Κατα σειρά ανάλυσης είναι τα Δυαδικα Προβλήματα (Binary classification problems), τα Προβλήματα πολλών κατηγοριών



Σχήμα 1.1: Binary Classification

(Multi-class classification problems) και τα Προβλήματα πολλαπλών ετικετών (Multi-label classification problems).

1.2.1 Δυαδικά προβλήματα

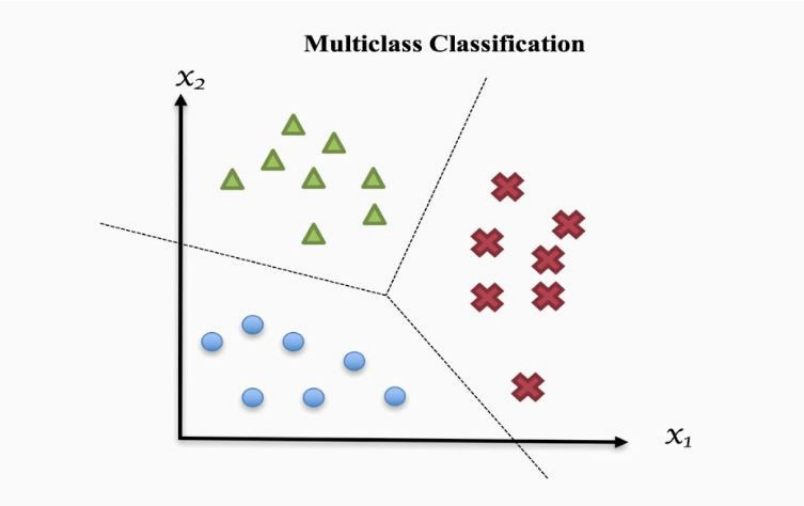
Όσον αφορά τα Δυαδικά προβλήματα, σε αυτά ανήκουν είδη προβλημάτων που παρουσιάζουν δύο ετικέτες κλάσης, όπου μπορούν να κατανεμηθούν τα στιγμιότυπα με βάση τα χαρακτηριστικά τους. Για παράδειγμα, η ανίχνευση ανεπιθύμητων μηνυμάτων ηλεκτρονικού ταχυδρομείου(”spam”, ”not spam”), πρόβλεψη για την καύση (”churn”, ”not churn”), η ανίχνευση καρκίνου(“cancer detected”, “cancer not detected”) κ.α.

Στην παραπάνω εικόνα 1.1, παρουσιάζεται ένα δυαδικό πρόβλημα κατηγοριοποιησης που αφορά την κατάταξη ενός μηνύματος ηλεκτρονικού ταχυδρομείου. Οι δύο κλάσεις που μπορεί να ανήκει ένα στιγμιότυπο του συγκεκριμένου συστήματος είναι ”spam” και ”not spam”. Επίσης, οι διακεκομμένες γραμμές αναπαριστούν τα όρια μεταξύ των κλάσεων. Συνήθως, σε αυτού του είδους προβλήματα οι δύο ετικέτες αφορούν την ίδια κατάσταση και είναι αντίθετες μεταξύ τους, όπως στο συγκεκριμένο πρόβλημα ένα e-mail αν είναι spam , δηλαδή αν ανήκει στην 1η κλάση, αυτομάτως αυτό σημαίνει ότι δεν ανήκει στην δεύτερη κλάση. Άρα, οι δύο κλάσεις είναι εννοιλογικά αντίθετες.

Τόσο στη μηχανική μάθηση όσο και στην εξόρυξη δεδομένων, έχουν αναπτυχθεί πολλές μέθοδοι για δυαδική κατηγοριοποίηση όπως Support Vector Machines(SVM) [5], Δέντρα Απόφασης, Δίκτυα Bayesian κ.α.

1.2.2 Προβλήματα πολλών κατηγοριών

Σε αυτήν την υποενότητα αναλύεται η δεύτερη κατηγορία προβλημάτων κατηγοριοποίησης, τα προβλήματα πολλών κατηγοριών (Multi-class Classification Problems). Ένα πρόβλημα κατηγοριοποίησης πολλών κατηγοριών παρουσιάζει διαφορετική πρόκληση από ένα πρόβλημα δυαδικής κατηγοριοποίησης και αποτελεί κατά κάποιο τρόπο επέκταση αυτού. Ειδικότερα, καλύπτει τις περιπτώσεις, όπου οι ετικέτες κλάσης είναι περισσότερες από δύο και τα στιγμιότυπα μπορούν να ανήκουν σε μία και μόνο κλάση. Για παράδειγμα, ένα σύνολο φρούτων μπορεί να είναι πορτοκάλια, μήλα ή αχλάδια. Η κατηγοριοποίηση



Σχήμα 1.2: Multi-class Classification

πολλαπλών κατηγοριών κάνει την υπόθεση ότι κάθε δείγμα αντιστοιχεί σε μία και μόνο μία ετικέτα: ένα φρούτο μπορεί να είναι είτε μήλο είτε αχλάδι, αλλά όχι και τα δύο ταυτόχρονα.

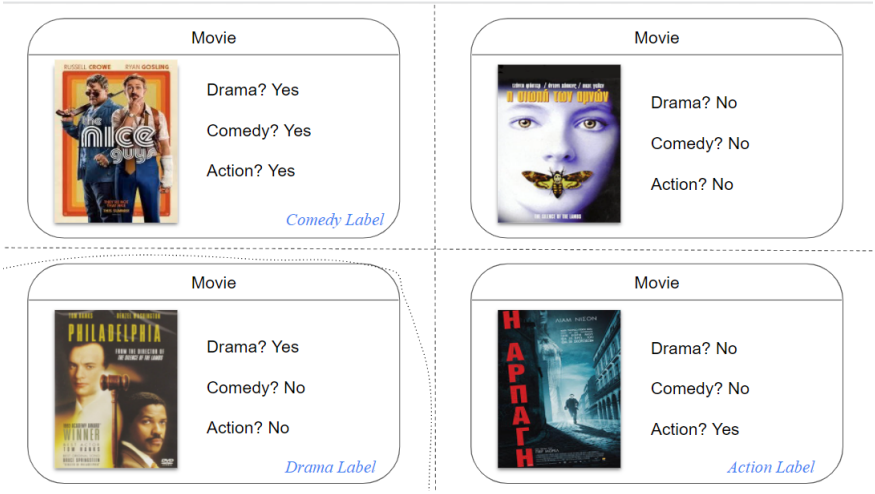
Παραπάνω στην εικόνα 1.2, φαίνεται η αναπαράσταση ενός multi-class κατηγοριοποιητή. Παρατηρείται πως υπάρχουν 3 διαφορετικές κλάσεις στον δισδιάστατο χώρο, όπου είναι τα τρίγωνα, τα σταυρουδάκια και οι κύκλοι, των οποίων τα όρια διαχωρίζονται από τις διακεκκομένες γραμμές. Επίσης, κάθε στιγμιότυπο μπορεί να ανήκει σε μία και μόνο μία κλάση, δηλαδή μπορεί να είναι ή κύκλος ή τετράγωνο ή σταυρουδάκι.

1.2.3 Προβλήματα πολλαπλών ετικετών

Η τελευταία και κυριότερη, για την παρούσα εργασία, κατηγορία είναι τα προβλήματα πολλαπλών ετικετών. Σε αυτού του είδους κατηγοριοποίηση, εξετάζονται προβλήματα που φέρουν παραπάνω από δύο ετικέτες κλάσης και κάθε στιγμιότυπο μπορεί να ανήκει σε καμία, μία ή και περισσότερες από αυτές. Το πλαίσιο πολλαπλών ετικετών λαμβάνει αυξημένη προσοχή και αφορά ευρύ φάσμα τομέων, συμπεριλαμβανομένης της κατηγοριοποίησης κειμένου, σκηνών, βίντεο και βιοπληροφορικής. Αυτό συμβαίνει, γιατί αποτελεί κατά κάποιο τρόπο επέκταση των δυαδικών και πολλών κατηγοριών κατηγοριοποιητών.

Στην παρακάτω εικόνα 1.3, παρουσιάζεται ένα απλό πρόβλημα πολλαπλών ετικετών, όπου κατηγοριοποιείται κάθε ταινία στο είδος της. Αυτό που είναι άξιο να παρατηρηθεί είναι το γεγονός ότι μια ταινία μπορεί είτε να ανήκει σε ένα μοναδικό είδος (μία μόνο κλάση), όπως η ταινία ”Philadelphia”, είτε να ανήκει και στα τρία είδη (τρεις κλάσεις), όπως η ταινία ”The Nice Guys”, είτε σε κανένα, όπως η ταινία ”Η σιωπή των αμνών”.

Μπορούμε να ομαδοποιήσουμε τις υπάρχουσες μεθόδους κατηγοριοποίησης πολλαπλών ετικετών σε δύο κύριες κατηγορίες: α) μέθοδοι μετασχηματισμού προβλήματος και β) μέθοδοι προσαρμογής αλγορίθμου. Ειδικότερα, μέθοδοι μετασχηματισμού προβλήματος ορίζονται οι μέθοδοι που μετασχηματίζουν το πρόβλημα κατηγοριοποίησης πολλαπλών ετικετών σε ένα ή περισσότερα προβλήματα κατηγοριοποί-



Σχήμα 1.3: Multi-Label Classification

ησης μονής ετικέτας ή παλινδρόμησης. Από την άλλη, οι μέθοδοι προσαρμογής αλγορίθμου επεκτείνουν συγκεκριμένους αλγόριθμους, προκειμένου να χειριστούν δεδομένα πολλαπών ετικετών απευθείας. Στο επόμενο Κεφάλαιο, αναλύονται λεπτομερώς και οι δύο κατηγορίες που προαναφέρθηκαν.

1.3 Ο αλγόριθμος κατηγοριοποίησης k εγγύτερων γειτόνων

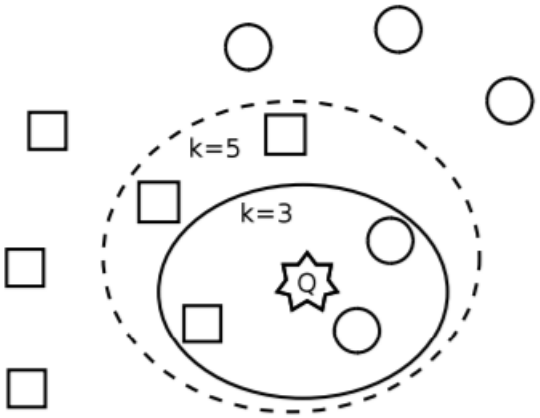
Ο αλγόριθμος k-εγγύτερων γειτόνων(k-NN) αποτελεί έναν από τους πιο απλούς, κατανοητούς και εύκολους στη χρήση επιβλεπόμενης μάθησης αλγόριθμους. Επιπλέον, είναι ικανός να αντιμετωπίσει προβλήματα τόσο κατηγοριοποίησης όσο και παλινδρόμησης(regression). Η συγκεκριμένη τεχνική εφαρμόζεται στη μηχανική μάθηση και στην εξόρυξη δεδομένων για λήψη αποφάσεων, καθώς εμφανίζει υψηλή αποτελεσματικότητα [6].

Όπως αναφέρθηκε και σε προηγούμενη ενότητα, ο k-NN ανήκει στους οκνηρούς κατηγοριοποιητές, επομένως δεν χτίζει κάποιο μοντέλο κατηγοριοποίησης και χρησιμοποιεί τα δεδομένα εκπαίδευσης για να κατηγοριοποιήσει νέα αταξινόμητα δεδομένα. Αναλυτικότερα, για την κατηγοριοποίηση ενός στιγμιοτύπου x εξετάζονται οι k εγγύτεροι γείτονες του, για τον προσδιορισμό της κλάσης του. Αυτό γίνεται μέσω της διαδικασίας της πλειοψηφίας, όπου ως ετικέτα κλάσης του x ορίζεται ως η πλειοψηφούσα κλάση των k πλησιέστερων γειτόνων του, οι οποίοι ανακτώνται βάσει μιας μετρικής απόστασης.

Σημειώνεται ότι, όταν k=1 τότε ο αλγόριθμος είναι γνωστός ως κατηγοριοποιητής πλησιέστερου γείτονα (1-NN rule). Πιο συγκεκριμένα, το στιγμιότυπο x κατατάσσεται στην ίδια κλάση με τον πιο κοντινό του γείτονα. Καθώς το μέγεθος του συνόλου δεδομένων εκπαίδευσης πλησιάζει το άπειρο, ο κατηγοριοποιητής πλησιέστερου γείτονα εγγυάται ποσοστό σφάλματος μικρότερο από το διπλάσιο του ποσοστού σφάλματος Bayes.

Η παρακάτω εικόνα 1.4, αναπαριστά ένα δυαδικό παράδειγμα κατηγοριοποίησης με τη χρήση του k-NN. Αναλυτικότερα, υπάρχουν δύο ετικέτες σε αυτό το σύνολο δεδομένων, οι κύκλοι και τα τετράγωνα και ένα αταξινόμητο στιγμιότυπο (Q), που θα πρέπει να εξακριβωθεί η κλάση του. Αν θέσουμε το k να ισούται με 3 (συνεχόμενη γραμμή κύκλου στο σχήμα), τότε το στιγμιότυπο Q θα καταταχθεί στην κλάση κύκλος, γιατί δύο από τους τρεις εγγύτερους γείτονες του είναι κύκλοι. Διαφορετικά, εάν θέσουμε το k

να ισούται με 5 (διακεκομμένη γραμμή κύκλου στο σχήμα), τότε το στιγμιότυπο Q θα καταταχθεί στην κλάση τετράγωνο, γιατί τρεις απο τους πέντε εγγύτερους γείτονες του είναι τετράγωνα [7].



Σχήμα 1.4: Παράδειγμα k-NN κατηγοριοποιητή για k=3 και k=5

Η απόδοση κατηγοριοποίησης με βάση τον k-NN εξαρτάται κυρίως από την τιμή της παραμέτρου k. Η υψηλότερη ακρίβεια κατηγοριοποίησης που μπορεί να επιτευχθεί λόγω του k, εξαρτάται από το διαθέσιμο σύνολο δεδομένων και ο καθορισμός του, συχνά απαιτεί δαπανηρές εργασίες προεπεξεργασίας δοκιμών και σφαλμάτων(trial-and-error). Αν και ο καθορισμός του k δεν ακολουθεί κάποιο γενικό κανόνα και η ”βέλτιστη” τιμή του μπορεί να διαφέρει τελείως από ένα σύνολο δεδομένων σε ένα άλλο, σύνολα δεδομένων με θόρυβο απαιτούν μεγαλύτερες τιμές του k, για να εξετάζονται περισσότεροι εγγύτεροι γείτονες. Ωστόσο, με αυτόν τον τρόπο δεν μπορούν να οριστούν επακριβώς τα όρια μεταξύ των διακριτών κλάσεων. Από την άλλη, ένας κατηγοριοποιητής καθίσταται ευαίσθητος στο θόρυβο με μικρότερες τιμές της παραμέτρου k. Επομένως, η κατηγοριοποίηση γίνεται περισσότερο ανακριβής σε περιπτώσεις, όπου στο σύνολο δεδομένων περιέχεται θόρυβος [7].

Είναι άξιο να αναφερθεί ότι, επειδή ο κατηγοριοποιητής k-NN χρησιμοποιεί μία μοναδική τιμή του k για ολόκληρη την κατηγοριοποίηση, ακόμα και η ”καλύτερη” τιμή του να βρεθεί, δεν θα είναι η ιδανική. Δηλαδή, διαφορετικές τιμές του k θεωρούνται ιδανικές σε διαφορετικό μέρος του χώρου δεδομένων. Συνεπώς, πολλές έρευνες για δυναμικό προσδιορισμό του k μπορούν να αναπτυχθούν, για να επιτύχουν μεγαλύτερη ακρίβεια από τον κατηγοριοποιητή k-NN με ιδανικότερο προσδιορισμό του k [8, 9].

Όσον αφορά, τα δυαδικά προβλήματα κατηγοριοποίησης (σύνολα δεδομένων με δύο κλάσεις), η τιμή της παραμέτρου k πρέπει να είναι μονός αριθμός, για να μην υπάρξει περίπτωση ισοψηφίας κατά την ψηφοφορία των εγγύτερων γειτόνων. Αντίθετα, στα μη δυαδικά προβλήματα, η τιμή του k μπορεί να είναι οποιαδήποτε. Σε αυτές τις περιπτώσεις, ο αλγόριθμος επιλέγει είτε μια τυχαία κλάση από αυτές που πλειοψήφησαν είτε την κλάση του πλησιέστερου γείτονα. Το γνωστό λογισμικό Weka [10] και πολλά άλλα εργαλεία λογισμικού εξόρυξης δεδομένων και μηχανικής μάθησης επιλέγουν να χρησιμοποιούν την πρώτη επιλογή.

Ακόμα ένα κύριο θέμα που θα πρέπει να διευθετυθεί είναι η επιλογή της μετρικής για τον υπολογισμό των αποστάσεων μεταξύ των στιγμιοτύπων. Βέβαια, οι τύποι δεδομένων των χαρακτηριστικών του συνόλου(attributes) θα πρέπει να υπάρχουν κατά νου για τη λήψη της απόφασης αυτής. Η Ευκλείδια απόσταση εφαρμόζεται κυρίως σε περιπτώσεις, όπου οι τύποι είναι πραγματικοί ή ακέραιοι αριθμοί. Ω-

στόσο, μπορούν να εφαρμοστούν κι άλλες μετρικές απόστασης όπως για παράδειγμα οι Mahalanobis, Manhatan, Minkowski, Chebyshev κ.α [11]. Ποικίλες μετρικές έχουν αναπτυχθεί και για την οργάνωση ονομαστικών χαρακτηριστικών (μη μετρικών χώρων). Ωστόσο, στη συγκεκριμένη εργασία τα πειράματα εφαρμόστηκαν πάνω σε σύνολα δεδομένων με χαρακτηριστικά να αποτελούν ακέραιους ή πραγματικούς αριθμούς. Για αυτό το λογο, ως μετρική απόστασης χρησιμοποιήθηκε η Ευκλείδια απόσταση. Συνεπώς, τα στιγμιότυπα που αποτελούνται από (*n*) χαρακτηριστικά, θεωρούνται σημεία δεδομένων (ή διανύσματα) στον n-διάστασεων ευκλείδειο μετρικό χώρο, και η ευκλείδεια απόσταση μεταξύ των σημείων *p* και *q* δίνεται από τον τύπο 1.1 [12].

$$d(p,q) = d(q,p) = \sqrt{(q_1{-}p_1)^2 + (q_2{-}p_2)^2 + ... + (q_n{-}p_n)^2} = \sqrt{\sum_{i=1}^{n}(q_i{-}p_i)^2} \tag{1.1}$$

Μια σημαντική παρατήρηση είναι ότι η τιμή απόστασης μπορεί να επηρεαστεί από τις διαφορετικές τιμές των χαρακτηριστικών στα στιγμιότυπα. Ακόμα και σε περιπτώσεις, όπου όλα τα χαρακτηριστικά έχουν την ίδια σημασία, τα χαρακτηριστικά με χαμηλότερο εύρος έχουν μικρότερη βαρύτητα στη μέτρηση της ευκλείδιας απόστασης, από τα χαρακτηριστικά με μεγαλύτερο εύρος τιμών. Για παράδειγμα, ας υποθέσουμε ότι το χαρακτηριστικό "μισθός" περιέχει τιμές μεταξύ 800 με 5000 και το χαρακτηριστικό "αριθμός παιδιών" περιέχει τιμές μεταξύ 0 με 6. Με τα δύο χαρακτηριστικά να έχουν την ίδια σημασία, το χαρακτηριστικό "μισθός" έχει μεγαλύτερη επιρροή στον υπολογισμό της απόστασης από το "αριθμός παιδιών". Ως εκ τούτου, θα πρέπει να γίνει κανονικοποίηση στις τιμές των χαρακτηριστικών σε ένα συγκεκριμένο αριθμητικό διάστημα(π.χ. [0, 1]). Υποθέτωντας ότι, το σύνολο δεδομένων συμπεριλαμβάνει (*n*) στιγμιότυπα και ένα χαρακτηριστικό (*e*) πρέπει να κανονικοποιηθεί στο διάστημα[0,1]. Η τιμή του χαρακτηριστικού του i-στου στιγμιότυπου i = 1, ...n κανονικοποιείται ως εξής:

$$normalized(e_i) = \frac{e_i{-}Emin}{Emax{-}Emin} \tag{1.2}$$

όπου *Emin* η μικρότερη τιμή του χαρακτηριστικού *e* και *Emax* η υψηλότερη. Η κανονικοποίηση δεδομένων είναι μια κοινή διαδικασία προεπεξεργασίας σε πολλές εργασίες εξόρυξης δεδομένων [7].

Με τον καιρό, αρκετές παραλλαγές του k-NN κατηγοριοποιητή έχουν προταθεί. Η μία και πιο σημαντική είναι ο σταθμισμένης-απόστασης(distance-weighted) k-NN κανόνας, που χρησιμοποιεί μία σταθμισμένης-απόστασης(distance-weight) μέθοδο για να δώσει μεγαλύτερο βάρος στους κοντινότερους γείτονες απ' ότι στους μακρυνότερους. Ο κοντινότερος γείτονας έχει βάρος ένα, ενώ ο μακρυνότερος μηδέν. Τα βάρη όλων των υπόλοιπων γειτόνων κυμαίνονται σε αυτό το διάστημα. Έτσι, ένα νέο στιγμιότυπο κατηγοριοποιείται με βάση μια σταθμισμένη ψηφοφορία: καταχωρείται στην κλάση με το μεγαλύτερο άθροισμα βαρών.

1.4 Μειονεκτήματα του αλγορίθμου κατηγοριοποίησης k εγγύτερων γειτόνων

Όπως έχει ήδη αναφερθεί, ο k-NN κατηγοριοποιητής παρουσιάζει μεγάλη αποτελματικότητα και απόδοση, καθώς και πολλά άλλα πλεονεκτήματα. Μερικά από αυτά είναι ο μηδενικός χρόνος που αφιερώνει κατά την εκπαίδευση των στιγμιοτύπων, τα νέα δεδομένα μπορούν να προστεθούν απρόσκοπτα που δεν

θα επηρεάσουν την ακρίβεια του αλγορίθμου, είναι πολύ εύκολο να εφαρμοστεί, καθώς απαιτούνται μόνο δύο παράμετροι, η τιμή του k και η μετρική απόστασης [13]. Παρ'όλα αυτά, υπάρχουν μειονεκτήματα, που σε κάποιες περιπτώσεις τον καθιστούν αναποτελεσματικό.

Ένα από τα κύρια θέματά του είναι το υψηλό υπολογιστικό κόστος που παρουσιάζει. Αυτό συμβαίνει, γιατί ο k-NN ως αλγόριθμος υπολογίζει όλες τις αποστάσεις μεταξύ των αταξινόμητων στιγμιοτύπων με αυτών που περιλαμβάνει το σύνολο δεδομένων εκπαίδευσης. Όταν το σύνολο δεδομένων είναι αρκετά μεγάλο, τότε το μειονέκτημα αυτό καθιστά τη χρήση του πολλές φορές χρονοβόρα και απαγορευτική. Για παράδειγμα, υποθέτωντας ότι σε ένα σύστημα κατηγοριοποίησης είναι αποθηκευμένα 100.000 στιγμιότυπα εκπαίδευσης και θα πρέπει να κατηγοριοποιηθούν 10.000 αταξινόμητα στιγμιότυπα με τη χρήση του k-NN. Εφαρμόζοντας απλά μαθηματικά, προκύπτει το συμπέρασμα ότι ο κατηγοριοποιητής θα πρέπει να υπολογίσει 1 δισεκατομμύριο αποστάσεις. Αν και στις μέρες μας, τα υπολογιστικά συστήματα είναι εφοδιασμένα με δυνατούς επεξεργαστές, τέτοιου είδους υπολογισμοί δεν παύουν να είναι χρονοβόροι και απαράδεκτοι σε περιπτώσεις που ο διαθέσιμος χρόνος είναι περιορισμένος. Όσον αφορα το υπολογιστικό κόστος, δεν εξαρτάται μόνο από το μέγεθος του συνόλου δεδομένων εκπαίδευσης, αλλά και από τις διαστάσεις του. Όσο υψηλότερη είναι η διάσταση των δεδομένων, τόσο περισσότεροι είναι οι υπολογισμοί που εκτελούνται για τον υπολογισμό της απόστασης.

Ακόμη ένα μειονέκτημα του k-NN κατηγοριοποιητή είναι η υψηλή απαίτηση αποθηκευτικού χώρου, για τα δεδομένα εκπαίδευσης. Αντίθετα από τους πρόθυμους κατηγοριοποιητές, οι οποίοι έπειτα από την κατασκευή του μοντέλου κατηγοριοποίησης μπορούν να αφαιρέσουν το σύνολο δεδομένων εκπαίδευσης, ο k-NN χρειάζεται τα δεδομένα αυτά καθ'όλη τη διάρκεια εκτέλεσης του. Για το λόγο αυτό, υπολογιστικά συστήματα με μεγάλη κύρια μνήμη είναι απαραίτητα, αν όχι ιδανικά για τη σωστή και αποτελεσματική λειτουργία του k-NN, για την αποθήκευση των δεδομένων εκπαίδευσης.

Μία τελευταία αδυναμία του k-NN, αλλά εξίσου σημαντική είναι η ευαισθησία σε θόρυβο που παρουσιάζει, όπως και πολλοί άλλοι αλγόριθμοι. Πιο συγκεκριμένα, η ποιότητα των δεδομένων εκπαίδευσης επηρεάζει σε μεγάλο βαθμό την ακρίβεια. Δεδομένα θορύβου και εσφαλμένης σήμανσης, καθώς και ακραίες τιμές και αλληλεπικαλύψεις μεταξύ των περιοχών δεδομένων διαφορετικών κατηγοριών, οδηγούν σε λιγότερο ακριβή κατηγοριοποίηση. Μία λύση στο πρόβλημα αυτό αποτελεί η χρήση υψηλών τιμών της παραμέτρου k, ώστε να εξετάζονται μεγαλύτερες γειτονιές. Ωστόσο αυτή η μέθοδος δεν θεωρείται η ιδανική, εφόσον υποδηλώνει μεγάλο αριθμό εκτελέσης δοκιμών και σφαλμάτων(trial-and-error) για τον καθορισμό της κατάλληλης τιμής k [7].

Αυτές οι αδυναμίες αποτελούν ενεργό ερευνητικό πρόβλημα και έχουν προσελκύσει το ενδιαφέρον της ερευνητικής κοινότητας εξόρυξης δεδομένων. Σε περιπτώσεις κατηγοριοποίησης μονής ετικέτας, οι παραπάνω αδυναμίες μπορούν να καλυφθούν με τη εφαρμογή κάποιας τεχνική μείωσης δεδομένων εκπαίδευσης ή κάποιας μεθόδου δεικτοδότησης. Στην επόμενη ενότητα, αναλύονται οι τεχνικές μείωσης δεδμένων, οι οποίες αποτελούν κύριο αντικείμενο ανάλυσης στη συγκεκριμένη εργασία.

1.5 Τεχνικές μείωσης δεδομένων (DRT)

Όπως έχει ήδη αναφερθεί στην προηγούμεη ενότητα, οι Τεχνικές μείωσης δεδομένων(DRT) μπορούν να εφαρμοστούν για να βελτιστοποιήσουν την απόδοση του k-NN κατηογριοποιητή. Γενικότερα, η μείωση

δεδομένων είναι μια διαδικασία που μειώνει τον όγκο των αρχικών δεδομένων και τα αντιπροσωπεύει σε πολύ μικρότερο όγκο. Οι τεχνικές αυτές έχουν νόημα, εφόσον ο χρόνος που απαιτείται για τη μείωση δεδομένων δεν επισκιάζει τον χρόνο που εξοικονομείται από την εξόρυξη δεδομένων στο μειωμένο σύνολο δεδομένων.

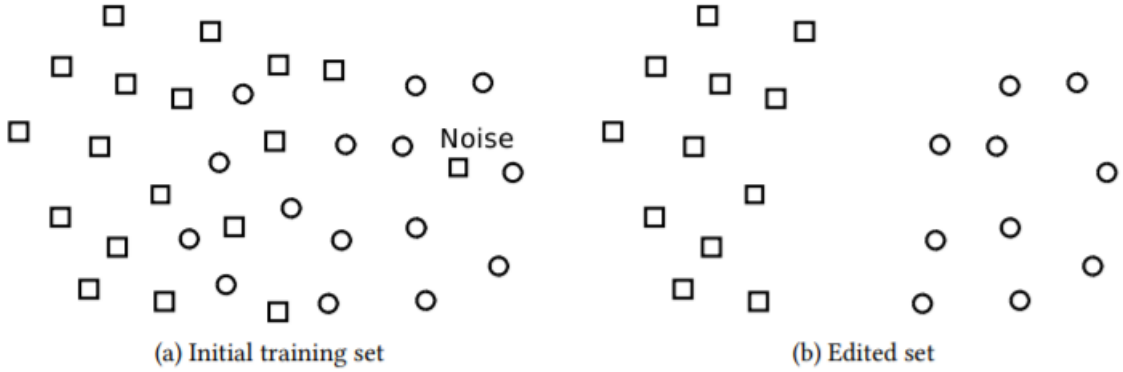
Πιο κατανοητά, η συλλογή δεδομένων από διαφορετικές αποθήκες δεδομένων για ανάλυση, οδηγεί σε τεράστιο όγκο δεδομένων. Είναι δύσκολο για έναν αναλυτή δεδομένων να χειριστεί αυτόν τον μεγάλο όγκο δεδομένων. Επίσης, είναι δύσκολο να εκτελεστούν σύνθετα ερωτήματα σχετικά με τον τεράστιο όγκο δεδομένων, καθώς χρειάζεται πολύς χρόνος και μερικές φορές καθίσταται ακόμη και αδύνατο να παρατηρούνται τα επιθυμητά δεδομένα. Γι' αυτό η μείωση των δεδομένων γίνεται αναγκαία. Επιπλέον, δεν επηρεάζει το αποτέλεσμα που λαμβάνεται από την εξόρυξη δεδομένων που σημαίνει ότι το αποτέλεσμα της εξόρυξης δεδομένων πριν και μετά από τη μείωση δεδομένων είναι το ίδιο (ή σχεδόν το ίδιο). Η μόνη διαφορά εμφανίζεται στην αποτελεσματικότητα της εξόρυξης δεδομένων, καθώς η ίδια αυξάνεται [14].

Υπάρχουν δύο κύριες κατηγορίες των τεχνικών μείωσης δεδομένων που αφορούν την κατηγοριοποίηση: (i) Μείωση στιγμιοτύπων (item reduction) και (ii) Μείωση διαστάσεων (dimensionality reduction). Αυτή η εργασία αφορά την πρώτη κατηγορία. Επίσης, οι αλγόριθμοι μείωσης στιγμιοτύπων χωρίζονται σε δύο ομάδες: i) αλγόριθμοι επιλογής προτύπων και (ii) αλγόριθμοι παραγωγής προτύπων. Οι αλγόριθμοι επιλογής προτύπων επιλέγουν κάποια στιγμιότυπα από ολόκληρο το σύνολο δεδομένων εκπαίδευσης, τα οποία ορίζονται ως αντιπροσωπευτικά (ή πρότυπα), ενώ οι αλγόριθμοι παραγωγής προτύπων συνοψίζουν παρόμοια στιγμιότυπα εκπαίδευσης και δημιουργούν νέα αντικείμενα, τα οποία χρησιμοποιούν ως πρότυπα. Στην πραγματικότητα, συγκεκριμένες περιοχές δεδομένων του πολυδιάστατου χώρου αντιπροσωπεύονται από ένα πρότυπο. Στόχος και των δύο αλγόριθμων είναι να επιλέξουν μόνο τα στιγμιότυπα που βρίσκονται κοντά στα όρια των κλάσεων.

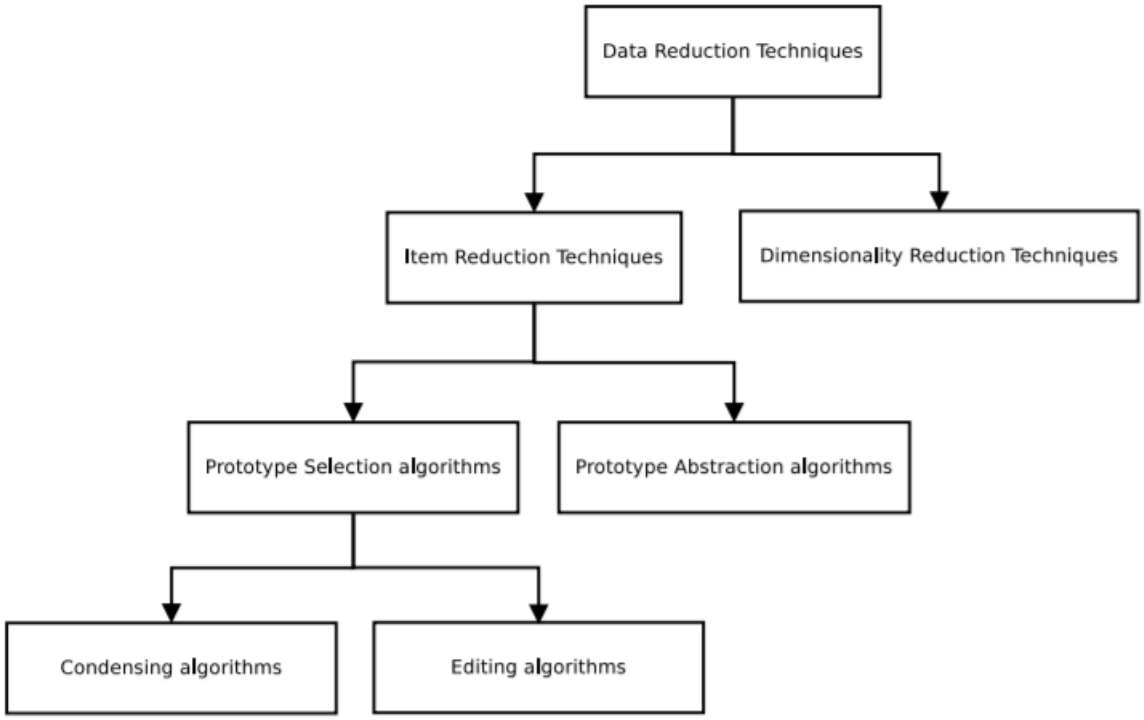
Οι αλγόριθμοι επιλογής προτύπων χωρίζονται σε δύο υποκατηγορίες: (i) αλγόριθμοι συμπύκνωσης (condensing) και (ii) αλγόριθμοι επεξεργασίας (editing). Οι πρώτοι έχουν το ίδιο κίνητρο με τους αλγόριθμους παραγωγής προτύπων. Έχουν ως στόχο, τη δημιουργία μικρών αντιπροσωπευτικών συνόλων των αρχικών δεδομένων εκπαίδευσης, τα οποία καλούνται συνήθως σύνολα συμπύκνωσης(condensing set). Η χρήση ενός συνόλου συμπύκνωσης έχει τα πλεονεκτήματα του χαμηλού κόστους, υπολογιστικών απαιτήσεων και αποθήκευσης, ενώ δεν επηρεάζει αρνητικά την ακρίβεια κατηγοριοποίησης. Από την άλλη πλευρά, οι αλγόριθμοι επεξεργασίας στοχεύουν στη βελτίωση της ακρίβειας παρά στην επίτευξη υψηλών ποσοστών μείωσης. Αυτό επιτυγχάνεται, με την προσπάθεια βελτίωσης της ποιότητας των δεδομένων εκπαίδευσης αφαιρώντας το θόρυβο, τα ακραία στιγμιότυπα, τα στιγμιότυπα που δεν φέρουν ετικέτα και εξομαλύνοντας τα όρια των αποφάσεων μεταξύ κλάσεων (βλέπε σχήμα 1.5).

Ιδανικά, ένας αλγόριθμος επεξεργασίας δημιουργεί ένα επεξεργασμένο σύνολο εκπαίδευσης χωρίς αλληλεπικάλυψη μεταξύ των κλάσεων. Το σχήμα 1.6 βοηθά στην κατανόηση της ιεραρχίας όλων των κατηγοριών που έχουν αναφερθεί στη συγκεκριμένη ενότητα. Άξιο αναφοράς αποτελεί το γεγονός, ότι ορισμένοι αλγόριθμοι συμπύκνωσης υιοθετούν την ιδέα της επεξεργασίας και ονομάζονται υβριδικοί.

Κάποια από τα κριτήρια που μπορούν να αξιολογήσουν τις τεχνικές μείωσης δεδομένων είναι το ποσοστό μείωσης (reduction rate), η ακρίβεια κατηγοριοποίησης (accuracy) και το υπολογιστικό κόστος προεπεξεργασίας (preprocessing computational cost). Όσον αφορά το ποσοστό μείωσης, εξηγεί πόσο



Σχήμα 1.5: Εξομάλυνση ορίων αποφάσεων και αφαίρεση θορύβου



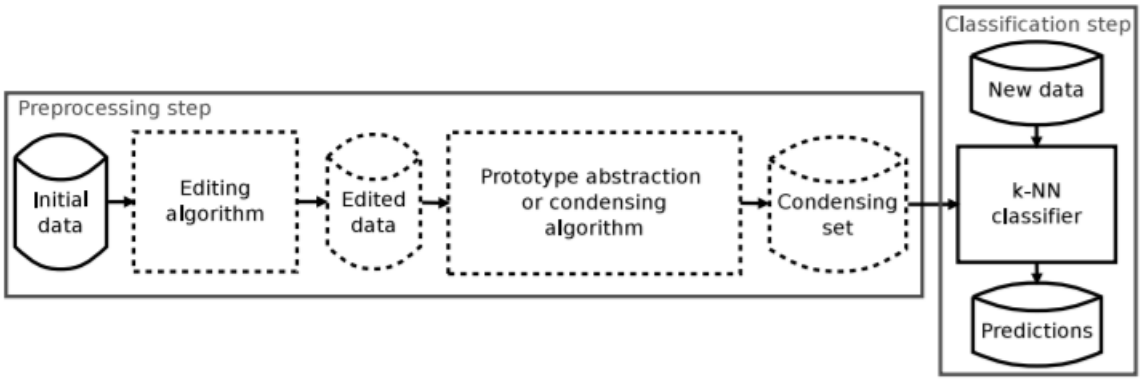
Σχήμα 1.6: Ιεραρχική κατηγοριοποίηση κατηγοριών DRT

πιο μικρό έγινε το μέγεθος του συνόλου συμπύκνωσης, σε σχέση με το μέγεθος του αρχικού συνόλου εκπαίδευσης. Με μαθηματική έννοια, είναι ο λόγος του αριθμού των απορριφθέντων στιγμιοτύπων έναντι του αριθμού των αρχικών στιγμιοτύπων του συνόλου εκπαίδευσης. Σύμφωνα με αυτά, προκύπτει το συμπέρασμα ότι, όσο υψηλότερος είναι ο ρυθμός μείωσης, τόσο πιο γρήγορη είναι η κατηγοριοποίηση με τον αλγόριθμο k-NN. Επόμενο κριτήριο είναι η ακρίβεια κατηγοριοποίησης, που επιτυγχάνεται από τον k-NN, όταν εκτελείται πάνω στο σύνολο συμπύκνωσης. Το τρίτο κριτήριο είναι το υπολογιστικό κόστος της προεπεξεργασίας, δηλαδή το κόστος που απαιτείται για την δημιουργία του συνόλου συμπύκνωσης. Για ορισμένους τομείς, ένα κριτήριο μπορεί να είναι πιο κρίσιμο από ένα άλλο. Ωστόσο, όλα αυτά πρέπει να πληρούν ορισμένες ελάχιστες απαιτήσεις.

Αν και, ο αλγόριθμος επεξεργασίας έχει έναν τελείως διαφορετικό στόχο από τις άλλες τεχνικές μείωσης

δεδομένων, μπορεί να εφαρμοστεί για τη βελτίωση της απόδοσης, αυξάνοντας τα ποσοστά μείωσης ή και τα επίπεδα ακρίβειας. Ειδικότερα, τα ποσοστά μείωσης πολλών αλγορίθμων παραγωγής προτύπων εξαρτώνται από το επίπεδο θορύβου στα δεδομένα εκπαίδευσης. Γι' αυτό, πολλοί αλγόριθμοι παραγωγής προτύπων απέχουν από την επίτευξη υψηλών ποσοστών μείωσης λόγω των υψηλών επιπέδων θορύβου στο σύνολο εκπαίδευσης. Ως συμπέρασμα αυτών προκύπτει ότι, όσο υψηλότερο είναι το επίπεδο θορύβου, τόσο χαμηλότεροι είναι οι ρυθμοί μείωσης που επιτυγχάνονται. Επομένως, η αποτελεσματική εφαρμογή τέτοιων αλγορίθμων απαιτεί την αφαίρεση θορύβου από τα δεδομένα, δηλαδή την εφαρμογή ενός αλγόριθμου επεξεργασίας εκ των προτέρων. Ως εκ τούτου, ένας αλγόριθμος επεξεργασίας θα πρέπει να εκτελείται πάνω σε ένα σύνολο εκπαίδευσης με θόρυβο, για να βελτιωθεί η ακρίβεια ή να υπάρχει αποτελεσματικότερη εφαρμογή των αλγορίθμων παραγωγής προτύπων [7].

Το σχήμα 1.7 απεικονίζει τη διαδικασία κατηγοριοποίησης k-NN μέσω της μείωσης δεδομένων. Η ολοκληρωμένη διαδικασία αποτελείται από δύο στάδια, την προεπεξεργασία, η οποία είναι προαιρετική και την κατηγοριοποίηση. Κυρίως, υπάρχουν τέσσερις πιθανοί τύποι προεπεξεργασίας: (i) χωρίς προεπεξεργασία, (ii) μόνο επεξεργασία, (iii) μόνο συμπύκνωση και (iv) τόσο επεξεργασία όσο και συμπύκνωση. Ο πρώτος τύπος εφαρμόζεται σε περιπτώσεις, όπου το σύνολο δεδομένων εκπαίδευσης είναι μικρό σε μέγεθος και δεν περιέχει θόρυβο και ακραία δεδομένα (outliers). Ο δεύτερος τύπος εφαρμόζεται σε περιπτώσεις, όπου το σύνολο δεδομένων εκπαίδευσης είναι μικρό και περιέχει θόρυβο. Ο τρίτος τύπος χρησιμοποιείται σε περιπτώσεις, όπου το σύνολο δεδομένων εκπαίδευσης είναι μεγάλο και δεν περιέχει θόρυβο. Τέλος, ο τέταρτος τύπος χρησιμοποιείται σε περιπτώσεις, όπου το σύνολο δεδομένων εκπαίδευσης είναι μεγάλο σε μέγεθος και περιέχει θόρυβο. Και οι τέσσερις τύποι προεπεξεργασίας είναι διαθέσιμοι στο WebDR1 [7].



Σχήμα 1.7: Διαδικασία κατηγοριοποίησης k-NN μέσω της μείωσης δεδομένων

Μια ολοκληρωμένη διαδικασία προεπεξεργασίας μείωσης δεδομένων στοχεύει στη δημιουργία ενός συνόλου συμπύκνωσης που δεν περιέχει θόρυβο, διατηρώντας ή δημιουργώντας για κάθε κατηγορία επαρκή αριθμό προτύπων που είναι απαραίτητα για την κατηγοριοποίηση των k πλησιέστερων γειτόνων [7].

1.6 Κίνητρο και Συνεισφορά

Η συγκεκριμένη εργασία επικεντρώνεται τόσο στις τεχνικές μείωσης δεδομένων όσο και στον κατηγοριοποιητή k-NN. Όπως έχει ήδη προαναφερθεί στις παραπάνω ενότητες, οι αδυναμίες του κατηγοριοποιητή

k-NN μπορούν να επιλυθούν χρησιμοποιώντας μια τεχνική μείωσης δεδομένων. Όμως, οι περισσότερες από αυτές καλύπτουν περιπτώσεις είτε δυαδικών προβλημάτων είτε προβλημάτων πολλών κατηγοριών. Υπάρχουν λίγες ερευνητικές προσεγγίσεις που έχουν αναπτυχθεί για μείωση δεδομένων σε περιπτώσεις προβλημάτων πολλαπλών ετικετών.

Ειδικότερα, ο κατηγοριοποιητής k-NN σταματά να είναι οκνηρός όταν χρησιμοποιείται σε συνδυασμό με έναν αλγόριθμο επιλογής ή παραγωγής προτύπων. Στην πραγματικότητα, το σύνολο συμπύκνωσης αποτελεί ένα μοντέλο κατηγοριοποίησης. Στην κατηγοριοποίηση πολλαπλών ετικετών, εάν χρησιμοποιείται η μέθοδος μετασχηματισμού προβλήματος BR, πρέπει να κατασκευάζεται ένα σύνολο συμπύκνωσης για κάθε ετικέτα. Επομένως, ο στόχος της μείωσης δεδομένων δεν επιτυγχάνεται και ο κατηγοριοποιητής k-NN πρέπει να αναζητά πλησιέστερους γείτονες σε κάθε σύνολο συμπύκνωσης, για να εκτελεί κάθε μεμονωμένη πρόβλεψη ετικέτας. Ως εκ τούτου, το υπολογιστικό κόστος παραμένει υψηλό.

Αυτή η παρατήρηση καθιστά σαφές ότι οι αλγόριθμοι επιλογής και παραγωγής προτύπων πρέπει να προσαρμοστούν, έτσι ώστε να μπορούν να χρησιμοποιηθούν για σύνολα δεδομένων με πολλές ετικέτες και αυτό είναι το κίνητρο πίσω από το παρόν έργο. Αυτό το άρθρο προτείνει έναν αλγόριθμο παραγωγής προτύπων για σύνολα δεδομένων πολλαπλών ετικετών. Αποτελεί μια παραλλαγή του αλγόριθμου μείωσης δεδομένων μέσω διαχωρισμού χώρου (Reduction By Partitining Space) RSP3, ο οποίος είναι ένας γρήγορος και μη παραμετρικός αλγόριθμος παραγωγής προτύπων για κατηγοριοποίηση μονής ετικέτας. Ο προτεινόμενος αλγόριθμος ονομάζεται Multilabel RSP3 (MSP3). Ο MRSP3 κληρονομεί όλες τις επιθυμητές ιδιότητες του RSP3 και κατασκευάζει ένα σύνολο συμπύκνωσης πολλαπλών ετικετών που στη συνέχεια χρησιμοποιείται από τον κατηγοριοποιητή BRkNN.

Η πειραματική μελέτη που πραγματοποιήθηκε δείχνει ότι ο MRSP3 επιτυγχάνει αξιοσημείωτα ποσοστά μείωσης χωρίς να επηρεάζει την μετρική Hamming-Loss. Για την αξιολόγηση του προτεινόμενου αλγόριθμου χρησιμοποιείται η μετρική Hamming-loss, καθώς είναι κατάλληλη για δεδομένα πολλαπλών ετικετών. Σύμφωνα με αυτήν, ο συνδυασμός του MRSP3 με τον κατηγοριοποιητή BRk-NN παράγει θετικά αποτελέσματα, καθώς οι τιμές της hamming-loss δεν επηρεάζονται δραματικά και τα ποσοστά μείωσης σε αρκετά σύνολα δεδομένων ξεπερνούν το 50%.

1.7 Η Οργάνωση της διπλωματικής

Όσον αφορά την οργάνωση αυτής της διπλωματικής εργασίας, η ίδια αποτελείται συνολικά από 7 Κεφάλαια. Εκτός από τις προηγούμενες ενότητες αυτού του κεφαλαίου, σε αυτή την ενότητα γίνεται μια σύντομη ανασκόπηση για το περιεχόμενο των υπόλοιπων κεφαλαίων.

Στο δέυτερο κεφάλαιο, αναλύονται λεπτομερώς οι μέθοδοι μετασχηματισμού προβλήματος πολλαπλών ετικετών. Είναι σημαντικό αυτό το κομμάτι, καθώς η πλήρης κατανόηση του είναι απαραίτητη για την κατανόηση των πειραμάτων και των διαδικασιών που ακολουθήθηκαν. Επιπλέον μία ενότητα αυτού του κεφαλαίου αφιερώνεται στη λεπτομερή περιγραφή του **BRk-NN** κατηγοριοποιητή, ο οποίος χρησιμοποιήθηκε στα πειράματα αντί του κλασσικού k-NN, αφού είναι ιδανικός για προβλήματα πολλαπλών ετικετών. Τέλος, στην τελευταία ενότητα αναφέρονται μετρικές, που είτε είναι κατάλληλες για σύνολα δεδομένων πολλαπλών ετικετών, όπως η **hamming-loss**, είτε όχι, όπως accuracy, recall κ.α.

Στο τρίτο κεφάλαιο, γίνεται μια μικρή ανασκόπηση των αλγόριθμων μείωσης δεδομένων για δεδομένα

πολλαπλών ετικετών. Αναλύονται διάφοροι αλγόριθμοι που έχουν προταθεί στο παρελθόν για τέτοιου είδους προβλήματα, καθώς και διάφορες μετρικές και τεχνικές που οι ίδιοι υιοθετούν.

Στο τέταρτο κεφάλαιο, γίνεται λεπτομερής ανάλυση των αλγορίθμων μείωσης δεδομένων με διαχωρισμό χώρου (Reduction by Space Partitioning). Αρχικά, περιγράφεται ο αλγόριθμος Chen and Jozwik**(CJA)** και στη συνέχεια οι 3 διαφορετικές επεκτάσεις αυτού που είναι οι **RSP1, RSP2** και **RSP3**. Παρουσιάζονται η βασική ιδέα αυτών, τα μειονεκτήματα και πλεονεκτήματα τους. Είναι ένα από τα πιο σημαντικά κεφάλαια, καθώς όλη η εργασία βασίζεται στον αλγόριθμο RSP3 για τη δημιουργία μιας παραλλαλής αυτού, ώστε να ταιριάζει σε προβλήματα πολλαπλών ετικετών.

Το πέμπτο κεφάλαιο αφορά τα πειράματα που εκτελέστηκαν κατά την διάρκεια της εργασίας. Ειδικότερα, περιγράφεται αναλυτικά ο αλγόριθμος μείωσης δεδομένων με διαχωρισμό χώρου για προβλήματα πολλαπλών ετικετών**(MRSP3)**, που δημιουργήθηκε στα πλαίσια της εργασίας. Δίνεται κομμάτι κώδικα όσον αφορά την υλοποίηση αυτού και των BRk-NNa και BRk-NNb. Το κεφάλαιο αυτό, αποτελεί την ουσία όλης της έρευνας, αφού υλοποιείται και προγραμματιστικά όλη η θεωρία αυτής.

Στο έκτο κεφάλαιο, αναγράφονται όλα τα αποτελέσματα των πειραμάτων που εκτελέστηκαν. Πιο συγκεκριμένα, περιγράφονται τα σύνολα δεδομένων που χρησιμοποιήθηκαν για κατηγοριοποίηση, δηλαδή η έννοια τους και τα χαρακτηριστικά τους. Επίσης, αναφέρεται το υπολογιστικό περιβάλλον, όπου εκτελέστηκαν τα πειράματα. Επιπλεόν, περιγράφεται η μέθοδος επικύρωσης μοντέλου που χρησιμοποιήθηκε, σε αυτή την περίπτωση η μέθοδος **Cross-validation**.

Τέλος, στο έβδομο κεφάλαιο, συνοψίζονται τα συμπεράσματα που έχουν προκύψει από όλη τη διεξαγωγή της έρευνας. Επίσης προτείνονται ιδέες για μελλοντική έρευνα σχετικά με την συγκεκριμένη εργασία για μείωση δεδομένων με διαχωρισμό του χώρου για σύνολα δεδομένων πολλαπλών ετικετών.

Κεφάλαιο 2ο: Προβλήματα πολλαπλών ετικετών

Μπορούμε να ομαδοποιήσουμε τις υπάρχουσες μεθόδους κατηγοριοποίησης πολλαπλών ετικετών σε δύο κύριες κατηγορίες: α) μέθοδοι μετασχηματισμού προβλήματος (problem transformation methods) και β) μέθοδοι προσαρμογής αλγορίθμου (algorithm adaptation methods). Οι μέθοδοι μετασχηματισμού προβλήματος μετασχηματίζουν το πρόβλημα κατηγοριοποίησης πολλαπλών ετικετών σε ένα ή περισσότερα προβλήματα κατηγοριοποίησης μονής ετικέτας, για τα οποία υπάρχει και μια τεράστια βιβλιογραφία αλγορίθμων μάθησης. Από την άλλη, μέθοδοι προσαρμογής αλγορίθμου καλούνται, αυτές που τροποποιούν υπάρχοντες αλγόριθμους μονής ετικέτας, προκειμένου να εφαρμοστούν σε δεδομένα πολλαπλών ετικετών απευθείας [15]. Η συγκεκριμένη εργασία ασχολείται μόνο με την πρώτη κατηγορία μεθόδων.

Ειδικότερα, με τις μεθόδους μετασχηματισμού προβλημάτων, ένα πρόβλημα πολλών ετικετών μετατρέπεται σε μία ή περισσότερες μονές ετικέτες (δηλ. δυαδικό ή πολλών κατηγοριών) προβλήματα. Με αυτόν τον τρόπο, χρησιμοποιούνται κατηγοριοποιητές μονής ετικέτας και οι προβλέψεις τους για τη μονή ετικέτα μετατρέπονται σε προβλέψεις πολλαπλών ετικετών. Ο μετασχηματισμός προβλήματος είναι ελκυστικός λόγω της επεκτασιμότητας και της ευελιξίας: οποιοσδήποτε κατηγοριοποιητής μονής ετικέτας μπορεί να χρησιμοποιηθεί.

Παρακάτω αναλύονται οι πιο σημαντικές και ευρέως χρησιμοποιημένες μέθοδοι μετασχηματισμού προβλήματος. Αναλύεται τόσο η σημασίας τους, όσο και η συνεισφορά τους στο έργο της εξόρυξης δεδομένων.

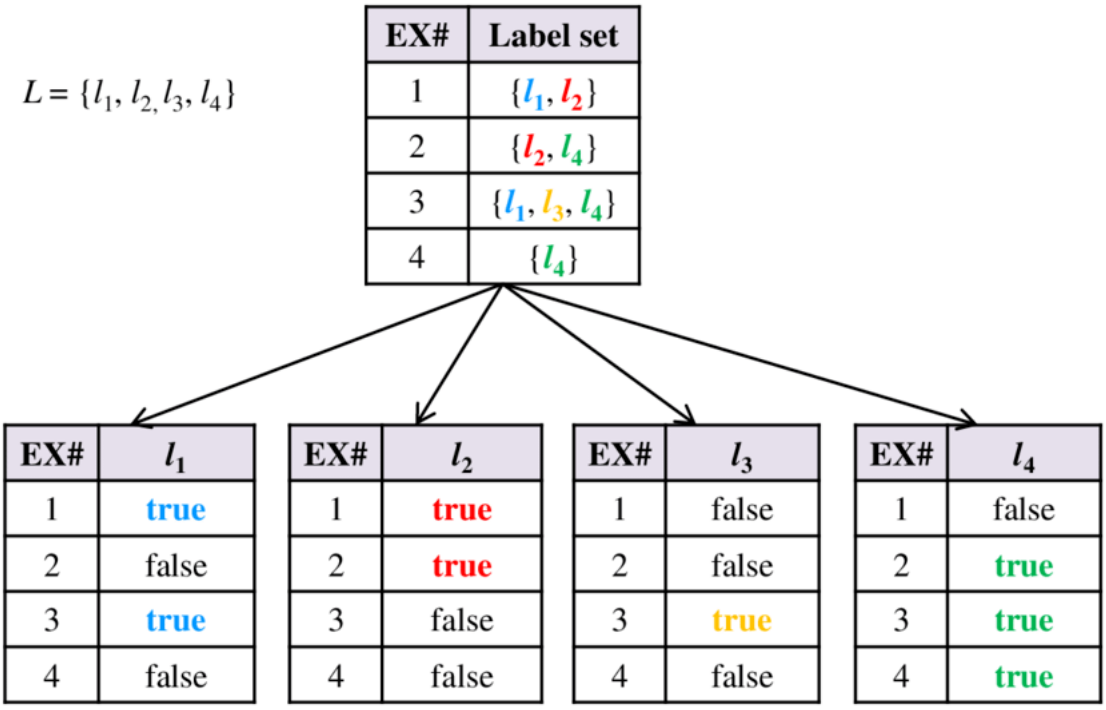
2.1 Binary Relevance

Η πιο κοινή μέθοδος μετασχηματισμού προβλήματος είναι η μέθοδος δυαδικής συνάφειας (Binary Relevance). Η BR αποτελεί πολύ γνωστό πλαίσιο για την κατηγοριοποίηση πολλαπλών ετικετών, το οποίο θεωρεί κάθε ετικέτα κατηγορίας ως πρόβλημα δυαδικής κατηγοριοποίησης. Είναι η απλούστερη στρατηγική, αλλά είναι πιο αποτελεσματική από ό,τι φαίνεται με την πρώτη ματιά.

Η δυαδική συνάφεια είναι μία από τις πιο δημοφιλής μεθόδους μετασχηματισμού, που χρησιμοποιούν q δυαδικούς κατηγοριοποιητές (q = $\left|B\right|$, συνολικός αριθμός κλάσεων (B) σε ένα σύνολο δεδομένων), έναν για κάθε ετικέτα. Αναλυτικότερα, η BR μετατρέπει το αρχικό σύνολο δεδομένων σε σύνολα δεδομένων q, όπου κάθε σύνολο δεδομένων περιέχει όλα τα στιγμιότυπα του αρχικού συνόλου δεδομένων και εκπαιδεύει έναν κατηγοριοποιητή σε καθένα από αυτά τα σύνολα δεδομένων. Αν ένα συγκεκριμένο στιγμιότυπο περιέχει την ετικέτα B_j = "1", αυτό σημαίνει πως ανήκει σε αυτήν την κλάση. Διαφορετικά, αν περιέχει την ετικέτα B_j = "0", αυτό σημαίνει πως το συγκεκριμένο στιγμιότυπο δεν ανήκει σε αυτήν την κλάση. Για την κατηγοριοποίηση ενός νέου αταξινόμητου στιγμιοτύπου, υπολογίζεται η ένωση των ετικετών που προβλέπονται θετικά από τους q κατηγοριοποιητές. Η BR χρησιμοποιείται σε πολλές πρακτικές εφαρμογές, αλλά μπορεί να χρησιμοποιηθεί μόνο σε εφαρμογές που οι ετικέτες είναι ανεξάρτητες μεταξύ τους, κάτι που αποτελεί τον μοναδικό περιορισμό του [16].

Στο Σχήμα 2.1, δίνεται ένα παράδειγμα για την κατανόηση της λειτουργίας της μεθόδου BR. Συγκεκριμένα, υπάρχουν 4 στιγμιότυπα (1,2,3,4) στο σύνολο δεδομένων εκπαίδευσης και 4 διαφορετικές κλάσεις (l_1,l_2,l_3,l_4), όπου μπορούν να κατανεμηθούν τα στιγμιότυπα στο σύστημα μας. Παρατηρείται ότι,

υπάρχει διαφορετικός πίνακας για κάθε ετικέτα και ότι σε καθένα από αυτούς απεικονίζονται όλα τα στιγμιότυπα. Επιπλέον, όπου η τιμή της ετικέτας είναι “true”, σημαίνει ότι το συγκεκριμένο στιγμιότυπο ανήκει σε αυτή την κλάση, διαφορετικά αν είναι “false”, ότι δεν ανήκει σε αυτή την κλάση. Όσον αφορά την κατηγοριοποίηση ενός νέου στιγμιοτύπου, σε κάθε πίνακα, δηλαδή διαφορετικό σύνολο δεδομένων, εκτελείται ένας δυαδικός κατηγοριοποιητής. Η ένωση των προβλέψεων “true” όλων των κατηγοριοποιητών απεικονίζεται στον πάνω πίνακα και είναι αυτό που θα καθορίσει σε ποιες κλάσεις θα ανήκει το νέο στιγμιότυπο.



Σχήμα 2.1: Τρόπος Λειτουργίας Binary Relevance-BR

Υπάρχουν ιδιότητες της μεθόδου BR, που αξίζει να αναφερθούν. Αρχικά, μία από αυτές είναι η εννοιολογική της απλότητα. Συγκεκριμένα, η μέθοδος BR είναι μια προσέγγιση πρώτης τάξης, η οποία χτίζει το μοντέλο κατηγοριοποίησης ετικέτα-ετικέτα και έτσι αγνοεί την ύπαρξη άλλων ετικετών κλάσης. Γι’ αυτό, η πολυπλοκότητα μοντελοποίησης της είναι γραμμική με τον αριθμό των ετικετών κλάσης στο χώρο της ετικέτας. Δεύτερον, η BR εμπίπτει στην κατηγορία μέθοδοι μετασχηματισμού προβλήματος, οι οποίες επιλύουν το πρόβλημα κατηγοριοποίησης με πολλές ετικέτες μετατρέποντάς το σε άλλα καθιερωμένα σενάρια μάθησης (δυαδική κατηγοριοποίηση σε αυτήν την περίπτωση). Κατά συνέπεια, η δυαδική συνάφεια δεν περιορίζεται σε συγκεκριμένες τεχνικές μάθησης και μπορεί να εδραιωθεί σε οποιονδήποτε αλγόριθμο δυαδικής μάθησης με διαφορετικά χαρακτηριστικά. Τρίτον, η δυαδική συνάφεια βελτιστοποιεί κατά μέσο όρο τις μετρήσεις αξιολόγησης με βάση πολλές ετικέτες, οι οποίες αξιολογούν την απόδοση του συστήματος εκμάθησης σε κάθε ετικέτα κλάσης ξεχωριστά και στη συνέχεια επιστρέφεται η μέση τιμή για όλες τις ετικέτες κλάσης. Επομένως, η πραγματική μέτρηση πολλαπλών ετικετών που βελτιστοποιείται εξαρτάται από τη δυαδική απώλεια, η οποία ελαχιστοποιείται από τον αλγόριθμο δυαδικής μάθησης. Τέλος, η δυαδική συνάφεια μπορεί εύκολα να προσαρμοστεί στην εκμάθηση δεδομένων πολλαπλών ετικετών με ετικέτες που λείπουν, όπου οι πληροφορίες επισήμανσης για στιγμιότυπα εκπαίδευσης είναι ελλιπείς λόγω παραγόντων, όπως το υψηλό κόστος επισήμανσης, η ανθρώπινη απρο-

σεξία στην εκχώρηση ετικετών κ.λπ. Για την αντιμετώπιση αυτής της κατάστασης, η μέθοδος BR μπορεί να παράγει ένα δυαδικό σύνολο εκπαίδευσης, αποκλείοντας τα δεδομένα, των οποίων οι πληροφορίες επισήμανσης δεν είναι διαθέσιμες [17].

2.2 BRKNN

Όπως έχει ήδη αναφερθεί στην προηγούμενη ενότητα, η Binary Relevance (BR) είναι η πιο διαδεδομένη μέθοδος μετασχηματισμού προβλήματος για κατηγοριοποίηση πολλαπλών ετικετών. Ο BRkNN είναι μια προσαρμογή του αλγορίθμου k-NN που είναι εννοιολογικά ισοδύναμη με τη χρήση BR σε συνδυασμό με τον αλγόριθμο k-NN. Επομένως, αντί να εφαρμοστεί ο BRkNN, θα μπορούσαμε να χρησιμοποιήσουμε τις υπάρχουσες υλοποιήσεις του BR και k-NN. Ωστόσο, το πρόβλημα κατά τη σύνδεση BR με kNN είναι ότι η ίδια διαδικασία υπολογισμού των k πλησιέστερων γειτόνων θα εκτελεστεί |L| φορές, όσες και το πλήθος των ετικετών. Προς αποφυγήν των περιττών υπολογισμών που απαιτούν χρόνο, ο BRkNN επεκτείνει τον k-NN αλγόριθμο, έτσι ώστε να γίνονται ανεξάρτητες προβλέψεις για κάθε ετικέτα, ακολουθώντας μία μοναδική αναζήτηση των k πλησιέστερων γειτόνων. Με αυτόν τον τρόπο το BRkNN είναι |L| φορές πιο γρήγορο από BR σε συνδυασμό με τον k-NN, γεγονός που θα μπορούσε να είναι κρίσιμο σε τομείς με μεγάλο σύνολο ετικετών και απαιτήσεων για χαμηλούς χρόνους απόκρισης [18]. Ο BRkNN υλοποιείται στο λογισμικό κατηγοριοποίησης πολλαπλών ετικετών MULAN [19].

Οι BRk-NNa και BRk-NNb είναι οι δύο επεκτάσεις του βασικού αλγόριθμου BRkNN, οι οποίες βασίζονται στον υπολογισμό των τιμών εμπιστευτικότητας (confidence) για κάθε ετικέτα $\lambda \in L$. Η εμπιστευτικότητα για μια ετικέτα μπορεί να αποκτηθεί εύκολα λαμβάνοντας υπόψιν το ποσοστό των k πλησιέστερων γειτόνων που την περιλαμβάνουν. Επισήμως, έστω ότι Y_j, j = 1. . . k, το σύνολο των ετικετών των k πλησιέστερων γειτόνων ενός νέου στιγμιοτύπου x. Η εμπιστευτικότητα c μιας ετικέτας $\lambda \in L$ ισούται με:

$$c = \frac{1}{k} \sum_{j=1}^{k} I_{Y_j}(\lambda) \tag{2.1}$$

όπου $I_{Y_j} : L \rightarrow 0, 1$ είναι μια συνάρτηση που εξάγει 1 εάν η ετικέτα εισόδου λ ανήκει στο σύνολο Y_j διαφορετικά 0 και ονομάζεται συνάρτηση δεικτών στη θεωρία συνόλων.

Η πρώτη επέκταση του BRkNN ονομάζεται BRkNN-a και εκχωρεί τις ετικέτες που έχουν εκχωρηθεί σε τουλάχιστον από τους μισούς γείτονες. Επιπλέον, ελέγχει εάν ο BRkNN εξάγει το κενό σύνολο, επειδή καμία από τις ετικέτες $\lambda \in L$ δεν περιλαμβάνεται τουλάχιστον στους μισούς από τους k πλησιέστερους γείτονες. Εάν ισχύει αυτή η συνθήκη, τότε εξάγει την ετικέτα με την υψηλότερη εμπιστευτικότητα. Αντιμετωπίζει έτσι ένα γενικό μειονέκτημα της BR, που δεν έχει αναπτυχθεί στο παρελθόν: καθώς κάθε ετικέτα προβλέπεται ανεξάρτητα με την BR, υπάρχει πιθανότητα το κενό σύνολο να δοθεί ως η συνολική έξοδος. Η δεύτερη επέκταση του BRkNN ονομάζεται BRkNN-b και εκχωρεί τις πιο δημοφιλείς ετικέτες m των γειτόνων, δηλαδή αυτές με την υψηλότερη τιμή εμπιστευτικότητας, όπου m είναι ο μέσος αριθμός ετικετών που έχουν εκχωρηθεί στους γείτονες του στιγμιοτύπου.

2.3 Label Powerset

Η πιο φυσική προσέγγιση μετασχηματισμού πρόβληματος είναι η μέθοδος Label Powerset (LP), που δημιουργεί μια νέα κατηγορία για κάθε συνδυασμό ετικετών και στη συνέχεια λύνει το πρόβλημα χρησιμοποιώντας προσεγγίσεις κατηγοριοποίησης πολλαπλών κλάσεων. Το κύριο μειονέκτημα αυτής της προσέγγισης αυτής είναι η εκθετική αύξηση του αριθμού των κλάσεων, καθώς η χειρότερη περίπτωση αριθμού νέων κλάσεων είναι 2^L, όπου L το σύνολο των διαφορετικών κλάσεων. Αυτό οδηγεί σε πολλές δημιουργημένες κλάσεις με πολύ λίγες εμφανίσεις, που με τη σειρά του οδηγεί σε υπερβολική τοποθέτηση (overfitting) [20]. Για να επιλυθεί αυτό το ζήτημα, η Read [21] έχει αναπτύξει μια μέθοδο μετασχηματισμού κλαδεμένων προβλημάτων (PPT), η οποία επιλέγει μόνο τις μετασχηματισμένες ετικέτες που εμφανίζονται περισσότερο από έναν προκαθορισμένο αριθμό φορών. Μια άλλη μέθοδος label power-set είναι το HOMER [22], το οποίο κατασκευάζει πρώτα μια ιεραρχία των πολλαπλών ετικετών και στη συνέχεια κατασκευάζει έναν κατηγοριοποιητή για τα σύνολα ετικετών σε κάθε κόμβο της ιεραρχίας. Αναλυτικότερα, η μέθοδος LP λαμβάνει υπόψιν τις συσχετίσεις μεταξύ ετικετών δημιουργώντας έναν μοναδικό κατηγοριοποιητή, στον οποίο κάθε μοναδικός συνδυασμός των ετικετών αποτελούν μία μόνο ετικέτα. Το σύνολο όλων των συνδυασμών είναι επίσης γνωστό ως το σύνολο ισχύος (Powerset) του L και συμβολίζεται με P(L) [23].

Για καλύτερη κατανόηση της λειτουργίας της μεθόδου LP, ακολουθεί ένα απλό και εύκολο παράδειγμα. Στον Πίνακα 2.1, απεικονίζονται τα δεδομένα πολλαπλών ετικετών, ενώ στον Πίνακα 2.2 είναι τα μετασχηματισμένα δεδομένα. Παρατηρείται ότι, στον πρώτο πίνακα το στιγμιότυπο 1 ανήκει σε δύο κλάσεις, τις B1 και B2. Σύμφωνα με τη μέθοδο, η νέα κλάση που θα σχηματιστεί θα είναι η B1,2, όπως αναγράφεται στον δεύτερο πίνακα. Για τον ίδιο λόγο αντίστοιχα, θα σχηματιστούν οι νέες ετικέτες B1,2,3, B2,4 και B1,2,5. Έτσι, το πρόβλημα από πρόβλημα πολλαπλών ετικετών (multi-label) μετασχηματίζεται σε πρόβλημα πολλών κατηγοριών(multi-class). Είναι άξιο να αναφερθεί ότι, έπειτα από την εφαρμογή της μεθόδου LP στα δεδομένα, δεν υπάρχει απώλεια πληροφορίας κι επίσης θεωρείται ότι υπάρχει συσχέτιση μεταξύ των δεδομένων και των ετικετών. Επιπλέον, ένα πρόβλημα πολλών κατηγοριών δίνει τη δυνατότητα εφαρμογής κατανομών πιθανότητας, για να γίνει η κατάταξη μεταξύ των ετικετών. 'Ετσι, όταν εισέρχεται ένα νέο στιγμιότυπο στο σύστημα, για παράδειγμα το στιγμιότυπο 6, που ανήκει στις κλάσεις B1, B2 και B4, πρέπει να κατανεμηθεί σύμφωνα με τον Πίνακα 2.2, οπότε θα πρέπει να υπολογιστούν οι πιθανότητες. Η έξοδος της μεθόδου LP θα είναι η πιο πιθανή κλάση. Όσον αφορά τους άσσους και τα μηδενικά στον Πίνακα 2.2, επισημαίνουν την ύπαρξη της συγκεκριμένης κλάσης στην νέα ετικέτα που έχει δημιουργηθεί. Για να βρεθεί η συσχέτιση μεταξύ των ετικετών, υπολογίζεται για κάθε ετικέτα η συνολική της πιθανότητα.

Πίνακας 2.1: Πρόβλημα πολλαπλών ετικετών

Ex	Attribute	Labelset
1	A1	B1,B2
2	A2	B1,B2,B3
3	A3	B4
4	A4	B1,B2,B5
5	A5	B2,B4

Πίνακας 2.2: Πρόβλημα πολλών κατηγοριών

c	P(c|x)	B1	B2	B3	B4	B5
B1,B2	0.4	1	1	0	0	0
B1,B2,B3	0.2	1	1	1	0	0
B4	0.1	0	0	0	1	0
B1,B2,B5	0.0	1	1	0	0	1
B2,B4	0.3	0	1	0	1	0
Συνολική Πιθανότητα	**1.0**	**0.6**	**0.9**	**0.2**	**0.4**	**0.0**

Η ιεραρχία των κατηγοριοποιητών πολλαπλών ετικετών (HOMER) [22] είναι ένας αλγόριθμος για αποτελεσματική και υπολογιστικά αποδοτική εκμάθηση πολλών ετικετών σε τομείς με μεγάλο αριθμό ετικετών. Το HOMER κατασκευάζει μια ιεραρχία κατηγοριοποιητών με πολλές ετικέτες, καθεμία από τις οποίες ασχολείται με ένα πολύ μικρότερο σύνολο ετικετών σε σύγκριση με το Q (ο συνολικός αριθμός ετικετών) και μια πιο ισορροπημένη κατανομή στιγμιοτύπων. Αυτό οδηγεί σε βελτιωμένη πρόβλεψη απόδοσης και επίσης σε γραμμική εκπαίδευση και λογαριθμικές πολυπλοκότητες δοκιμών σε σχέση με το Q. Μία από τις κύριες διαδικασίες στο HOMER είναι η ομοιόμορφη κατανομή ενός συνόλου ετικετών σε υποδιαιρέσεις k disjoint έτσι ώστε παρόμοιες ετικέτες να τοποθετούνται μαζί και να διαφέρουν μεταξύ τους. Η καλύτερη προγνωστική απόδοση αναφέρεται χρησιμοποιώντας έναν αλγόριθμο ισορροπημένου k προσαρμοσμένο για HOMER [22]. Το HOMER είναι μια υπολογιστικά αποδοτική μέθοδος κατηγοριοποίησης πολλαπλών ετικετών, ειδικά σχεδιασμένη για μεγάλα σύνολα δεδομένων πολλαπλών ετικετών [24].

2.4 Random k-Labelsets

Παραπάνω αναφέρθηκε ένα κύριο μειονέκτημα της μεθόδου Label Powerset, που είναι η εκθετική αύξηση του συνόλου των νέων κλάσεων. Μια προσπάθεια αντιμετώπισης αυτού αποτελεί η τυχαία μέθοδος K-labelsets, που ονομάζεται επίσης RAKEL, η οποία εξακολουθεί να αξιοποιεί τη βασική μέθοδο. Η ιδέα της μεθόδου αυτή είναι η δημιουργία label powerset κατηγοριοποιητών, αλλά μόνο για σύνολα ετικετών k κάθε φορά, το οποίο k μπορεί να είναι τρία, τέσσερα κ.α. Ένα label powerset ορίζεται να είναι υπεύθυνο μόνο, για παράδειγμα, για τέσσερις ετικέτες και στη συνέχεια, να δημιουργείται ένα ολόκληρο σύνολο από αυτές. Σύμφωνα με αυτά, δημιουργούνται label powerset κατηγοριοποιητές, όπου ο καθένας είναι υπεύθυνος για τις ετικέτες k και είναι J από αυτούς συνολικά. Γενικά, αντί να υπάρχει ένας μεγάλος κατηγοριοποιητής πολλαπλών κλάσεων, θα έχουμε έναν αριθμό μεγάλων κατηγοριοποιητών πολλαπλών κλάσεων, όπου ο χώρος εξόδου του καθενός θα είναι το δυναμικό υποσύνολο του συνόλου των ετικετών. [25]

Η μέθοδος RAKEL προσπαθεί να αντιμετωπίσει ορισμένα από τα προβλήματα της μεθόδου Label powerset, αλλά έχει το κόστος της εισαγωγής μερικών υπερ-παραμέτρων. Ειδικότερα, ορίζεται η υπερ-παράμετρος J, που είναι ο αριθμός των συνόλων που θα χρησιμοποιήσουμε και η υπερ-παράμετρος k, που είναι ο αριθμός των ετικετών που υπάρχουν σε κάθε label powerset κατηγοριοποιητή [25]. Μια απλή διαδικασία ψηφοφορίας καθορίζει το τελικό σύνολο ετικετών για ένα συγκεκριμένο στιγμιότυπο. [24].

Στο Σχήμα 2.2, απεικονίζεται ένα παράδειγμα αυτής της μεθόδου, για καλύτερη κατανόηση. Συγκεκριμένα, το μέγεθος των υποσυνόλων ορίζεται να είναι 3, επομένως k = 3, ενώ το πλήθος των τυχαίων

υποσυνόλων να είναι 7, επομένως n = 7. Για την κατηγοριοποίηση ενός νέου στιγμιοτύπου, οι προβλέψεις του κατηγοριοποιητή (multi-label με χρήση μεθόδου LP) για κάθε ένα από τα μοντέλα που έχουν παραχθεί αναπαρίστανται στο τμήμα των προβλέψεων. Τέλος, υπολογίζεται πρώτα η πιθανότητα για κάθε ετικέτα (average votes), κι έπειτα χρησιμοποιείται κάποιο κατώφλι (στη συγκεκριμένη περίπτωση 0,5) για να καθοριστεί η τελική πρόβλεψη για το νέο στιγμιότυπο.

model	labelsets (k=3)	predictions					
		λ1	λ2	λ3	λ4	λ5	λ6
h1	λ1, λ2, λ6	1	0	-	-	-	-
h2	λ2, λ3, λ4	-	1	1	0	-	-
h3	λ3, λ5, λ6	-	-	0	-	0	1
h4	λ2, λ4, λ5	-	0	-	0	0	-
h5	λ1, λ4, λ5	1	-	-	0	1	-
h6	λ1, λ2, λ6	1	0	1	-	-	-
h7	λ1, λ2, λ6	0	-	-	1	-	0
	Average votes	3/4	1/4	2/3	1/4	1/3	2/3
	Final prediction	1	0	1	0	0	1

Σχήμα 2.2: Παράδειγμα τρόπου λειτουργίας μεθόδου RAKEL

2.5 Classifier Chain

Η προσέγγιση Classifier Chain (CC) είναι μια μέθοδος μετασχηματισμού προβλήματος που βασίζεται στη binary relevance μέθοδο. Σύμφωνα με την προτεινόμενη μέθοδο, ένας απλός δυαδικός κατηγοριοποιητής συσχετίζεται με καθεμία από τις προκαθορισμένες ετικέτες στο σύνολο δεδομένων και όλοι αυτοί οι κατηγοριοποιητές είναι συνδεδεμένοι σε μια αλυσίδα με σειρά. Ο χώρος χαρακτηριστικών κάθε κατηγοριοποιητή στην αλυσίδα επεκτείνεται με τις συσχετίσεις ετικετών 0/1 όλων των προηγούμενων κατηγοριοποιητών. Έτσι, κάθε απόφαση κατηγοριοποίησης για μια συγκεκριμένη ετικέτα στην αλυσίδα αυξάνεται από όλες τις προηγούμενες δυαδικές προβλέψεις σχετικότητας στην αλυσίδα. Με αυτόν τον τρόπο λαμβάνονται υπόψιν οι συσχετίσεις μεταξύ των ετικετών. Η μέθοδος CC έχει αποδειχθεί ότι βελτιώνει την ακρίβεια κατηγοριοποίησης της μεθόδου BR σε έναν αριθμό κανονικών (όχι μεγάλου μεγέθους) συνόλων δεδομένων. Παρ'όλα αυτά, υπάρχουν κάποια σημεία στα οποία παρουσιάζει αδυναμίες και είναι καλό να αντιμετωπιστούν.

Ένα από τα μειονεκτήματα αυτής της μεθόδου, που σημειώνουν οι συγγραφείς, είναι ότι η ίδια η σειρά της αλυσίδας επηρεάζει την ακρίβεια. Αυτό μπορεί να λυθεί είτε με ευρετική επιλογή της σειράς των μελών της αλυσίδας ή χρησιμοποιώντας ένα σύνολο κατηγοριοποιητών αλυσίδας. Οποιαδήποτε από αυτές τις λύσεις αυξάνει τις απαιτήσεις υπολογιστικού χρόνου. Ένα άλλο μειονέκτημα αυτής της προσέγγισης είναι ότι σε σύνολα δεδομένων που περιέχουν πολλά χαρακτηριστικά, η επίδραση της προτεινόμενης αύξησης της ετικέτας στον μεγάλο χώρο χαρακτηριστικών είναι πολύ μικρή. Μπορεί ακόμη και να παραμεληθεί σε σύνολα δεδομένων, όπου ο αριθμός των χαρακτηριστικών είναι πολύ υψηλότερος από τον αριθμό των ετικετών [23].

Ουσιαστικά, το CC είναι μια μέθοδος BR και μετατρέπει τα προβλήματα πολλαπλών ετικετών σε προβλήματα μιας ετικέτας. Διαφέρει όμως από την BR, επειδή ο χώρος χαρακτηριστικών για κάθε δυαδικό μοντέλο επεκτείνεται με τη συνάφεια της ετικέτας 0/1 όλων των προηγούμενων ταξινομητών. Έτσι,

σχηματίζεται μια αλυσίδα κατηγοριοποιητή. Η διαδικασία εκπαίδευσης εκπαιδεύει τον πρώτο κατηγοριοποιητή για την ετικέτα l_i από το αρχικό σετ εκπαίδευσης D, κι έτσι αυξάνει το χώρο χαρακτηριστικών από τις παρουσίες των τιμών ετικέτας εκπαίδευσης. Στη συνέχεια, εκπαιδεύει έναν επόμενο κατηγοριοποιητή για την επόμενη ετικέτα. Προφανώς, τα πάνω βήματα μπορούν να επαναληφθούν τόσες φορές όσο είναι το πλήθος των κατηγοριοποιητών ετικέτας, που μπορούν να εκπαιδευτούν. Στο Σχήμα 2.3, απεικονίζεται μια διαδικασία πρόβλεψης με ένα παράδειγμα.

$h : X \rightarrow$	l
$h_1 : [0, 1, 0, 1, 0, 0, 1, 1, 0]$	1
$h_1 : [0, 1, 0, 1, 0, 0, 1, 1, 0, 1]$	0
$h_1 : [0, 1, 0, 1, 0, 0, 1, 1, 0, 1, 0]$	0
$h_1 : [0, 1, 0, 1, 0, 0, 1, 1, 0, 1, 0, 0]$	1
$h_1 : [0, 1, 0, 1, 0, 0, 1, 1, 0, 1, 0, 0, 1]$	0

Σχήμα 2.3: Διαδικασία πρόβλεψης CC μεθόδου

Η σειρά της ίδιας της αλυσίδας είναι φυσικό να επηρεάζει την ακρίβεια κατηγοριοποίησης. Υπάρχει η πιθανή επίδραση της διάδοσης σφαλμάτων κατά μήκος της αλυσίδας κατά το χρόνο κατηγοριοποίησης, όταν ένας (ή περισσότεροι) των πρώτων κατηγοριοποιητών κάνουν κακή πρόβλεψη. Αυτό παρατηρήθηκε επίσης από ορισμένους συγγραφείς, οι οποίοι πρότειναν διάφορες μεθόδους για την επίλυσή του, αλλά εξακολουθούν να έχουν αστάθεια [26].

Σε αυτήν τη μέθοδο, όπως ειπώθηκε και παραπάνω, η σειρά των κατηγοριοποιητών στην αλυσίδα είναι πολύ σημαντική για την πρόβλεψη ακρίβειας και αυτό οδηγεί σε σημαντική επιρροή στα αποτελέσματα των προβλέψεων. Για το λόγο αυτό, στην έρευνα προτείνεται ένας βελτιωμένος CC αλγόριθμος, που συνδυάζεται με τον αλγόριθμο k-means, για την επιβεβαίωση της σειράς των δυαδικών κατηγοριοποιητών. Αυτός ο αλγόριθμος, που ονομάζεται αλγόριθμος Km-CC, διασφαλίζει ότι οι σωστές συσχετίσεις μεταδίδονται επίμονα και μαζικά για τη βελτίωση της ακρίβειας των προηγούμενων προβλέψεων. Ειδικότερα, επιλεγμένα στιγμιότυπα διατηρούν τις περισσότερες από τις σημαντικές πληροφορίες του αρχικού συνόλου εκπαίδευσης. Στην πράξη, οι καλύτερες τεχνικές επιλογής στιγμιοτύπων θα πρέπει να είναι σε θέση να ανιχνεύουν και να αγνοούν θορυβώδη, τροποποιημένα και παραπλανητικά στγμιότυπα. Στον Km-CC, η ποιότητα της εκπαίδευσης επηρεάζει την ακρίβεια αναγνώρισης και αυξάνει την απόδοση του επόμενου προς επιλογή στιγμιοτύπου. Η απόδοση πρόβλεψης και η ακρίβεια κατηγοριοποίησης μπορούν να αυξηθούν ως αποτέλεσμα της επιλογής στιγμιοτύπου, με την αφαίρεση θορυβώδων και παραπλανητικών στιγμιοτύπων [26].

2.6 Μετρικές Αξιολόγησης για σύνολα δεδομένων πολλαπλών ετικετών

Είναι σημαντικό να αναφερθεί ότι υπάρχουν δύο κύριες μέθοδοι στην κατηγοριοποίηση δεδομένων πολλαπλών ετικετών: η κατηγοριοποίηση πολλαπλών ετικετών (MLC) και η κατάταξη ετικετών (Label Ranking-LR). Η MLC επικεντρώνεται στην εκπαίδευση μοντέλου, το οποίο να μπορεί να παράγει ένα διχοτομημένο συνόλο ετικετών με σχετικές και άσχετες ετικέτες σε σχέση με ένα νέο στιγμιότυπο [15].

Στην εξόρυξη δεδομένων πολλαπλών ετικετών τόσο η MLC όσο και η LR αποτελούν σημαντικό κομμάτι. Για παράδειγμα, σε μια μηχανή αναζήτησης, στην οποία ο χρήστης θα αναζητήσει μία συγκεκριμένη είδηση, στόχος είναι στα αποτελέσματα να εμφανίζονται μόνο τα ενδιαφέροντα άρθρα και κυρίως στην κορυφή της λίστας. Κάτι τέτοιο θα μπορούσε να επιτευχθεί με μεθόδους που θα έχουν την δυνατότητα να εξορύξουν τόσο μια κατάταξη όσο και να δημιουργήσουν μια διχοτόμηση (bipartition) ετικετών από δεδομένα πολλαπλών ετικετών [15].

Η ρύθμιση της κατηγοριοποίησης πολλαπλών ετικετών (MLC) η οποία, σε αντίθεση με τη συμβατική κατηγοριοποίηση (μονής ετικέτας), επιτρέπει σε ένα στιγμιότυπο να ανήκει σε πολλές κλάσεις ταυτόχρονα, έχει λάβει ολοένα και μεγαλύτερη προσοχή στη μηχανική μάθηση τα τελευταία έτη. Συγκεκριμένα, διάφορες προσεγγίσεις, που έχουν προταθεί, στοχεύουν στην εκμετάλλευση εξαρτήσεων μεταξύ ετικετών κλάσης. Παρ' όλο που ο ίδιος ο στόχος είναι σαφώς αξιόλογος και εμπειρικά πολλές προσεγγίσεις έχουν πράγματι αποδείξει ότι αυτός βελτιώνει την προγνωστική απόδοση, η διεξοδική θεωρητική ανάλυση του MLC εξακολουθεί να λείπει [27].

Η κατηγοριοποίηση πολλαπλών ετικετών αξιολογείται με διαφορετικές μετρικές σε σχέση με την συνηθισμένη κατηγοριοποίηση μονής ετικέτας. Υπάρχουν τρεις κύριες κατηγορίες αυτών των μετρικών οι i) example-based ii) label-based και iii) ranking-based. Στην πρώτη κατηγορία, ανήκουν μετρικές οι οποίες λαμβάνουν υπόψιν το μέσο όρο των διαφορών μεταξύ των πραγματικών και των προβλεπόμενων ετικετών επί όλου του συνόλου δοκιμής(testing set). Η δεύτερη κατηγορία περιλαμβάνει τις μετρικές, που για να υπολογίσουν το μέσο όρο για κάθε ετικέτα χρησιμοποιούν ξεχωριστές αξιολογήσεις για κάθε ετικέτα, δηλαδή αποσυνθέτουν τη διαδικασία της αξιολόγησης. Τέλος, η τρίτη κατηγορία μετρικών, εμφανίζεται συχνά στη βιβλιογραφία, όμως δεν έχει άμεση σχέση με την κατηγοριοποίηση δεδομένων πολλαπλών ετικετών [28]. Στην συγκεκριμένη ενότητα αναλύονται μερικές από τις μετρικές που έχουν προταθεί στο παρελθόν για την αξιολόγηση κατηγοριοποιητών πολλαπλών ετικετών.

Ειδικότερα, μερικές από τις πιο βασικές example-based μετρικές αξιολόγησεις αποτελούν οι Hamming loss, Accuracy, Subset accuracy, Precision, Recall και F_1. Η Hamming loss είναι αυτή που χρησιμοποιήθηκε ως μετρική αξιολόγησης των αλγορίθμων που εκτελέστηκαν στα πειράματα της συγκεκριμένης εργασίας, γι' αυτό το λόγο γίνεται λεπτομερής ανάλυση αυτής. Όμως, και οι υπόλοιπες μετρικές θα περιγραφούν για λόγους πληρότητας και γνώσης, καθώς είναι βασικές στην εξόρυξη δεδομένων.

Όσον αφορά την **Hamming loss**, η ίδια δίνει βάση στη συμμετρική διαφορά μεταξύ των πραγματικών ετικετών του δείγματος και των ετικετών που προβλέπει ο κατηγοριοποιητής για ένα σύνολο δεδομένων δοκιμής και ο τύπος της ορίζεται ως εξής:

$$Hamming - Loss = \frac{1}{m}\sum_{i=1}^{m}\frac{|\, Y_i \Delta Z_i \,|}{M} \tag{2.3}$$

όπου Δ είναι η συμμετρική διαφορά των δυο συνόλων, η οποία ισοδυναμείται με την δυαδική (XOR) λογική σχέση, Y_i το σύνολο των πραγματικών ετικετών του δείγματος και Z_i το σύνολο των ετικετών που προβλέπει ο κατηγοριοποιητής. Από τον τύπο προκύπτει ότι, όσο πιο μικρή είναι η τιμή του HL, τόσο πιο καλός είναι ο κατηγοριοποιητής.

Είναι σημαντικό να αναφερθεί ότι η εσφαλμένη κατηγοριοποίηση δεδομένων πολλαπλών ετικετών δεν

θεωρείται σωστή ή λάθος. Μια πρόβλεψη που δεν περιέχει κανένα υποσύνολο των πραγματικών κλάσεων (ετικετών) θα πρέπει να θεωρείται χειρότερη από μια πρόβλεψη που περιέχει ένα από αυτά. Έτσι, η ακρίβεια δεν αποτελεί ιδανικό μέτρο, καθώς δεν υπολογίζει ορθά τα σωστά ταξινομημένα στιγμιότυπα. Με την Hamming Loss υπολογίζεται η απώλεια, που παράγεται στη συμβολοσειρά bits των ετικετών κατά τη διάρκεια της πρόβλεψης, εκτελώντας μια πράξη XOR μεταξύ των πραγματικών και των προβλεπόμενων ετικετών και στη συνέχεια βρίσκοντας το μέσο όρο σε όλο το σύνολο δεδομένων [12].

Παρακάτω δίνεται ένα ολοκληρωμένο παράδειγμα αυτής της μετρικής, για την καλύτερη και πλήρης κατανόηση της. Υποθέτωντας οτι, ορίζεται ένα σύνολο δεδομένων με δύο (2) στιγμιότυπα και δύο (2) ετικέτες σε κάθε στιγμιότυπο τότε αν:

- **Περίπτωση 1η**: Πραγματικά στιγμιότυπα ίδια με αυτά που προβλέφτηκαν.

  Έστω οτι τα πραγματικά είναι: $Actual = 0, 1, 1, 1$ και αυτά που προβλέφθηκαν είναι: $Predicted = 0, 1, 1, 1$, τοτε έχουμε: $Actual \oplus Predicted = 0, 0, 0, 0$. Άρα η τιμή της Hamming Loss είναι: **HL=0.0**

- **Περίπτωση 2η**: Πραγματικά στιγμιότυπα τελείως διαφορετικά με αυτά που προβλέφτηκαν.

  Έστω οτι τα πραγματικά είναι: $Actual = 0, 1, 1, 1$ και αυτά που προβλέφθηκαν είναι: $Predicted = 1, 0, 0, 0$, τοτε έχουμε: $Actual \oplus Predicted = 1, 1, 1, 1$. Άρα η τιμή της Hamming Loss είναι: **HL=4/(2*2)=1**

- **Περίπτωση 3η**: Πραγματικά στιγμιότυπα μερικώς διαφορετικά με αυτά που προβλέφτηκαν.

  Έστω οτι τα πραγματικά είναι: $Actual = 0, 1, 1, 1$ και αυτά που προβλέφθηκαν είναι: $predicted = 0, 0, 0, 1$, τοτε έχουμε: $Actual \oplus Predicted = 0, 1, 1, 0$. Άρα η τιμή της Hamming Loss είναι: **HL = (1 + 1)/(2*2)=0.5**

[12].

Όσον αφορά την ακρίβεια (accuracy) ή ακρίβεια υποσυνόλου (subset accuracy), ισούται με τον λόγο των σωστά προβλεπόμενων ετικετών προς το πλήθος των συνολικών προβλεπόμενων ετικετών. Αυτό το μέτρο αξιολόγησης θεωρείται αρκετά αυστηρό, καθώς προϋποθέτει ότι το σύνολο τον προβλεπόμενων ετικετών (predicted set of labels) θα πρέπει να ταυτίζεται πλήρως με τις πραγματικές ετικέτες του συνόλου δοκιμής (true set of labels). Ο τύπος της ορίζεται ως εξής:

$$Accuracy = \frac{1}{m} \sum_{i=1}^{m} I(Z_i = Y_i) \tag{2.4}$$

όπου I(true) = 1 and I(false) = 0.

Οι υπόλοιπες μετρικές ορίζονται ως εξής:

$$Precision = \frac{1}{m} \sum_{i=1}^{m} \frac{(Y_i \cap Z_i)}{Z_i} \tag{2.5}$$

$$Recall = \frac{1}{m}\sum_{i=1}^{m}\frac{(Y_i \cap Z_i)}{Y_i} \tag{2.6}$$

$$F_1 = \frac{1}{m}\sum_{i=1}^{m}\frac{2 \mid Y_i \cap Z_i \mid}{\mid Z_i \mid + \mid Y_i \mid} \tag{2.7}$$

Για την αξιολόγηση ενός δυαδικού κατηγοριοποιητή συνήθως χρησιμοποιούνται μετρικές όπως η ακρίβεια, η ορθότητα (precision) και η ευαισθησία (recall). Οι macro-averaging και micro-averaging [29] είναι τιμές μέσων όρων που βοηθούν στον υπολογισμό όλων αυτών των μετρικών για όλες τις ετικέτες. Οι λειτουργίες αυτές υπολογίζονται σημαντικά για ακρίβεια και ευαισθησία κατά μέσο όρο, όπως και ο αρμονικός μέσος όρος τους (F-measure) στην επιστήμη ανάκτησης πληροφορίας (Information Retrieval) [30].

Υποθέτωντας ότι μία αξιολόγηση σε ένα δυαδικό πρόβλημα είναι το $B(tp, tn, fp, fn)$ και στηρίζεται στον υπολογισμό του αριθμού των προβλέψεων που είναι είτε θετικά αληθή (true positives-tp) είτε αρνητικά αληθή (true negatives-tn) είτε θετικά ψευδή (false positives-fp) είτε αρνητικά ψευδή (false negatives-fn). Αν τα $tp_\lambda, tn_\lambda, fp_\lambda, fn_\lambda$ αναπαριστούν αντίστοιχα τον αριθμό των θετικά αληθή, αρνητικά αληθή, θετικά ψευδή και αρνητικά ψευδή προβλέψεων για μία ετικέτα (λ), τότε για το (B) οι μετρικές macro-averaged και micro-averaged, υπολογίζονται ως εξής:

$$Bmacro = \frac{1}{q}\sum_{\lambda=1}^{q} B(tp_\lambda, tn_\lambda, fp_\lambda, fn_\lambda) \tag{2.8}$$

$$Bmicro = B(\sum_{\lambda=1}^{q} tp_\lambda, \frac{1}{q}\sum_{\lambda=1}^{q} tn_\lambda, \frac{1}{q}\sum_{\lambda=1}^{q} fp_\lambda, \frac{1}{q}\sum_{\lambda=1}^{q} fn_\lambda) \tag{2.9}$$

Πρέπει να σημειωθεί ότι τα αποτελέσματα που προκύπτουν είτε από τον μικρο-μέσο όρο (micro-averaging) είτε από τον μάκρο-μέσο όρο (macro-averaging) είναι τα ίδια για ορισμένα μέτρα, όπως η ακρίβεια (accuracy), για άλλα μέτρα, όπως η ορθότητα (precision) και η ευαισθησία (recall), αυτά διαφέρουν. Πρέπει να τονιστεί επίσης ότι, οι (μακρο(macro), μικρο(micro)), ακρίβεια (accuracy) και απώλεια Hamming Loss λαμβάνουν τιμές στο διάστημα [0,1], αφού η απώλεια Hamming Loss είναι στην πραγματικότητα το μέσο δυαδικό σφάλμα κατηγοριοποίησης [15].

Κεφάλαιο 3ο: Επισκόπηση αλγορίθμων μείωσης δεδομένων για σύνολα δεδομένων πολλαπλών ετικετών

Ενώ το μεγαλύτερο μέρος της έρευνας αφιερώνεται στην κατηγοριοποίηση πολλαπλών ετικετών, στο ζήτημα του υπολογιστικού κόστους σε μεγάλα σύνολα δεδομένων εκπαίδευσης πολλαπλών ετικετών εστιάζουν ελάχιστες εργασίες. Υπάρχουν λίγες έρευνες που εστιάζουν στην επιτάχυνση των οκνηρών κατηγοριοποιητών σε μεγάλα σύνολα δεδομένων εκπαίδευσης πολλαπλών ετικετών και ακόμη λιγότεροι που αφορούν τους αλγόριθμους συμπύκνωσης για αυτόν τον τύπο συνόλων δεδομένων. Σε αυτήν την ενότητα, εξετάζονται τα περιορισμένα σχετικά έργα για γρήγορη κατηγοριοποίηση πολλαπλών ετικετών.

3.1 Αλγόριθμοι βασιζόμενοι στον ENN κανόνα

Οι συγγραφείς στο [31] προτείνουν έναν νέο αλγόριθμο, τον οποίο καλούν MLeNN (MultiLabel edited Nearest Neighbor). Είναι ένας ευρετικός αλγόριθμος υποδειγματοληψίας πολλαπλών ετικετών που βασίζεται στον γνωστό κανόνα Wilson's Edited Nearest Neighbor (ENN rule) και επικεντρώνεται στην επεξεργασία μη ισορροπημένων συνόλων δεδομένων. Ο MLeNN διευθετεί δύο ζητήματα, που προκύπτουν από την εφαρμογή υποδειγματοληψίας σε δεδομένα πολλαπλών ετικετών, τον τρόπο που επιλέγονται οι υποψήφιοι και τον τρόπο εξέτασης των ταξικών διαφορών μεταξύ αυτών και των γειτόνων τους. Το πρώτο ζήτημα επιλύεται περιορίζοντας πρώτα τα στιγμιότυπα που μπορούν να λειτουργήσουν ως υποψήφιοι σε εκείνα στα οποία δεν εμφανίζεται καμία ετικέτα μειοψηφίας, ενώ το δεύτερο ορίζοντας μια μετρική απόστασης για την εύρεση διαφοράς μεταξύ οποιουδήποτε ζεύγους ετικετών. Επίσης, ο προτεινόμενος αλγόριθμος κατά την εφαρμογή του χρησιμοποιεί την μετρική IRLbl, με την οποία εκτιμά το επίπεδο ανισορροπίας για κάθε ετικέτα και την MeanIR, για την εκτίμηση του ολικού επιπέδου ανισορροπίας.

Στο άρθρο [32] αναπτύσσεται μία μέθοδος, η οποία βασίζεται στην επιλογή προτύπου χρησιμοποιώντας και αυτή τον κανόνα πλησιέστερου γείτονα. Επιπλέον, ο προτεινόμενος αλγόριθμος χρησιμοποιεί Hamming loss για να προσδιορίσει την εμφάνιση θορύβου. Η ιδέα πίσω από τον αλγόριθμο είναι απλή: οι περιπτώσεις με υψηλή απώλεια Hamming πιθανότατα βρίσκονται κοντά στα όρια αποφάσεων και για αυτόν τον λόγο πρέπει να αφαιρεθούν, όπως στην περίπτωση του κανόνα ENN. Τέλος, το πλεονέκτημα που έχει αυτή η μέθοδος είναι ότι απαιτεί ελάχιστο αποθηκευτικό χώρο και μειώνει τον χρόνο λειτουργίας των αρχικών αλγόριθμων κατηγοριοποίησης. Και τα δύο προαναφερθέντα έργα αφορούν την επεξεργασία δεδομένων. Έτσι, δεν εξετάζονται περαιτέρω στην παρούσα εργασία.

3.2 Αλγόριθμοι βασιζόμενοι σε τοπικά σύνολα

Σε αυτήν την εργασία [33] προτείνεται προσαρμογή της έννοιας του τοπικού συνόλου σε δεδομένα πολλαπλών ετικετών, καθώς και δυο νέοι αλγόριθμοι, ο ένας για επεξεργασία του συνόλου δεδομένων και ο άλλος για μείωση μεγέθους του συνόλου δεδομένων. Οι προτεινόμενοι αλγόριθμοι επιλογής προτύπου LSBo και LSSm [34] είναι προσαρμοσμένοι και βασίζονται στην έννοια των τοπικών συνόλων. Ο LSBo συμπυκνώνει τα δεδομένα εκπαίδευσης, ενώ ο LSSm τα επεξεργάζεται. Ειδικότερα, ο LSSm αφαιρεί τα στιγμιότυπα που παρουσιάζουν βλαβερότητα μεγαλύτερη από τη χρησιμότητά τους. Επίσης, είναι πολύ αποτελεσματικός για την ανίχνευση θορύβου, καθώς οι θορυβώδεις περιπτώσεις έχουν μεγάλη βλάβη και πολύ χαμηλή χρησιμότητα. Για τους ίδιους λόγους, αφαιρεί επίσης περιπτώσεις από

επικαλυπτόμενες περιοχές και μερικές περιπτώσεις πολύ κοντά σε στενά και ακανόνιστα σύνορα. Ο LSBo αποτελεί επέκταση της μεθόδου Cluster Border με τη διαφορά ότι αντί να χρησιμοποιεί τον ENN rule για φίλτρο θορύβου, χρησιμοποιεί τον LSSm. Στη συγκεκριμένη έρευνα οι δύο νέοι αλγόριθμοι που χρησιμοποιούνται, οι οποίοι βασίζονται στους LSSm και LSBo, ονομάζονται αντίστοιχα HDLSSm και HDLSBo. Αξίζει να σημειωθεί ότι ο HDLSSm έχει μία τιμή κατωφλίου για τον υπολογισμό του τοπικού συνόλου, ενώ ο HDLSBo έχει δύο κατώτατα όρια, ένα για το αρχικό βήμα φιλτραρίσματος (χρησιμοποιώντας HDLSSm) και άλλο για τους τοπικούς υπολογισμούς.

Σε προβλήματα με μία ετικέτα, το τοπικό σύνολο ενός στιγμιότυπου x αποτελείται από όλα τα στιγμιότυπα ίδιας κλάσης, που η απόσταση τους από το x είναι μικρότερη από αυτή του πλησιέστερου στιγμιοτύπου διαφορετικής κλάσης. Σε σύνολα δεδομένων πολλαπλών ετικετών, οι συγγραφείς ορίζουν ότι εκεί δεν χρειάζεται ένα τοπικό σύνολο να περιέχει το ίδιο ακριβώς labelset. Το σύνολο ετικετών των στιγμιοτύπων σε ένα τοπικό σύνολο μπορεί να διαφέρει ελαφρώς. Οι συγγραφείς χρησιμοποιούν το Hamming Loss επάνω στο σύνολο εικτετών για να υπολογίσουν τη διαφορά μεταξύ αυτών. Εάν η τιμή Hamming Loss μεταξύ των ετικετών δύο στιγμιοτύπων είναι μεγαλύτερη από ένα προκαθορισμένο κατώφλι, τότε θεωρείται ότι τα δύο στιγμιότυπα ανήκουν σε διαφορετικές κλάσεις. Οι προτεινόμενοι αλγόριθμοι PS δεν είναι παραμετρικοί και η απόδοσή τους εξαρτάται από το προκαθορισμένο κατώφλι. Επομένως, δεν εξετάζονται περαιτέρω στη συγκεκριμένη εργασία.

3.3 Συνδυασμός μεθόδων μετασχηματισμού προβλήματος με αλγόριθμους επιλογής προτύπου

Στην έρευνα [35] χρησιμοποποιούνται οι BR, LP και άλλες μέθοδοι μετασχηματισμού σε συνδυασμό με αλγόριθμους επιλογής προτύπων μιας ετικέτας, για την απόκτηση νέων αλγορίθμων επιλογής προτύπων για σύνολα δεδομένων πολλαπλών ετικετών. Στην περίπτωση του BR και των παραλλαγών του, ο προτεινόμενος αλγόριθμος αποτελείται από 3 στάδια: i) εκτέλεση επιλογής πρότυπου κατά τη συλλογή ενός μετασχηματισμένου συνόλου δεδομένων μονής ετικέτας, ii) συνδυασμός αποτελεσμάτων υπολογίζοντας ένα κατώφλι, και iii) τελική επιλογή των στιγιοτύπων. Αναλυτικότερα, πρώτα ο αλγόριθμος αντιγράφει το αρχικό σύνολο δεδομένων |L| φορές, όπου L το πλήθος των ετικετών. Κάθε αντίγραφο αφορά μια διαφορετική ετικέτα l και περιέχει όλες τις εμφανίσεις του αρχικού συνόλου δεδομένων εκπαίδευσης, με την ένδειξη “1” εάν το στιγμιότυπο φέρει την ετικέτα l και ως “0” διαφορετικά. Στη συνέχεια, χρησιμοποιείται ένας αλγόριθμος επιλογής προτύπων σε κάθε αντίγραφο και δημιουργεί |L| συμπυκνωμένα σύνολα, ένα για κάθε ετικέτα. Κάθε φορά που επιλέγεται ένα στιγμιότυπο, λαμβάνει μια ψήφο που συσσωρεύεται σε ένα διάνυσμα με τις ψήφους για όλες τις εμφανίσεις. Τέλος, ο αλγόριθμος δημιουργεί ένα πλήρες σύνολο συμπύκνωσης επιλέγοντας όλες τις περιπτώσεις με αριθμό ψήφων που υπερβαίνει το προκαθορισμένο όριο. Όπως αναφέρθηκε προηγουμένως, δίνεται έμφαση σε προσεγγίσεις χωρίς παραμέτρους. Ως εκ τούτου, αυτές οι στρατηγικές δεν εξετάζονται περαιτέρω.

3.4 Αλγόριθμος MLkNN σε GPU

Τελευταίο αλλά εξίσου σημαντικό, σε αυτό το άρθρο [36] προτείνεται μια πολύ αποτελεσματική εφαρμογή του δημοφιλούς ταξινομητή k-Nearest Neighbor (MLkNN) [16] πολλαπλών ετικετών σε GPU. Είναι γωνστό ότι, ο MLk-NN παρουσιάζει υψηλή υπολογιστική πολυπλοκότητα όταν ασχολείται με μεγάλο

αριθμό είτε στιγμιοτύπων είτε χαρακτηριστικών, γεγονός που περιορίζει την πρακτική χρηστικότητα του. Προκειμένου να μετριαστεί αυτό το πρόβλημα, προτείνεται η εφαρμογή τεσσάρων τμημάτων του MLkNN σε GPU, που αποτελείται από: (i) υπολογισμό προηγούμενων πιθανοτήτων, (ii) υπολογισμό μεταγενέστερων πιθανοτήτων, (iii) αναζήτηση k-NN και (iv) πρόβλεψη εξόδου πολλαπλών ετικετών. Στην προσαρμογή GPU του αρχικού αλγορίθμου MLkNN, ο υπολογισμός προηγούμενων πιθανοτήτων χωρίζεται σε δύο μικρότερους πυρήνες που στη συνέχεια εκτελούνται. Πρώτον, χρησιμοποιώντας ετικέτες στιγμιοτύπων από το σύνολο εκπαίδευσης, εκτελείται ένας πυρήνας που υπολογίζει αθροίσματα μετρήσεων για κάθε ετικέτα. Στην προτεινόμενη προσαρμογή GPU του αλγορίθμου ML-kNN, η διαδικασία υπολογισμού των μεταγενέστερων πιθανοτήτων χωρίζεται σε τέσσερις φάσεις: (i) εύρεση γειτονιάς για κάθε νέο στιγμιότυπο x, (ii) μέτρηση συχνότητας, (iii) άθροιση μετρήσεων και (iv) υπολογισμός των τελικών θετικών πιθανοτήτων. Στην επόμενη φάση, το τμήμα πρόβλεψης περιλαμβάνει τον υπολογισμό της γειτονιάς για κάθε στιγμιότυπο δοκιμής. Στη συνέχεια, εκτελούνται δύο πυρήνες προκειμένου να ληφθεί απόφαση σχετικά με τις περιπτώσεις και το προβλεπόμενο σύνολο ετικετών τους. Ως βήμα προεπεξεργασίας, ο πυρήνας χρησιμοποιείται για τη μέτρηση συχνότητας γειτόνων για το δεδομένο στιγμιότυπο που αντιστοιχεί στη δεδομένη ετικέτα από το σύνολο ετικετών. Αυτή η μέθοδος επιτρέπει την επίτευξη σημαντικών επιταχύνσεων σε σύνολα δεδομένων μεγάλης κλίμακας πολλαπλών ετικετών, επιτρέποντας την επέκταση των τομέων χρηστικότητας MLk-NN.

Κεφάλαιο 4ο: Αλγόριθμοι παραγωγής προτύπων

Όπως έχει ήδη αναφερθεί στο πρώτο Κεφάλαιο, αν και οι αλγόριθμοι παραγωγής προτύπων έχουν ίδιο κίνητρο με τους αλγόριθμους συμπύκνωσης, διαφέρουν στον τρόπο με τον οποίο δημιουργούν το συμπυκνωμένο σύνολο. Σε αντίθεση με τους αλγόριθμους συμπύκνωσης, που επιλέγουν κάποια "πραγματικά" δεδομένα εκπαίδευσης ως πρότυπα, οι αλγόριθμοι παραγωγής προτύπων δημιουργούν νέα πρότυπα, συνοψίζοντας τα παρόμοια σιγμιότυπα. Στην πραγματικότητα, ένας k-NN κατηγοριοποιητής που υιοθετεί την ιδέα της παραγωγής προτύπων, εκτελείται πάνω σε ένα τεχνητό σύνολο εκαπίδευσης. Σε αυτό το Κεφάλαιο αναλύονται μερικοί από τους πιο γνωστούς αλγόριθμους παραγωγής προτύπων.

4.1 Αλγόριθμος Chen και Jozwik

Οι Chen και Jozwik πρότειναν το 1996 έναν ευρέως γνωστό και αποτελεσματικό αλγόριθμο αραγωγής προτύπων. Ο Chen και Jozwik αλγόριθμος (CJA) [37] είναι παραμετρικός, καθώς ο χρήστης θα πρέπει να καθορίσει το μέγεθος του συμπυκνωμένου συνόλου που επιθυμεί. Αρχικά, ο CJA ανακτά τα πιο απομακρυσμένα αντικείμενα, x και y του συνόλου δεδομένων εκπαίδευσης. Η απόσταση μεταξύ των x και y ορίζει την διάμετρο του συνόλου δεδομένων. Έτσι, βασιζόμενος σε αυτά τα δύο στιγμιότυπα, ο CJA διαιρεί το σύνολο δεδομένων εκπαίδευσης σε δύο υποσύνολα, το S_x, το οποίο περιέχει τα στιγμιότυπα που βρίσκονται πιο κοντά στο x και το S_y, το οποίο περιέχει τα στιγμιότυπα που βρίσκονται πιο κοντά στο y. Στη συνέχεια, ο CJA επιλέγει να διαιρέσει το υποσύνολο, στο οποίο συμπεριλαμβάνονται στιγμιότυπα που ανήκουν σε περισσότρες από μία κλάσεις, δηλαδή αυτό που είναι ανομειογενές (non-homogeneous subset). Αν και τα δύο υποσύνολα είναι ανομειογενή, τότε επιλέγεται πρώτα αυτό με τη μεγαλύτερη διάμετρο. Αν όλα τα υποσύνολα που δημιουργούνται έπειτα από διαίρεση είναι ομοιογενή, τότε ο CJA συνεχίζει να εκτελείται διαιρώντας αυτά και επιλέγοντας πάντα πρώτο αυτό με τη μεγαλύτερη διάμετρο. Η διαδικασία αυτή συνεχίζει μέχρι ο αριθμός των υποσυνόλων να ισούται με τον αριθμό των υποσυνόλων που επιθυμεί κι έχει ορίσει ο χρήστης, όπως αναφέρθηκε παραπάνω. Στο τέλος, για κάθε δημιουργημένο υποσύνολο S, ο CJA υπολογίζει τον μέσο όρο των στιγμιοτύπων στο S και δημιουργεί ένα μέσο στιγμιότυπο, το οποίο θα σηματοδοτεί την ετικέτα από την πλειοψηφούσα κλάση στο S. Τα μέσα στιγμιότυπα, που έχουν δημιουργηθεί, αποτελούν το τελικό συμπυκνωμένο σύνολο δεδομένων.

Το μέσο στιγμιότυπο m για κάθε υποσύνολο S, υπολογίζεται από τον μέσο όρο των t τιμών των χαρακτηριστικών των στοιχείων $x = 1, 2, ..., \mid S \mid$, που ανήκουν στο S. Ως εκ τούτου, τα t μέσα χαρακτηριστικά $m.d_j$, υπολογίζονται ως εξής:

$$m.d_j = \frac{1}{S} \sum_{x_i \epsilon S} x_i.d_j, j = 1, 2, ..., t \tag{4.1}$$

Ο Αλγόριθμος 1 παραθέτει σε ψευδοκώδικα μια πιθανή υλοποίηση του CJA. Παρατηρείται ότι, δέχεται ως παράμετρο το σύνολο δεδομένων εκπαίδευσης (TS) και τον αριθμό των προτύπων, n, που θα δημιουργηθούν. Ο αλγόριθμος χρησιμοποιεί μια δομή δεδομένων για να αποθηκεύσει τα δημιουργημένα υποσύνολα. Αρχικά, ολόκληρο το TS αποτελεί ένα υποσύνολο και αποθηκεύεται στο TS (γραμμές 1-2). Στη συνέχεια, το ανομοιογενές υποσύνολο C με τη μεγαλύτερη διάμετρο διαιρείται σε δύο υποσύνολα (γραμμές 4-8). Αν όλα τα υποσύνολα είναι ομοιογενή, τότε ο CJA το ομοιογενές υποσύνολο C με τη

Algorithm 1 CJA

Input: TS, n
Output: CS

1: $S \leftarrow \varnothing$
2: add(S, TS)
3: **for** $i = 2$ to n **do**
4: $C \leftarrow$ select the non-homogeneous subset $\in S$ with the largest diameter
5: **if** $C == \varnothing$ {All subsets are homogeneous} **then**
6: $C \leftarrow$ select the homogeneous subset $\in S$ with the largest diameter
7: **end if**
8: $(S_x, S_y) \leftarrow$ divide C into two subsets
9: add(S, S_x)
10: add(S, S_y)
11: remove(S, C)
12: **end for**
13: $CS \leftarrow \varnothing$
14: **for** each subset $T \in S$ **do**
15: $r \leftarrow$ compute the mean instance by averaging the instances in T
16: $r.label \leftarrow$ find the most common class label in T
17: $CS \leftarrow CS \cup \{r\}$
18: **end for**
19: **return** CS

μεγαλύτερη διάμετρο (γραμμές 5-7). Και τα δύο υποσύνολα προστίθενται στο S, ενώ το C αφαιρείται (γραμμές 9-11). Η διαδικασία της κατασκευής υποσυνόλων εκτελείται μέχρι ο αριθμός των υποσυνόλων που δημιουργούνται να ισούται με το n (γραμμή 3). Το τελευταίο βήμα είναι ο μέσος υπολογισμός (ή παραγωγή προτύπου) για κάθε υποσύνολο και η συμπερίληψή του στο συμπυκνωμένο σύνολο CS (γραμμές 13-18).

Ο CJA επιλέγει το επόμενο προς διαίρεση υποσύνολο εξετάζοντας την διάμετρο του. Αυτό συμβαίνει με τη σκεψη ότι, ένα υποσύνολο με μεγάλη διάμετρο πιθανότατα περιλαμβάνει περισσότερα δεδομένα εκπαίδευσης. Επομένως, αν αυτό το υποσύνολο διαιρεθεί πρώτο, επιτυγχάνεται μεγαλύτερο ποσοστό μείωσης (reduction rate). Μία επιθυμητή ιδιότητα είναι ότι ο CJA κατασκευάζει το ίδιο συμπυκνωμένο υποσύνολο ανεξάρτητα από την κατάταξη των δεδομένων στο σύνολο εκπαίδευσης. Ωστόσο, στον τρόπο λειτουργίας του έχει δύο σημεία που θεωρούνται αδύναμα. Το πρώτο είναι ότι ο αλγόριθμος είναι παραμετρικός, καθώς ο χρήστης θα πρέπει να ορίζει ως παράμετρο το πλήθος των υποσυνόλων που επιθυμεί να δημιουργηθούν. Αυτό συνήθως περιλαμβάνει δαπανηρές διαδικασίες δοκιμής-και-σφάλματος(trial-and-error). Σε ορισμένους τομείς, αυτή η ιδιότητα ίσως αποδεικύεται επιθυμητή, καθώς επιτρέπει σε κάποιον να ελέγχει το μέγεθος των συμπυκνωμένων υποσυνόλων. Ωστόσο, απαγορεύει τον αυτόματο καθορισμό του μεγέθους του συμπυκνωμένου συνόλου σύμφωνα με τη φύση των διαθέσιμων δεδομένων. Η δεύτερη αδυναμία που παρουσιάζει ο αλγόριθμος CJA είναι ότι τα στιγμιότυπα που δεν ανήκουν στην πιο κοινή κλάση του υποσυνόλου, δεν αντιπροσωπεύονται στο συμπυκνωμένο σύνολο. Αυτό συμβαίνει, γιατί το μέσο στιγμιότυπο κάθε υποσυνόλου λαμβάνει ως ετικέτα την πιο κοινή κλάση κι έτσι στιγμιότυπα που ανήκουν σε διαφορετική κλάση απλά αγνοούνται.

4.2 Reduction by Space Partitioning

Ο αλγόριθμος των Chen και Jozwik αποτελεί τον πρόγονο της οικογένειας των Μείωση με Διαχωρισμού Χώρου (Reduction by Space Partitioning-RSP) αλγορίθμων [38], η οποία διαθέτει τρεις δημοφιλείς παραγωγής προτύπων γνωστοί και ως RSP1, RSP2 και RSP3. Οι αλγόριθμοι RSP1 και RSP2 είναι παρόμοιοι τόσο μεταξύ τους, όσο και με τον CJA. Είναι και οι δύο παραμετρικοί και δέχονται ως παράμετρο το μέγεθος των συμπυκνωμένων συνόλων. Σε αντίθεση με αυτούς, ο RSP3 καθορίζει αυτόματα το μέγεθος του συμπυκνωμένου συνόλου ανάλογα με τη φύση των δεδομένων και με τον θόρυβο που περιέχεται σε αυτά. Παρακάτω παρουσιάζονται αναλυτικότερα οι συγκεκριμένοι αλγόριθμοι.

4.2.1 RSP1

Ο RSP1 αποτελεί την πρώτη παραλλαγή του CJA και επικεντρώνεται στην αντιμετώπιση του δεύτερου μειονεκτήματος που παρουσιάζει αυτός. Για υπενθύμιση, το μειονέκτημα αυτό αφορά την αγνόηση των στιγμιοτύπων, που ανήκουν σε διαφορετική από την κοινή κλάση του υποσυνόλου, για τη δημιουργία προτύπων. Ειδικότερα, ο τρόπος που λειτουργεί ο RSP1 είναι να υπολογίζει τόσα μέσα στιγμιότυπα όσες και οι διαφορετικές κλάσεις που περιέχονται μέσα στο υποσύνολο. Επομένως, υπολογίζει ως μέσο όρο τα στιγμιότυπα που ανήκουν σε κάθε κλάση μέσα στο υποσύνολο, επιλύοντας έτσι με επιτυχία την αδυναμία του CJA. Προφανώς, το συμπυκωνμένο σύνολο που εξάγεται από τον RSP1 είναι μεγαλύτερο από αυτό που προκύπτει από τον CJA. Ωστόσο, προσπαθεί να βελτιώσει την ακρίβεια αφού λαμβάνει υπόψη όλα τα δεδομένα του συνόλου εκπαίδευσης.

Ο RSP1 όπως έχει ήδη αναφερθεί, είναι παραμετρικός, καθώς δέχεται ως είσοδο τον αριθμό του μεγέθους των υποσυνόλων που επιθυμεί να δημιουργηθούν. Επίσης, ακόμη ένα κοινό σημείο που έχουν οι δύο αυτοί αλγόριθμοι είναι το κριτήριο επιλογής του υποσυνόλου, που πρόκειται να διαιρεθεί. Αναλυτικότερα, ο RSP1 επιλέγει το επόμενο υποσύνολο που θα διαιρέσει με βάση τη διάμετρο του, όπως ακριβώς και ο CJA. Επιλέγεται το υποσύνολο με τη μεγαλύτερη διάμετρο, καθώς στηρίζεται στην ιδέα ότι ένα υποσύνολο με μεγάλη διάμετρο περιέχει περισσότερα δεδομένα κι έτσι επιτυγχάνεται μεγαλύτερο ποσοστό μείωσης.

4.2.2 RSP2

Ο RSP2 αποτελεί την δεύτερη παραλλαγή του CJA και διαφέρει μόνο σε ένα σημείο με τον RSP1. Το σημείο αυτό είναι το κριτήριο επιλογής του επόμενου προς διαίρεση υποσυνόλου. Σε αντίθεση με τους CJA και RSP1, που επιλέγουν με βάση τη διάμετρο του υποσυνόλου, ο RSP2 υιοθετεί ως κριτήριο τον υψηλότερο βαθμό επικάλυψης. Ο βαθμός επικάλυψης ισούται με τον λόγο της μέσης απόστασης μεταξύ στιγμιοτύπων που ανήκουν σε διαφορετικές κλάσεις και της μέσης απόστασης μεταξύ στιγμιοτύπων που ανήκουν στην ίδια κλάση. Αυτό το κριτήριο υποθέτει ότι τα στιγμιότυπα που ανήκουν σε μία συγκεκριμένη κλάση βρίσκονται όσο το δυνατόν πιο κοντά το ένα με το άλλο, ενώ τα στιγμιότυπα που ανήκουν σε διαφορετικές κλάσεις βρίσκονται όσο το δυνατόν πιο μακριά το ένα από το άλλο και στοχεύει στην μεγαλύτερη μείωση όγκου των δεδομένων. Σύμφωνα με την έρευνα [38], η επιλογή υποσυνόλου με βάση τον βαθμό επικάλυψης είναι καλύτερη σε σχέση με την διάμετρο.

4.2.3 RSP3

Ο RSP3 αποτελεί την τρίτη και τελευταία παραλλαγή του CJA και διαφέρει σε αρκετά σημεία από τους υπόλοιπους αλγόριθμους. Ο RSP3 υιοθετεί την έννοια της ομοιογένειας. Πιο συγκεκριμένα, διαιρεί τα υποσύνολα που δεν είναι ομοιογενή και τερματίζει μέχρι να γίνουν ομοιογενή, δηλαδή να περιέχονται μόνο τα στιγμιότυπα που ανήκουν σε μία συγκεκριμένη κλάση. Επιπλέον, μπορεί να χρησιμοποιήσει ως κριτήριο επιλογής του επόμενου προς διαίρεση υποσυνόλου είτε τη μεγαλύτερο διάμετρο είτε τον υψηλότερο βαθμό επικάλυψης. Στην πραγματικότητα, αφού όλα τα ανομοιογενή υποσύνολα έχουν διαιρεθεί, η επιλογή του κριτηρίου δεν παίζει κάποιο ρόλο. Οποιοδήποτε μη ομοιογενές υποσύνολο μπορεί να διαιρεθεί πρώτο. Είναι σημαντικό να αναφερθεί ότι, ο RSP3 είναι ο μοναδικός RSP αλγόριθμος, συμπεριλαμβανομένου και του CJA, ο οποίος καθορίζει αυτόματα το μέγεθος του συμπυκνωμένου συνόλο, γι' αυτό και δεν είναι παραμετρικός. Συνεπώς, ο RSP3 εξαλείφει και τα δύο αδύναμα σημεία του CJA. Παρατηρείται επίσης ότι, όπως στους CJA, RSP1 και RSP2, το συμπυκνωμένο σύνολο που δημιουργείται από τον RSP3 δεν εξαρτάται από τη σειρά δεδομένων στο σύνολο εκπαίδευσης.

Ο Αλγόριθμος 2 παραθέτει σε ψευδοκώδικα την υλοποίηση του RSP3. Αναλυτικότερα, χρησιμοποιείται μια απλή δομή δεδομένων S, για να συγκεντρώσει τα μη επεξεργασμένα υποσύνολα. Αρχικά, ολόκληρο το σύνολο δεδομένων εκπαίδευσης (TS) αποτελεί ένα μη επεξεργαμσένο υποσύνολο και τοποθετείται μέσα στο S (γραμμή 2). Σε κάθε επανάληψη, ο RSP3 επιλέγει το υποσύνολο C με την υψηλότερη τιμή στο κριτήριο διαχωρισμού (γραμμή 5) και ελέγχει αν το C είναι ομοιογενές ή όχι. Αν είναι ομοιογενές, το μέσο στιγμιότυπο υπολογίζεται από το μέσο όρο των στιγμιοτύπων στο C και τοποθετείται στο συμπυκνωμένο σύνολο (CS) ως πρότυπο (γραμμές 6-9). Διαφορετικά, αν το C είναι ανομοιογενές, τότε αυτό χωρίζεται σε δύο υποσύνολα τα D_1 και D_2 (γραμμή 11) όπως ακριβώς γίνεται στον CJA. Στη συνέχεια, αυτά τα δύο υποσύνολα, που δημιουργήθηκαν, προστίθενται στο S και το C αφαιρείται από το S (γραμμές 12-15). Η επανάληψη επανέλαβε-μέχρι (repeat-until loop) συνεχίζει να εκτελείται μέχρι το S να μείνει άδειο (γραμμή 16), δηλαδή όλα τα υποσύνολα να είναι ομοιογενή.

Algorithm 2 RSP3

Input: TS
Output: CS

1: $S \leftarrow \varnothing$
2: add(S, TS)
3: $CS \leftarrow \varnothing$
4: **repeat**
5: $C \leftarrow$ select a subset $\in S$
6: **if** C is homogeneous **then**
7: $r \leftarrow$ calculate the mean instance by averaging the instances in C
8: $r.label \leftarrow$ class of instances in C
9: $CS \leftarrow CS \cup \{r\}$
10: **else**
11: $(D_1, D_2) \leftarrow$ divide C into two subsets
12: add(S, D_1)
13: add(S, D_2)
14: remove(S, C)
15: **end if**
16: **until** IsEmpty(S)
17: **return** CS

Σύμφωνα με τον RSP3, παρατηρείται ότι δημιουργεί ελάχιστα πρότυπα για την εκπροσώπηση των περιοχών που δεν είναι κοντά στα σύνορα των κλάσεων και πολλά πρότυπα για την εκπροσώπηση των περιοχών που είναι κοντά στα σύνορα των κλάσεων. Είναι σίγουρο πως, το ποσοστό μείωσης που επιτυγχάνεται από τον RSP3 εξαρτάται βαθιά από τα επίπεδα θορύβου στα δεδομένα. Πιο συγκεκριμένα, όσο υψηλότερο είναι το επίπεδο θορύβου στα δεδομένα, τόσο μικρότερα είναι και τα υποσύνολα που δημιουργούνται και ως συνάρτηση, τόσο χαμηλότερο ποσοστό μείωσης επιτυγχάνεται. Είναι σημαντικό να αναφερθεί ότι, βρίσκοντας τα πιο απομακρυσμένα στιγμιότυπα σε κάθε υποσύνολο υλοποιείται υπολογισμός όλων των αποστάσεων μεταξύ των στιγμιοτύπων μεσα στο υποσύνολο. Επομένως, είναι δαπανηρές διαδικασίες που επιδεινώνουν ολόκληρο το κόστος προεπεξεργασίας του αλγορίθμου. Για το λόγο αυτό σε περιπτώσεις, όπου το σύνολο δεδομένων είναι μεγάλο, το μειονέκτημα αυτό καθιστά την εφαρμογή του απαγορευτική.

Κεφάλαιο 5ο: Προτεινόμενοι Αλγόριθμοι

5.1 Αλγόριθμος MRSP3

Στο προηγούμενο κεφάλαιο, έγινε λεπτομερής ανάλυση των αλγόριθμων παραγωγής προτύπου CJA, RSP1, RSP2 και RSP3. Ο αλγόριθμος που προτείνεται σε αυτήν την εργασία βασίζεται στον αλγόριθμο RSP3 και ονομάζεται αλγόριθμος μείωσης δεδομένων με διαχωρισμό χώρου για προβλήματα πολλαπλών ετικετών **MRSP3**. Όπως έχει ήδη αναφερθεί, ο αλγόριθμος RSP3 είναι προσαρμοσμένος για να εφαρμόζεται σε δεδομένα μονής ετικέτας, κάτι που καθιστά τη χρήση του ακατάλληλη για δεδομένα πολλαπλών ετικετών. Με τον καιρό, έγιναν προσπάθειες δημιουργίας αλγορίθμων που να είναι προσαρμοσμένοι για τέτοιου είδους προβλήματα, ωστόσο η βιβλιογραφία που αφορά αυτούς τους αλγόριθμους παραμένει περιορισμένη. Ως εκ τούτου, αυτό αποτέλεσε την αφορμή για τη σύλληψη της ιδέας της ανάπτυξης ενός νέου αλγόριθμου, ο οποίος να εφαρμόζεται σε δεδομένα πολλαπλών ετικετών.

Η επιλογή του αλγόριθμου RSP3 ως βάση για τον νέο αλγόριθμο που προτείνεται, έγινε επειδή είναι ο πιο γνωστός στη βιβλιογραφία αλγόριθμος παραγωγής προτύπων, ενώ παράλληλα είναι αποτελεσματικός σε σχέση με τους υπόλοιπους. Ας θυμηθούμε ότι, ο αλγόριθμος RSP3 αντιμετωπίζει και τα δύο ελαττώματα του αλγόριθμου CJA και κατ' επέκταση όλων των υπολοίπων, που είναι η παραμετροποίηση και η αγνόηση της κλάσης των στιγμιοτύπων που αποτελούν τη μειοψηφία σε μία ομάδα. Ειδικότερα, ο RSP3 δεν δέχεται ως παράμετρο το πλήθος των ομάδων που επιθυμεί ο χρήστης να δημιουργηθούν, διαιρεί τα υποσύνολα δεδομένων μέχρις ότου να είναι όλα ομοιογενή. Έτσι, επιλύεται όχι μόνο το πρώτο μειονέκτημα, αλλά και το δεύτερο, καθώς κάθε νέα δημιουργημένη ομάδα αποτελείται από στιγμιότυπα της ίδιας κλάσης, επομένως ακόμα και οι μειοψηφίες που υπήρχαν με τον αλγόριθμο CJA, με τον RSP3 αντιπροσωπεύονται πλήρως.

Ο προτεινόμενος αλγόριθμος MRSP3 υιοθετεί όλα τα χαρακτηριστικά του RSP3, όμως διαφέρει στον τρόπο υλοποίησης καθώς ακολουθεί μία διαφορετική προσέγγιση ως προς το τι είναι ομοιογένεια. Συνεπώς, η διαφορά του RSP3 και του προτεινόμενου MRSP3 αφορά το κριτήριο ομοιογένειας που χρησιμοποιεί ο κάθε αλγόριθμος. Ο MRSP3 θεωρεί ότι μία ομάδα είναι ομοιογενής, όταν όλα τα στιγμιότυπα που ανήκουν σε αυτή έχουν τουλάχιστον μια κοινή κλάση και όχι όλες οι κλάσεις που ανήκουν δύο στιγμιότυπα να ταυτίζονται μεταξύ τους. Επιπλέον, το πρότυπο που δημιουργείται ως αντιπρόσωπος μιας ομάδας, φέρει ως εικέτες την κοινή κλάση των στιγμιοτύπων και κλάσεις, στις οποίες ανήκουν τουλάχιστον τα μισά στιγμιότυπα.

Ο Αλγόριθμος 3 παραθέτει σε ψευδοκώδικα την υλοποίηση του MRSP3. Αναλυτικότερα, χρησιμοποιείται μια απλή δομή δεδομένων S, για να συγκεντρώσει τα μη επεξεργασμένα υποσύνολα. Αρχικά, ολόκληρο το σύνολο δεδομένων εκπαίδευσης (TS) αποτελεί ένα μη επεξεργαμσένο υποσύνολο και τοποθετείται μέσα στο S(γραμμή 2). Σε κάθε επανάληψη, ο MRSP3 επιλέγει το υποσύνολο C με την υψηλότερη τιμή στο κριτήριο διαχωρισμού (γραμμή 5) και ελέγχει αν το C είναι ομοιογενές ή όχι. Αν είναι ομοιογενές, το μέσο στιγμιότυπο υπολογίζεται από το μέσο όρο των στοιχείων στο C και τοποθετείται στο συμπυκνωμένο σύνολο (CS) ως πρότυπο (γραμμές 6-9). Διαφορετικά, αν το C είναι ανομοιογενές, τότε αυτό χωρίζεζται σε δύο υποσύνολα τα $D1$ και $D2$ (γραμμή 11), όπως ακριβώς γίνεται στον CJA. Στη συνέχεια, αυτά τα δύο υποσύνολα, που δημιουργήθηκαν, προστίθενται στο S και το C αφαιρείται από το S (γραμμές 12-15). Η επανάληψη επανέλαβε-μέχρι (repeat-until loop) συνεχίζει να εκτελείται μέχρι

το S να μείνει άδειο (γραμμή 16), δηλαδή όλα τα υποσύνολα να είναι ομοιογενή.

Algorithm 3 MRSP3

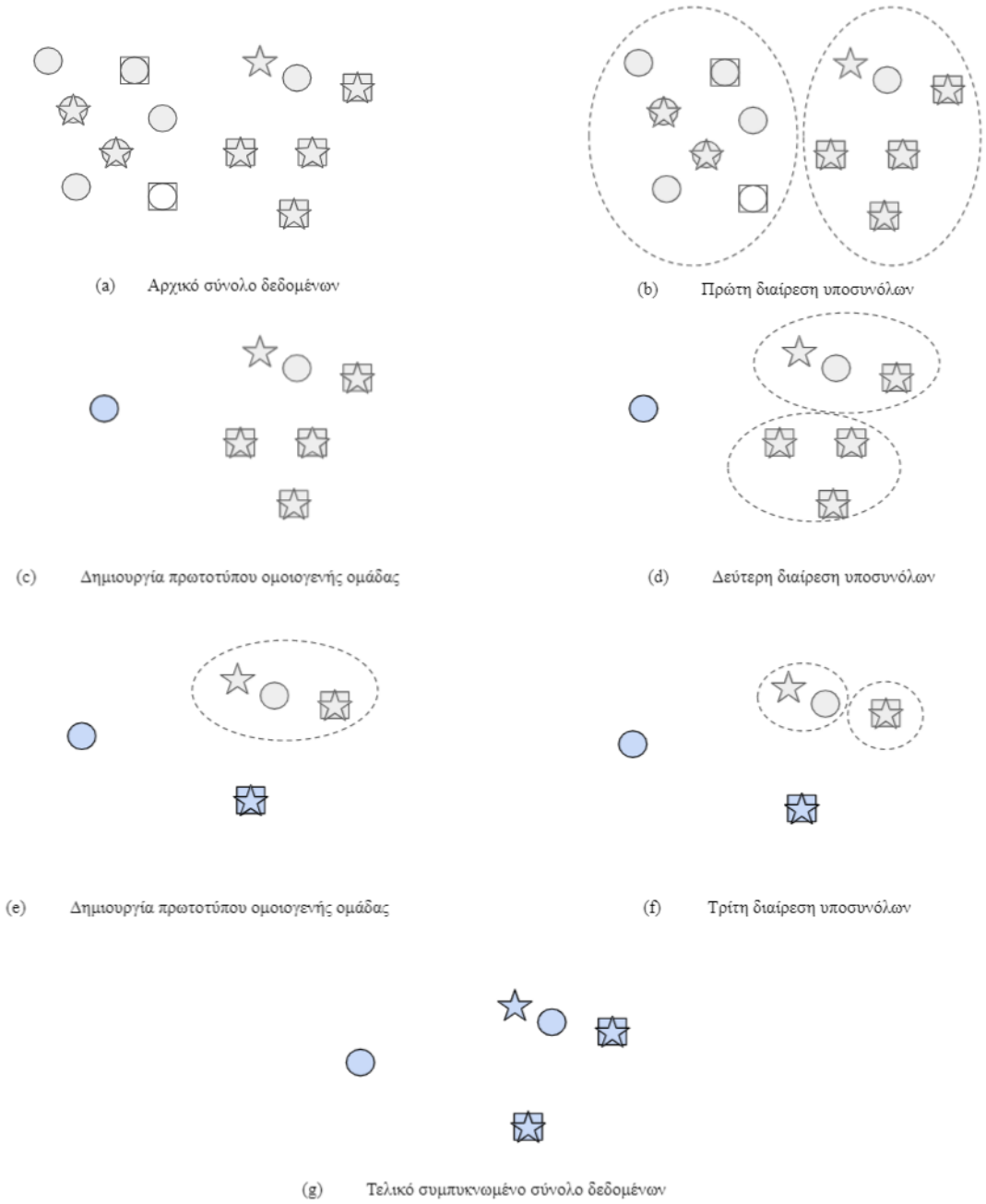
Input: TS
Output: CS

```
1:  S ← ∅
2:  add(S, TS)
3:  CS ← ∅
4:  repeat
5:      C ← select a subset ∈ S
6:      if C is homogeneous then
7:          r ← calculate the mean instance by averaging the instances in C
8:          r.label ← common class of instances in C and class of half of instances in C
9:          CS ← CS ∪ {r}
10:     else
11:         (D1, D2) ← divide C into two subsets
12:         add(S, D1)
13:         add(S, D2)
14:         remove(S, C)
15:     end if
16: until IsEmpty(S)
17: return CS
```

Παρακάτω στο Σχήμα 5.1 απεικονίζεται η εφαρμογή του MRSP3 σε ένα σύνολο δεδομένων πολλαπλών ετικετών, το οποίο βρίσκεται σε ένα δισδιάστατο χώρο δεδομένων. Στο συγκεκριμένο παράδειγμα, οι ετικέτες που μπορεί να έχει ένα στιγμιότυπο είναι τρεις: κύκλος, τετράγωνο και αστέρι. Όπως είναι γνωστό, σε ένα πρόβλημα πολλαπλών ετικετών ένα στιγμιότυπο μπορεί να ανήκει σε παραπάνω από μία κλάσεις, όπως για παράδειγμα εδώ ένα στιγμιότυπο είναι και κύκλος και αστέρι. Ο τρόπος λειτουργίας του αλγορίθμου πειγράφεται με απλά βήματα τα οποία συνοδεύονται και με εικόνες, για την καλύτερη κατανόηση του.

Πρώτα, έχουμε το αρχικό σύνολο δεδομένων στον δισδιάστατο χώρο πάνω στο οποίο θα εκτελεστεί ο αλγόριθμος MRSP3 (βήμα (a)). Αρχικά, ο MRSP3 υπολογίζει την απόσταση μεταξύ των πιο απομακρυσμένων στιγμιοτύπων του συνόλου δεδομένων. Φαίνεται πως τα πιο απομακρυσμένα στιγμιότυπα είναι ο κύκλος, που βρίσκεται πάνω και τέρμα αριστερά με το τετράγωνο-αστέρι πάνω και τέρμα δεξιά. Έτσι, όποιο στιγμιότυπο βρίσκεται πιο κοντά στον κύκλο, ανήκει στο ίδιο υποσύνολο με αυτό, όπως το ίδιο συμβαίνει και με όποιο στιγμιότυπο βρίσκεται πιο κοντά στο τετράγωνο-αστέρι. Με αυτόν τον τρόπο, δημιουργούνται δύο υποσύνολα (βήμα (b)). Έπειτα, ελέγχει τυχαία πρώτα το υποσύνολο αριστερά αν είναι ομοιογενές. Εφόσον, όλα τα στιγμιότυπα είναι κύκλοι, τότε έχουν μια κοινή κλάση την κλάση "κύκλος", άρα το υποσύνολο αυτό είναι ομοιογενές. Στη συνέχεια, ο MRSP3 υπολογίζει τον μέσο όρο των στιγμιοτύπων, για να δημιουργήσει το πρότυπο του υποσυνόλου και του θέτει ως ετικέτα μόνο την ετικέτα "κύκλος", καθώς καμία άλλη ετικέτα δεν φέρουν τουλάχιστον τα μισά στιγμιότυπα (βήμα (c)). Την ίδια διαδικασία εκτελεί ο MRSP3 και για το δεξί υποσύνολο. Έτσι, προκύπτουν τα δύο υποσύνολα που φαίνονται στο βήμα (d), με βάση τα δύο πιο απομακρυσμένα στιγμιότυπα. Ελέγχοντας τυχαία πρώτα το κάτω υποσύνολο, προκύπτει ότι είναι ομοιογενές, καθώς όλα τα στιγμιότυπα έχουν κοινές κλάσεις τις κλάσεις "τετράγωνο" και "αστέρι". Έτσι, υπολογίζεται το πρότυπο για αυτό το υποσύνολο, το οποίο

φέρει ως ετικέτες “τετράγωνο” και “αστέρι” (βήμα (e)). Τέλος, ο MRSP3 ελέγχει το πάνω υποσύνολο αν είναι ομοιογενές. Αυτό όμως δεν είναι κι έτσι διασπάται σε δύο υποσύνολα, όπως φαίνεται στο βήμα (f). Εφόσον, στο δεξί υποσύνολο ανήκει ένα στιγμιότυπο, αυτό θα είναι και το πρότυπο του. Όσον αφόρά το αριστερό υποσύνολο, το οποίο δεν είναι ομοιογενές διασπάται σε δύο υποσύνολα, τα οποία έχουν το καθένα από ένα στιγμιότυπο για αυτό και αυτά τα στιγμιότυπα είναι και τα πρότυπά τους. Το τελικό συμπυκνωμένο σύνολο απεικονίζεται στο βήμα (g), όπου πλέον το σύνολο δεδομένων έχει μειωθεί αρκετά και η κάθε ετικέτα εκπροσωπείται. Αυτός είναι ο τρόπος λειτουργίας του νέου αλγόριθμου MRSP3.



Σχήμα 5.1: Παράδειγμα εφαρμογής αλγόριθμου MRSP3

5.2 Υλοποίηση MRSP3

Ο αλγόριθμος MRSP3 υλοποιήθηκε σε γλώσσα C++, επειδή βασίστηκε σε μία υπάρχουσα υλοποίηση του αλγόριθμου RSP3. Ελάχιστες ήταν οι αλλαγές που έπρεπε να γίνουν, καθώς οι δύο αλγόριθμοι δεν διαφέρουν πολύ. Παρακάτω, παρατίθεται το πιο σημαντικό κομμάτι του κώδικα του αλγορίθμου MRSP3, που αποτελεί την ουσιαστική διαφορά από τον αλγόριθμο RSP3.

Αρχικά στο Σχήμα 5.2, παρουσιάζεται το κομμάτι του κώδικα του MRSP3 που αφορά την ομοιογένεια ενός συνόλου, η οποία αποτυπώνεται στη μέθοδο $homogenity$, Αρχικά, δέχεται ως παραμέτρους τον πίνακα $TrainData$, ο οποίος περιέχει όλα τα στιγμιότυπα εκπαίδευσης, την μεταβλητή $partitions$, η οποία ισούται με τον αριθμό των υποσυνόλων, τον πίνακα hom, ο οποίος είναι τύπου boolean και είναι αποθηκευμένα σε αυτόν όλα τα υποσύνολα με τιμές “true” ή “false”, ανάλογα με το αν είναι ομοιογενή ή όχι. Τέλος, δέχεται ως παράμετρο την μεταβλητή $dataNumV$, η οποία εκφράζει τον αριθμό των στιγμιοτύπων. Έπειτα, γίνεται αρχικοποίηση ορισμένων μεταβλητών, οι οποίες θα χρειαστούν παρακάτω, όπως η fp, που δηλώνει τον αριθμό των στιγμιοτύπων που ανήκουν στο ίδιο υποσύνολο. Στις γραμμές 283-287, υπολογίζεται το πλήθος των στιγμιοτύπων που ανήκουν στο ίδιο υποσύνολο, δηλαδή το fp. Στη συνέχεια στη γραμμή 291, γίνεται έλεγχος αν το υποσύνολο έχει μόνο ένα στιγμιότυπο. Αν ναι, τότε στον πίνακα hom αποθηκεύτεται για αυτό το υποσύνολο η τιμή “true”. Διαφορετικά, εκτελείται το $else$ στη γραμμή 249 που είναι και αυτό, στο οποίο πρέπει να δωθεί η μεγαλύτερη προσοχή. Αρχικά, αρχικοποιείται ο μετρητής cl για κάθε κλάση, στον οποίο αποθηκεύτεται το πλήθος των “1” που υπάρχουν για κάθε κλάση. Έπειτα, με την επανάληψη for στις γραμμές 302-309, ελέγχω εάν τα στιγμιότυπα σε ένα υποσύνολο έχουν μία κοινή κλάση. Μέσα στην for αρχικά, ελέγχεται πόσα στιγμιότυπα ανήκουν στο ίδιο υποσύνολο (γραμμές 303-304). Στη συνέχεια, στην επόμενη for (γραμμές 305-306) αθροίζονται για κάθε κλάση οι τελευταίες στήλες των συνόλων δεδομένων με τιμές “0” και “1” και αποθηκεύονται στον πίνακα cl. Για την καλύτερη κατανόηση της λειτουργίας του cl, παρακάτω αναλύεται ένα μικρό παράδειγμα. Έπειτα, στις γραμμές 310-312, ελέγχεται αν κάποια από τις κλάσεις έχει άθροισμα τόσο, όσα και τα στιμιότυπα που βρίσκονται στο υποσύνολο αυτό. Αν ναι, τότε το υποσύνολο αυτό είναι ομοιογενές, διαφορειτικά όχι.

Στον Πίνακα 5.1, παρουσιάζεται ένα παράδειγμα χρήσης του cl πίνακα που χρησιμοποιείται στον παραπάνω αλγόριθμο. Είναι πολύ σημαντική η πλήρης κατανόηση του, γιατί με βάση αυτό προκύπτει το αποτέλεσμα αν ένα υποσύνολο είναι ομοιεγενές ή όχι. Υποθέτωντας ότι, έχουμε 4 στιγμιότυπα σε ένα σύνολο δεδομένων και έχουν τις αντίστοιχες ετικέτες που αναγράφονται στον πίνακα. Όπου υπάρχει τιμή “1”, το στιγμιότυπο φέρει αυτή την ετικέτα, διαφορετικά όχι. Για τον υπολογισμό του cl κάθε κλάσης, απλά αθροίζονται κάθετα οι στήλες, ώστε να προκύψει ο αριθμός των στιγμιοτύπων που ανήκουν σε κάθε κλάση. Τέλος, εφόσον υπολογιστεί ολόκληρος ο πίνακας cl, αν έστω και μία τιμή του ισούται με τον αριθμό των στιγμιοτύπων, στην συγκεκριμένη περίπτωση με 4, τότε το υποσύνολο είναι ομοιεγενές. Με βάση τον MRSP3, η ετικέτα που θα εκχωρηθεί στο πρότυπο που θα δημιουργηθεί, θα είναι μόνο η L3, καθώς αυτή έχει τιμή στον Πίνακα 5.1 μεγαλύτερη από το μισό του πλήθους των γειτόνων.

5.3 Υλοποίηση BRK-NNa

Ο BRk-NNa είναι ένας αλγόριθμος κατηγοριοποίησης, ο οποίος επεκτείνει τον k εγγύτερων γειτόνων (k-NN) αλγόριθμο, ώστε να εφαρμόζεται σε δεδομένα πολλαπλών ετικετών. Είναι ουσιαστικά ο συνδυα-

```
void homogenity(TrainItem trainData[], int partitions, bool hom[], int dataNumV){
   int i, j, fp, p, countp;
   bool flag;
   int *cl = new int[CLASSES];

   for (p=1;p<=partitions; p++){
      if (hom[p-1]){
         continue;
      }
      fp=0;
      for (i=0; i<dataNumV; i++) {
         if (trainData[i].groupid==p){
            fp++;
            if (fp>1)
               break;
         }
      }

      if (fp<=1){
         hom[p-1]=true;
      }
      else{
         //check if the current partition is homogeneous
         flag=false;

         for (i = 0; i<CLASSES; i++){
            cl[i] = 0;
         }
         countp=0; //count the items in partition
         for (i=0; i<dataNumV; i++){
            if (trainData[i].groupid==p){
                countp++;
                for (j = 0; j<CLASSES; j++){
                    cl[j] += trainData[i].classAttr[j];
                }
            }
         }
         for (i = 0; i<CLASSES; i++){
            if (cl[i] == countp){
                flag = true;
                break;
            }
         }

         hom[p-1]=flag;
      }
   }
}
```

Σχήμα 5.2: Κομμάτι κώδικα του MRSP3

Πίνακας 5.1: Παράδειγμα υπολογισμού του πίνακα cl

#Ex	L1	L2	L3	L4	L5
1	0	1	1	0	0
2	0	0	1	0	0
3	1	0	1	0	1
4	0	1	1	1	1
cl	**1**	**2**	**4**	**1**	**2**

σμός k-NN με την μέθοδο μετασχηματισμού προβλήματος binary relevance (BR). Αυτή η έκδοση του κατηγοριοποιητή εκχωρεί τις ετικέτες που έχουν εκχωρηθεί σε τουλάχιστον από τους μισούς γείτονες. Η υλοποίηση του έγινε σε python, γιατί υπάρχει έτοιμη στο scikit.learn (http://scikit.ml/api/skmultilearn.ada-

pt.brknn.html). Παρακάτω στο Σχήμα 5.3, απεικονίζεται ο κώδικας του BRk-NNa και ακολουθεί πλήρης επεξήγηση του.

Αρχικά, έχει γίνει εισαγωγή των δεδομένων και στις μεταβλητές X_train και y_train έχουν αποθηκευτεί τα χαρακτηριστικά και οι τιμές των ετικετών του συνόλου δεδομένων εκπαίδευσης αντίστοιχα. Επίσης, στις μεταβλητές X_test και y_test έχουν αποθηκευτεί τα χαρακτηριστικά και οι τιμές των ετικετών του συνόλου δεδομένων δοκιμής αντίστοιχα. Πρώτα αρχικοποιείται ο κατηγοριοποιητής, δέχοντας ως παράμετρο το k, για τον προσδιορισμο του πλήθους των εξεταζόμενων γειτόνων. Στη συνέχεια, με την μέθοδο fit, εκπαιδεύονται τα δεδομένα και με την μέθοδο predict γίνεται η πρόβλεψη των ετικετών του συνόλου δεδομένων δοκιμής. Τα αποτελέσματα αποθηκεύονται στη μεταβλητή $predictions$, ώστε να χρησιμοποιηθούν στην αξιολόγηση του μοντέλου κατηγοριοποίησης. Έπειτα, με την μέθοδο hamming_loss, η οποία δέχεται ως παράμετρο τις πραγματικές τιμές ετικετών του συνόλου δοκιμής και τις τιμές που προβλέφθηκαν, υπολογίζεται η τιμή της μετρικής Hamming-Loss. Τέλος, υπολογίζονται οι μέσοι όροι των τιμών Hamming-Loss και CPU time.

```
27      classifier = BRkNNaClassifier(k=1)
28      print("Classifier: ", classifier)
29
30            # train
31      classifier.fit(X_train, y_train)
32
33      # predict
34      predictions = classifier.predict(X_test)
35
36      #Hamming-Loss of predict(X)
37      hl = hamming_loss(y_test, predictions)
38      sum_hl = sum_hl + hl
39
40      #CPU time
41      time = p.cpu_times()[0]
42      cpu_time = cpu_time + time
43
44  print("Mean CPU time: ", cpu_time/5 , "Mean hamming-loss: ", sum_hl/5)
```

Σχήμα 5.3: Υλοποίηση κώδικα του BRk-NNa

5.4 Υλοποίηση BRK-NNb

Ο BRk-NNb είναι ένας αλγόριθμος κατηγοριοποίησης, ο οποίος επεκτείνει τον k εγγύτερων γειτόνων (k-NN) αλγόριθμο, ώστε να εφαρμόζεται σε δεδομένα πολλαπλών ετικετών. Είναι ουσιαστικά ο συνδυασμός k-NN με την μέθοδο μετασχηματισμού προβλήματος binary relevance (BR). Αυτή η έκδοση του ταξινομητή εκχωρεί τις πιο δημοφιλείς ετικέτες m των γειτόνων, όπου m είναι ο μέσος αριθμός ετικετών που έχουν εκχωρηθεί στους γείτονες του στγμιοτύπου. Η υλοποίηση του έγινε σε python, γιατί υπάρχει έτοιμη στο scikit.learn (http://scikit.ml/api/skmultilearn.adapt.brknn.html). Παρακάτω στο Σχήμα 5.4, απεικονίζεται ο κώδικας του BRk-NNb και ακολουθεί πλήρης επεξήγηση του.

Αρχικά, έχει γίνει εισαγωγή των δεδομένων και στις μεταβλητές X_train και y_train έχουν αποθηκευτεί τα χαρακτηριστικά και οι τιμές των ετικετών του συνόλου δεδομένων εκπαίδευσης αντίστοιχα. Επίσης, στις μεταβλητές X_test και y_test έχουν αποθηκευτεί τα χαρακτηριστικά και οι τιμές των ετικετών του συνόλου δεδομένων δοκιμής αντίστοιχα. Έτσι, πρώτα αρχικοποιείται ο κατηγοριοποιητής, δέχοντας ως παράμετρο το k, για τον προσδιορισμο του πλήθους των εξεταζόμενων γειτόνων. Στη συνέχεια, με την μέθοδο fit, εκπαιδεύονται τα δεδομένα και με την μέθοδο predict γίνεται η πρόβλεψη των ετικετών του συνόλου δεδομένων δοκιμής. Τα αποτελέσματα αποθηκεύονται στη μεταβλητή $predictions$, ώστε να χρησιμοποιηθούν στην αξιολόγηση του μοντέλου κατηγοριοποίησης. Έπειτα, με την μέθοδο hamming_loss, η οποία δέχεται ως παράμετρο τις πραγματικές τιμές ετικετών του συνόλου δοκιμής και τις τιμές που προβλέφθηκαν, υπολογίζεται η τιμή της μετρικής Hamming-Loss. Τέλος, υπολογίζονται οι μέσοι όροι των τιμών Hamming-Loss και CPU time.

```
27      classifier = BRkNNbClassifier(k=1)
28      print("Classifier: ", classifier)
29
30            # train
31      classifier.fit(X_train, y_train)
32
33      # predict
34      predictions = classifier.predict(X_test)
35
36      #Hamming-Loss of predict(X)
37      hl = hamming_loss(y_test, predictions)
38      sum_hl = sum_hl + hl
39
40      #CPU time
41      time = p.cpu_times()[0]
42      cpu_time = cpu_time + time
43
44  print("Mean CPU time: ", cpu_time/5 , "Mean hamming-loss: ", sum_hl/5)
```

Σχήμα 5.4: Υλοποίηση κώδικα του BRk-NNb

Κεφάλαιο 6ο: Πειραματική Μελέτη

Η συγκεκριμένη έρευνα και τα πειράματα, που εκτελέστηκαν στα πλαίσια αυτής, αφορούν την κατηγοριοποίηση δεδομένων πολλαπλών ετικετών. Αφιερώθηκε αρκετός χρόνος στην έυρεση κατάλληλων συνόλων δεδομένων, για την γρηγορότερη και αποτελεσματικότερη εξαγωγή συμπερασμάτων. Επιπλέον, για την επίτευξη στόχου της παρούσας εργασίας σημαντικό ρόλο κατείχαν η επιλογή λογισμικού και γλώσσας προγραμματισμού για την υλοποίηση των αλγορίθμων. Όλοι αυτοί οι παράγοντες παρουσιάζονται σε αυτό το κεφάλαιο και ολοκληρώνουν την εικόνα της έρευνας και των διαδικασιών που ακολουθήθηκαν.

6.1 Πειραματική Διαμόρφωση

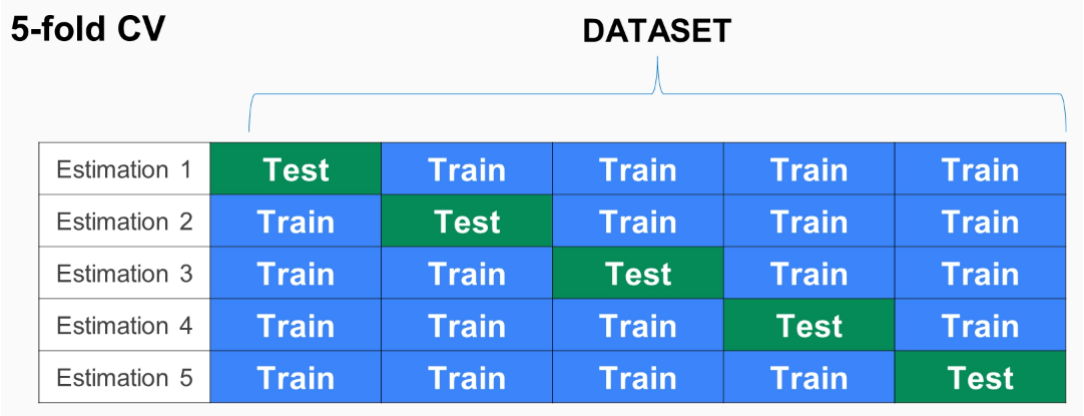
Αρχικά, για την καλύτερη εικόνα των πειραμάτων πρέπει να ανφερθεί πως τα χαρακτηριστικά των συνόλων δεδομένων είναι αριθμητικά, ενώ οι ετικέτες είναι δυαδικές, δηλαδή έχουν είτε την τιμή “0” είτε την τιμή “1”. Ως γνωστόν, η τιμή “0” υπάρχει όταν η ετικέτα απουσιάζει από το συγκεκριμένο στιγμιότυπο, ενώ η τιμή “1” όταν παρουσιάζεται στο συγκεκριμένο στιγμιότυπο. Όμως, για το λόγο ότι καθένα χαρακτηριστικό έχει διαφορετικό εύρος τιμών, αυτό οδηγεί σε λανθασμένη κατηγοριοποίηση, καθώς έτσι αποδίδει διαφορετική βαρύτητα σε αυτό. Έτσι, η πρώτη τεχνική που εφαρμόστηκε πάνω στα χαρακτηριστικά των δεδομένων ήταν η κανονικοποίηση, για να ανήκουν όλα στο ίδιο εύρος τιμών, στην συγκεκριμένη περίπτωση στο διάστημα [0,1]. Αυτή η διαδικασία υλοποιήθηκε για όλα τα σύνολα δεδομένων με την συγγραφή και εκτέλεση συγκεκριμένου προγράμματος σε γλώσσα C. Με αυτόν τον τρόπο, όλα τα χαρακτηριστικά καθε συνόλου δεδομένων κατέχουν πλεόν το ίδιο βάρος χωρίς την αλλαγή της σημασίας των τιμών τους.

Επόμενο βήμα αποτελεί η επιλογή μεθόδου επικύρωσης του μοντέλου που χρησιμοποιήθηκε στα πειράματα. Η μέθοδος που επιλέχθηκε είναι η 5-fold-cross-validation. Γενικότερα, η Cross-Validation είναι μια στατιστική μέθοδος αξιολόγησης και σύγκρισης αλγορίθμων μάθησης διαιρώντας τα δεδομένα σε δύο τμήματα: το ένα χρησιμοποιείται για να μάθει ή να εκπαιδεύσει ένα μοντέλο και το άλλο χρησιμοποιείται για την επικύρωση του μοντέλου. Η απόδοση κάθε αλγορίθμου σε κάθε fold μπορεί να παρακολουθείται χρησιμοποιώντας κάποια προκαθορισμένη μέτρηση απόδοσης, όπως ακρίβεια (accuracy) ή hamming-loss για προβλήματα κατηγοριοποίησης ή το μέσο τετραγωνικό σφάλμα (MSE) για προβλήματα παλινδρόμησης [39].

Η τεχνική k-folds είναι η πιο γνωστή και κατανοητή μορφή της μεθόδου cross-validation, καθώς οδηγεί σε ένα λιγότερο προκατειλημμένο μοντέλο σε σύγκριση με άλλες μεθόδους. Αυτό συμβαίνει, γιατί διασφαλίζει ότι κάθε σημείο από το αρχικό σύνολο δεδομένων έχει την ευκαιρία να εμφανιστεί στο σύνολο εκπαίδευσης και στο σύνολο δοκιμής. Η διαδικασία έχει μια μόνο παράμετρο που ονομάζεται k που αναφέρεται στον αριθμό ομάδων στις οποίες πρέπει να διαιρεθεί ένα δείγμα δεδομένων. Ως εκ τούτου, η διαδικασία συχνά ονομάζεται k-fold cross-validation. Εάν επιλεγεί μια τιμή για το k που δεν μπορεί να διαιρέσει ομοιόμορφα το δείγμα δεδομένων, τότε μια ομάδα θα περιέχει το υπόλοιπο των δειγμάτων. Είναι προτιμότερο να χωριστεί το δείγμα δεδομένων σε ομάδες k με τον ίδιο αριθμό δειγμάτων, έτσι ώστε να είναι όλα ισοδύναμα [40]. Η μέθοδος αυτή αρχικά, κατανέμει τα δεδομένα τυχαία σε ισομεγέθη τμήματα ή πτυχές (folds). Στη συνέχεια πραγματοποιούνται επαναλήψεις εκπαίδευσης (training) και ε-

πικύρωσης (validation) έτσι ώστε σε κάθε επανάληψη διαφορετικό τμήμα των δεδομένων να παραμένει για επικύρωση ενώ οι υπόλοιπες k-1 πτυχές να χρησιμοποιούνται για εκπαίδευση. Αυτή είναι μια από τις καλύτερες προσεγγίσεις εάν υπάρχουν περιορισμένα δεδομένα εισόδου. Στην εξόρυξη δεδομένων και μηχανική εκμάθηση η 10-fold cross-validation (k = 10) είναι η πιο κοινή [39].

Στην παρακάτω εικόνα 6.1, απεικονίζεται ένα σχήμα της μεθόδου k-fold cross-validation, για την καλύτερη κατανόηση της. Παρατηρείται λοιπόν, ότι σε κάθε επανάληψη χρησιμοποιείται το μεγαλύτο μέρος του συνόλου δεδομένων για εκπαίδευση και το υπόλοιπο μέρος για επικύρωση. Επίσης, σε κάθε επανάληψη τα μέρη των δεδομένων αλλάζουν διαδοχικά ώστε να υπάρχει αμεροληψία στις προβλέψεις και στην αξιολόγηση του μοντέλου. Τέλος, όπως αναφέρθηκε και παραπάνω, τα τμήματα στα οποία χωρίζονται τα δεδομένα είναι ίσα μεταξύ τους, για να προκύψουν ορθότερα αποτελέσματα.



Σχήμα 6.1: Παράδειγμα 5-fold Cross-Validation μεθόδου

Τα πειράματα της έρευνας εκτελέστηκαν σε υπολογιστή με επεξεργαστή **Intel Core i3-5005U 2.00 GHz** και **μνήμη RAM 4.00 GB**. Επίσης χρησιμοποιήθηκε το **PyCharm**, ένα ολοκληρωμένο περιβάλλον ανάπτυξης που χρησιμοποιείται στον προγραμματισμό υπολογιστών, σε γλώσσα **python 3.9**, για την υλοποίηση του αλγόριθμου BRk-NN. Για την υλοποίηση αυτή χρειάστηκε να γίνει εισαγωγή ορισμένων βιβλιοθηκών από το **scikit-learn**. Μερικές από αυτές είναι η "sklearn.metrics import hamming_loss" για τον υπολογισμό της μετρικής hamming-loss, η "psutil" για τον χρόνο CPU που χρειάστηκε ο αλγόριθμος για να εκτελεστεί, η "skmultilearn.adapt import BRkNNaClassifier" για την εισαγωγη του BRk-NN αλγόριθμου, η "numpy" για την χρήση πινάκων και η "scipy.sparse import csc_matrix" για τη χρήση matrix πινάκων.

Τέλος, για την ολοκλήρωση της περιγραφής του τρόπου εκτέλεσης των πειραμάτων πρέπει να σημειωθεί ότι επειδή χρησιμοποιήθηκε η μέθοδος 5-folds τόσο ο αλγόριθμος BRK-NN όσο και ο MRSP3 εκτελούνται 5 φορές ο καθένας και αποτελούνται απο τα ζεύγη του συνόλου δεδομένων (εκπαίδευσης και επικύρωσης). Σε κάθε επανάληψη υπολογίζεται η τιμή της μετρικής Hamming-Loss και στο τέλος υπολογίζεται ο μέσος όρος των 5 τιμών που προέκυψαν από κάθε επανάληψη, για τη συνολική απώλεια. Επίσης, εκτός από την hamming-loss υπολογίζεται με τον ίδιο ακριβώς τρόπο και ο CPU χρόνος που χρειάστηκε ο αλγόριθμος για να εκτελεστεί. Επιπλέον, επισημαίνεται ότι ο αλγόριθμος MRSP3 υλοποιήθηκε σε C++, επειδή βασίστηκε σε μία ήδη υπάρχουσα υλοποίση του RSP3 και η τροποποίηση του θεωρήθηκε εύκολη.

6.2 Περιγραφή Συνόλων Δεδομένων

Τα σύνολα δεδομένων που χρησιμοποιούνται στις πειραματικές μελέτες αυτής της διπλωματικής είναι σύνολα πολλαπλών ετικετών, καθώς όλη η εργασία αφορά την κατηγοριοποίηση αυτών. Τα σύνολα αυτά απαιτούν κάποια χαρακτηριστικά, όπως να είναι δυαδικά (binary) και να έχουν πολλαπλές κλάσεις (multi-label). Επιπλέον, για την εκτέλεση του αλγόριθμου μείωσης δεδομένων MRSP3 και του k-εγγύτερων γειτόνων BRk-NN χρησιμοποιώντας την Ευκλείδεια απόσταση είναι απαραίτητο όλα τα χαρακτηριστικά να είναι αριθμητικά δεδομένα. Με αυτόν τον τρόπο, διευκολύνεται η αξιολόγηση των αποτελεσμάτων μεταξύ διαφορετικών συνόλων δεδομένων.

Στις παρακάτω ενότητες γίνεται περιγραφή των συνόλων δεδομένων που χρησιμοποιήθηκαν στα πειράματα. Εννέα σύνολα δεδομένων πολλαπλών ετικετών χρησιμοποιήθηκαν συνολικά, τα οποία είναι διαθέσιμα στο Mulan [19] και στην ιστοσελίδα [41].

6.2.1 Σύνολο Δεδομένων CAL500

Το Computer Audition Lab (CAL500) είναι ένα σύνολο δεδομένων μουσικής, που αποτελείται από 502 τραγούδια. Το κάθε τραγούδι σχολιάστηκε χειροκίνητα από τουλάχιστον τρεις σχολιαστές, οι οποίοι χρησιμοποίησαν ένα λεξιλόγιο 174 ετικετών σχετικά με τις σημασιολογικές έννοιες. Αυτές οι ετικέτες καλύπτουν 6 σημασιολογικές κατηγορίες: ενορχήστρωση, φωνητικά χαρακτηριστικά, είδη, συναισθήματα, ακουστική ποιότητα του τραγουδιού και όροι χρήσης. Για να δημιουργηθούν νέες σημασιολογικές ετικέτες, πληρώθηκαν 66 προπτυχιακοί μαθητές, για να σχολιάσουν το μουσικό σώμα με τις σημασιολογικές έννοιες του λεξιλογίου. Οι συμμετέχοντες επιβραβεύτηκαν με 10$ ανά ώρα για να ακούσουν και να σχολιάσουν μουσική σε έναν υπολογιστή στο εργαστήριο του πανεπιστημίου. Η διεπαφή σχολιασμού που βασίζεται σε υπολογιστή περιείχε ένα MP3 player και μια φόρμα HTML. Το έντυπο HTML, αποτελούταν από ένα ή περισσότερα κουτιά (ratio buttons) ή πλαίσια ελέγχου (check boxes) για καθεμία από τις 135 έννοιες. Κάθε σχολιασμός διαρκούσε περίπου 5 λεπτά και οι περισσότεροι συμμετέχοντες ανέφεραν ότι η ακρόαση και η εμπειρία σχολιασμού ήταν απολαυστική. Συλλέχτηκαν τουλάχιστον τρεις σημασιολογικοί σχολιασμοί για καθένα από τα 500 τραγούδια και συνολικά 1708 σχολιασμοί.

Για κάθε τραγούδι δημιουργήθηκε μια συλλογή από ανθρώπινους σχολιασμούς, όπου κάθε σχολιασμός είναι ένα διάνυσμα αριθμών που εκφράζουν την απάντηση ενός θέματος σε ένα σύνολο λέξεων. Για κάθε λέξη, ο σχολιαστής έχει απαντήσει με 1 εάν πιστεύει ότι το τραγούδι αντιπροσωπευέται από τη λέξη και 0 εάν δεν είναι σίγουρος. Στη συνέχεια, συλλέχθηκαν όλοι οι σχολιασμοί για κάθε τραγούδι και συμπιέστηκαν σε ένα μόνο διάνυσμα σχολιασμού παρατηρώντας το επίπεδο συμφωνίας πάνω από όλους τους σχολιαστές. Τα τελικά σημασιολογικά βάρη είναι:

$$[y]_i = max(0, [\frac{\#(PositiveVotes) - \#(NegativeVotes)}{\#(Annotations)}]_i). \tag{5.1}$$

Για παράδειγμα, για ένα συγκεκριμένο τραγούδι, αν τέσσερις σχολιαστές το επισημάνουν με τιμές +1, +1, 0, -1, τότε το $[y]_i = \frac{1}{4}$. Τα σημασιολογικά βάρη χρησιμοποιούνται για τον υπολογισμό των παραμέτρων. Για σκοπούς αξιολόγησης, δημιουργείται επίσης ένα δυαδικό διάνυσμα σχολιασμού για κάθε τραγούδι. Για να δημιουργηθεί αυτό το διάνυσμα, ένα τραγούδι έχει ως ετικέτα μια λέξη, αν τουλάχιστον δύο άτομα ψηφίσουν για τη λέξη και υπάρχει ένα υψηλό επίπεδο συμφωνίας μεταξύ όλων των θεμάτων.

Αυτό διασφαλίζει ότι, κάθε θετική ετικέτα είναι αξιόπιστη. Τέλος, αφαιρούνται όλες τις λέξεις που αντιπροσωπεύονται από λιγότερα από πέντε τραγούδια. Αυτό μειώνει το σύνολο των 237 λέξεων σε σύνολο 174 λέξεων [42]. Αναλυτικότερα, τα χαρακτηριστικά του συνόλου δεδομένων είναι σύμφωνα με τον πίνακα 6.1

Πίνακας 6.1: Σύνολο Δεδομένων CAL500

Τομέας	Στιγμιότυπα	Γνωρίσματα	Ετικέτες	Πληθικότητα(Cardinality)	Πυκνότητα (Density)	Διακριτότητα (Distinct)
Μουσική	502	68	174	26.044	0.15	502

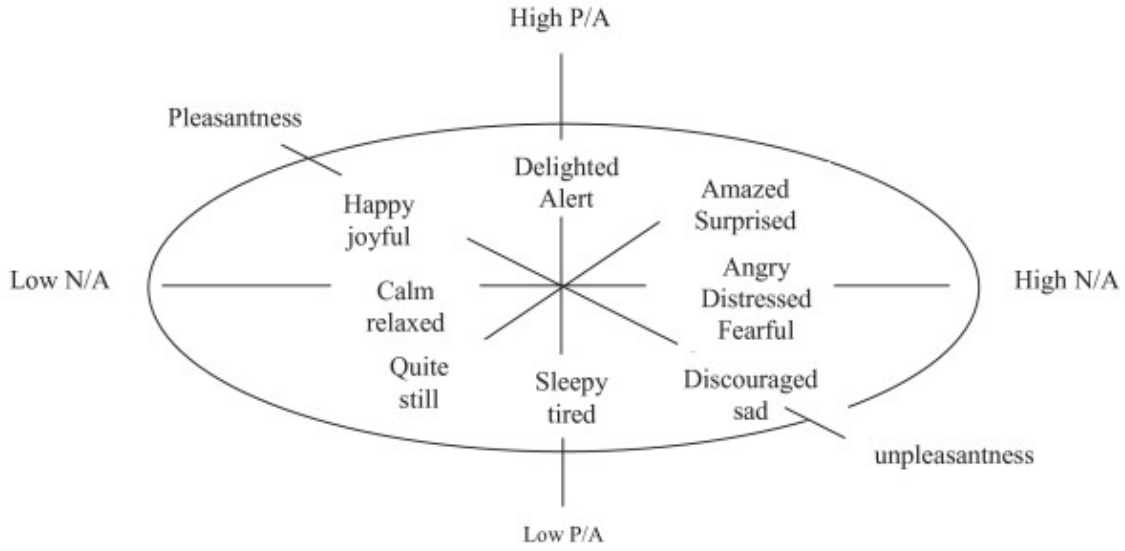
6.2.2 Σύνολο Δεδομένων Emotions

Το σύνολο δεδομένων Emotions αποτελείται από 100 τραγούδια 7 διαφορετικών ειδών μουσικής: κλασσική, ρέγκε, χιπ-χοπ, techno και τζαζ. Για τη δημιουργία της συλλογής επιλέχθηκαν 3 τραγούδια από κάθε αλμπουμ, που στο σύνολο έφταναν τα 233 μουσικά άλμπουμ. Στη συνέχεια, για κάθε τραγούδι, εξαιρώντας 30 δευτερόλεπτα από την αρχή του, έγινε επιλογή τμήματος του διάρκειας 30 δευτερολέπτων. Τα τμήματα που πρόεκυψαν αποθηκεύτηκαν και μετατράπηκαν σε αρχεία με ρυθμό δειγματοληψίας 22.050 HZ, 16-bit ανά δείγμα. Παρακάτω περιγράφονται τα χαρακτηριστικά του αρχείου ήχου και η διαδικασία επισήμανσης συναισθημάτων. Έγινε χρήση του εργαλείου Marsyas [43] για να εξαχθούν τα χαρακτηριστικά. Τα χαρακτηριστικά αυτά χωρίζονται σε δύο κατηγορίες: τα ρυθμικά και τα ηχοχρώματος.

Ρυθμικά χαρακτηριστικά: Τα ρυθμικά χαρακτηριστικά προκύπτουν από την εξαγωγή περιοδικών αλλαγών από ένα ιστόγραμμα ρυθμού. Αναλυτικότερα, εκτελείται ένας αλγόριθμος που προσδιορίζει τις κορυφές του ιστογράμματος λαμβάνοντας υπόψη την αυτοσυσχέτιση. Στη συνέχεια, επιλέγονται οι δύο υψηλότερες κορυφές και υπολογίζονται τα πλάτη τους, οι BMPs (ρυθμοί ανά λεπτό-beats per minute) και η υψηλή προς χαμηλή αναλογία των BPM. Επιπλέον, αθροίζονται τα ιστογράμματα μεταξύ 40-90, 90-140 και 140-250 BPM αντίστοιχα, για να υπολογιστούν τρία χαρακτηριστικά. Από ολόκληρη την διαδικασία που περιγράφηκε δημιουργούνται 8 ρυθμικά χαρακτηριστικά.

Χαρακτηριστικά ηχοχρώματος: Οι συντελεστές με συχνότητα Mel Frequency Cepstral (MFCC) χρησιμοποιούνται για αναγνώριση ομιλίας και μοντελοποίηση μουσικής [44]. Για την παραγωγή MFCC χαρακτηριστικών, το σήμα διαιρείται σε καρέ και για κάθε καρέ υπολογίζεται το φάσμα πλάτους. Στη συνέχεια, υπολογίζεται ο λογάριθμος του και μετατρέπεται σε κλίμακα Mel. Τέλος, εφαρμόζεται διακριτός μετασχηματισμός συνημιτόνου. Επιλέγονται τα πρώτα 13 MFCCs. Επιπλέον, χρησιμποιώντας το βραχυπρόθεσμο μετασχηματισμό Fourier (Short-Term Fourier Transform-FFT) παράγεται ένα άλλο σύνολο τριών χαρακτηριστικών που σχετίζονται με υφές ηχοχρώματος. Για καθένα από τα 16 χαρακτηριστικά (13 MFCC, 3 FFT) υπολογίζονται μέση τιμή (mean), τυπική απόκλιση (std), μέση τυπική απόκλιση (mean std) και τυπική απόκλιση τυπικής απόκλισης (std std) σε όλα τα καρέ. Έτσι, με την πραπάνω διαδικασία που περιγράφηκε δημιοιυργείται ένα σύνολο 64 χαρακτηριστικών ηχοχρώματος [45].

Για την εκχώρηση ετικετών σε δεδομένα με συναισθήματα χρησιμοποιήθηκε το μοντέλο Tellegen-Watson-Clark. Επιλέχθηκε το συγκεκριμένο μοντέλο, επειδή ο συναισθηματικός χώρος της μουσικής μπερδεμένος με πολλά συναισθήματα και μία εφαρμογή μουσικής που στηρίζεται στη διάθεση, θα ήταν ιδανικό να συνδυάζεται με μια σειρά από διαθέσεις και συναισθήματα. Για να επιτευχθεί αυτό, χωρίς όμως να χρησιμοποιείται τεράστιος αριθμός ετικετών, διατηρήθηκαν μόνο έξι κύριες συναισθηματικές ομάδες από



Σχήμα 6.2: Μοντέλο Tellegen-Watson-Clark

αυτό το μοντέλο, οι οποίες απεικονίζονται στο Σχήμα 6.2. Τα κλιπ ήχου σχολιάστηκαν από τρεις άνδρες ειδικούς ηλικίας 20, 25 και 30 ετών από τη Σχολή Μουσικών Σπουδών. Μόνο τα τραγούδια με εντελώς πανομοιότυπη επισήμανση από όλους τους ειδικούς διατηρήθηκαν για μετέπειτα πειραματισμό. Αυτή η διαδικασία οδήγησε σε ένα τελικό σύνολο δεδομένων σχολιασμού 593 τραγουδιών. Πιθανοί λόγοι για αυτήν την απροσδόκητα υψηλή συμφωνία των ειδικών είναι το μικρό μήκος και το κοινό τους υπόβαθρο. Οι προτεινόμενες ετικέτες απεικονίζονται στον Πίνακα 6.2, όπου η τελευταία στήλη του δείχνει τον αριθμό στιγμιοτύπων που σχολιάζονται με κάθε ετικέτα [45]. Τέλος, στον Πίνακα 6.3 απεικονίζεται η σύνοψη του περιεχομένου του συνόλου δεδομένων emotions.

Πίνακας 6.2: Περιγραφή ετικετών του συνόλου δεδομένων emotions

Ετικέτα	Περιγραφή	Αριθμός στιγμιοτύπων
L1	Amazed-surprised	173
L2	Happy-pleased	166
L3	Relaxing-calm	264
L4	Quiet-still	148
L5	Sad-lonely	168
L6	Angry-fearful	189

Πίνακας 6.3: Σύνολο Δεδομένων Emotions

Τομέας	Στιγμιότυπα	Γνωρίσματα	Ετικέτες	Πληθικότητα(Cardinality)	Πυκνότητα (Density)	Διακριτότητα (Distinct)
Μουσική	593	72	6	1.869	0.311	27

6.2.3 Σύνολο δεδομένων Water quality

Το σύνολο δεδομένων water quality πηγάζει από έρευνες που μελετώνται από το υδρο-μετεωρολογικό ινστιτούτο της Σλοβενίας, οι οποίες αφορούν την ποιότητα των υδάτων των ποταμών της Σλοβενίας και

διατηρεί μια βάση δεδομένων για την ποιότητα του νερού. Τα στοιχεία που συλλέχθηκαν αφορούν μια περίοδο εξαετίας (1990-1995). Τα βιολογικά δείγματα συλλέγονται δυο φορές το χρόνο, μία φορά το καλοκαίρι και μία φορά το χειμώνα, ενώ τα φυσικά και χημικά δείγματα συλλέγονται συχνότερα (περίοδοι μεταξύ των μετρήσεων που διαρκούν από έναν έως αρκετούς μήνες) για κάθε μέρος δειγματοληψίας.

Τα φυσικά και χημικά δείγματα περιλαμβάνουν τις μετρούμενες τιμές των 16 διαφορετικών παραμέτρων: βιολογική ζήτηση οξυγόνου (BOD), ηλεκτρική αγωγιμότητα, συγκεντρώσεις από Cl, CO2, NH4, PO4, SiO2, NO2, NO3 και διαλυμένο οξυγόνο (O2), χημική ζήτηση οξυγόνου (K2Cr2O7 και KMnO4), αλκαλικότητα (pH), κορεσμός οξυγόνου, θερμοκρασία του νερού και ολική σκληρότητα. Στα βιολογικά δείγματα περιλαμβάνονται μια λίστα από ταξινομικές βαθμίδες που βρίκονται στο σημείο δειγματοληψίας και την πυκνότητά τους. Η συχνότητα εμφάνισης (πυκνότητα) κάθε υφιστάμενης κατηγοριοποίησης καταγράφεται από έναν ειδικό βιολόγο σε τρία διαφορετικά ποιοτικά επίπεδα: 1 = δευτερευόντως, 3 = συχνά και 5 = άφθονα. Τα δεδομένα αποθηκεύονται σε σχεσιακή βάση δεδομένων που εκπροσωπείται από την Prolog. Σύμφωνα με την Prolog ορολογία κάθε σχέση είναι κατηγόρημα και κάθε πλειάδα είναι γεγονός. Ειδικότερα, τα χημικά δείγματα έχουν ως κατηγόρημα: (Τοποθεσία, Έτος, Μήνας, Ημέρα, Λίστα 16 τιμών) και αποτελούνται από 2580 γεγονότα. Όσον αφορά τα βιοογικά δείγματα, αυτά έχουν ως κατηγόρημα: (Τοποθεσία, Ημέρα, Μήνας, Έτος, Λίστα από ταξινομικές βαθμίδες) και αποτελούνται από 1106 γεγονότα.

Γενικά, το σύνολο δεδομένων είναι αρκετά ξεκάθαρο, αλλά όχι ιδανικό. Στις φυσικοχημικές μετρήσεις, δεκατέσσερις από αυτές περιέχουν τιμές που λείπουν. Επιπρόσθετα, αν και οι βιολογικές μετρήσεις συνήθως γίνονται ακριβώς την ίδια ημέρα με κάποια φυσικοχημική μέτρηση, για 43 βιολογικές μετρήσεις, φυσικοχημικά δεδομένα για την ίδια ημέρα δεν είναι διαθέσιμα. Δεδομένου ότι αυτή η αδυναμία των δεδομένων είναι πολύ περιορισμένη, τα δείγματα με τις τιμές που λείπουν στα πειράματα απλώς αγνοήθηκαν. Έτσι ως αποτέλεσμα αυτού, δημιουργείται ένα σύνολο δεδομένων με 1060 δείγματα νερού τα οποία συμπληρώνονται βιολογικές και φυσικοχημικές πληροφορίες που είναι διαθέσιμες [46]. Τέλος, στον Πίνακα 6.4 απεικονίζεται η σύνοψη του περιεχομένου του συνόλου δεδομένων emotions.

Πίνακας 6.4: Σύνολο Δεδομένων Water quality

Τομέας	Στιγμιότυπα	Γνωρίσματα	Ετικέτες	Πληθικότητα(Cardinality)	Πυκνότητα (Density)	Διακριτότητα (Distinct)
Χημεία	1060	16	14	5.073	0.362	0.778

6.2.4 Σύνολο δεδομένων Scene

Το σύνολο δεδομένων Scene περιέχει 2407 εικόνες όπου κάθε εικόνα σχολιάζεται με έως και 6 έννοιες, όπως παραλία, βουνό, πεδίαδα, πτώση φυλλώματος, ηλιοβασίλεμα και αστική σκηνή. Κάθε εικόνα περιγράφεται με 294 οπτικά αριθμητικά χαρακτηριστικά και αυτά τα χαρακτηριστικά παρουσιάζονται με χρωματικές στιγμές στο χρωματικό χώρο Luv. Πριν από αυτό το στάδιο, κάθε εικόνα χωρίζεται σε 49 μπλοκ χρησιμοποιώντας πλέγμα με 7 σειρές από 7 στήλες. Έτσι, τα χαρακτηριστικά κάθε εικόνας είναι $2 \times 3 \times 7 \times 7 = 294$ [47]. Τέλος, στον Πίνακα 6.5 απεικονίζεται η σύνοψη του περιεχομένου του συνόλου δεδομένων scene.

Πίνακας 6.5: Σύνολο Δεδομένων Scene

Τομέας	Στιγμιότυπα	Γνωρίσματα	Ετικέτες	Πληθικότητα(Cardinality)	Πυκνότητα (Density)	Διακριτότητα (Distinct)
Εικόνα	2407	294	6	1.074	0.179	15

6.2.5 Σύνολο δεδομένων Yeast

Το σύνολο δεδομένων Yeast σχηματίζεται από δεδομένα έκφρασης μικρο-συστοιχίας και φυλογενετικά προφίλ με 1500 γονίδια στο σύνολο εκπαίδευσης και 917 στο σύνολο δοκιμών. Επίσης, τα γνωρίσματα αυτού του συνόλου είναι 103 και το πλήθος των ετικετών είναι 14. Κάθε γονίδιο σχετίζεται με ένα σύνολο λειτουργικών τάξεων των οποίων το μέγιστο μέγεθος μπορεί να είναι πιθανώς περισσότερο από 190. Αυτό το σύνολο δεδομένων έχει ήδη αναλυθεί με προσέγγιση δύο κατηγοριών και είναι γνωστό ότι είναι δύσκολη. Προκειμένου να καταστεί ευκολότερη, χρησιμοποιήθηκε η γνωστή δομή των λειτουργικών τάξεων. Το σύνολο των τάξεων είναι πράγματι δομημένο σε ένα δέντρο του οποίου τα φύλλα είναι οι λειτουργικές κατηγορίες. Δεδομένου ένα γονίδιο, γνωρίζοντας ποια άκρη να πάρει από το ένα επίπεδο στο άλλο οδηγείται απευθείας σε ένα φύλλο κι έτσι σε μια λειτουργική τάξη. Δεδομένου ότι ένα γονίδιο μπορεί να έχει πολλές λειτουργικές τάξεις, αυτό είναι ένα πρόβλημα με πολλές ετικέτες: ένα γονίδιο σχετίζεται με διαφορετικές άκρες [48]. Τέλος, στον Πίνακα 6.6 απεικονίζεται η σύνοψη του περιεχομένου του συνόλου δεδομένων yeast.

Πίνακας 6.6: Σύνολο Δεδομένων Yeast

Τομέας	Στιγμιότυπα	Γνωρίσματα	Ετικέτες	Πληθικότητα(Cardinality)	Πυκνότητα (Density)	Διακριτότητα (Distinct)
Βιολογία	2417	103	14	4.237	0.303	198

6.2.6 Σύνολο δεδομένων Birds

Τα πουλιά έχουν λειτουργήσει ευρέως ως δείκτες βιοποικιλότητας, επειδή παρέχουν κρίσιμες υπηρεσίες οικοσυστήματος, ανταποκρίνονται γρήγορα στις αλλαγές, είναι σχετικά εύκολο να εντοπιστούν και μπορεί να αντικατοπτρίζουν αλλαγές σε χαμηλότερα τροφικά επίπεδα (π.χ. έντομα, φυτά). Δυστυχώς, η συλλογή δεδομένων σχετικά με τις τάσεις στον πληθυσμό των πουλιών μαστίζεται από προβλήματα κακής εκπροσώπησης δείγματος σε απομακρυσμένες περιοχές, προκατάληψη παρατηρητών, ανιχνευσιμότητα και ιδίως το απαγορευτικό κόστος της δειγματοληψίας σε μεγάλες χωρικές και χρονικές κλίμακες. Αυτά τα προβλήματα θα μπορούσαν να βελτιωθούν σε κάποιο βαθμό με τη χρήση αυτοματοποιημένων ακουστικών ερευνών. Ωστόσο, η πολυπλοκότητα του τραγουδιού των πτηνών, ο θόρυβος που υπάρχει στους περισσότερους βιοτόπους και το ταυτόχρονο τραγούδι που εμφανίζεται σε πολλές κοινότητες πουλιών καθιστούν την αυτοματοποιημένη αναγνώριση ειδών μια δύσκολη εργασία.

Αυτό το πρόβλημα διατυπώνεται στο πλαίσιο πολλαπλών σημείων (MIML) για εποπτευόμενη κατηγοριοποίηση. Σε αυτήν την εφαρμογή, τα αντικείμενα που πρέπει να ταξινομηθούν είναι ηχογραφήσεις, τα μέρη είναι τμήματα του φασματογράφου που αντιστοιχούν στις συλλαβές του ήχου των πουλιών που περιγράφονται από ένα διάνυσμα χαρακτηριστικών από ακουστικές ιδιότητες και οι ετικέτες είναι τα είδη που υπάρχουν.

Για τη συλλογή ήχου σε HJA χρησιμοποιούνται 13 Wildlife Acoustics Συσκευές εγγραφής Song Meter SM1. Ο ήχος εγγράφεται στα 16 kHz. Το αποτέλεσμα της εφαρμογής το FFT είναι ένα φασματογρά-

Πίνακας 6.7: Είδη πουλιών Συνόλου Δεδομένων Birds

Κωδικός	Όνομα
BRCR	Brown Creeper
PAWR	Pacific Wren
PSFL	Pacific-slope Flycatcher
RBNU	Red-breasted Nuthatch
DEJU	Dark-eyed Junco
OSFL	Olive-sided Flycatcher
HETH	Hermit Thrush
CBCH	Chestnut-backed Chickadee
VATH	Varied Thrush
HEWA	Hermit Warbler
SWTH	Swainson's Thrush
HAFL	Hammond's Flycatcher
WETA	Western Tanager
BHGB	Black-headed Grosbeak
GCKI	Golden Crowned Kinglet
WAVI	Warbling Vireo
MGWA	MacGillivray's Warbler
STJA	Stellar's Jay
CONI	Common Nighthawk

φημα με συχνότητες από 0-8 kHz. Αυτό το εύρος είναι αρκετό για να συλλάβει τους περισσότερους ήχους πουλιών στο HJA. Είναι πιθανό να παραλείπονται ορισμένοι ήχοι πουλιών λόγω αυτής της συχνότητας δειγματοληψίας, αλλά οι προτεινόμενες μέθοδοι εξακολουθούν να λειτουργούν καλά για τα είδη που προσδιορίζονται στο σύνολο δεδομένων. Κατά συνέπεια, το σύνολο δεδομένων αποτελείται από ένα αντιπροσωπευτικό δείγμα με 645 εγγραφές ήχου 10 δευτερολέπτων από έξι τοποθεσίες, όλες εντός του εύρους από 5:00 π.μ. έως 5:20 π.μ. (τα πουλιά είναι πολύ δραστήρια αυτή τη στιγμή ημέρα), στις 31/5/2009. Πολλές από τις ηχογραφήσεις περιλαμβάνουν πολλαπλά είδη πουλιών που φωνάζουν ταυτόχρονα. Για αυτό, προσδιορίζεται χειροκίνητα το σύνολο των ειδών που υπάρχουν σε κάθε εγγραφή 10 δευτερολέπτων. Υπάρχουν 19 είδη πουλιών στις ηχογραφήσεις που εξετάστηκαν (Πίνακας 6.7). Τέλος, στον Πίνακα 6.8 απεικονίζεται η σύνοψη του περιεχομένου του συνόλου δεδομένων birds.

Πίνακας 6.8: Σύνολο Δεδομένων Birds

Τομέας	Στιγμιότυπα	Γνωρίσματα	Ετικέτες	Πληθικότητα(Cardinality)	Πυκνότητα (Density)	Διακριτότητα (Distinct)
Ήχος	645	260	19	1.014	0.053	133

6.2.7 Σύνολο δεδομένων Image

Το σύνολο δεδομένων Image αποτελείται από 2.000 εικόνες που φέρουν ως ετικέτες είδη όπως δέντρα, βουνά, θάλασσα, ηλιοβασίλεμα και έρημο. Τ ο συγκεκριμένο σύνολο χαρακτηρίζεται ως πολλαπλών ετικετών, καθώς περίπου το 22% των εικόνων συσχετίζονται με πολλαπλές ετικέτες ταυτόχρονα, φτάνοντας τις 1,24 ετικέτες κατά μέσο όρο. Επίσης, κάθε εικόνα εκφράζεται με ένα διάνυσμα χαρακτηριστικών. Ειδικότερα, κάθε έγχρωμη εικόνα τροποποιείται πρώτα στον χώρο CIE Luv, που είναι ένας ομοιόμορφος χρωματικός χώρος, έτσι ώστε χρωματικές διαφορές που γίνονται αντιληπτές να υποδεικύουν τις ευκλεί-

δεις αποστάσεις σε αυτόν τον χρωματικό χώρο. Έπειτα από αυτή τη διαδικασία, η εικόνα διαιρείται σε 49 μπλοκ χρησιμοποιώντας πλέγμα 7 × 7, όπου σε κάθε μπλοκ υπολογίζονται η πρώτη και η δεύτερη στιγμή (μέση και διακύμανση) κάθε ζώνης, που εκφράζουν την εικόνα χαμηλής ανάλυσης και τα υπολογιστικά φθηνά χαρακτηριστικά υφής αντίστοιχα. Ως τελικό βήμα, κάθε εικόνα μετατρέπεται σε διανύσματα διαστάσεων 49 × 3 × 2 = 294 διαστάσεων. Τέλος, στον Πίνακα 6.9 απεικονίζεται η σύνοψη του περιεχομένου του συνόλου δεδομένων image.

Πίνακας 6.9: Σύνολο Δεδομένων Image

Τομέας	Στιγμιότυπα	Γνωρίσματα	Ετικέτες	Πληθικότητα(Cardinality)	Πυκνότητα (Density)	Διακριτότητα (Distinct)
Εικόνα	2.000	294	5	1.236	0.247	0.625

6.2.8 Σύνολο δεδομένων Mediamill

Το σύνολο δεδομένων mediamill αφορά το σύστημα Mediamill, μια σημασιολογική μηχανή αναζήτησης βίντεο που τροφοδοτείται από ανάλυση πολυμέσων. Το σύστημα αυτό επιτρέπει στους χρήστες να αλληλεπιδρούν σε εννοιολογικό επίπεδο, σε αντίθεση στο παραδοσιακό επίπεδο δεδομένων. Το σύστημα αυτό αποτελείται από τρία επίπεδα: σημασιολογική αρχιτεκτονική ευρετηρίασης (Semantic Indexing), σημασιολογική αναζήτηση (Semantic Querying) και σημασιολογικές απεικονίσεις (Semantic Visualizations). Η κεντρική υπόθεση στη σημασιολογική αρχιτεκτονική ευρετηρίασης είναι ότι η μετάδοση κάθε βίντεο είναι το αποτέλεσμα μιας διαδικασίας συγγραφής. Όταν θέλουμε να εξαγάγουμε σημασιολογία από την μετάδοση ενός ψηφιακού βίντεο, χρειάζεται να αντιστραφεί αυτή η διαδικασία συγγραφής. Στην σημασιολογική αναζήτηση, ευρετηριάζονται αρχεία βίντεο ειδήσεων με ένα άνευ προηγουμένου λεξικό 100 σημασιολογικών εννοιών, με βάση την αρχιτεκτονική σημασιολογικού εντοπισμού. Το λεξικό περιέχει έννοιες που σχετίζονται με οχήματα, σπορ, αντικείμενα και ρυθμίσεις, καθώς και συγκεκριμένα άτομα, όπως ο Hu Jintao και ο Tony Blair. Έτσι, οι χρήστες μπορούν να εξερευνήσουν το αρχείο βίντεο χρησιμοποιώντας ερώτημα κατά έννοια, π.χ. ανακτώντας όλα τα πλάνα που περιέχουν ανθρώπους που βαδίζουν. Τέλος, οι σημασιολογικές απεικονίσεις βασίζονται στις παραδοσιακές απεικονίσεις των αποτελεσμάτων ανάκτησης βίντεο, καλούμενα ως storyboard, τα οποία ουσιαστικά απεικονίζουν τα αποτελέσματα ενός ερωτήματος ως πλέγμα βασικών καρέ [49].

Το σύνολο δεομένων mediamill περιέχει προ-υπολογισμένα χαρακτηριστικά πολυμέσων χαμηλού επιπέδου από 85 ώρες μετάδοσης διεθνών βίντεο ειδήσεων του TRECVID 2005/2006. Επιπλέον, αυτό το σύνολο δεδομένων περιέχει αραβικές, κινέζικες και αμερικανικές εκπομπές ειδήσεων που καταγράφηκαν κατά τη διάρκεια του Νοέμβριου το 2004 και για τον σχολιασμό των περιεχομένων τους χρησιμοποιήθηκαν πολλαπλές ετικέτες. Ο σχολιασμός των δεδομένων mediamill επεκτάθηκε στις τρέχουσες 101 έννοιες από έναν χειριστικό σχολιασμό των 39 ετικετών από το TRECVID 2005 [50]. Τέλος, στον Πίνακα 6.10 απεικονίζεται η σύνοψη του περιεχομένου του συνόλου δεδομένων mediamill.

Πίνακας 6.10: Σύνολο Δεδομένων Mediamill

Τομέας	Στιγμιότυπα	Γνωρίσματα	Ετικέτες	Πληθικότητα(Cardinality)	Πυκνότητα (Density)	Διακριτότητα (Distinct)
Βίντεο	43.907	120	101	4.376	0.043	6.555

6.2.9 Σύνολο δεδομένων CHD

Το σύνολο δεδομένων της στεφανιαίας νόσου-Coronary Heart Disease (CHD) αποτελείται από 555 στιγμιότυπα, τα οποία στην πραγματικότητα αντιπροσωπεύουν τους ασθενείς. Μεταξύ των 555 ασθενών, 265 ασθενείς είναι άνδρες και 290 ασθενείς είναι γυναίκες. Επιπλέον, τα συμπτώματα που συλλέχθηκαν για διάγνωση έρευνας περιλαμβάνουν 8 διαστάσεις όπως κρύο ή ζεστό, εφίδρωση, κεφάλι, σώμα κ.α. και συνολικά 125 συμπτώματα, τα οποία αποτελούν τα χαρακτηριστικά του συνόλου δεδομένων. Υπάρχουν 15 σύνδρομα στη διάγνωση διαφοροποίησης, από τα οποία επιλέγονται 6 κοινά πρότυπα στη μελέτη, συμπεριλαμβανομένων: z1 Σύνδρομο ανεπάρκειας της καρδιάς qi, z2 Σύνδρομο ανεπάρκειας της καρδιάς yang, z3 Σύνδρομο ανεπάρκειας της καρδιάς yin, z4 Σύνδρομο στασιμότητας Q1, z5 Σύνδρομο φλέγματος Turbid και z6 σύνδρομο στάσης αίματος. Αρχικά, υπήρχαν 52 συμπτώματα (χαρακτηριστικά) στο σύνολο δεδομένων, 3 εκ των οποίων θεωρήθηκαν περιττά (όπως το οίδημα), γι' αυτό και εξαιρέθηκαν από το τελικό σύνολο δεδομένων. Επομένως, το ολοκληρωμένο και τελικό σύνολο δεδομένων CHD αποτελείται απο 49 χαρακτηριστικά. Επιπλεόν, κάθε στιγμιότυπο μπορεί να έχει απο 0 έως 5 ετικέτες. Ο μέσος όρος αριθμός ετικετών του δείγματος είναι 2,58 [51]. Τέλος, στον Πίνακα 6.11 απεικονίζεται η σύνοψη του περιεχομένου του συνόλου δεδομένων CHD.

Πίνακας 6.11: Σύνολο Δεδομένων CHD

Τομέας	Στιγμιότυπα	Γνωρίσματα	Ετικέτες	Πληθικότητα(Cardinality)	Πυκνότητα (Density)	Διακριτότητα (Distinct)
Ιατρική	555	49	6	2.580	0.430	0.531

Στον παρακάτω Πίνακα 6.12, γίνεται σύνοψη όλων των συνόλων δεδομένων πολλαπλών ετικετών που χρησιμοποιήθηκαν για την εκτέλεση πειραμάτων, τα οποία αναλύθηκαν λεπτομερώς παραπάνω. Αναγράφονται τα πιο σημαντικά στοιχεία αυτών των συνόλων.

Πίνακας 6.12: Σύνοψη συνόλων δεδομένων πολλαπλών ετικετών

Όνομα	Τομέας	Στιγμιότυπα	Γνωρίσματα	Ετικέτες
CAL500	Μουσική	502	68	174
Emotions	Μουσική	593	72	6
Water quality	Χημεία	1060	16	14
Scene	Εικόνα	2407	294	6
Yeast	Βιολογία	2417	103	14
Birds	Ήχος	645	260	19
CHD	Ιατρική	555	49	6
Image	Εικόνα	2000	294	5
Mediamill	Βίντεο	43907	120	101

6.3 Αποτελέσματα Πειραμάτων

6.3.1 Διαδικασία εκτέλεσης πειραμάτων

Αρχικά, περιγράφεται η διαδικασία εκτέλεσης πειραμάτων σε απλά και κατανοητά βήματα. Στο πρώτο βήμα, συλλέχθηκαν όλα τα σύνολα δεδομένων, έπειτα από κανονικοποίηση των τιμών των χαρακτηριστικών τους, ώστε να κυμαίνονται στο ίδιο εύρος τιμών. Στο δέυτερο βήμα, υλοποιήθηκαν σε γλώσσα

python οι αλγόριθμοι BRk-NNa και BRk-NNb, οι οποίοι χρησιμοποιήθηκαν για την κατηγοριοποίηση των δεδομένων πολλαπλών ετικετών. Επιπλέον, υλοποιήθηκε ο αλγόριθμος MRSP3 σε γλώσσα C++, για την μείωση των δεδομένων. Έπειτα ως τρίτο βήμα, εκτελέστηκε ο BRk-NNa πάνω στα αρχικά σύνολα δεδομένων και από τα αποτελέσματα που προέκυψαν υπολογίστηκαν οι τιμές hamming-loss και CPU time, οι οποίες τιμές αποθηκεύτηκαν σε ένα αρχείο. Στη συνέχεια, στο τέταρτο βήμα, εκτελέστηκε ο αλγόριθμος MRSP3 σε όλα τα σύνολα δεδομένων πολλαπλών ετικετών, για να μειώσει τον όγκο των δεδομένων. Από τα αποτελέσματα αυτής της εφαρμογής, υπολογίστηκαν οι τιμές reduction rate, η οποία δηλώνει το ποσοστό μείωσης που πέτυχε ο MRSP3 και computations, η οποία δηλώνει το πλήθος των αποστάσεων που υπολόγισε ο αλγόριθμος. Έπειτα από αυτό, εκτελέστηκε ξανά ο αλγόριθμος BRk-NNa στα νέα, πλέον μειωμένα σε όγκο, δεδομένα, ώστε να γίνει σύγκριση των τιμών και να εξαχθούν τα συμπεράσματα σχετικά με την αξιολόγηση των αλγορίθμων. Σημειώνεται ότι, η ίδια ακριβώς διαδικασία από το τρίτο βήμα και μετά ακολουθήθηκε και για τον αλγόριθμο κατηγοριοποίησης BRk-NNb.

Επίσης, τα πειράματα εκτελούνται χρησιμοποιώντας ως αλγόριθμο μείωσης δεδομένων όχι των MRSP3, αλλά τον MRHC. Ο λόγος εκτέλεσης των πειραμάτων και με τον αλγόριθμο MRHC είναι η σύγκριση των δύο αλγορίθμων. Ο αλγόριθμος MRHC αποτελεί μια παραλλάγή του αλγορίθμου RHC για σύνολα δεδομένων πολλαπλών ετικετών. Ο αλγόριθμος MRHC προτάθηκε πρόσφατα και αποτελεί αποτέλεσμα διπλωματικής εργασίας που παρουσιάστηκε στο τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος [52].

Ειδικότερες λεπτομέρειες σχετικά με τα πειράματα είναι οι τιμές της παραμέτρου k, για τις οποίες εκτελέστηκαν οι αλγόριθμοι BRk-NNa και BRk-NNb. Οι τιμές που επιλέχθηκαν τυχαία ήταν k = 1, k = 5, k = 9, k = 13 και k = 17. Επομένως, εκτελέστηκαν (30) πειράματα συνολικά για κάθε σύνολο δεδομένων, αναλυτικά (5) με την εφαρμογή του BRk-NNa, (5) με την εφαρμογή του BRk-NNb, (5) με τον συνδυασμό BRkNNa - MRSP3, (5) με τον συνδυασμό BRkNNb - MRSP3, (5) με τον συνδυασμό BRkNNa - MRHC και (5) με τον συνδυασμό BRkNNb - MRHC. Στην παρακάτω ενότητα, παρουσιάζονται με μορφή πινάκων και αναλύονται τα αποτελέσματα των πειραμάτων.

6.3.2 Παρουσίαση και Ανάλυση των αποτελεσμάτων

Στον Πίνακα 6.13, απεικονίζονται τα αποτελέσματα για το σύνολο δεδομένων **CAL500**. Όσον αφορά τους δύο αλγορίθμους BRk-NNa και BRk-NNb, είναι ξεκάθαρο πως αποδίδει καλύτερα ο πρώτος από τον δέυτερο, καθώς οι τιμές Hamming-Loss και χρόνος CPU είναι χαμηλότερες στις μετρήσεις του πρώτου αλγορίθμου. Επιπλέον παρατηρείται ότι, ο αλγόριθμος MRSP3 κατάφερε να μειώσει το σύνολο δεδομένων κατα 76% περίπου, άρα λειτούργησε με μεγάλη επιτυχία. Ο συνδυασμός BRKNNa-MRSP3 αποδίδει καλύτερα σε σχέση με τον BRk-NNa που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες και ο χρόνος CPU κυμαίνεται από 1,2-1,36 στην πρώτη περίπτωση, ενώ στην δέυτερη από 1,15-1,45. Ο συνδυασμός BRKNNb-MRSP3 αποδίδει καλύτερα σε σχέση με τον BRk-NNb που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες και ο χρόνος CPU κυμαίνεται από 1,25-1,84 στην πρώτη περίπτωση, ενώ στην δέυτερη από 1,41-2,05. Επομένως, ο σκοπός της εργασίας εξυπηρετείται επιτυχώς.

Όσον αφορά τον αλγόριθμο MRHC, παρατηρείται ότι, κατάφερε να μειώσει το σύνολο δεδομένων κατα 40% περίπου, άρα λειτούργησε ικανοποιητικά. Ο συνδυασμός BRKNNa-MRHC αποδίδει χειρότερα

σε σχέση με τον BRk-NNa που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες, ενώ ο χρόνος CPU κυμαίνεται από 1,56-1,97 στην πρώτη περίπτωση, ενώ στην δεύτερη από 1,15-1,45. Ο συνδυασμός BRKNNb-MRHC αποδίδει ισοδύναμα σε σχέση με τον BRk-NNb που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες και ο χρόνος CPU κυμαίνεται από 1,68-2 στην πρώτη περίπτωση, ενώ στην δεύτερη από 1,41-2,05. Επομένως, ο αλγόριθμος λειτούργησε μετρίως.

Πίνακας 6.13: Αποτελέσματα για το σύνολο δεδομένων CAL500

Αλγόριθμος	Haming-Loss	Reduction Rate (%)	Computations
BRK-NNa (k=1)	0.197	-	-
BRK-NNa (k=5)	0.153	-	-
BRK-NNa (k=9)	0.146	-	-
BRK-NNa (k=13)	0.144	-	-
BRK-NNa (k=17)	0.141	-	-
BRKNNa-MRSP3 (k=1)	0.150	76.36	340001.41
BRKNNa-MRSP3 (k=5)	0.140	76.36	340001.41
BRKNNa-MRSP3 (k=9)	0.138	76.36	340001.41
BRKNNa-MRSP3 (k=13)	0.138	76.36	340001.41
BRKNNa-MRSP3 (k=17)	0.137	76.36	340001.41
BRKNNa-MRHC (k=1)	0.170	40.5	712827.8
BRKNNa-MRHC (k=5)	0.145	40.5	712827.8
BRKNNa-MRHC (k=9)	0.141	40.5	712827.8
BRKNNa-MRHC (k=13)	0.139	40.5	712827.8
BRKNNa-MRHC (k=17)	0.139	40.5	712827.8
BRK-NNb (k=1)	0.288	-	-
BRK-NNb (k=5)	0.289	-	-
BRK-NNb (k=9)	0.288	-	-
BRK-NNb (k=13)	0.288	-	-
BRK-NNb (k=17)	0.287	-	-
BRKNNb-MRSP3 (k=1)	0.212	76.36	340001.41
BRKNNb-MRSP3 (k=5)	0.215	76.36	340001.41
BRKNNb-MRSP3 (k=9)	0.215	76.36	340001.41
BRKNNb-MRSP3 (k=13)	0.214	76.36	340001.41
BRKNNb-MRSP3 (k=17)	0.215	76.36	340001.41
BRKNNb-MRHC (k=1)	0.241	40.5	712827.8
BRKNNb-MRHC (k=5)	0.251	40.5	712827.8
BRKNNb-MRHC (k=9)	0.253	40.5	712827.8
BRKNNb-MRHC (k=13)	0.255	40.5	712827.8
BRKNNb-MRHC (k=17)	0.256	40.5	712827.8

Στον Πίνακα 6.14, απεικονίζονται τα αποτελέσματα για το σύνολο δεδομένων **Water quality**. Όσον αφορά τους δύο αλγορίθμους BRk-NNa και BRk-NNb, είναι ξεκάθαρο πως αποδίδει καλύτερα ο πρώτος από τον δέυτερο, καθώς οι τιμές Hamming-Loss και χρόνος CPU είναι χαμηλότερες στις μετρήσεις του πρώτου αλγορίθμου. Επιπλέον παρατηρείται ότι, ο αλγόριθμος MRSP3 κατάφερε να μειώσει το σύνολο δεδομένων κατα 60% περίπου, άρα λειτούργησε με μεγάλη επιτυχία. Ο συνδυασμός BRKNNa-MRSP3 αποδίδει ισοδύναμα σε σχέση με τον BRk-NNa που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής

Hamming-Loss είναι χαμηλότερες, ενώ ο χρόνος CPU κυμαίνεται από 1,3-1,56 στην πρώτη περίπτωση, ενώ στην δεύτερη από 1,22-1,5. Ο συνδυασμός BRKNNb-MRSP3 αποδίδει καλύτερα σε σχέση με τον BRk-NNb που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες και ο χρόνος CPU κυμαίνεται από 1,82-1,95 στην πρώτη περίπτωση, ενώ στην δεύτερη από 1,9-2,15. Επομένως, ο σκοπός της εργασίας εξυπηρετείται μερικώς.

Όσον αφορά τον αλγόριθμο MRHC, παρατηρείται ότι, κατάφερε να μειώσει το σύνολο δεδομένων κατα 40% περίπου, άρα λειτούργησε ικανοποιητικά. Ο συνδυασμός BRKNNa-MRHC αποδίδει χειρότερα σε σχέση με τον BRk-NNa που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες, ενώ ο χρόνος CPU κυμαίνεται από 1,44-1,83 στην πρώτη περίπτωση, ενώ στην δεύτερη από 1,22-1,5. Ο συνδυασμός BRKNNb-MRHC αποδίδει καλύτερα σε σχέση με τον BRk-NNb που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss δεν επηρεάζονται δραματικά και ο χρόνος CPU κυμαίνεται από 1,77-2 στην πρώτη περίπτωση, ενώ στην δεύτερη από 1,9-2,15. Επομένως, ο αλγόριθμος λειτούργησε καλώς.

Πίνακας 6.14: Αποτελέσματα για το σύνολο δεδομένων Water quality

Αλγόριθμος	Hamming-Loss	Reduction Rate (%)	Computations
BRK-NNa (k=1)	0.385	-	-
BRK-NNa (k=5)	0.350	-	-
BRK-NNa (k=9)	0.336	-	-
BRK-NNa (k=13)	0.333	-	-
BRK-NNa (k=17)	0.328	-	-
BRKNNa-MRSP3 (k=1)	0.368	59.906	1559593.75
BRKNNa-MRSP3 (k=5)	0.339	59.906	1559593.75
BRKNNa-MRSP3 (k=9)	0.331	59.906	1559593.75
BRKNNa-MRSP3 (k=13)	0.329	59.906	1559593.75
BRKNNa-MRSP3 (k=17)	0.331	59.906	1559593.75
BRKNNa-MRHC (k=1)	0.379	40.640	369904.8
BRKNNa-MRHC (k=5)	0.340	40.640	369904.8
BRKNNa-MRHC (k=9)	0.330	40.640	369904.8
BRKNNa-MRHC (k=13)	0.327	40.640	369904.8
BRKNNa-MRHC (k=17)	0.326	40.640	369904.8
BRK-NNb (k=1)	0.457	-	-
BRK-NNb (k=5)	0.450	-	-
BRK-NNb (k=9)	0.443	-	-
BRK-NNb (k=13)	0.441	-	-
BRK-NNb (k=17)	0.438	-	-
BRKNNb-MRSP3 (k=1)	0.442	59.906	1559593.75
BRKNNb-MRSP3 (k=5)	0.440	59.906	1559593.75
BRKNNb-MRSP3 (k=9)	0.435	59.906	1559593.75
BRKNNb-MRSP3 (k=13)	0.432	59.906	1559593.75
BRKNNb-MRSP3 (k=17)	0.43	59.906	1559593.75
BRKNNb-MRHC (k=1)	0.442	40.640	369904.8
BRKNNb-MRHC (k=5)	0.443	40.640	369904.8
BRKNNb-MRHC (k=9)	0.440	40.640	369904.8
BRKNNb-MRHC (k=13)	0.436	40.640	369904.8
BRKNNb-MRHC (k=17)	0.440	40.640	369904.8

Στον Πίνακα 6.15, απεικονίζονται τα αποτελέσματα για το σύνολο δεδομένων **Scene**. Όσον αφορά τους δύο αλγορίθμους BRk-NNa και BRk-NNb, είναι ξεκάθαρο πως αποδίδει καλύτερα ο πρώτος από τον δέυτερο, καθώς οι τιμές Hamming-Loss και χρόνος CPU είναι χαμηλότερες στις μετρήσεις του πρώτου αλγορίθμου. Επιπλέον παρατηρείται ότι, ο αλγόριθμος MRSP3 κατάφερε να μειώσει το σύνολο δεδομένων κατα 40% περίπου, άρα λειτούργησε ικανοποιητικά. Ο συνδυασμός BRKNNa-MRSP3 αποδίδει καλύτερα σε σχέση με τον BRk-NNa που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες ή ίδιες και ο χρόνος CPU κυμαίνεται από 3,7-4,1 στην πρώτη περίπτωση, ενώ στην δέυτερη από 4,84-5,11. Ο συνδυασμός BRKNNb-MRSP3 αποδίδει καλύτερα σε σχέση με τον BRk-NNb που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες και ο χρόνος CPU κυμαίνεται από 4,28-4,56 στην πρώτη περίπτωση, ενώ στην δέυτερη από 5,17-5,63. Επομένως, ο σκοπός της εργασίας εξυπηρετείται επιτυχώς.

Όσον αφορά τον αλγόριθμο MRHC, παρατηρείται ότι, κατάφερε να μειώσει το σύνολο δεδομένων κατα 85% περίπου, άρα λειτούργησε με επιτυχία. Ο συνδυασμός BRKNNa-MRHC αποδίδει καλύτερα σε σχέση με τον BRk-NNa που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι υψηλοτερες ελάχιστα, ενώ ο χρόνος CPU κυμαίνεται από 2,82-3,15 στην πρώτη περίπτωση, ενώ στην δέυτερη από 4,84-5,11. Ο συνδυασμός BRKNNb-MRHC αποδίδει καλύτερα σε σχέση με τον BRk-NNb που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss δεν επηρεάζονται δραματικά και ο χρόνος CPU κυμαίνεται από 2,74-3,39 στην πρώτη περίπτωση, ενώ στην δέυτερη από 5,17-5,63. Επομένως, ο αλγόριθμος λειτούργησε επιτυχώς.

Στον Πίνακα 6.16, απεικονίζονται τα αποτελέσματα για το σύνολο δεδομένων **Yeast**. Όσον αφορά τους δύο αλγορίθμους BRk-NNa και BRk-NNb, είναι ξεκάθαρο πως αποδίδει καλύτερα ο πρώτος από τον δέυτερο, καθώς οι τιμές Hamming-Loss και χρόνος CPU είναι χαμηλότερες στις μετρήσεις του πρώτου αλγορίθμου. Επιπλέον παρατηρείται ότι, ο αλγόριθμος MRSP3 κατάφερε να μειώσει το σύνολο δεδομένων κατα 53% περίπου, άρα λειτούργησε με επιτυχία. Ο συνδυασμός BRKNNa-MRSP3 αποδίδει ισοδύναμα σε σχέση με τον BRk-NNa που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι υψηλότερες και ο χρόνος CPU κυμαίνεται από 2,37-2,79 στην πρώτη περίπτωση, ενώ στην δέυτερη από 2,45-2,73. Ο συνδυασμός BRKNNb-MRSP3 αποδίδει καλύτερα σε σχέση με τον BRk-NNb που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες και ο χρόνος CPU κυμαίνεται από 2,97-3,66 στην πρώτη περίπτωση, ενώ στην δέυτερη από 3,56-4,65. Επομένως, ο σκοπός της εργασίας εξυπηρετείται μερικώς.

Όσον αφορά τον αλγόριθμο MRHC, παρατηρείται ότι, κατάφερε να μειώσει το σύνολο δεδομένων κατα 52% περίπου, άρα λειτούργησε με επιτυχία. Ο συνδυασμός BRKNNa-MRHC αποδίδει χειρότερα σε σχέση με τον BRk-NNa που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι υψηλοτερες, ενώ ο χρόνος CPU κυμαίνεται από 2,18-3 στην πρώτη περίπτωση, ενώ στην δέυτερη από 2,45-2,73. Ο συνδυασμός BRKNNb-MRHC αποδίδει καλύτερα σε σχέση με τον BRk-NNb που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες και ο χρόνος CPU κυμαίνεται από 2,74-3 στην πρώτη περίπτωση, ενώ στην δέυτερη από 3,56-4,65. Επομένως, ο αλγόριθμος λειτούργησε καλώς.

Στον Πίνακα 6.17, απεικονίζονται τα αποτελέσματα για το σύνολο δεδομένων **Emotions**. Όσον αφορά τους δύο αλγορίθμους BRk-NNa και BRk-NNb, είναι ξεκάθαρο πως αποδίδει καλύτερα ο πρώτος

Πίνακας 6.15: Αποτελέσματα για το σύνολο δεδομένων Scene

Αλγόριθμος	Hamming-Loss	Reduction Rate (%)	Computations
BRK-NNa (k=1)	0.133	-	-
BRK-NNa (k=5)	0.120	-	-
BRK-NNa (k=9)	0.120	-	-
BRK-NNa (k=13)	0.122	-	-
BRK-NNa (k=17)	0.122	-	-
BRKNNa-MRSP3 (k=1)	0.126	39.325	6811157
BRKNNa-MRSP3 (k=5)	0.117	39.325	6811157
BRKNNa-MRSP3 (k=9)	0.121	39.325	6811157
BRKNNa-MRSP3 (k=13)	0.125	39.325	6811157
BRKNNa-MRSP3 (k=17)	0.128	39.325	6811157
BRKNNa-MRHC (k=1)	0.126	85.13	480151.6
BRKNNa-MRHC (k=5)	0.122	85.13	480151.6
BRKNNa-MRHC (k=9)	0.127	85.13	480151.6
BRKNNa-MRHC (k=13)	0.142	85.13	480151.6
BRKNNa-MRHC (k=17)	0.153	85.13	480151.6
BRK-NNb (k=1)	0.289	-	-
BRK-NNb (k=5)	0.286	-	-
BRK-NNb (k=9)	0.284	-	-
BRK-NNb (k=13)	0.283	-	-
BRK-NNb (k=17)	0.283	-	-
BRKNNb-MRSP3 (k=1)	0.286	39.325	6811157
BRKNNb-MRSP3 (k=5)	0.286	39.325	6811157
BRKNNb-MRSP3 (k=9)	0.284	39.325	6811157
BRKNNb-MRSP3 (k=13)	0.282	39.325	6811157
BRKNNb-MRSP3 (k=17)	0.282	39.325	6811157
BRKNNb-MRHC (k=1)	0.287	85.13	480151.6
BRKNNb-MRHC (k=5)	0.285	85.13	480151.6
BRKNNb-MRHC (k=9)	0.281	85.13	480151.6
BRKNNb-MRHC (k=13)	0.280	85.13	480151.6
BRKNNb-MRHC (k=17)	0.278	85.13	480151.6

από τον δέυτερο, καθώς οι τιμές Hamming-Loss και χρόνος CPU είναι χαμηλότερες στις μετρήσεις του πρώτου αλγορίθμου. Επιπλέον παρατηρείται ότι, ο αλγόριθμος MRSP3 κατάφερε να μειώσει το σύνολο δεδομένων κατα 46% περίπου, άρα λειτούργησε ικανοποητικά. Ο συνδυασμός BRKNNa-MRSP3 αποδίδει ισοδύναμα σε σχέση με τον BRk-NNa που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες, ενώ ο χρόνος CPU κυμαίνεται από 1,07-1,24 στην πρώτη περίπτωση, ενώ στην δέυτερη από 1,17-1,54. Ο συνδυασμός BRKNNb-MRSP3 αποδίδει καλύτερα σε σχέση με τον BRk-NNb που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες και ο χρόνος CPU κυμαίνεται από 1,63-1,8 στην πρώτη περίπτωση, ενώ στην δέυτερη από 1,75-2,11. Επομένως, ο σκοπός της εργασίας εξυπηρετείται μερικώς.

Όσον αφορά τον αλγόριθμο MRHC, παρατηρείται ότι, κατάφερε να μειώσει το σύνολο δεδομένων κατα 65% περίπου, άρα λειτούργησε με επιτυχία. Ο συνδυασμός BRKNNa-MRHC αποδίδει χειρότερα σε σχέση με τον BRk-NNa που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι υ-

Πίνακας 6.16: Αποτελέσματα για το σύνολο δεδομένων Yeast

Αλγόριθμος	Hamming-Loss	Reduction Rate (%)	Computations
BRK-NNa (k=1)	0.294	-	-
BRK-NNa (k=5)	0.203	-	-
BRK-NNa (k=9)	0.195	-	-
BRK-NNa (k=13)	0.195	-	-
BRK-NNa (k=17)	0.195	-	-
BRKNNa-MRSP3 (k=1)	0.225	53.020	3885707.5
BRKNNa-MRSP3 (k=5)	0.204	53.020	3885707.5
BRKNNa-MRSP3 (k=9)	0.202	53.020	3885707.5
BRKNNa-MRSP3 (k=13)	0.203	53.020	3885707.5
BRKNNa-MRSP3 (k=17)	0.205	53.020	3885707.5
BRKNNa-MRHC (k=1)	0.232	51.850	1341732.8
BRKNNa-MRHC (k=5)	0.202	51.850	1341732.8
BRKNNa-MRHC (k=9)	0.202	51.850	1341732.8
BRKNNa-MRHC (k=13)	0.202	51.850	1341732.8
BRKNNa-MRHC (k=17)	0.203	51.850	1341732.8
BRK-NNb (k=1)	0.384	-	-
BRK-NNb (k=5)	0.363	-	-
BRK-NNb (k=9)	0.348	-	-
BRK-NNb (k=13)	0.338	-	-
BRK-NNb (k=17)	0.331	-	-
BRKNNb-MRSP3 (k=1)	0.353	53.020	3885707.5
BRKNNb-MRSP3 (k=5)	0.331	53.020	3885707.5
BRKNNb-MRSP3 (k=9)	0.324	53.020	3885707.5
BRKNNb-MRSP3 (k=13)	0.318	53.020	3885707.5
BRKNNb-MRSP3 (k=17)	0.312	53.020	3885707.5
BRKNNb-MRHC (k=1)	0.361	51.850	1341732.8
BRKNNb-MRHC (k=5)	0.340	51.850	1341732.8
BRKNNb-MRHC (k=9)	0.328	51.850	1341732.8
BRKNNb-MRHC (k=13)	0.319	51.850	1341732.8
BRKNNb-MRHC (k=17)	0.312	51.850	1341732.8

ψηλοτερες, ενώ ο χρόνος CPU κυμαίνεται από 1,36-1,63 στην πρώτη περίπτωση, ενώ στην δέυτερη από 1,17-1,54. Ο συνδυασμός BRKNNb-MRHC αποδίδει καλύτερα σε σχέση με τον BRk-NNb που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss δεν επηρεάζονται δραματικά και ο χρόνος CPU κυμαίνεται από 1,33-1,69 στην πρώτη περίπτωση, ενώ στην δέυτερη από 1,75-2,11. Επομένως, ο αλγόριθμος λειτούργησε καλώς.

Στον Πίνακα 6.18, απεικονίζονται τα αποτελέσματα για το σύνολο δεδομένων **Birds**. Όσον αφορά τους δύο αλγορίθμους BRk-NNa και BRk-NNb, είναι ξεκάθαρο πως αποδίδει καλύτερα ο πρώτος από τον δέυτερο, καθώς οι τιμές Hamming-Loss και χρόνος CPU είναι χαμηλότερες στις μετρήσεις του πρώτου αλγορίθμου. Επιπλέον παρατηρείται ότι, ο αλγόριθμος MRSP3 κατάφερε να μειώσει το σύνολο δεδομένων κατα 32% περίπου, άρα λειτούργησε ικανοποητικά. Ο συνδυασμός BRKNNa-MRSP3 αποδίδει καλύτερα σε σχέση με τον BRk-NNa που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι υψηλότερες, όμως ο χρόνος CPU κυμαίνεται από 1,44-1,74 στην πρώτη περίπτωση, ενώ στην

Πίνακας 6.17: Αποτελέσματα για το σύνολο δεδομένων Emotions

Αλγόριθμος	Hamming-Loss	Reduction Rate (%)	Computations
BRK-NNa (k=1)	0.244	-	-
BRK-NNa (k=5)	0.207	-	-
BRK-NNa (k=9)	0.197	-	-
BRK-NNa (k=13)	0.195	-	-
BRK-NNa (k=17)	0.192	-	-
BRKNNa-MRSP3 (k=1)	0.227	46.442	298838.812
BRKNNa-MRSP3 (k=5)	0.203	46.442	298838.812
BRKNNa-MRSP3 (k=9)	0.195	46.442	298838.812
BRKNNa-MRSP3 (k=13)	0.191	46.442	298838.812
BRKNNa-MRSP3 (k=17)	0.200	46.442	298838.812
BRKNNa-MRHC (k=1)	0.226	65.730	65750.4
BRKNNa-MRHC (k=5)	0.203	65.730	65750.4
BRKNNa-MRHC (k=9)	0.205	65.730	65750.4
BRKNNa-MRHC (k=13)	0.215	65.730	65750.4
BRKNNa-MRHC (k=17)	0.225	65.730	65750.4
BRK-NNb (k=1)	0.400	-	-
BRK-NNb (k=5)	0.380	-	-
BRK-NNb (k=9)	0.379	-	-
BRK-NNb (k=13)	0.382	-	-
BRK-NNb (k=17)	0.384	-	-
BRKNNb-MRSP3 (k=1)	0.380	46.442	298838.812
BRKNNb-MRSP3 (k=5)	0.362	46.442	298838.812
BRKNNb-MRSP3 (k=9)	0.360	46.442	298838.812
BRKNNb-MRSP3 (k=13)	0.362	46.442	298838.812
BRKNNb-MRSP3 (k=17)	0.363	46.442	298838.812
BRKNNb-MRHC (k=1)	0.380	65.730	65750.4
BRKNNb-MRHC (k=5)	0.376	65.730	65750.4
BRKNNb-MRHC (k=9)	0.382	65.730	65750.4
BRKNNb-MRHC (k=13)	0.387	65.730	65750.4
BRKNNb-MRHC (k=17)	0.389	65.730	65750.4

δέυτερη από 1,59-1,94. Ο συνδυασμός BRKNNb-MRSP3 αποδίδει χειρότερα σε σχέση με τον BRk-NNb που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι υψηλότερες, ενώ ο χρόνος CPU κυμαίνεται από 1,53-1,71 στην πρώτη περίπτωση, ενώ στην δέυτερη από 1,48-1,71. Επομένως, ο σκοπός της εργασίας εξυπηρετείται μερικώς.

Όσον αφορά τον αλγόριθμο MRHC, παρατηρείται ότι, κατάφερε να μειώσει το σύνολο δεδομένων κατα 42% περίπου, άρα λειτούργησε ικανοποιητικά. Ο συνδυασμός BRKNNa-MRHC αποδίδει καλύτερα σε σχέση με τον BRk-NNa που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι υψηλοτερες, ενώ ο χρόνος CPU κυμαίνεται από 1,47-1,76 στην πρώτη περίπτωση, ενώ στην δέυτερη από 1,59-1,94. Ο συνδυασμός BRKNNb-MRHC αποδίδει χειρότερα σε σχέση με τον BRk-NNb που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες και ο χρόνος CPU κυμαίνεται από 1,35-1,85 στην πρώτη περίπτωση, ενώ στην δέυτερη από 1,48-1,71. Επομένως, ο αλγόριθμος λειτούργησε καλώς.

Πίνακας 6.18: Αποτελέσματα για το σύνολο δεδομένων Birds

Αλγόριθμος	Hamming-Loss	Reduction Rate (%)	Computations
BRK-NNa (k=1)	0.096	-	-
BRK-NNa (k=5)	0.084	-	-
BRK-NNa (k=9)	0.088	-	-
BRK-NNa (k=13)	0.088	-	-
BRK-NNa (k=17)	0.088	-	-
BRKNNa-MRSP3 (k=1)	0.131	31.815	160227.594
BRKNNa-MRSP3 (k=5)	0.098	31.815	160227.594
BRKNNa-MRSP3 (k=9)	0.098	31.815	160227.594
BRKNNa-MRSP3 (k=13)	0.098	31.815	160227.594
BRKNNa-MRSP3 (k=17)	0.098	31.815	160227.594
BRKNNa-MRHC (k=1)	0.093	42.700	79763.8
BRKNNa-MRHC (k=5)	0.090	42.700	79763.8
BRKNNa-MRHC (k=9)	0.093	42.700	79763.8
BRKNNa-MRHC (k=13)	0.095	42.700	79763.8
BRKNNa-MRHC (k=17)	0.097	42.700	79763.8
BRK-NNb (k=1)	0.244	-	-
BRK-NNb (k=5)	0.238	-	-
BRK-NNb (k=9)	0.237	-	-
BRK-NNb (k=13)	0.234	-	-
BRK-NNb (k=17)	0.384	-	-
BRKNNb-MRSP3 (k=1)	0.583	31.815	160227.594
BRKNNb-MRSP3 (k=5)	0.671	31.815	160227.594
BRKNNb-MRSP3 (k=9)	0.671	31.815	160227.594
BRKNNb-MRSP3 (k=13)	0.362	31.815	160227.594
BRKNNb-MRSP3 (k=17)	0.363	31.815	160227.594
BRKNNb-MRHC (k=1)	0.179	42.700	79763.8
BRKNNb-MRHC (k=5)	0.174	42.700	79763.8
BRKNNb-MRHC (k=9)	0.174	42.700	79763.8
BRKNNb-MRHC (k=13)	0.175	42.700	79763.8
BRKNNb-MRHC (k=17)	0.177	42.700	79763.8

Στον Πίνακα 6.19, απεικονίζονται τα αποτελέσματα για το σύνολο δεδομένων **CHD**. Όσον αφορά τους δύο αλγορίθμους BRk-NNa και BRk-NNb, είναι ξεκάθαρο πως αποδίδει καλύτερα ο πρώτος από τον δεύτερο, καθώς οι τιμές Hamming-Loss και χρόνος CPU είναι χαμηλότερες στις μετρήσεις του πρώτου αλγορίθμου. Επιπλέον παρατηρείται ότι, ο αλγόριθμος MRSP3 κατάφερε να μειώσει το σύνολο δεδομένων κατα 53% περίπου, άρα λειτούργησε με επιτυχία. Ο συνδυασμός BRKNNa-MRSP3 αποδίδει χειρότερα σε σχέση με τον BRk-NNa που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss για κάποιες τιμές του k είναι υψηλότερες ενώ άλλες χαμηλότερες, ενώ ο χρόνος CPU κυμαίνεται από 1,13-1,55 στην πρώτη περίπτωση, ενώ στην δέυτερη από 0,96-1,23. Ο συνδυασμός BRKNNb-MRSP3 αποδίδει καλύτερα σε σχέση με τον BRk-NNb που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι υψηλότερες, όμως ο χρόνος CPU κυμαίνεται από 1,55-1,73 στην πρώτη περίπτωση, ενώ στην δέυτερη από 1,45-1,97. Επομένως, ο σκοπός της εργασίας εξυπηρετείται μερικώς.

Όσον αφορά τον αλγόριθμο MRHC, παρατηρείται ότι, κατάφερε να μειώσει το σύνολο δεδομένων κατα

65% περίπου, άρα λειτούργησε με επιτυχία. Ο συνδυασμός BRKNNa-MRHC αποδίδει χειρότερα σε σχέση με τον BRk-NNa που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες, ενώ ο χρόνος CPU κυμαίνεται από 1,34-1,54 στην πρώτη περίπτωση, ενώ στην δέυτερη από 0,96-1,23. Ο συνδυασμός BRKNNb-MRHC αποδίδει χειρότερα σε σχέση με τον BRk-NNb που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss δεν επηρεάζονται δραματικά και ο χρόνος CPU κυμαίνεται από 1,65-2 στην πρώτη περίπτωση, ενώ στην δέυτερη από 1,45-1,97. Επομένως, ο αλγόριθμος λειτούργησε μετρίως.

Πίνακας 6.19: Αποτελέσματα για το σύνολο δεδομένων CHD

Αλγόριθμος	Hamming-Loss	Reduction Rate (%)	Computations
BRK-NNa (k=1)	0.368	-	-
BRK-NNa (k=5)	0.330	-	-
BRK-NNa (k=9)	0.311	-	-
BRK-NNa (k=13)	0.309	-	-
BRK-NNa (k=17)	0.301	-	-
BRKNNa-MRSP3 (k=1)	0.356	53.303	224104.797
BRKNNa-MRSP3 (k=5)	0.326	53.303	224104.797
BRKNNa-MRSP3 (k=9)	0.322	53.303	224104.797
BRKNNa-MRSP3 (k=13)	0.311	53.303	224104.797
BRKNNa-MRSP3 (k=17)	0.306	53.303	224104.797
BRKNNa-MRHC (k=1)	0.356	65.470	65892.2
BRKNNa-MRHC (k=5)	0.320	65.470	65892.2
BRKNNa-MRHC (k=9)	0.307	65.470	65892.2
BRKNNa-MRHC (k=13)	0.299	65.470	65892.2
BRKNNa-MRHC (k=17)	0.300	65.470	65892.2
BRK-NNb (k=1)	0.388	-	-
BRK-NNb (k=5)	0.339	-	-
BRK-NNb (k=9)	0.345	-	-
BRK-NNb (k=13)	0.339	-	-
BRK-NNb (k=17)	0.338	-	-
BRKNNb-MRSP3 (k=1)	0.383	53.303	224104.797
BRKNNb-MRSP3 (k=5)	0.342	53.303	224104.797
BRKNNb-MRSP3 (k=9)	0.344	53.303	224104.797
BRKNNb-MRSP3 (k=13)	0.345	53.303	224104.797
BRKNNb-MRSP3 (k=17)	0.342	53.303	224104.797
BRKNNb-MRHC (k=1)	0.381	65.470	65892.2
BRKNNb-MRHC (k=5)	0.345	65.470	65892.2
BRKNNb-MRHC (k=9)	0.348	65.470	65892.2
BRKNNb-MRHC (k=13)	0.345	65.470	65892.2
BRKNNb-MRHC (k=17)	0.348	65.470	65892.2

Στον Πίνακα 6.20, απεικονίζονται τα αποτελέσματα για το σύνολο δεδομένων **Image**. Όσον αφορά τους δύο αλγορίθμους BRk-NNa και BRk-NNb, είναι ξεκάθαρο πως αποδίδει καλύτερα ο πρώτος από τον δέυτερο, καθώς οι τιμές Hamming-Loss και χρόνος CPU είναι χαμηλότερες στις μετρήσεις του πρώτου αλγορίθμου. Επιπλέον παρατηρείται ότι, ο αλγόριθμος MRSP3 κατάφερε να μειώσει το σύνολο δεδομένων κατα 35% περίπου, άρα λειτούργησε ικανοποιητικά. Ο συνδυασμός BRKNNa-MRSP3 αποδίδει

καλύτερα σε σχέση με τον BRk-NNa που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες και ο χρόνος CPU κυμαίνεται από 3,43-3,95 στην πρώτη περίπτωση, ενώ στην δέυτερη από 3,74-4,22. Ο συνδυασμός BRKNNb-MRSP3 αποδίδει καλύτερα σε σχέση με τον BRk-NNb που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες και ο χρόνος CPU κυμαίνεται από 3,86-4,29 στην πρώτη περίπτωση, ενώ στην δέυτερη από 4,67-5,13. Επομένως, ο σκοπός της εργασίας εξυπηρετείται επιτυχώς.

Όσον αφορά τον αλγόριθμο MRHC, παρατηρείται ότι, κατάφερε να μειώσει το σύνολο δεδομένων κατα 72% περίπου, άρα λειτούργησε με επιτυχία. Ο συνδυασμός BRKNNa-MRHC αποδίδει καλύτερα σε σχέση με τον BRk-NNa που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες, ενώ ο χρόνος CPU κυμαίνεται από 2,7-3,19 στην πρώτη περίπτωση, ενώ στην δέυτερη από 3,74-4,22. Ο συνδυασμός BRKNNb-MRHC αποδίδει καλύτερα σε σχέση με τον BRk-NNb που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss δεν επηρεάζονται δραματικά και ο χρόνος CPU κυμαίνεται από 3-3,41 στην πρώτη περίπτωση, ενώ στην δέυτερη από 4,67-5,13. Επομένως, ο αλγόριθμος λειτούργησε επιτυχώς.

Στον Πίνακα 6.21, απεικονίζονται τα αποτελέσματα για το σύνολο δεδομένων **Mediamill**. Όσον αφορά τους δύο αλγορίθμους BRk-NNa και BRk-NNb, είναι ξεκάθαρο πως αποδίδει καλύτερα ο πρώτος από τον δέυτερο, καθώς οι τιμές Hamming-Loss και χρόνος CPU είναι χαμηλότερες στις μετρήσεις του πρώτου αλγορίθμου. Επιπλέον παρατηρείται ότι, ο αλγόριθμος MRSP3 κατάφερε να μειώσει το σύνολο δεδομένων κατα 69% περίπου, άρα λειτούργησε με επιτυχία. Ο συνδυασμός BRKNNa-MRSP3 αποδίδει ξεκάθαρα καλύτερα σε σχέση με τον BRk-NNa που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss δεν επηρεάζονται δραματικά, ενώ ο χρόνος CPU κυμαίνεται από 35-37 στην πρώτη περίπτωση, ενώ στην δέυτερη από 85-111. Ο συνδυασμός BRKNNb-MRSP3 αποδίδει ξεκάθαρα καλύτερα σε σχέση με τον BRk-NNb που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες και ο χρόνος CPU κυμαίνεται από 48-52 στην πρώτη περίπτωση, ενώ στην δέυτερη από 108-121. Επομένως, ο σκοπός της εργασίας εξυπηρετείται επιτυχώς.

Όσον αφορά τον αλγόριθμο MRHC, παρατηρείται ότι, κατάφερε να μειώσει το σύνολο δεδομένων κατα 55% περίπου, άρα λειτούργησε με επιτυχία. Ο συνδυασμός BRKNNa-MRHC αποδίδει καλύτερα σε σχέση με τον BRk-NNa που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες και ο χρόνος CPU κυμαίνεται από 47-59 στην πρώτη περίπτωση, ενώ στην δέυτερη από 85-111. Ο συνδυασμός BRKNNb-MRHC αποδίδει καλύτερα σε σχέση με τον BRk-NNb που εκτελείται μοναδικά, καθώς οι τιμές της μετρικής Hamming-Loss είναι χαμηλότερες και ο χρόνος CPU κυμαίνεται από 57-63 στην πρώτη περίπτωση, ενώ στην δέυτερη από 108-121. Επομένως, ο αλγόριθμος λειτούργησε επιτυχώς.

Στον παρακάτω Πίνακα 6.22 απεικονίζονται οι μέσοι όροι των hamming-loss, CPU time και reduction rate για κάθε σύνολο δεδομένων για όλες τις τιμές k για τις οποίες εκτελέστηκαν οι MRSP3 και MRHC. Ο πίνακας αυτός βοηθά στην εξαγωγή συμπεράσματος για το ποιος τελικά αλγόριθμος από τους δύο αποδίδει καλύτερα. Για να καταλήξουμε στον καλύτερο αλγόριθμο, αρκεί να ελέγξουμε για κάθε σύνολο δεδομένων τους μέσους όρους.

Αρχικά, για το σύνολο CAL500, παρατηρούμε ότι ο MRSP3 επιτυγχάνει μεγαλύτερο ποσοστό μείωσης (RR) σε σχέση με τον MRHC. Επίσης, λόγω αυτού οι τιμές hamming-loss και CPU time είναι χαμηλό-

Πίνακας 6.20: Αποτελέσματα για το σύνολο δεδομένων Image

Αλγόριθμος	Hamming-Loss	Reduction Rate (%)	Computations
BRK-NNa (k=1)	0.298	-	-
BRK-NNa (k=5)	0.278	-	-
BRK-NNa (k=9)	0.276	-	-
BRK-NNa (k=13)	0.272	-	-
BRK-NNa (k=17)	0.272	-	-
BRKNNa-MRSP3 (k=1)	0.284	35.650	6410363.5
BRKNNa-MRSP3 (k=5)	0.270	35.650	6410363.5
BRKNNa-MRSP3 (k=9)	0.266	35.650	6410363.5
BRKNNa-MRSP3 (k=13)	0.262	35.650	6410363.5
BRKNNa-MRSP3 (k=17)	0.260	35.650	6410363.5
BRKNNa-MRHC (k=1)	0.288	71.710	419476.4
BRKNNa-MRHC (k=5)	0.259	71.710	419476.4
BRKNNa-MRHC (k=9)	0.256	71.710	419476.4
BRKNNa-MRHC (k=13)	0.254	71.710	419476.4
BRKNNa-MRHC (k=17)	0.253	71.710	419476.4
BRK-NNb (k=1)	0.345	-	-
BRK-NNb (k=5)	0.330	-	-
BRK-NNb (k=9)	0.328	-	-
BRK-NNb (k=13)	0.331	-	-
BRK-NNb (k=17)	0.332	-	-
BRKNNb-MRSP3 (k=1)	0.339	35.650	6410363.5
BRKNNb-MRSP3 (k=5)	0.325	35.650	6410363.5
BRKNNb-MRSP3 (k=9)	0.326	35.650	6410363.5
BRKNNb-MRSP3 (k=13)	0.332	35.650	6410363.5
BRKNNb-MRSP3 (k=17)	0.334	35.650	6410363.5
BRKNNb-MRHC (k=1)	0.344	71.710	419476.4
BRKNNb-MRHC (k=5)	0.328	71.710	419476.4
BRKNNb-MRHC (k=9)	0.332	71.710	419476.4
BRKNNb-MRHC (k=13)	0.331	71.710	419476.4
BRKNNb-MRHC (k=17)	0.334	71.710	419476.4

τερες, επομένως ο MRSP3 αποδίδει καλύτερα σε σχέση με τον MRHC για αυτό το σύνολο δεδομένων. Για το επόμενο σύνολο Scene, ο MRSP3 παρουσιάζει χαμηλότερο ποσοστό μείωσης σε σχέση με τον MRHC. Βέβαια η μέση τιμή hamming-loss είναι χαμηλότερη στον πρώτο, ενώ ο μέσος χρόνος CPU είναι καλύτερος στον δεύτερο. Επομένως, για αυτό το σύνολο δεδομένων αποδίδει καλύτερα ο αλγόριθμος MRHC. Για το σύνολο Yeast, παρατηρείται ότι, οι δύο αλγόριθμοι αποδίδουν σχεδόν ισοδύναμα, καθώς επιτυγχάνουν σχεδόν το ίδιο ποσοστό μείωσης, με μικρή διαφορά μεγαλύτερο ο MRSP3. Επιπλέον, η τιμή hamming-loss είναι ίδια και για τους δύο, ενώ καλύτερο χρόνο CPU επιτυγχάνει ο MRSP3. Επομένως, είναι ίσοι αλλά με μικρή διαφορά ο MRSP3 είναι αποδοτικότερος. Στο σύνολο δεδομένων Birds παρατηρείται ότι, ο MRHC αποδίδει καλύτερα σε σχέση με τον MRSP3, καθώς παρουσιάζει μεγαλύτερο ποσοστό μείωσης. Επιπλέον η μέση τιμή hamming-loss είναι χαμηλότερη στον MRHC σε σχέση με τον MRSP3, ενώ ο χρόνος CPU στον πρώτο είναι ελάχιστα μεγαλύτερος. Για το σύνολο δεδομένων Image, καλύτερο ποσοστό μείωσης παρουσιάζει ο MRHC, καθώς τιμή hamming-loss και CPU χρόνο. Επομένως, είναι ξεκάθαρο πως ο MRHC αποδίδει καλύτερα σε σχέση με τον MRSP3. Για το σύνολο

Πίνακας 6.21: Αποτελέσματα για το σύνολο δεδομένων Mediamill

Αλγόριθμος	Hamming-Loss	Reduction Rate (%)	Computations	CPU time
BRK-NNa (k=1)	0.041	-	-	85.084
BRK-NNa (k=5)	0.033	-	-	92.763
BRK-NNa (k=9)	0.032	-	-	111.094
BRK-NNa (k=13)	0.032	-	-	92.563
BRK-NNa (k=17)	0.032	-	-	94.481
BRKNNa-MRSP3 (k=1)	0.035	68.565	2604417792	35.869
BRKNNa-MRSP3 (k=5)	0.032	68.565	2604417792	37.506
BRKNNa-MRSP3 (k=9)	0.032	68.565	2604417792	37.494
BRKNNa-MRSP3 (k=13)	0.032	68.565	2604417792	37.850
BRKNNa-MRSP3 (k=17)	0.032	68.565	2604417792	36.906
BRKNNa-MRHC (k=1)	0.038	55.86	806819021.2	47.394
BRKNNa-MRHC (k=5)	0.032	55.86	806819021.2	59.741
BRKNNa-MRHC (k=9)	0.032	55.86	806819021.2	58.869
BRKNNa-MRHC (k=13)	0.032	55.86	806819021.2	48.503
BRKNNa-MRHC (k=17)	0.032	55.86	806819021.2	52.000
BRK-NNb (k=1)	0.085	-	-	108.081
BRK-NNb (k=5)	0.086	-	-	117.841
BRK-NNb (k=9)	0.086	-	-	116.497
BRK-NNb (k=13)	0.086	-	-	121.881
BRK-NNb (k=17)	0.086	-	-	119.581
BRKNNb-MRSP3 (k=1)	0.073	68.565	2604417792	48.363
BRKNNb-MRSP3 (k=5)	0.074	68.565	2604417792	49.109
BRKNNb-MRSP3 (k=9)	0.074	68.565	2604417792	49.575
BRKNNb-MRSP3 (k=13)	0.074	68.565	2604417792	52.319
BRKNNb-MRSP3 (k=17)	0.074	68.565	2604417792	50.800
BRKNNb-MRHC (k=1)	0.079	55.86	806819021.2	57.441
BRKNNb-MRHC (k=5)	0.080	55.86	806819021.2	62.784
BRKNNb-MRHC (k=9)	0.080	55.86	806819021.2	62.466
BRKNNb-MRHC (k=13)	0.080	55.86	806819021.2	62.366
BRKNNb-MRHC (k=17)	0.081	55.86	806819021.2	63.603

δεδομένων CHD, αποδοτικότερος προκύπτει ότι είναι ο MRHC σε σχέση με τον MRSP3, καθώς έχει υψηλότερο ποσοστό μείωσης και μικρότερες τιμές για hamming-loss και CPU χρόνο. Για το σύνολο δεδομένων Water-quality (wq) αποδοτικότερος είναι ο MRSP3, καθώς επιτυγχάνει μεγαλύτερο ποσοστό μείωσης, ίδια τιμή hamming-loss και καλύτερο χρόνο CPU. Για το σύνολο δεδομένων Emotions, ο MRHC είναι αποδοτικότερος σε σχέση με τον MRSP3, καθώς παρουσιάζει μεγαλύτερο ποσοστό μείωσης. Όμως, έχει μεγαλύτερες τιμές σε hamming-loss και CPU χρόνο. Τέλος, για το σύνολο δεδομένων Mediamill ο MRSP3 αποδίδει προφανώς καλύτερα από τον MRHC, καθώς το ποσοστό μείωσης που παρουσιάζει είναι υψηλότερο. Επιπλέον, η τιμή hamming-loss είναι ίδια και για τους δύο αλγόριθμους, όμως ο MRSP3 εκτελείται σε αρκετά μικρότερο χρόνο σε σχέση με τον MRHC.

Συνοψίζοντας όλα τα παραπάνω, προκύπτει ότι σε πέντε (5) από τα εννέα (9) σύνολα δεδομένων ο MRHC αποδίδει καλύτερα σε σχεση με τον MRSP3. Σε κάποια σύνολα δεδομένων, οι δύο αλγόριθμοι παρουσιάζουν μεγάλες διαφορές, ενώ σε κάποια άλλα είναι σχεδόν ισοδύναμοι. Όμως, το σημαντικό είναι ότι και οι δύο αλγόριθμοι επιτυγχάνουν τον σκοπό της εργασίας αυτής, δηλαδή να μειώσουν τον όγκο των δεδομένων διατηρώντας ή επηρεάζοντας ελάχιστα την απώλεια hamming-loss και εκτελώντας την κατηγοριοποίηση σε μικρότερο χρόνο.

Πίνακας 6.22: Σύνοψη μέσων τιμών για MRSP3 και MRHC

Σύνολο δεδομένων	MRSP3-HL	MRSP3-CPU	MRSP3-RR	MRHC-HL	MRHC-CPU	MRHC-RR
CAL500	0.141	1.292	76.359	0.147	1.774	40.500
Scene	0.123	3.936	39.325	0.134	3.025	85.130
Yeast	0.208	2.605	53.020	0.208	2.871	51.850
Birds	0.105	1.567	31.815	0.093	1.610	42.700
Image	0.268	3.572	35.650	0.262	2.983	71.710
CHD	0.324	1.410	53.303	0.316	1.466	65.470
WQ	0.340	1.404	59.906	0.340	1.614	40.640
Emotions	0.203	1.402	46.442	0.215	1.529	65.730
Mediamill	0.033	37.125	68.565	0.033	53.301	55.860

Κεφάλαιο 7ο: Συμπεράσματα και Μελλοντική έρευνα

Με το πέρασμα των χρόνων, έχουν προταθεί αρκετές τεχνικές μείωσης δεδομένων, οι οποίες πολλές από αυτές είναι διαθέσιμες στη βιβλιογραφία. Βέβαια, το μεγαλύτερο μέρος αυτών αφορά τα προβλήματα κατηγοριοποίησης μονής ετικέτας. Είναι λιγοστές οι επιστημονικές μελέτες που ερευνούν τα προβλήματα κατηγοριοποίησης δεδομένων πολλαπλών ετικετών. Αλγόριθμοι κατηγοριοποίησης όπως ο k-NN, ο οποίος αποτελεί κύριο μέρος των πειραμάτων αυτής της εργασίας, με το υψηλό υπολογιστικό κόστος που παρουσιάζει σε δεδομένα μεγάλου μεγέθους καθιστά την χρήση του αναποτελεσματική. Έτσι, όλες αυτές οι σκέψεις και οι προβληματισμοί αποτέλεσαν το κίνητρο για την συγγραφή και την εκπόνηση αυτής της έρευνας.

Σύμφωνα με όσα αναφέρθηκαν, στόχος αυτής της διπλωματικής ήταν η ανάπτυξη νέων τεχνικών μείωσης δεδομένων, οι οποίες βασίζονται σε ήδη υπάρχουσες τεχνικές και προσαρμόζονται για προβλήματα δεδομένων πολλαπλών ετικετών και δεν απαιτούν μετασχηματισμό του προβλήματος. Επιπλέον, ένας ακόμη στόχος ήταν η προσπάθεια συνδυασμού αυτών των αλγορίθμων μείωσης δεδομένων με τον οκνηρό κατηγοριοποιητή k-NN, έτσι ώστε η κατηγοριοποίηση να εκτελείται πάνω σε ένα μειωμένου μεγέθους σύνολο δεδομένων πολλαπλών ετικετών, μειώνοντας με αυτόν τον τρόπο το υπολογιστικό κόστος που απαιτεί. Βέβαια, η επίτευξη αυτών των στόχων συνεπάγεται με την διατήρηση της ακρίβειας σε υψηλά επίπεδα. Όμως, επειδή η ακρίβεια σε σύνολα δεδομένων πολλαπλών ετικετών θεωρείται ακατάλληλη, καθώς χωρίζει τις προβλέψεις για ένα στιγμιότυπο σε ”ολικώς σωστές” και ”ολικώς λανθασμένες”, έγινε χρήση μίας άλλης μετρικής που είναι ιδανική για τέτοιες περιπτώσεις. Η μετρική αυτή είναι η Hamming-Loss, σύμφωνα με την οποία οι προβλέψεις μπορούν να είναι είτε ”ολικώς σωστές” είτε ”μερικώς σωστές” είτε ”ολικώς λανθασμένες”. Σε αντίθεση με την ακρίβεια, οι ιδανικές τιμές hamming-loss είναι αυτές που πλησιάζουν κοντά στο μηδέν, καθώς η μετρική αυτή εκφράζει την απώλεια που παράγεται κατά τη διάρκεια της πρόβλεψης.

Επομένως, για την επιτυχή ολοκλήρωση όλων των στόχων που αναφέρθηκαν παραπάνω, αναπτύχθηκε ο αγλόριθμος MRSP3, ο οποίος είναι επέκταση του γνωστού αλγόριθμου μείωσης δεδομένων RSP3, ώστε να εφαρμόζεται ιδανικά σε προβλήματα δεδομένων πολλαπλών ετικετών. Στον αλγόριθμο MRSP3 μία ομάδα θεωρείται ομοιογενής, όταν όλα τα στιγμιότυπα αυτής έχουν τουλάχιστον μια κοινή κλάση. Επίσης, χρησιμοποιήθηκαν οι αλγόριθμοι κατηγοιοποίησης BRk-NNa και BRk-NNb, για την κατηγοριοποίηση δεδομένων πολλαπλών ετικετών, καθώς ο k-NN αφορά μόνο τα προβλήματα μονής ετικέτας. Οι συγκεκριμένοι αλγόριθμοι υπάρχουν έτοιμοι υλοποιημένοι σε γλώσσα python και είναι διαθέσιμοι μέσω της βιβλιοθήκης skmultilearn.

Τα πειράματα που εκτελέστηκαν κατά τη διάρκεια της διπλωματικής εργασίας, εφαρμόστηκαν σε εννέα σύνολα δεδομένων πολλαπλών ετικετών, τα οποία έχουν περιγραφεί λεπτομερώς στο προηγούμενο κεφάλαιο. Ως μετρική απόστασης χρησιμοποιήθηκε η Ευκλείδια απόσταση και ως μετρική αξιολόγησης ορίστηκε η Hamming-loss. Τα συμπεράσματα που προέκυψαν από τις πειραματικές μελέτες απέδειξαν πως τα σύνολα δεδομένων πολλαπλών ετικετών παρουσίασαν μεγάλα ποσοστά μείωσης, εφορμόζοντας τον αλγόριθμο MRSP3 και πως η απώλεια hamming-loss είτε δεν επηρεάστηκε είτε εμφάνισε ελάχιστη μείωση, γεγονός θετικό για την έρευνα. Επομένως, ο MRSP3 καθώς και οι αλγόριθμοι κατηγοριοποίησης BRk-NNa και BRk-NNb ολοκοκλήρωσαν με επιτυχία τον στόχο όλων αυτών των πειραμάτων.

Όμως, όπως είναι ήδη γνωστό, αν και υπάρχουν πολλοί προτεινόμενοι αλγόριθμοι μείωσης δεδομένων που μπορούν να χρησιμοποιηθούν σε συνδυασμό με τον k-NN κατηγοριοποιητή, δεν είναι κατάληλλοι για την αντιμετώπιση προβλημάτων δεδομένων πολλαπλών ετικετών. Έτσι, υπάρχει αρκετός ελεύθερος χώρος για την ανάπτυξη παρόμοιων και εκτεταμένων ερευνών, που μπορούν να συμπληρώσουν την παγκόσμια βιβλιογραφία και επιστήμη. Προτείνεται λοιπόν, η επικέντρωση σε ήδη υπάρχοντες αλγόριθμους και στην τροποποίηση τους, έτσι ώστε να εφαρμόζουν επιτυχώς σε τέτοιου είδους προβλήματα, καθώς και η εύρεση νέων τεχνικών που θα μειώσουν τα μειονεκτήματα του κατηγοριοποιητή k-NN, ώστε να επεκταθεί η εφαρμογή του σε περισσότερα επίπεδα προβλημάτων κατηγοριοποίησης.

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] Χ.Μακρής and Β.Μεγαλοοικονόμου, "Εξόρυξη Δεδομένων και Αλγόριθμοι Μάθησης."

[2] J. Fürnkranz, *Decision Tree*, pp. 263–267. Boston, MA: Springer US, 2010.

[3] W. S. Sarle, "Neural networks and statistical models," 1994.

[4] I. Rish, "An empirical study of the naïve bayes classifier," *IJCAI 2001 Work Empir Methods Artif Intell*, vol. 3, 01 2001.

[5] M. Hearst, S. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their Applications*, vol. 13, no. 4, pp. 18–28, 1998.

[6] O. Harrison, "Machine learning basics with the knearest neighbors algorithm.."

[7] S. Ougiaroglou, "Algorithms and techniques for efficient and effective nearest neighbours classification.," 2014.

[8] S. Ougiaroglou, A. Nanopoulos, A. Papadopoulos, Y. Manolopoulos, and T. Welzer, "Adaptive k-nearest-neighbor classification using a dynamic number of nearest neighbors," pp. 66–82, 09 2007.

[9] S. Ougiaroglou, G. Evangelidis, and K. Diamantaras, *Dynamic k-NN Classification Based on Region Homogeneity*, pp. 27–37. 08 2020.

[10] I. Witten, M. Hall, E. Frank, G. Holmes, B. Pfahringer, and P. Reutemann, "The weka data mining software: An update," *SIGKDD Explorations*, vol. 11, pp. 10–18, 11 2009.

[11] M. M. Deza and E. Deza, *Encyclopedia of Distances*. Springer Berlin Heidelberg, 2009.

[12] □. Φιλιππάκης, "Τεχνικές Μείωσης Δεδομένων για Σύνολα Δεδομένων Πολλαπλών Ετικετών," 2021.

[13] N. Kumar, "Advantages and disadvantages of knn algorithm in machine learning."

[14] T. Neha, "Data reduction."

[15] G. Tsoumakas and I. Katakis, "Multi-label classification: An overview," *International Journal of Data Warehousing and Mining*, vol. 3, pp. 1–13, Jan. 2007.

[16] J. M. Nareshpalsingh and P. H. N. Modi, "Multi-label classification methods: A comparative study," 2017.

[17] M.-L. ZHANG, Y.-K. LI, and X. G. , "Binary relevance for multi-label learning: An overview," 2017.

[18] E. Spyromitros, G. Tsoumakas, and I. Vlahavas, "An empirical study of lazy multilabel classification algorithms," in *Artificial Intelligence: Theories, Models and Applications* (J. Darzentas, G. A. Vouros, S. Vosinakis, and A. Arnellos, eds.), (Berlin, Heidelberg), pp. 401–406, Springer Berlin Heidelberg, 2008.

[19] G. Tsoumakas, I. Katakis, and I. Vlahavas, *Mining Multi-label Data*, pp. 667–685. 07 2010.

[20] M. Pushpa and S. Karpagavalli, "Multi-label classification: Problem transformation methods in tamil phoneme classification," 2017.

[21] J. Read, "A pruned problem transformation method for multi-label classification," p. 143–150, 2008.

[22] G. Tsoumakas, I. Katakis, and I. Vlahavas, "Effective and efficient multilabel classification in domains with large number of labels," p. 30–44, 2008.

[23] L. Rokach, A. Schclar, and E. Itach, "Ensemble methods for multi-label classification,"

[24] G. Madjarov, D. Kocev, D. Gjorgjevikj, and S. Džeroski, "An extensive experimental comparison of methods for multi-label learning," 2012.

[25] Domino, "Classify all the things (with multiple labels)," July 23, 2018.

[26] Z. Yu, Q. Wang, Y. Fan, H. Dai, and d Meikang Qiu, "An improved classifier chain algorithm for multi-label classification of big data analysis," 2015.

[27] K. Dembczyński, W. Waegeman, W. Cheng, and E. Hüllermeier, "Regret analysis for performance metrics in multi-label classification: The case of hamming and subset zero-one loss," in *Machine Learning and Knowledge Discovery in Databases* (J. L. Balcázar, F. Bonchi, A. Gionis, and M. Sebag, eds.), pp. 280–295, Springer Berlin Heidelberg, 2010.

[28] . Κωνσταντίνα, "Τεχνικές Μηχανικής Μάθησης για Ροές Δεδομένων με Πολλαπλές Ετικέτες," 2016.

[29] Y. Yang, "An evaluation of statistical approaches to text categorization," *Inf. Retr.*, vol. 1, p. 69–90, May 1999.

[30] G. Tsoumakas, I. Katakis, and I. Vlahavas, *Mining Multi-label Data*, pp. 667–685. Boston, MA: Springer US, 2010.

[31] F. Charte, A. J. Rivera, M. J. del Jesus, and F. Herrera, "Mlenn: A first approach to heuristic multilabel undersampling," in *Intelligent Data Engineering and Automated Learning – IDEAL 2014* (E. Corchado, J. A. Lozano, H. Quintián, and H. Yin, eds.), (Cham), pp. 1–9, Springer International Publishing, 2014.

[32] S. Kanj, F. Abdallah, T. Denœux, and K. Tout, "Editing training data for multi-label classification with the k-nearest neighbor rule," *Pattern Analysis and Applications*, vol. 19, 02 2015.

[33] . Arnaiz-González, J.-F. Díez-Pastor, J. Rodríguez, and C. García-Osorio, "Local sets for multi-label instance selection," *Applied Soft Computing*, vol. 68, 04 2018.

[34] E. Leyva Miranda, A. González-Muñoz, and R. Pérez, "Three new instance selection methods based on local sets: A comparative study with several approaches from a bi-objective perspective," *Pattern Recognition*, vol. 48, 04 2015.

[35] Álvar Arnaiz-González, J.-F. Díez-Pastor, J. J. Rodríguez, and C. García-Osorio, "Study of data transformation techniques for adapting single-label prototype selection algorithms to multi-label learning," *Expert Systems with Applications*, vol. 109, pp. 114–130, 2018.

[36] P. Skryjomski, B. Krawczyk, and A. Cano, "Speeding up k-nearest neighbors classifier for large-scale multi-label learning on gpus," *Neurocomputing*, vol. 354, pp. 10–19, 2019. Recent Advancements in Hybrid Artificial Intelligence Systems.

[37] C. H. Chen and A. Jóundefinedwik, "A sample set condensation algorithm for the class sensitive artificial neural network," *Pattern Recogn. Lett.*, vol. 17, p. 819–823, July 1996.

[38] J. Sánchez, "High training set size reduction by space partitioning and prototype, volume = 37, journal = Pattern Recognition," pp. 1561–1564, 01 2004.

[39] P. Refaeilzadeh, L. Tang, and H. Liu, *Cross-Validation*, pp. 532–538. Boston, MA: Springer US, 2009.

[40] J. Brownlee, "How to configure k-fold cross-validation."

[41] J. M. Moyano, "Multi-label classification dataset repository," 2020.

[42] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet, "Semantic annotation and retrieval of music and sound effects," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 2, pp. 467–476, 2008.

[43] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 5, pp. 293–302, 2002.

[44] B. Logan, "Mel frequency cepstral coefficients for music modeling," *Proc. 1st Int. Symposium Music Information Retrieval*, 11 2000.

[45] K. Trochidis, G. Tsoumakas, G. Kalliris, and I. Vlahavas, "Multi-label classification of music into emotions," vol. 2011, pp. 325–330, 01 2008.

[46] H. Blockeel, S. Džeroski, and J. Grbović, "Simultaneous prediction of multiple chemical parameters of river water quality with tilde," in *Principles of Data Mining and Knowledge Discovery* (J. M. Żytkow and J. Rauch, eds.), (Berlin, Heidelberg), pp. 32–40, Springer Berlin Heidelberg, 1999.

[47] M. R. Boutell, J. Luo, X. Shen, and C. M. Brown, "Learning multi-label scene classification," *Pattern Recognition*, vol. 37, no. 9, pp. 1757–1771, 2004.

[48] A. Elisseeff and J. Weston, "A kernel method for multi-labelled classification," in *Advances in Neural Information Processing Systems* (T. Dietterich, S. Becker, and Z. Ghahramani, eds.), vol. 14, MIT Press, 2002.

[49] C. Snoek, M. Worring, J. Gemert, J.-M. Geusebroek, D. Koelma, G. NGuyên, O. Rooij, and F. Seinstra, "Mediamill: exploring news video archives based on learned semantics," pp. 225–226, 01 2005.

[50] G. Nasierding and A. Kouzani, "Comparative evaluation of multi-label classification methods," 05 2012.

[51] H. Shao, G.-Z. Li, G. Liu, and Y. Wang, "Symptom selection for multi-label data of inquiry diagnosis in traditional chinese medicine," *Science China Information Sciences*, vol. 56, 05 2012.

[52] S. Ougiaroglou, P. Filippakis, and G. Evangelidis, "Prototype generation for multi-label nearest neighbours classification," 2021.

[53] A. Clare and R. D. King, "Knowledge discovery in multi-label phenotype data," in *Principles of Data Mining and Knowledge Discovery* (L. De Raedt and A. Siebes, eds.), pp. 42–53, Springer Berlin Heidelberg, 2001.

[54] R. E. Schapire and Y. Singer, "BoosTexter: A Boosting-based System for Text Categorization," *Machine Learning*, vol. 39, no. 2/3, pp. 135–168, 2000.

[55] Y. Freund and R. E. Schapire, "A desicion-theoretic generalization of on-line learning and an application to boosting," in *Computational Learning Theory* (P. Vitányi, ed.), (Berlin, Heidelberg), pp. 23–37, Springer Berlin Heidelberg, 1995.

[56] D. Stojanova, M. Ceci, A. Appice, and S. Džeroski, "Network regression with predictive clustering trees," in *Machine Learning and Knowledge Discovery in Databases* (D. Gunopulos, T. Hofmann, D. Malerba, and M. Vazirgiannis, eds.), pp. 333–348, Springer Berlin Heidelberg, 2011.

[57] M.-L. Zhang and Z.-H. Zhou, "A k-nearest neighbor based algorithm for multi-label classification," in *2005 IEEE International Conference on Granular Computing*, vol. 2, pp. 718–721 Vol. 2, 2005.

[58] X. Luo and A. N. Zincir-Heywood, "Evaluation of two systems on multi-class multi-label document classification," in *Foundations of Intelligent Systems*, pp. 161–169, Springer Berlin Heidelberg, 2005.

[59] A. K. McCallum, "Multi-label text classification with a mixture model trained by em," in *AAAI 99 Workshop on Text Learning*, 1999.

[60] S. Godbole and S. Sarawagi, "Discriminative methods for multi-labeled classification," in *Advances in Knowledge Discovery and Data Mining* (H. Dai, R. Srikant, and C. Zhang, eds.), pp. 22–30, Springer Berlin Heidelberg, 2004.

[61] C. Sammut and G. I. Webb, eds., *Stacked Generalization*, pp. 912–912. Boston, MA: Springer US, 2010.

[62] F. Thabtah, P. Cowling, and Y. Peng, "Mmac: a new multi-class, multi-label associative classification approach," in *Fourth IEEE International Conference on Data Mining (ICDM'04)*, pp. 217–224, 2004.