Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων
Τομεας Τεχνολογιας Πληροφορικης και Υπολογιστων

Προστασία της Ιδιωτικότητας Κατά τη Δημοσίευση Δεδομένων με Λειτουργικές Εξαρτήσεις

Διπλωματικη Εργασια

του

ΝΙΚΟΛΑΟΥ ΚΟΥΛΟΥΡΗ

Επιβλέπων: Ιωάννης Βασιλείου
Καθηγητής Ε.Μ.Π.

Εργαστηριο Συστηματων Βασεων Γνωσεων και Δεδομενων
Αθήνα, Ιούλιος 2015

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών
Εργαστήριο Συστημάτων Βάσεων Γνώσεων και Δεδομένων

Προστασία της Ιδιωτικότητας Κατά τη Δημοσίευση Δεδομένων με Λειτουργικές Εξαρτήσεις

Διπλωματικη Εργασια

του

ΝΙΚΟΛΑΟΥ ΚΟΥΛΟΥΡΗ

Επιβλέπων: Ιωάννης Βασιλείου
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 2 Ιουλίου 2015.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Ιωάννης Βασιλείου	Κώστας Κοντογιάννης	Γιώργος Στάμου
Καθ. Ε.Μ.Π.	Καθ. Ε.Μ.Π.	Καθ. Ε.Μ.Π.

Αθήνα, Ιούλιος 2015

(Υπογραφή)

.....

Νικολαος Κουλουρης
Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Ευχαριστίες

Αρχικά θα ήθελα να ευχαριστήσω τον καθηγητή του ΕΜΠ κύριο Ιωάννη Βασιλείου, για την ευκαιρία που μου έδωσε να ασχοληθώ με ένα τόσο ενδιαφέρον αντικείμενο στην παρούσα διπλωματική εργασία.

Στη συνέχεια ευχαριστώ ιδιαίτερα τον κ. Μανώλη Τερροβίτη, ερευνητή στο Ινστιτούτο Πληροφοριακών Συστημάτων, που με εισήγαγε στο πρόβλημα της προστασίας της ιδιωτικότητας. Επιπρόσθετα, οφείλω ένα μεγάλο ευχαριστώ στον Γιάννη Λιαγούρη, μεταδιδακτορικό ερευνητή στο ETH Zurich, για την άρτια και μεθοδική συνεργασία μας, την καθοδήγηση και την επιστημονική στήριξη του για την συγγραφή της παρούσας εργασίας.

Τέλος, θα ήθελα να ευχαριστήσω την οικογένεια μου και τους ανθρώπους που ήταν κοντά μου καθ'όλη τη διάρκεια της εργασίας αυτής και γενικότερα των προπτυχιακών μου σπουδών. Η στήριξη τους ήταν ανεκτίμητη.

Περίληψη

Η διαφύλαξη της ιδιωτικότητας κατά τη δημοσίευση δεδομένων έχει αποκτήσει ιδιαίτερο ενδιαφέρον τα τελευταία χρόνια, λόγω της ανάγκης των διαφόρων φορέων να μοιράζονται δεδομένα που περιέχουν ευαίσθητες πληροφορίες για φυσικά ή νομικά πρόσωπα. Τα στοιχεία που περιλαμβάνονται σε αυτά τα δεδομένα είναι πολύτιμα για πληθώρα επιχειρήσεων, πανεπιστημίων και οργανισμών λόγω της στατιστικής αξίας τους και της αναγκαιότητας τους για ιατρικές μελέτες, κοινωνιολογικές μελέτες, πειραματικές αξιολογήσεις αλγορίθμων από την έρευνα κτλ.

Ακόμη και αν τα δεδομένα δημοσιοποιηθούν χωρίς τα χαρακτηριστικά που προσδιορίζουν άμεσα ένα άτομο, η αποκάλυψη ευαίσθητων πληροφοριών για ένα άτομο μπορεί να επιτευχθεί με χρήση λογικού συμπερασμού (reasoning) . Ο αντίπαλος με χρήση των λειτουργικών εξαρτήσεων των δεδομένων μπορεί να συμπεράνει υπονοούμενη γνώση η οποία είναι ευαίσθητη.

Ο κύριος σκοπός αυτής της εργασίας είναι η διερεύνηση αυτού του προβλήματος. Ορίζεται φορμαλιστικά το πρόβλημα που προκύπτει στην ανωνυμοποίηση δεδομένων με λειτουργικές εξαρτήσεις, προτείνεται ένας πρωτότυπος αλγόριθμος που εγγυάται την προστασία της ιδιωτικότητας σε αυτό το σενάριο και αξιολογείται πειραματικά με πραγματικά δεδομένα η πρακτικότητα του αλγόριθμου.

Λέξεις Κλειδιά

ανωνυμοποίηση, ιδιωτικότητα, λειτουργικές εξαρτήσεις, γράφος εξαρτήσεων

Abstract

Privacy preservation during data publishing has gained considerable attention during the last years due to the need of several organizations to share their data without revealing sensitive information about real people or legal entities included in them. The records included in these datasets are valuable for a variety of enterprises, universities and organizations because of their statistical value and the necessity for them in medical studies, social studies, experimental evaluation of algorithms in research etc.

Even if the data are published without the characteristics that identify an entity directly, disclosure of sensitive information can be achieved through the use of reasoning over the data. An opponent can use the functional dependencies of the data to infer implicit knowledge which is sensitive.

The main purpose of this work is to investigate this problem. The problem of anonymizing data with functional dependencies is defined in a principled way, a novel algorithm that guarantees the privacy of the data is proposed and its practicality with real-life data is experimentally evaluated.

Keywords

anonymization, privacy, functional dependencies, dependency graph

Περιεχόμενα

Κατάλογος Σχημάτων

Κατάλογος Πινάκων

Κεφάλαιο 1

Εισαγωγή

Η συλλογή ψηφιακών πληροφοριών από κυβερνήσεις, οργανισμούς και ιδιώτες έχει δημιουργήσει τεράστιες ευκαιρίες για λήψη αποφάσεων βασισμένη στη γνώση και τις πληροφορίες αυτές. Υπάρχει ανάγκη για ανταλλαγή και δημοσίευση πληροφοριών μεταξύ διαφορετικών οργανισμών λόγω αμοιβαίου συμφέροντος ή λόγω ορισμένων κανονισμών που απαιτούν τη δημοσίευση συγκεκριμένων δεδομένων. Όμως, τα δεδομένα στην αρχική τους μορφή συνήθως περιέχουν ευαίσθητες πληροφορίες για άτομα και η δημοσίευση τους μπορεί να οδηγήσει στην παραβίαση της ιδιωτικότητας τους.

Στα παραδοσιακά πληροφοριακά συστήματα, ο τρόπος που εξασφαλίζεται η προστασία της ιδιωτικότητας είναι συνήθως ο ακόλουθος τρόπος: Ο διαχειριστής του συστήματος ορίζει τις όψεις ασφαλείας (security views) χειροκίνητα. Κατά τον καθορισμό των όψεων ασφαλείας, ο διαχειριστής ουσιαστικά ορίζει ποιο μέρος του συνόλου των δεδομένων μπορεί να είναι δημόσιο, συνήθως ακολουθώντας την αρχή των 'ελάχιστων προνομίων' που ορίζει ότι οι χρήστες θα πρέπει να έχουν τα ελάχιστα δικαιώματα που απαιτούνται για να κάνουν τη δουλειά τους. Αυτό απαιτεί καλή κατανόηση του σχήματος της βάσης δεδομένων και της σημασιολογίας των δεδομένων. Στη περίπτωση των σύνθετων λειτουργικών εξαρτήσεων και ενημερώσεων, η πλήρης κατανόηση του σχήματος των δεδομένων δεν μπορεί να θεωρείται δεδομένη, ως εκ τούτου, ο χειροκίνητος ορισμός των όψεων ασφαλείας είναι επιρρεπής σε λάθη.

1.1 Αντικείμενο της διπλωματικής

Ο χώρος που κινείται αυτή η διπλωματική είναι η εύρεση αλγορίθμων για την ανωνυμοποίηση δεδομένων με λειτουργικές εξαρτήσεις κατά τη δημοσίευση τους ώστε να προστατεύεται η ιδιωτικότητα.

Σε αντίθεση με τα παραδοσιακά πληροφοριακά συστήματα, ακολουθούμε μια διαφορετική προσέγγιση για την ανωνυμοποίηση των δεδομένων. Συγκεκριμένα:

- Ορίζουμε το μέρος των δεδομένων το οποίο είναι ευαίσθητο, άρα και πρέπει να μείνει κρυφό, και αφήνουμε το σύστημα να αποφασίσει ποιο είναι το μέρος της βάσης δεδομένων που μπορούμε να δημοσιεύσουμε ώστε να μην μπορεί να συναχθεί ως συμπέρασμα ευαίσθητη πληροφορία.

- Σε ένα διαφορετικό σενάριο, ορίζουμε το μέρος των δεδομένων το οποίο είναι ευαίσθητο και, επίσης, ορίζουμε το ελάχιστο μέρος των δεδομένων που χρειάζεται ο χρήστης ώστε να εκτελέσει τη δουλειά του. Τότε, το σύστημα πρέπει επαληθεύσει ότι οι πληροφορίες προς δημοσίευση δεν οδηγούν σε διαρροή δεδομένων σε σχέση με τη σημασιολογία των δεδομένων.

- Είσοδος σε αυτή τη διαδικασία ανωνυμοποίησης είναι ένα σύνολο από φορμαλιστικούς κανόνες, όπως μια οντολογία, οι οποίοι εκφράζουν εξαρτήσεις μεταξύ των δεδομένων ή και κοινή γνώση σχετικά με τα δεδομένα.

- Οι λειτουργικές εξαρτήσεις (functional dependencies) μπορούν να εκφραστούν με τη μορφή μιας οντολογίας, δηλαδή με ένα σύνολο φορμαλιστικών κανόνων, και αυτό μπορεί να γίνει αυτόματα με ένα υπάρχον αλγόριθμο εξόρυξης λειτουργικών εξαρτήσεων, ο οποίος αναγνωρίζει τυχόν εξαρτήσεις στα δεδομένα. Συσχετίσεις στα δεδομένα συνήθως οδηγούν σε διαρροή δεδομένων λόγω των διαδρομών συμπερασμού.

- Ολόκληρη η διαδικασία παρέχει αποδείξεις γιατί ένα μέρος των δεδομένων πρέπει να παραμείνει κρυφό. Με άλλα λόγια, πρέπει να μπορούμε να βρίσκουμε αποδοτικά μέσω του γράφου εξαρτήσεων ποιες είναι οι διαδρομές συμπερασμού που διακόπτουμε με κάθε αφαίρεση δεδομένων.

- Η αυτόματη διαδικασία απαιτεί συλλογιστική (reasoning) με τους κανόνες συμπερασμού, είτε με όλη τη βάση δεδομένων (bottom-up), είτε με ορισμένα στοιχεία των δεδομένων που έχουν οριστεί δημόσια και ευαίσθητα (top-down).

Ένα παράδειγμα χρήσης που αποτέλεσε και κίνητρο για ενασχόληση με το συγκεκριμένο πρόβλημα είναι αυτό που περιγράφεται στη συνέχεια. Έστω ότι σε ένα νοσοκομείο πρέπει να κατασκευαστεί ένα μέρος ενός πληροφοριακού συστήματος για τη γραμματεία του νοσοκομείου που θα έχει ορισμένες πληροφορίες για τους ασθενείς. Το ότι ένας ασθενής είναι άρρωστος, είναι ευαίσθητη πληροφορία. Έτσι, στα παραδοσιακά συστήματα, ο διαχειριστής θα όριζε μια όψη ασφαλείας χωρίς την πληροφορία ότι, για παράδειγμα, ο ασθενής Γιάννης είναι άρρωστος. Όμως, στη βάση δεδομένων και στην όψη υπάρχουν οι πληροφορίες ότι ο Γιάννης είναι μολυσμένος από τον ιό Α, και ότι εμβολιάστηκε το 1994. Αυτές οι πληροφορίες δεν είναι ευαίσθητες. Όμως, με χρήση λογικού συμπερασμού μπορεί κανείς να συμπεράνει ότι το εμβόλιο του Γιάννη δεν ισχύει πια, διότι τα εμβόλια του 1994 ήταν τύπου Χ και δεν ισχύουν πια (μπορούμε να οδηγηθούμε σε αυτό το συμπέρασμα με χρήση μιας οντολογίας που μοντελοποιεί ιατρική γνώση). Αφού ο Γιάννης δεν είναι πια εμβολιασμένος και έχει μολυνθεί με τον ιό Α, τότε μπορούμε να συμπεράνουμε ότι είναι άρρωστος. Ο διαχειριστής του συστήματος είναι δύσκολο να βρει αυτή τη διαρροή πληροφορίας χειροκίνητα. Η προσέγγιση μας μπορεί να βρει αυτό το μονοπάτι συμπερασμού με αυτόματο τρόπο και να μας πει ποια είναι η πληροφορία που πρέπει επιπρόσθετα να κρύψουμε ώστε να προστατεύεται η ιδιωτικότητα του Γιάννη.

Τέλος, η οντολογίας της SNOMED CT χρησιμοποιείται ευρέως για τη μοντελοποίηση ιατρικών δεδομένων και χρησιμοποιήθηκε και σε αυτή την εργασία για τη πειραματική αξιολόγηση. Ο συνδυασμός της SNOMED CT με τη βάση δεδομένων MIMIC II, που περιέχει

πραγματικούς ιατρικούς φακέλους ασθενών αποτελούν ένα πραγματικό παράδειγμα χρήσης του προβλήματος που παρουσιάζουμε σε αυτή την εργασία και στο οποίο η διασφάλιση της ιδιωτικότητας φαίνεται επιτακτική.

1.2 Συνεισφορά αυτής της Εργασίας

Αυτή η δουλειά προτείνει έναν αλγόριθμο ανωνυμοποίησης δεδομένων που ανταπεξέρχεται στο σενάριο της προσέγγισης που ακολουθούμε. Η βασικότερη συνεισφορά της εργασίας αυτής είναι:

1. Ο ορισμός του προβλήματος ανωνυμοποίησης δεδομένων υπό λειτουργικές εξαρτήσεις. Παρουσίαση μια πρωτότυπης προσέγγισης του προβλήματος της προστασίας της ιδιωτικότητας στο συγκεκριμένο σενάριο.

2. Ο ορισμός του γράφου εξαρτήσεων. Το μοντέλο του γράφου εξαρτήσεων παρουσιάζει ομοιότητες με το μοντέλο της προέλευσης δεδομένων (data provenance), όμως εφαρμόζεται σε διαφορετικό πεδίο, κατά την διαδικασία του λογικού συμπερασμού. Ακόμη, διαφοροποιείται η υλοποίηση του γράφου εξαρτήσεων.

3. Η ανάπτυξη και υλοποίηση αλγορίθμου ανωνυμοποίησης δεδομένων με λειτουργικές συσχετίσεις που επιλύει το πρόβλημα της προστασίας της ιδιωτικότητας στο πρόβλημα μας.

1.3 Οργάνωση του τόμου

Η εργασία αυτή είναι οργανωμένη σε οκτώ κεφάλαια ως εξής:

Στο Κεφάλαιο 2 δίνεται το θεωρητικό υπόβαθρο και οι βασικές έννοιες που σχετίζονται με τη διπλωματική αυτή. Αρχικά περιγράφεται η έννοια της ιδιωτικότητας και οι απειλές κατά αυτής, στη συνέχεια παρατίθενται οι βασικοί μετασχηματισμοί δεδομένων για ανωνυμοποίηση και τέλος δίνεται η έννοια της οντολογίας και της συλλογιστικής ανάλυσης σε σχεσιακά δεδομένα.

Στο Κεφάλαιο 3 παρουσιάζεται η βιβλιογραφία που αφορά σχετικές εργασίες για τη προστασία της ιδιωτικότητας κατά τη δημοσίευση δεδομένων για διαφορετικά είδη δεδομένων και σε διαφορετικά σενάρια.

Στο Κεφάλαιο 4 παρουσιάζονται οι αρχικές σκέψεις και το κίνητρο για την επίλυση του προβλήματος, στη συνέχεια ορίζεται φορμαλιστικά το πρόβλημα με το οποίο ασχολούμαστε σε αυτή την εργασία.

Στο Κεφάλαιο 5 περιγράφεται ο αλγόριθμος που προτείνεται για την κατασκευή του γράφου συσχετίσεων και για την ανωνυμοποίηση δεδομένων με λειτουργικές εξαρτήσεις.

Το Κεφάλαιο 6 αναφέρεται στη πειραματική διαδικασία που εκτελέστηκε με σκοπό την αξιολόγηση της αποδοτικότητας του αλγορίθμου, καθώς και τα αποτελέσματα που προέκυψαν σχετικά με τον χρόνο εκτέλεσης, την διασφάλιση της ιδιωτικότητας και την διατήρηση της χρήσιμης πληροφορίας.

Στο Κεφάλαιο 7 καταγράφονται οι τεχνικές λεπτομέρειες υλοποίησης του αλγορίθμου που αναπτύχθηκε, αναλύονται οι βασικές μέθοδοι τους και περιγράφονται οι βασικές δομές δεδομένων που χρησιμοποιήθηκαν.

Τέλος, το Κεφάλαιο 8 συνοψίζει τη συνεισφορά της εργασίας αυτής, παρουσιάζει κάποια συμπεράσματα και προτείνει νέα θέματα και προβλήματα ως συνέχεια της δουλειάς αυτής.

Κεφάλαιο 2

Θεωρητικό υπόβαθρο

Σε αυτό το κεφάλαιο θα παρουσιάσουμε συνοπτικά το θεωρητικό υπόβαθρο που κρίνεται απαραίτητο ώστε να είναι ευκολότερη η κατανόηση των εννοιών με τις οποίες πραγματεύεται η εργασία αυτή στα επόμενα κεφάλαια. Στη συνέχεια αποσαφηνίζεται η έννοια της ιδιωτικότητας και της ανωνυμοποίησης, και παρουσιάζονται οι βασικές αρχές των μεθόδων μετασχηματισμού των αρχικών δεδομένων, όπως είναι οι μέθοδοι της γενίκευσης, της απομάκρυνσης όρων και της προσθήκης θορύβου. Τέλος, δίνουμε κάποιους ορισμούς για τις έννοιες της οντολογίας και της συλλογιστικής ανάλυσης.

2.1 Ιδιωτικότητα

Η έννοια της προστασίας της ιδιωτικότητας (privacy),που δεν πρέπει να συγχέεται με την έννοια της ασφάλειας (security), έγκειται στην αποτροπή κάποιου, που συνήθως αποκαλείται *εισβολέας* ή *αντίπαλος*, από το να αποκτήσει πρόσθετες και ευαίσθητες γνώσεις για μια οντότητα του πραγματικού κόσμου (π.χ. ένα άτομο), αναλύοντας τις εγγραφές που περιλαμβάνονται σε ένα σύνολο δημοσιευμένων δεδομένων. Στη γενική περίπτωση, υποθέτουμε ότι κάποια γνώση είναι γνωστή ήδη στον αντίπαλο πριν από τη δημοσίευση των δεδομένων. Αυτή η γνώση, είναι το *γνωστικό υπόβαθρο* (background knowledge) του εισβολέα. Δημοσιεύοντας οποιαδήποτε γνώση, πρέπει να μετασχηματίζουμε τα αρχικά δεδομένα ώστε να προστατεύσουμε την ιδιωτικότητα των οντοτήτων, δηλαδή ώστε ο αντίπαλος να μην μπορεί να εξάγει πληροφορίες που είναι ευαίσθητες για της οντότητες μας. Το γνωστικό υπόβαθρο του αντιπάλου, οι ευαίσθητες πληροφορίες για μια οντότητα, ο τύπος της επίθεσης που μπορεί κανείς να εκτελέσει με βάση τα δημοσιευμένα στοιχεία και η εγγύηση που θέλουμε να παρέχεται από τη μέθοδο ανωνυμοποίησης ποικίλλουν ανάλογα με την εφαρμογή και ορίζουν σε κάθε περίπτωση ένα διαφορετικό μοντέλο ιδιωτικότητας.

Σε ένα τέτοιο μοντέλο, τα δεδομένα που δημοσιεύονται μπορούν να χωριστούν σε τρεις κατηγορίες. Στα *άμεσα αναγνωριστικά* (π.χ. Ταχυδρομικός Κώδικάς ή Α.Φ.Μ), γνωρίζοντας τα οποίο κάποιος μπορεί να αναγνωρίσει άμεσα πληροφορίες για μια οντότητα και γι'αυτόν τον λόγο πρέπει να απομακρύνονται πάντα πριν τη δημοσίευση των δεδομένων. Στα *εν δυνάμει αναγνωριστικά* (Quasi Identifiers - QIs), τα οποία μπορούν να χρησιμοποιηθούν από

Σχήμα 2.1: Συνδυασμός Δεδομένων για αναγνώριση οντότητας

τον αντίπαλο για να συνδέσει ένα μέρος της πληροφορίας στα δημοσιευμένα δεδομένα με μια οντότητα του πραγματικού κόσμου. Πριν συνεχίσουμε, αξίζει να αναφέρουμε ότι το γνωστικό υπόβαθρο ενός αντιπάλου μπορεί να μην περιορίζεται μόνο στη γνώση QIs για μερικές οντότητες, αλλά μπορεί να έχει στατιστικές πληροφορίες για τα αρχικά δεδομένα (π.χ. στατιστική κατανομή ατόμων που πάσχουν από κάποια ασθένεια) ή να γνωρίζει τον αλγόριθμο που έχει χρησιμοποιηθεί για την ανωνυμοποίηση. Σε αυτή τη περίπτωση θα πρέπει τα δεδομένα να προστατεύονται και από τη τεχνική της αντίστροφης επίθεσης (reverse-engineering attack). Τέλος, υπάρχουν τα *ευαίσθητα στοιχεία* (Sensitive Items - SIs), όπου είναι τα στοιχεία τα οποία είναι κρυφά από έναν εισβολέα και θέλουμε να τα κρατήσουμε κρυφά. Σε πολλές σύνθετες περιπτώσεις δεν είναι απλό να διαχωρίσουμε τα δεδομένα σε αυτές τις τρεις κατηγορίες, συγκεκριμένα σε ρεαλιστικά σενάρια είναι δύσκολη η διάκριση μεταξύ των QIs και των SIs. Σε άλλα σύνθετα σενάρια, κάποια ευαίσθητα στοιχεία μπορεί να είναι γνωστά σε μερικούς εισβολείς και να μπορούν να χρησιμοποιηθούν ως εν δυνάμει αναγνωριστικά, οπότε και το πρόβλημα της ανωνυμοποίησης γίνεται εξαιρετικά δύσκολο.

Στη συνέχεια παραθέτουμε μερικά πραγματικά παραδείγματα για να φανεί τόσο η χρησιμότητα την ανωνυμοποίησης των δεδομένων κατά τη δημοσίευση, όσο και για να γίνουν πιο κατανοητές οι έννοιες που ορίστηκαν προηγουμένως. Σύμφωνα με μια έρευνα που πραγματοποιήθηκε το 2000 από τη καθηγήτρια Sweeney, το 87% του πληθυσμού των Ηνωμένων Πολιτειών της Αμερικής, μπορεί να προσδιοριστεί μοναδικά μόνο με χρήση του ταχυδρομικού κώδικα, του φύλου και της ημερομηνίας γεννήσεώς [42]. Το σύνολο των χαρακτηριστικών αυτών αποτελούν εν δυνάμει αναγνωριστικά QIs, γιατί χρησιμοποιούνται από έναν εισβολέα για να εξαχθούν ευαίσθητες πληροφορίες για πολίτες των Η.Π.Α. Η ίδια μελέτη αναφέρει ότι υπάρχει και ένα άλλο σύνολο στοιχείων, ακόμα πιο γενικό, με βάση το οποίο μπορούν να προσδιοριστεί μοναδικά περίπου ο μισός πληθυσμός των Η.Π.Α. Αυτό το σύνολο είναι ο δήμος

ή η πόλη κατοικίας, το φύλο και η ημερομηνία γέννησης. Η Sweeny σε ένα επόμενο άρθρο της, παρουσίασε μια πραγματική απειλή της ιδιωτικότητας κατά του κυβερνήτη της Μασαχουσέτης [43]. Με έναν απλό συνδυασμό από στοιχεία από δυο διαφορετικές βάσεις δεδομένων, όπου η πρώτη αφορούσε τα εκλογικά στοιχεία πολιτών από δημόσιους καταλόγους, και η δεύτερη ιατρικά δεδομένα από οργανισμούς ασφάλισης, ήταν αρκετά για να ανευρεθεί ο ιατρικός φάκελος του κυβερνήτη. Αυτό έγινε με συνδυασμό από στοιχεία από τις δυο βάσεις, όπως φαίνεται στο σχήμα 2.1. Το σύνολο των στοιχείων ξεχωριστά σε κάθε σύνολο δεδομένων, δεν αναγνωρίζει μοναδικά τον ιδιοκτήτη της οντότητας, αλλά ο συνδυασμός τους αποτελεί ένα εν δυνάμει αναγνωριστικό, το οποίο προσδιορίζει είτε τον μοναδικό ιδιοκτήτη είτε ένα μικρό σύνολο πιθανών ιδιοκτητών της εγγραφής.

2.2 Απειλές κατά της Ιδιωτικότητας

2.2.1 Αποκάλυψη της Συμμετοχής

Το πρώτο είδος απειλής της ιδιωτικότητας είναι η αποκάλυψη της συμμετοχής (membership disclosure). Σε αυτή την περίπτωση ο αντίπαλος, ανεξαρτήτως του γνωστικού του υποβάθρου, δεν θα πρέπει να βρίσκεται σε θέση να μπορεί να συμπεράνει με υψηλή βεβαιότητα ότι η εγγραφή που αντιστοιχεί σε μια συγκεκριμένη οντότητα συμπεριλαμβάνεται στα δεδομένα που έχουν δημοσιευτεί. Αυτή η απειλή εμφανίζεται όταν τα αρχεία που δημοσιεύονται έχουν επιλεγεί με κάποιο κριτήριο το οποίο αποτελεί ευαίσθητη πληροφορία και με κάποιο τρόπο είναι γνωστή πληροφορία στον αντίπαλο.

Ένα τυπικό παράδειγμα είναι όταν ένας αντίπαλος γνωρίζει ότι ένας πολίτης μιας συγκεκριμένης χώρας έχει μολυνθεί από τον ιό του HIV και τα δημοσιευμένα δεδομένα περιλαμβάνουν όλους τους πολίτες αυτής της χώρα με HIV. Τότε, προφανώς, ο αντίπαλος γνωρίζει ότι μια εγγραφή των δεδομένων ανήκει στον πολίτη που γνωρίζει. Αξίζει να επισημάνουμε ότι στα περισσότερα προβλήματα ιδιωτικότητας, θεωρείται ότι η ύπαρξη της εγγραφής μιας οντότητας στο σύνολο των δεδομένων είναι γνωστή στον αντίπαλο από την αρχή.

2.2.2 Αποκάλυψη της Ταυτότητας

Το δεύτερο είδος απειλής κατά της ιδιωτικότητας στη δημοσίευση δεδομένων είναι η αποκάλυψη της ταυτότητας (identity disclosure). Σε αυτή την περίπτωση ο εισβολέας, ανεξαρτήτως του γνωστικού του υπόβαθρου, δεν θα πρέπει να βρίσκεται σε θέση να συνδέσει με υψηλό βαθμό βεβαιότητας μια συγκεκριμένη εγγραφή σε μια οντότητα. Εδώ, υποθέτουμε ότι ο επιτιθέμενος γνωρίζει ήδη ότι η εγγραφή της οντότητας που τον ενδιαφέρει βρίσκεται ήδη στα δεδομένα.

Η πιο γνωστή τεχνική ανωνυμοποίησης που έχει στόχο την πρόληψη από την απειλή της αποκάλυψης της ταυτότητας είναι η *k-ανωνυμία* [43]. Το αποτέλεσμα των αλγορίθμων *k-ανωνυμοποίησης, έχουν ως αποτέλεσμα ένα σύνολο δεδομένων, στο οποίο ο αντίπαλος δεν είναι ικανός να συνδέσει μια εγγραφή με λιγότερες από k (k > 1) διακριτές οντότητες του πραγματικού κόσμου. Με αυστηρούς όρους, αυτό σημαίνει ότι το ίδιο σύνολο τιμών QIs*

εμφανίζεται σε τουλάχιστον k εγγραφές στο δημοσιευμένο σύνολο δεδομένων, ώστε να μην μπορούν να προσδιοριστούν με υψηλό βαθμό βεβαιότητας.

Ένα τυπικό παράδειγμα αυτής της απειλής, είναι το παράδειγμα που παραθέσαμε στην προηγούμενη ενότητα με τον κυβερνήτη της Μασαχουσέτης. Σε αυτό το παράδειγμα, ο εισβολέας, με χρήση του εκλογικού καταλόγου ως γνωστικό υπόβαθρο, συνέδεσε μια εγγραφή στα ιατρικά δεδομένα ασφαλιστικών ταμείων με την οντότητα του κυβερνήτη. Αν τα δεδομένα των ασφαλιστικών ταμείων είχαν ανωνυμοποιηθεί με κάποιο αλγόριθμο k-ανωνυμοποίησης, τότε ο αντίπαλος θα μπορούσε να συνδέσει τη συγκεκριμένη με τουλάχιστον k οντότητες, μια εκ των οποίων θα ήταν ο Κυβερνήτης.

2.2.3 Αποκάλυψη Χαρακτηριστικών

Το τρίτο είδος απειλής κατά της ιδιωτικότητας στη δημοσίευση δεδομένων είναι η αποκάλυψη χαρακτηριστικών (attribute disclosure). Σε αυτή την περίπτωση ο εισβολέας, ανεξαρτήτως του γνωστικού του υπόβαθρου, δεν θα πρέπει να βρίσκεται σε θέση να χρησιμοποιήσει τις εγγραφές στα δημοσιευμένα δεδομένα προκειμένου να ανακαλύψει με υψηλή πιθανότητα νέα SIs για μια οντότητα του πραγματικού κόσμου. Ομοίως με την αποκάλυψη της ταυτότητας, η ύπαρξη της εγγραφής μιας οντότητας στα δημοσιευμένα δεδομένα θεωρείται ότι είναι γνωστή στον εισβολέα από την αρχή. Ακόμη, η αποκάλυψη της ταυτότητας οδηγεί προφανώς σε αποκάλυψη χαρακτηριστικών, το αντίθετο όμως δεν ισχύει, καθώς μπορεί να συμβεί αποκάλυψη χαρακτηριστικών με ή χωρίς αποκάλυψη της ταυτότητας.

Η πιο γνωστή τεχνική ανωνυμοποίησης που έχει ως στόχο την πρόληψη από την απειλή της αποκάλυψης χαρακτηριστικών είναι η l-διαφορετικότητα [36]. Το αποτέλεσμα των αλγορίθμων l-διαφορετικότητας, έχουν ως αποτέλεσμα ένα σύνολο δεδομένων, στο οποίο ο αντίπαλος δεν είναι ικανός να συνδέσει λιγότερα από l διαφορετικά SIs με μια οντότητα, ή διαφορετικά, σε όλες τις εγγραφές που έχουν τις ίδιες τιμές στο σύνολο QIs, υπάρχουν τουλάχιστον l διαφορετικά SIs. Οι περιορισμοί που εμφανίζει η l-διαφοροποίηση στη προστασία από την αποκάλυψη χαρακτηριστικών έχει επισημανθεί [35]. Έτσι, εισάγεται ένα νέο μοντέλο προστασίας της ιδιωτικότητας πέρα από την k-ανωνυμία και την l-διαφορετικότητα. Αυτή η προσέγγιση, όμως, προϋποθέτει ότι η κατανομή ενός SI σε κάθε k-ανώνυμη ομάδα εγγραφών είναι κοντά στην κατανομή του SI στο σύνολο των δεδομένων.

Ένα παράδειγμα για αυτό το είδος εισβολής είναι αν σε ένα σύνολο με δημοσιευμένα δεδομένα για ασθενείς που έχουν τον ιό του HIV, μπορούμε να συμπεράνουμε, για παράδειγμα, ότι όλες οι γυναίκες που είναι χορεύτριες και είναι άνω των 30 ετών έχουν HIV. Τότε ένας εισβολέας θα μπορούσε να εφαρμόσει αυτά τα δεδομένα σε δημογραφικά στοιχεία και να συμπεράνει ότι κάποιες οντότητες του πραγματικού κόσμου έχουν τον ιό του HIV με υψηλή βεβαιότητα.

2.2.4 Γενικές Παρατηρήσεις

Προφανώς, η προστασία από την αποκάλυψη της συμμετοχής προστατεύει και από την αποκάλυψη της ταυτότητας και από την αποκάλυψη χαρακτηριστικών. Με άλλα λόγια, αν ο

αντίπαλος δεν μπορεί να συμπεράνει αν τα δημοσιευμένα δεδομένα συμπεριλαμβάνουν την εγγραφή μια συγκεκριμένης οντότητας, τότε δεν μπορεί να συνδέσει και καμία εγγραφή ή κάποιο στοιχείο με οποιαδήποτε γνωστή οντότητα. Από την άλλη, η πρόληψη από την αποκάλυψη της ταυτότητας δεν προστατεύει επίσης και από την αποκάλυψη χαρακτηριστικών. Ένα παράδειγμα αυτής της περίπτωσης είναι η λεγόμενη επίθεση ομοιογένειας. Αυτό το είδος επίθεσης συμβαίνει όταν όλες οι εγγραφές μιας k-ανώνυμης ομάδας περιέχουν τα ίδια SIs. Τότε, ένας εισβολέας, ο οποίος γνωρίζει ότι η εγγραφή μιας οντότητας περιλαμβάνεται στα δεδομένα, μπορεί εύκολα να συμπεράνει ότι τα συγκεκριμένα SIs σχετίζονται και με την οντότητα που τον ενδιαφέρει. Διαισθητικά, η προστασία από την αποκάλυψη χαρακτηριστικών είναι ισχυρότερη από την προστασία από την αποκάλυψη της ταυτότητας, ενώ και οι δυο είναι πιο αδύναμες σε σχέση με την προστασία από την αποκάλυψη της συμμετοχής.

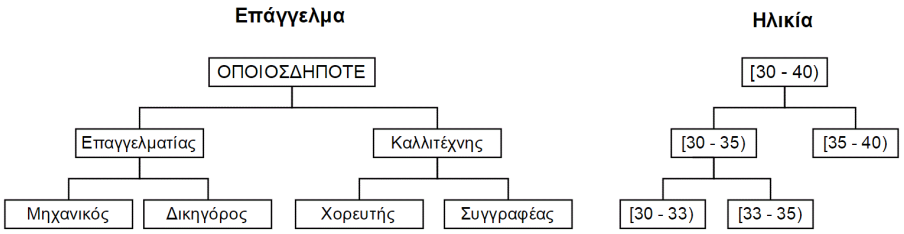
2.3 Ποιότητα Ανωνυμοποιημένων Αποτελεσμάτων

Η προστασία της ιδιωτικότητας συνδέεται αντίστροφα με την ποιότητα των δημοσιευμένων δεδομένων και η σχέση αυτή έχει τονιστεί από τα πρώτα κιόλας στάδια της έρευνας στον τομέα της ιδιωτικότητας. Διαισθητικά, όσο πιο ισχυρή είναι η εγγύηση της ιδιωτικότητας, δηλαδή, όσο πιο δύσκολο είναι για έναν εισβολέα να συμπεράνει πρόσθετες πληροφορίες από τα δεδομένα, τόσο μικρότερη είναι και η χρησιμότητα των δημοσιευμένων δεδομένων. Αυτό γίνεται εύκολα κατανοητό αν θεωρήσουμε έναν απλό χρήστη που θέλει να αναλύσει τα δεδομένα ως έναν αντίπαλο χωρίς κανένα γνωστικό υπόβαθρο. Συνέπεια αυτής της αντίστροφης σχέσης είναι ότι όλοι οι αλγόριθμοι ανωνυμοποίησης έχουν ως στόχο να διασφαλίσουν την ιδιωτικότητα προκαλώντας τη μικρότερη δυνατή απώλεια πληροφορίας στα τελικά δεδομένα.

Για την μέτρηση της ποιότητας συνήθως χρησιμοποιούνται οι τύποι που είναι γνωστοί ως *μετρικές χρησιμότητας*. Οι μετρικές χρησιμότητας ποικίλλουν μεταξύ των διαφορετικών περιπτώσεων χρήσεως αλλά όλες βασίζονται στη θεμελιώδη αρχή ότι ένα ανωνυμοποιημένο σύνολο δεδομένων πρέπει να διατηρεί τις στατιστικές ιδιότητες του αρχικού συνόλου δεδομένων για να είναι χρήσιμο. Αν λάβουμε υπόψιν ότι οι ιδιότητες αυτές αξιολογούνται με ειδικούς αλγορίθμους, η χρησιμότητα σε σχέση με ένα δεδομένο αλγόριθμο εξασφαλίζεται αν και μόνο αν η εφαρμογή του αλγορίθμου στα ανωνυμοποιημένα δεδομένα παράγει αποτελέσματα τα οποία είναι πολύ κοντά σε εκείνα που παράγει όταν εφαρμόζεται στο αρχικό σύνολο δεδομένων.

2.4 Τεχνικές Μετασχηματισμού Δεδομένων

Συνήθως, τα αρχικά δεδομένα δεν ικανοποιούν τις συγκεκριμένες απαιτήσεις ιδιωτικότητας που θέλουμε. Οπότε, τα αρχικά δεδομένα πρέπει να μετασχηματιστούν πριν δημοσιευτούν. Η ανωνυμοποίηση γίνεται εφαρμόζοντας μια ακολουθία από διαφορετικούς μετασχηματισμούς στον πίνακα με τα αρχικά δεδομένα. Οι μετασχηματισμοί αυτοί μπορεί να είναι μιας εκ των ακόλουθων μορφών: γενίκευση, απομάκρυνση όρων, αποσυσχέτιση, προσθήκη θορύβου. Στη συνέχεια αναλύουμε τις διαφορετικές αυτές μορφές μετασχηματισμού των αρχικών δεδομένων.



Σχήμα 2.2: Δέντρα ταξινόμησης για Επάγγελμα και Ηλικία

Πριν συνεχίσουμε, αξίζει να επισημάνουμε ότι, εκτός από την ανωνυμοποίηση μέσω μετασχηματισμού των δεδομένων πριν τη δημοσίευση, μια άλλη προσέγγιση είναι η απευθείας (ή διαδραστική) ανωνυμοποίηση, όπου το σύνολο των δεδομένων δεν δημοσιεύεται εξολοκλήρου. Αντ'αυτού, φιλοξενείται σε ένα δημόσιο διακομιστή όπου οι ενδιαφερόμενοι χρήστες μπορούν να θέσουν ερωτήματα, τα αποτελέσματα των οποίων ανωνυμοποιούνται κατά την αποτίμηση. Όμως, η προσέγγιση αυτή δεν έχει επικρατήσει επειδή: (α) η δυναμική ανωνυμοποίηση των απαντήσεων που επιστρέφονται από ένα τέτοιο σύστημα είναι ένα υπολογιστικά δύσκολο έργο (π.χ. εξαρτάται από τις προηγούμενες απαντήσεις που έχουν επιστραφεί από το σύστημα), και (β) η φιλοξενία των δεδομένων είναι συνήθως μια δαπανηρή διαδικασία που οι περισσότεροι οργανισμοί δεν μπορούν να αντέξουν οικονομικά.

2.4.1 Γενίκευση

Η γενίκευση (generalization) είναι η πρώτη τεχνική μετασχηματισμού που χρησιμοποιήθηκε για ανωνυμοποίηση δεδομένων και αρχικά παρουσιάστηκε μαζί με την έννοια της k-ανωνυμίας. Κάθε μετασχηματισμός γενίκευσης κρύβει κάποιες λεπτομέρειες στα QIs. Η αρχή της μεθόδου είναι να αντικαταστήσει κάποια αρχικά στοιχεία που μπορεί να οδηγήσουν σε παραβίαση της ιδιωτικότητας με γενικευμένα στοιχεία, δηλαδή ομάδες στοιχείων. Διαισθητικά, ο μετασχηματισμός αυτός κρύβει τα αρχικά (σπάνια) στοιχεία μέσα σε (συχνές) γενικεύσεις τους. Οι πιο γενικευμένες έννοιες προκύπτουν μέσα από δέντρα ταξινόμησης (taxonomy trees) των εννοιών, όπως αυτά στο σχήμα 2.2. Στο ίδιο σχήμα παρουσιάζεται και ένα αντιπροσωπευτικό παράδειγμα, όπου η αντικατάσταση των όρων *Δικηγόρος* και *Μηχανικός* αντικαθίστανται από τον πιο γενικό όρο *Επαγγελματίας*. Αυτές οι γενικεύσεις μπορεί να αντιστοιχούν είτε σε κατηγορίες εννοιολογικά αλληλένδετων στοιχείων, όπως στο προηγούμενο παράδειγμα, είτε σε κατηγορίες που ορίζονται αυθαίρετα από τον υπεύθυνο της ανωνυμοποίησης. Ένα γενικευμένο στοιχείο μπορεί να είναι γενίκευση ενός άλλου στοιχείου, και έτσι σχηματίζεται το ιεραρχικό δέντρο ταξινόμησης, όπου εμφανίζονται όλα τα επίπεδα γενίκευσης. Επειδή η ιεραρχία αυτή είναι ένα δέντρο, η αντικατάσταση στοιχείων κατά την ανωνυμοποίηση είναι μια ντετερμινιστική διαδικασία.

Με τη χρήση της ιεραρχίας γενίκευσης, τα στοιχεία γενικεύονται σε ένα υψηλότερο επίπεδο του δέντρου ταξινόμησης προκειμένου να διασφαλιστεί το κριτήριο της ιδιωτικότητας που επιθυμούμε. Εκτός από τη λεγόμενη ολική κωδικοποίηση, όπου όλες οι εμφανίσεις ενός στοιχείου είτε παραμένουν οι ίδιες είτε γενικεύονται στο ίδιο επίπεδο ιεραρχίας, υπάρχει και

η τοπική κωδικοποίηση, η οποία προσπαθεί να ελαχιστοποιήσει την απώλεια πληροφορίας κατά την ανωνυμοποίηση. Η τοπική κωδικοποίηση χρησιμοποιείται τόσο στην παραδοσιακή k-ανωνυμία, όσο και στη l-διαφορετικότητα. Σε αυτή την τεχνική, η γενίκευση μπορεί να είναι μερική, δηλαδή, διαφορετικές εμφανίσεις του ίδιου όρου μπορούν να γενικευτούν σε διαφορετικά επίπεδα ιεραρχίας ή και να μην γενικευτούν καθόλου.

2.4.2 Απομάκρυνση Όρων

Η απομάκρυνση όρων, η οποία μπορεί να απομακρύνει τόσο έναν ολόκληρο όρο όσο και κάποια στοιχεία αυτού, είναι μια άλλη τεχνική μετασχηματισμού που εισήχθη νωρίς στη μελέτη της ανωνυμοποίησης των δεδομένων. Η βασική ιδέα της μεθόδου είναι ότι αντικείμενα που σπάνια εμφανίζονται στα δεδομένα, είναι ορισμένες φορές προτιμότερο να αφαιρεθούν τελείως από τα δεδομένα. Γενικά, η απομάκρυνση όρων παράγει περισσότερη απώλεια πληροφορίας από τις υπόλοιπες μεθόδους μετασχηματισμού των δεδομένων, αλλά μπορεί να είναι πολύ χρήσιμη όταν το τμήμα των δεδομένων που θέλουμε να αποκρύψουμε είναι αμελητέο ως προς το σύνολο των δεδομένων. Έτσι, χρησιμοποιείται συνήθως σε συνδυασμό με άλλες μεθόδους μετασχηματισμού, τις περισσότερες φορές με γενίκευση και προσθήκη θορύβου. Για παράδειγμα, η απομάκρυνση όρων είναι πολύ χρήσιμη με γενίκευση διότι βοηθάει στο να ελαχιστοποιηθεί το επίπεδο γενίκευσης και τελικά να έχουμε μικρότερη απώλεια πληροφορίας στα δεδομένα. Σε ορισμένες εργασίες, η απομάκρυνση στοιχείων ακολουθείται από την προσθήκη νέων στη θέση τους. Σε αυτή την περίπτωση, τα στοιχεία που προστέθηκαν μπορεί να αντιστοιχούν στη μέση τιμή όσων απομακρύνθηκαν (όταν π.χ. τα στοιχεία είναι αριθμητικά).

2.4.3 Αποσυσχέτιση

Η αποσυσχέτιση, σε αντίθεση με τις δυο προηγούμενες τεχνικές μετασχηματισμού δεδομένων, δεν τροποποιεί τα δεδομένα, αλλά αποσυσχετίζει τη σχέση μεταξύ ορισμένων στοιχείων, μειώνοντας τη δομική πληροφορία που υπάρχει στο αρχικό σύνολο δεδομένων. Η βασική ιδέα της αποσυσχέτισης είναι να κρύψει το γεγονός ότι ορισμένα στοιχεία εμφανίζονται μαζί στην ίδια εγγραφή. Έτσι, ορισμένες προσεγγίσεις στοχεύουν στο να διαχωρίσουν τα στοιχεία οι συνδυασμοί των οποίων αποτελούν QIs, ενώ άλλες προσεγγίσεις στοχεύουν στο να διαχωρίσουν τα QIs από τα SIs. Ανεξάρτητα από τη συγκεκριμένη υλοποίηση, διαισθητικά η αποσυσχέτιση ισοδυναμεί με το σπάσιμο των εγγραφών του αρχικού συνόλου δεδομένων σε μικρότερες, οι οποίες μπορούν στη συνέχεια να δημοσιευτούν χωριστά ή με τη μορφή τελείως ανακατασκευασμένων αρχείων. Το μεγάλο πλεονέκτημα αυτής της μεθόδου είναι ότι δεν τροποποιεί τα δεδομένα.

2.4.4 Προσθήκη Θορύβου

Αν και έχει μελετηθεί εκτενώς από θεωρητική σκοπιά, η προσθήκη θορύβου είναι μια σχετικά νέα τεχνική στον τομέα της ανωνυμοποίησης. Παρουσιάστηκε αρχικά στο πλαίσιο του ελέγχου αποκάλυψης στατιστικών στοιχείων (statistical disclosure control) [48], και πιο πρόσφατα στα πλαίσια της διαφορικής ιδιωτικότητας (differential privacy) [17]. Η μέθοδος

αυτή εισάγει θόρυβο στα δεδομένα, συνήθως Gauss ή Laplace, με στόχο να νοθεύσει τις αρχικές εγγραφές και ταυτόχρονα να διατηρήσει (όσο το δυνατόν) τις στατιστικές ιδιότητες του αρχικού συνόλου δεδομένων. Με άλλα λόγια, η βασική ιδέα είναι η αντικατάσταση των ευαίσθητων τιμών s με $s+r$, όπου r είναι μια τυχαία τιμή που ακολουθεί κάποια κατανομή. Η τεχνική αυτή μπορεί να εφαρμοστεί τόσο σε αριθμητικά όσο και σε μη αριθμητικά δεδομένα.

Το πλεονέκτημα αυτής της μεθόδου σε σχέση με τις προηγούμενες τεχνικές μετασχηματισμού δεδομένων είναι ότι δεν εξαρτάται από το γνωστικό αντικείμενο του αντιπάλου. Γι'αυτό μπορεί να αντιμετωπίσει ισχυρούς αντιπάλους που έχουν ακόμα και στατιστικές πληροφορίες για τα αρχικά δεδομένα, αν και η προστασία σε αυτή τη περίπτωση δεν είναι πάντοτε εφικτή [29]. Ένα σημαντικό μειονέκτημα της μεθόδου είναι ότι συνήθως είναι πολύ αυστηρή, πράγμα που οδηγεί σε δημοσίευση ενός πολύ μικρού τμήματος των αρχικών δεδομένων που σε πολλές περιπτώσεις περιλαμβάνει μόνο πολύ συχνές εγγραφές.

2.5 Η έννοια της Οντολογίας

Σαφής ορισμός της έννοιας της οντολογίας δεν υπάρχει καθώς ο ορισμός διαφέρει ανάλογα με τον επιστημονικό τομέα όπου χρησιμοποιείται. Μια προσπάθεια σε έναν σύντομο ορισμό της έννοιας της οντολογίας είναι: *οντολογία είναι η τυπική περιγραφή ενός συνόλου πληροφοριών και των συσχετίσεων μεταξύ τους, ικανή να χρησιμοποιηθεί από υπολογιστές.*

Κάθε οντολογία αποτελείται από τη σύνθεση των εξής δυο τμημάτων:

- Το λεξιλόγιο (intensional knowledge) που αποτελείται από ονόματα εννοιών και σχέσεων. Η χρήση του λεξιλογίου είναι η περιγραφή της πληροφορίας που περιγράφει η οντολογία, ή διαφορετικά του 'κόσμου' που μοντελοποιεί.

- Ένα σύνολο επιπλέον γνώσης (extensional knowledge) σχετικά με την πληροφορία που περιγράφεται, το οποίο περιλαμβάνει δηλώσεις/ισχυρισμούς (assertions). Οι δηλώσεις αυτές αντιστοιχίζουν άτομα σε έννοιες και ζεύγη ατόμων ή ζεύγη σταθερών (literals) σε σχέσεις. Αυτό το τμήμα των οντολογιών είναι προαιρετικό.

Τα άτομα της οντολογίας αποτελούν, στην ουσία, τα αντικείμενα που θέλουμε να διαχειριστούμε και συχνά είναι αναγνωριστικά (URIs). Οι έννοιες του λεξιλογίου ισοδυναμούν με σύνολα ατόμων που μοιράζονται ένα τουλάχιστον κοινό χαρακτηριστικό και αναφέρονται συχνά ως κλάσεις (classes). Οι σχέσεις αντιστοιχούν σε σύνολα από ζεύγη ατόμων (ή ζεύγη ατόμου-σταθεράς) και ονομάζονται ιδιότητες (properties), επειδή ακριβώς προσδίδουν ιδιότητες στα άτομα συνδέοντάς τα μεταξύ τους (ή με κάποια σταθερά).

Οι οντολογίες χαρακτηρίζονται ως ο τυπικός προσδιορισμός της πληροφορίας ο οποίος εξασφαλίζει μια κοινή αντίληψη της περιοχής και παρέχει τη δυνατότητα συλλογιστικής ανάλυσης (reasoning), τόσο για την εξαγωγή νέων (υπονοούμενων) σχέσεων όσο και για τον έλεγχο της ισχύος ήδη υπαρχουσών (εύρεση αντιφάσεων). Η έννοια της τυπικότητας είναι καθοριστική για τη δυνατότητα των υπολογιστών να την αναλύσουν και άρα να διαχειριστούν αυτόματα και την ίδια την πληροφορία.

Ο όρος μοντελοποιημένη γνώση αναφέρεται στις θεωρίες αναπαράστασης γνώσης ως Βάση Γνώσης (Knowledge Base - KB) και αποτελεί θεμελιώδη έννοια την οποία θα χρησιμοποιούμε στο εξής. *Βάση Γνώσης (KB)* ονομάζουμε ένα σύνολο γνώσης που περιγράφεται (μοντελοποιείται) με χρήση ενός τυπικού φορμαλισμού ή αλλιώς μιας γλώσσας αναπαράστασης. Ακόμη, όταν αναφερόμαστε σε οντολογίες, εννοούμε Βάσεις Γνώσεις (KB) που έχουν γραφτεί με χρήση μιας τυπικής γλώσσας, όπως οι RDF και OWL.

Παράδειγμα. Στο σχήμα 2.3 βλέπουμε ένα παράδειγμα μιας μικρής οντολογίας. Η KB αυτή μοντελοποιεί τη γνώση για τον ιό Α και για έναν ασθενή με το όνομα *john*. Σε αυτή την οντολογία, τα $NotVaccinated$, $InfectedWithVirusA$, Ill, $VaccineTypeX$, και $Vaccinated1994$ αποτελούν απλές κλάσεις, ενώ, τα $InfectedWithVirusA \sqcap NotVaccinated$, $\exists Vaccinated.\{va\}$ και $\exists Vaccinated.VaccineTypeX$ είναι σύνθετες κλάσεις. $Vaccinated$ είναι ρόλος. Ο *john* και το *va* είναι άτομα (individuals). Το αξίωμα 1 δηλώνει ότι οι ασθενείς που έχουν μολυνθεί από τον ιό Α αλλά δεν είναι εμβολιασμένοι, είναι άρρωστοι. Το αξίωμα 2 δηλώνει ότι όλοι όσοι έχουν εμβολιαστεί με το εμβόλιο τύπου Χ, πρέπει να αντιμετωπίζονται σαν να μην έχουν εμβολιαστεί επαρκώς. Το αξίωμα 3 λέει ότι το εμβόλιο *va* είναι τύπου Χ. Τα αξιώματα 4 και 5 μας δίνουν πληροφορίες για τον *john*. Μας λένε ότι ο *john* εμβολιάστηκε το 1994 και ότι έχει μολυνθεί από τον ιό Α. Τέλος, το τελευταίο αξίωμα δηλώνει ότι όλοι όσοι εμβολιάστηκαν το 1994, εμβολιάστηκαν με το εμβόλιο *va*.

1. $InfectedWithVirusA \sqcap NotVaccinated \sqsubseteq Ill$
2. $\exists Vaccinated.VaccineTypeX \sqsubseteq NotVaccinated$
3. $\{va\} \sqsubseteq VaccineTypeX$
4. $\{john\} \sqsubseteq Vaccinated1994$
5. $\{john\} \sqsubseteq InfectedWithVirusA$
6. $Vaccinated1994 \sqsubseteq \exists Vaccinated.\{va\}$

Σχήμα 2.3: Παράδειγμα Οντολογίας

2.6 Συλλογιστική Ανάλυση

Οι γλώσσες αναπαράστασης δεν αποσκοπούν μόνο στη μοντελοποίηση της γνώσης, όπως αναφέραμε προηγουμένως, αλλά και στην εύκολη ανάλυσή της. Η ανάλυση αυτή περιλαμβάνει δύο σκέλη. Το πρώτο αφορά στον έλεγχο της ισχύος των αξιωμάτων της KB, ενώ το δεύτερο στην εξαγωγή νέων αξιωμάτων που προκύπτουν από τα υπάρχοντα μέσω μιας αλγοριθμικής διαδικασίας.

Ιδιαίτερο ενδιαφέρον έχει όταν η συλλογιστική ανάλυση χρησιμοποιείται για εξαγωγή νέων αξιωμάτων σε συνδυασμό με χρήση μιας οντολογίας. Αναλύοντας τα αξιώματα που υπάρχουν στη KB μέσω κατάλληλων αλγορίθμων μπορεί κάποιος να εξάγει υπονοούμενες πληροφορίες για τα άτομα, οι οποίες δεν υπάρχουν ρητά στα αρχικά δεδομένα. Στη βιβλιογραφία υπάρχουν αρκετοί αλγόριθμοι (π.χ. αλγόριθμοι Tableau [4]) οι οποίοι εκτελούν αναδρομικά τη συλλογιστική ανάλυση για εξαγωγή νέων αξιωμάτων και οι οποίοι έχουν εκτενή θεωρητική ανάλυση

[3] [4], όμως, οι λεπτομέρειες των αλγορίθμων είναι πέρα από τα πλαίσια αυτής της διπλωματικής. Αξίζει να αναφέρουμε ότι όσο περισσότερο εκφραστική είναι η γλώσσα αναπαράστασης της KB, τόσο δυσκολότερη, ως και αδύνατη είναι η πλήρης αλγοριθμική ανάλυση της γνώσης που μοντελοποιείται.

Παράδειγμα. Από τα αξιώματα που υπάρχουν στο παράδειγμα της προηγούμενης ενότητας (σχήμα 2.3), μπορεί κανείς να εξάγει επιπλέον υπονοούμενες πληροφορίες. Από τα αξιώματα 3 και 6, εξάγεται η γνώση ότι όσοι εμβολιάστηκαν το 1994, εμβολιάστηκαν με εμβόλιο τύπου X. Από αυτό το υπονοούμενο αξίωμα και το αξίωμα 4, καταλαβαίνουμε ότι ο *john* έχει εμβολιαστεί με εμβόλιο τύπου X. Αν συνδυάσουμε πάλι το νέο αξίωμα που παράχθηκε με το αξίωμα 2, τότε καταλαβαίνουμε ότι ο *john* δεν έχει εμβολιαστεί. Τελικώς, με χρήση του πρώτου αξιώματος συμπεραίνουμε ότι ο *john* είναι άρρωστος. Όλα τα νέα αξιώματα παράχθηκαν με λογικό συμπερασμό και με αυτόν τον τρόπο εξαγάγαμε υπονοούμενη γνώση που δεν υπήρχε ρητά στα αρχικά δεδομένα.

Κεφάλαιο 3

Σχετικές Εργασίες

Στο κεφάλαιο αυτό αρχικά γίνεται μια περιγραφή των αλγορίθμων ανωνυμοποίησης για διάφορες κατηγορίες δεδομένων, όπως για σχεσιακά δεδομένα και για πολυδιάστατα δεδομένα. Για κάθε κατηγορία δεδομένων, περιγράφουμε συνοπτικά αλγορίθμους που έχουν προταθεί στη βιβλιογραφία και πως διαφέρει η κάθε περίπτωση (το είδος των δεδομένων άρα και οι αλγόριθμοι αυτοί) από τη περίπτωση ανωνυμοποίησης που αναγνωρίσαμε και με την οποία ασχοληθήκαμε σε αυτή τη διπλωματική. Στη συνέχεια, αναλύουμε κάποιες προσεγγίσεις που είναι πιο κοντά με το πρόβλημα μας και παραθέτουμε που υπολείπονται έναντι της δικιάς μας προτεινόμενης λύσης.

3.1 Ανωνυμοποίηση Σχεσιακών Δεδομένων

Η μελέτη της προστασίας της ιδιωτικότητας ξεκίνησε από τη μελέτη δεδομένων από σχεσιακές βάσεις. Στο [43] προτείνεται η έννοια της k-ανωνυμίας ως τυπικό μοντέλο προστασίας των σχεσιακών δεδομένων. Στο ίδιο άρθρο προτείνονται κάποιες αρχικές τεχνικές που βασίζονται στην γενίκευση και την απομάκρυνση όρων για ανωνυμοποίηση των δεδομένων. Στο [37] απέδειξαν ότι η βέλτιστη k-ανωνυμοποίηση με γενίκευση και απομάκρυνση όρων, όταν τα QIs είναι πολυδιάστατα είναι ένα NP-δύσκολο πρόβλημα. Έτσι, πρότειναν δυο πολυωνυμικούς αλγορίθμους ως προς τον χρόνο που επιτυγχάνουν προσέγγιση ανεξάρτητη από το μέγεθος των δεδομένων. Ο προσεγγιστικός αλγόριθμος αυτός έχει πολυπλοκότητα $O(k \log k)$. Το όριο αυτό μειώθηκε και άλλο στο [40] , όπου οι συγγραφείς παρουσίασαν έναν $O(\log k)$ προσεγγιστικό αλγόριθμο για το πρόβλημα της k-ανωνυμίας και έδειξαν ότι είναι το κάτω όριο της πολυπλοκότητας για το πρόβλημα.

Ο αλγόριθμος Incognito [33] προσφέρει ένα μοντέλο που εγγυάται k-ανωνυμοποίηση μέσω ενός αλγορίθμου ολικής κωδικοποίησης. Ο αλγόριθμος αυτός βασίζεται στις εξής δυο βασικές ιδέες: στη από κάτω προς τα πάνω συνάθροιση (rollup) στις διαστάσεις την ανωνυμοποίησης των QIs και στην ιδέα του υπολογισμού της εκ των προτέρων. Παρ'ότι ο αλγόριθμος είναι εκθετικός ως προς το μέγεθος των QIs, σε μερικές περιπτώσεις μπορεί να είναι μέχρι και μιας τάξης μεγέθους γρηγορότερος και έτσι να είναι πρακτικός για ανωνυμοποίηση πλήρους κωδικοποίησης σε μεγάλες βάσεις δεδομένων. Ο αλγόριθμος Mondrian [34] βασίζεται σε

ένα μοντέλο πολλών διαστάσεων, το οποίο οδηγεί σε ανωνυμοποιήσεις υψηλότερου επιπέδου σύμφωνα με αρκετές διαφορετικές μετρικές. Ο αλγόριθμος αυτός είναι άπληστος και σύμφωνα με πειραματικά αποτελέσματα συχνά οδηγεί σε προτιμότερα αποτελέσματα από τους βέλτιστους αλγορίθμους που βασίζονται, όμως, σε μια διάσταση για την ανωνυμοποίηση. Η πολυσχεσιακή k-ανωνυμία προτάθηκε στο [38] και επεκτείνει την έννοια της k-ανωνυμίας σε πολλαπλές σχέσεις (πίνακες). Η ιδέα εδώ είναι ότι οι πληροφορίες που σχετίζονται με ένα συγκεκριμένο πρόσωπο βρίσκονται σε πολλές διαφορετικές σχέσεις (πίνακες) οι οποίες έχουν διαφορετικές λειτουργικές εξαρτήσεις (Functional Dependencies). Ο μετασχηματισμός των δεδομένων σε αυτή τη δουλειά στηρίζεται σε ολική κωδικοποίηση.

Οι παραπάνω προσεγγίσεις προστατεύουν από την αποκάλυψη της ταυτότητας, δηλαδή, δεν επιτρέπουν σε έναν εισβολέα να συνδέσει μια συγκεκριμένη εγγραφή με ένα συγκεκριμένο πρόσωπο. Ωστόσο, δεν εμποδίζουν τον εισβολέα από το να συνδέσει ορισμένα SIs με ένα πρόσωπο. Για να αντιμετωπιστεί αυτό το πρόβλημα, εισήχθη η έννοια της l-διαφορετικότητας [36]. Ο αλγόριθμος Anatomy [49] δεν γενικεύει ούτε απομακρύνει όρους από τα δεδομένα, όπως οι μέχρι τότε προσεγγίσεις, αντ'αυτού, αποσυσχετίζει τις εγγραφές και δημοσιεύει τα κομμάτια ξεχωριστά. Έτσι, ο αλγόριθμος αυτός διατηρεί σε μεγάλο βαθμό (σφάλμα μικρότερο από 10%) τις στατιστικές ιδιότητες των αρχικών δεδομένων, σε αντίθεση με τους υπόλοιπους αλγορίθμους που βασίζονται σε γενίκευση οι οποίοι δεν προτείνονται για ανάλυση των αποτελεσμάτων τους. Παρόλα αυτά, ο Anatomy περιορίζεται σε σχεσιακά δεδομένα με σαφή διάκριση μεταξύ QIs και SIs. Τέλος, η t-εγγύτητα [35] υιοθετεί μια σαφή διάκριση μεταξύ QIs και SIs σε ένα σχεσιακό πίνακα και εφαρμόζει ολική κωδικοποίηση.

3.2 Ανωνυμοποίηση Δεδομένων Συναλλαγών

Η έρευνα σχετικά με την ανωνυμοποίηση πολυδιάστατων δεδομένων ξεκίνησε από τον κίνδυνο εξόρυξης συχνών συνδυασμών στοιχείων από δεδομένα συναλλαγών (transactional data). Τα δεδομένα συναλλαγών αποτελούνται από τα αντικείμενα που αγοράστηκαν μαζί από κάποιο άτομο. Ακόμα και αν αφαιρεθούν τα προσωπικά στοιχεία του αγοραστή, τα υπόλοιπα δεδομένα μπορούν να είναι και πάλι επικίνδυνα για την ιδιωτικότητα του αγοραστή, καθώς ο επιτιθέμενος μπορεί να έχει μερική γνώση της συναλλαγής. Ενδιαφέρον είναι ότι τα δεδομένα συναλλαγών δεν έχουν καθορισμένη δομή και μπορούν να είναι πολυδιάστατα. Το [47] θεωρεί ένα σύνολο δεδομένων από συναλλαγές D, όπου κάθε συναλλαγή περιέχει μια σειρά από διαφορετικά αντικείμενα. Έστω S ένα σύνολο κανόνων συσχέτισης που μπορεί να δημιουργηθεί από το σύνολο των δεδομένων, και $S' \subset S$ είναι ένα σύνολο από κανόνες συσχέτισης (association rules) που πρέπει να μείνει κρυφό. Ο μετασχηματισμός των δεδομένων στο [47] βασίζεται στην προσθήκη ή στην απόκρυψη κάποιων από των υπαρχόντων στοιχείων από τα αρχικά δεδομένα D, λαμβάνοντας υπόψιν τη γνώση του S'. Όμως, συχνά προκύπτει ένας κίνδυνος στη δημοσίευση ανωνυμοποιημένων κανόνων συσχέτισης, τα λεγόμενα κανάλια συμπερασμού (inference channels), όπου μελετώνται στο [2]. Το πρόβλημα αυτό γίνεται εύκολα κατανοητό με το ακόλουθο παράδειγμα: υποθέτουμε έναν κανόνα $a_1 \bigwedge a_2 \bigwedge a_3 \Rightarrow a_4$, όπου τα a_i είναι στοιχεία, και ο κανόνας αυτός έχει υποστήριξη 80, και εμπιστοσύνη $98,7\%$.

Δηλαδή, ο κανόνας αυτός ισχύει για 80 στοιχεία, οπότε μπορούμε να υποθέσουμε ότι το νούμερο αυτό είναι αρκετά μεγάλο για να προστατέψουμε την ιδιωτικότητα κάποιου. Όμως, εξ'ορισμού μπορούμε να υπολογίσουμε την υποστήριξη του στοιχειοσυνόλου $a_1 \bigwedge a_2 \bigwedge a_3$ ως $80/0.987 \approx 81$. Οπότε μπορούμε να συμπεράνουμε ότι ο κανόνας $a_1 \bigwedge a_2 \bigwedge a_3 \bigwedge \neg a_4$ εμφανίζεται σε $81 - 80 = 1$ εγγραφές. Με άλλα λόγια, εάν ο επιτιθέμενος έχει αρνητικό υπόβαθρο γνώσης, δηλαδή, γνωρίζει ότι ένα άτομο συνδέεται με τα a_1, a_2, αλλά όχι με το a_3, τότε η ιδιωτικότητα αυτού του ατόμου βρίσκεται σε κίνδυνο. Στο [45] εισάγεται μια πιο χαλαρή εκδοχή της k-ανωνυμίας, σύμφωνα με την οποία ένας αντίπαλος έχοντας γνώση m στοιχείων της αρχικής εγγραφής, δεν μπορεί να διακρίνει αυτή την εγγραφή από $k-1$ άλλες. Προφανώς, δεν υπάρχει κάποια διάκριση μεταξύ QIs και SIs. Οι συγγραφείς παρέχουν ευριστικούς αλγορίθμους για την ανωνυμοποίηση των δεδομένων που βασίζονται σε γενίκευση, και χρησιμοποιούν τόσο ολική όσο και μερική κωδικοποίηση. Η προσέγγιση αυτή προστατεύει από την αποκάλυψη της ταυτότητας ενός ατόμου αλλά δεν μπορεί να χρησιμοποιηθεί για την προστασία από την αποκάλυψη χαρακτηριστικών. Το [50] εισάγει μια παρόμοια εγγύηση προστασίας της ιδιωτικότητας που ονομάζεται (h, k, p)-συνεκτικότητα και η οποία προστατεύει τόσο από την αποκάλυψη της ταυτότητας, όσο και από την αποκάλυψη χαρακτηριστικών. Όμως, οι μέθοδοι ανωνυμοποίησης βασίζονται στην απομάκρυνση όρων και απαιτούν την a priori γνώση των SIs. Ακόμη, όπως ήδη έχουμε αναφέρει, η απομάκρυνση όρων οδηγεί σε σημαντική απώλεια πληροφορίας.

Το [21] επεκτείνει το [49] για την επίτευξη της l-διαφορετικότητας σε δεδομένα συναλλαγών με ένα μεγάλο αριθμό αντικειμένων ανά συναλλαγή. Επειδή θεωρούνται τα QIs ξεχωριστά σύνολα από τα SIs, η βασική ιδέα του άρθρου είναι να δημιουργηθούν κλάσεις ισοδυναμίας όπου τα QIs δημοσιεύονται χωριστά από τα SIs και τους αριθμούς εμφανίσεων τους στα αρχικά δεδομένα, δίνοντας με αυτό τον τρόπο μια απλή λύση στο πρόβλημα της ανωνυμοποίησης. Τέλος, το [13] προβλέπει μια πιο πολύπλοκη εγγύηση της l-διαφορετικότητας για αραιά πολυδιάστατα δεδομένα που την ονομάζουν ρ-αβεβαιότητα. Σε αυτή, τα SIs μπορούν να χρησιμοποιηθούν και ως QIs. Ακόμα, ο αλγόριθμος που προτείνεται χρησιμοποιεί ολική κωδικοποίηση και απομάκρυνση όρων.

3.3 Ανωνυμοποίηση Δεδομένων Αναζήτησης από τον Ιστό

Η AOL δημοσίευσε σκόπιμα πολλά δεδομένα αναζήτησης το 2006 για ερευνητικούς σκοπούς. Όμως, ακολούθησε η αναγνώριση ορισμένων χρηστών [6] και έτσι δημιουργήθηκε ενδιαφέρον στην ερευνητική κοινότητα για την ανωνυμοποίηση δεδομένων αναζήτησης στο διαδίκτυο. Η προστασία της ιδιωτικότητας είναι δύσκολη υπόθεση σε αυτήν την περίπτωση για τους εξής λόγους [46]:

- Τα δεδομένα έχουν πάρα πολλές διαστάσεις και είναι πάρα πολύ αραιά. Τα δεδομένα περιέχουν 10 εκατομμύρια αναζητήσεις από 650000 μοναδικούς χρήστες. Το 90% των αναζητήσεων είναι μοναδικό. Έτσι, η ανωνυμοποίηση με αφαίρεση των μοναδικών όρων

θα οδηγούσε σε δημοσίευση μόνο του 50% των αρχικών δεδομένων, ενώ η απομάκρυνση των όρων που εμφανίζονται 3 φορές ή λιγότερες θα οδηγούσε σε δημοσίευση μόνο του 30% των αρχικών δεδομένων, με μόνο 3% του συνόλου των όρων.

- Τα λάθη των χρηστών (π.χ. ορθογραφικά λάθη ή ακρωνύμια) καθιστούν δύσκολη την αποσαφήνιση της σημασιολογίας των ερωτημάτων των χρηστών. Έτσι, η χρήση της μεθόδου της γενίκευσης για ανωνυμοποίηση είναι πρακτικά δύσκολη γιατί δεν είναι κατανοητή η σημασιολογία των ερωτημάτων των χρηστών.

- Η φύση των ερωτημάτων που θέτουν οι χρήστες σε συνδυασμό με τις διαφορετικές πηγές που μπορεί να αποκτήσει γνώση ο εισβολέας, κάνουν τη μοντελοποίηση του προβλήματος πολύ δύσκολη.

- Τα μεταδεδομένα που συνοδεύουν τα αρχεία αναζητήσεως (π.χ. το χρονικό στιγμιότυπο) κάνουν την ανωνυμοποίηση ακόμα πιο δύσκολη.

Υπάρχουν πολλές δουλειές που έχουν προσπαθήσει να βρούνε τα προβλήματα τα οποία καθιστούν τα δεδομένα της AOL ευάλωτα. Το [31] παρουσιάζει έναν αλγόριθμο που εγγυάται την διαφορική προστασία της ιδιωτικότητας μέσω της απομάκρυνσης όρων. Οι συγγραφείς προσφέρουν ένα πλαίσιο προστασίας της ιδιωτικότητας όμως η προτεινόμενη μέθοδος αντιμετωπίζει ένα από τα προβλήματα που αναφέραμε προηγουμένως. Συγκεκριμένα, η προτεινόμενη μέθοδος κρύβει εντελώς όλους τους σπάνιους όρους, οι οποίοι είναι η πλειονότητα των όρων στα δεδομένα της AOL. Στο [39] προτείνετε μια μέθοδος ανωνυμοποίησης για δεδομένα αναζητήσεων, όπου προστίθενται νέα στοιχεία στο σύνολο των δεδομένων τα οποία παρουσιάζουν όμοια κατανομή όπως τα αρχικά δεδομένα αλλά μπορούν να κρύψουν τους σπάνιους όρους.

3.4 Ανωνυμοποίηση Ημι-δομημένων Δεδομένων

Η ανωνυμοποίηση ημι-δομημένων δεδομένων (π.χ. XML, OWL) θέτει αρκετές τεχνικές προκλήσεις. Συγκεκριμένα, στα δεδομένα XML υπάρχει η ανάγκη για την αντιμετώπιση της πολύπλοκης δομικής πληροφορίας που υπάρχει στα δεδομένα και μπορεί να χρησιμοποιηθεί από τον αντίπαλο για να αποκαλύψει ευαίσθητες πληροφορίες. Ακόμη, υπάρχει το πρόβλημα των επαγωγικών επιθέσεων σε OWL δεδομένα, όπου οι επιθέσεις αυτές είναι αρκετά δύσκολο να αντιμετωπιστούν στη γενική περίπτωση επειδή απαιτούν κατάλληλους αλγορίθμους συλλογιστικής. Με άλλα λόγια, στα OWL δεδομένα, που μας ενδιαφέρουν, η αποκάλυψη ενός απλού γεγονότος μπορεί να μην οδηγήσει από μόνη της στη παραβίαση της ιδιωτικότητας, αλλά μπορεί να αποκαλύψει ευαίσθητες πληροφορίες αν συνδυαστεί με άλλα γνωστά γεγονότα ή με τις σχέσεις μιας οντολογίας που είναι γνωστές στον εισβολέα.

Στη συνέχεια, περιγράφουμε σχετικές δουλειές που υπάρχουν στον τομέα των ημι-δομημένων δεδομένων και αναγνωρίζουμε δυο βασικές κατευθύνσεις σχετικά με τις δουλειές αυτές. Μια κατεύθυνση είναι σχετικά με το πρόβλημα της ιδιωτικότητας σε δεδομένα με οντολογίες και συλλογιστικούς αλγορίθμους. Μια άλλη είναι σχετικά με διαδραστικά μοντέλα ανωνυμοποίησης ημι-δομημένων δεδομένων.

3.4.1 Ημι-δομημένα Δεδομένα, Οντολογίες και Αλγόριθμοι Συλλογιστική

Όταν χρησιμοποιούνται οντολογίες OWL για τη μοντελοποίηση δεδομένων υπάρχουν αρκετές τεχνικές δυσκολίες και το νέο πρόβλημα που ανακύπτει έχει αρκετό ερευνητικό ενδιαφέρον [15]. Οι οντολογίες OWL χρησιμοποιούνται εκτενώς σε κλινικά δεδομένα, όπου οντολογίες όπως η SNOMED CT είναι μέρος των νοσοκομειακών πληροφοριακών συστημάτων σε πολλές χώρες. Η διατήρηση της ιδιωτικότητας σε συστήματα βασισμένα σε οντολογίες είναι ιδιαίτερης σημασίας, ειδικά όταν υπάρχουν πολλοί διαφορετικοί χρήστες που έχουν πρόσβαση στο σύστημα και έχουν διαφορετικά προνόμια. Η μη έγκυρη διαρροή πληροφοριών από ιατρικά συστήματα μπορεί να αποβεί καταστροφική για κυβερνήσεις, νοσοκομεία αλλά κυρίως για τους ίδιους τους ασθενείς. Στις σχεσιακές βάσεις, ο έλεγχος της πρόσβασης στα δεδομένων γινόταν με διαφορετικές όψεις των δεδομένων για διαφορετικούς χρήστες με διαφορετικά προνόμια. Όμως, μια όψη χωρίς τα κρυφά γεγονότα δεν είναι αρκετή σε συστήματα με οντολογίες.

Στο [10] παρουσιάζουν ένα μοντέλο ιδιωτικότητας, το οποίο είναι ευαίσθητο σε επιθέσεις κατά της ιδιωτικότητας Βάσεων Γνώσης (KB), και παρουσιάζουν μια μέθοδο κατασκευής ασφαλών όψεων KB. Και εδώ αναγνωρίζεται το πρόβλημα ότι η απόκρυψη μόνο των μυστικών δεν είναι αρκετή για την προστασία της ιδιωτικότητας σε αυτού του είδους δεδομένα. Οπότε, προτείνουν ένα πιο ισχυρό μοντέλο ιδιωτικότητας το οποίο λαμβάνει υπόψιν του τόσο τα ίδια τα γεγονότα που πρέπει να μείνουν μυστικά, όσο και την μετα-πληροφορία που προκύπτει μέσω των οντολογιών. Η βασική ιδέα για την κατασκευή των ασφαλών όψεων KB είναι η προσέγγιση του γνωστικού υποβάθρου που χρησιμοποιεί ο εισβολέας ώστε να μειωθεί η πολυπλοκότητα κατασκευής των όψεων. Ο σκοπός του αλγορίθμου που παρουσιάζουν είναι όμοιος με το σκοπό του αλγορίθμου αυτής της διπλωματικής, όμως το μοντέλο που παρουσιάζεται σε αυτό το άρθρο είναι ένας μηχανισμός απαντήσεως ερωτημάτων. Ακόμη, το μοντέλο αυτό χρησιμοποιεί προσεγγίσεις του γνωστικού υποβάθρου του αντιπάλου. Μια άλλη διαφορά είναι ότι στις περισσότερες περιπτώσεις ο αλγόριθμος που προτείνουν έχει εκθετικό χρόνο.

Στο [23], οι συγγραφείς ερευνούν της ασφάλεια της ιδιωτικότητας για τους ιδιοκτήτες πληροφοριακών συστημάτων οι οποίοι θέλουν να μοιραστούν ένα μέρος των δεδομένων τους. Ορίζουν το πρόβλημα της ιδιωτικότητα ως μια νέα πτυχή των καθαρά λογικών και συλλογιστικών προβλημάτων και υιοθετούν το διαδραστικό μοντέλο ανωνυμίας ως λύση του προβλήματος. Στο [25], οι ίδιοι συγγραφείς ερευνούν το πρόβλημα της ιδιωτικότητας που προκύπτει κατά την επαναχρησιμοποίηση οντολογιών. Η OWL επιτρέπει την εισαγωγή μιας οντολογίας, K_h, στην οντολογία K_v. Ο reasoner, τότε απλώς συγχωνεύει τα αξιώματα των δυο οντολογιών. Όμως, μπορεί η οντολογία K_h να περιέχει πληροφορίες που είναι ευαίσθητες. Για να λυθεί το πρόβλημα αυτό, οι συγγραφείς προτείνουν η οντολογία K_h να είναι διαθέσιμη μέσω ενός μαντείου (oracle), το οποίο θα επιτρέπει την χρήση της K_h μέσω queries. Στη συνέχεια, ερευνούν τους αλγορίθμους που μπορούν να χρησιμοποιηθούν που μπορούν να χρησιμοποιηθούν για την εισαγωγή μέσω ερωτημάτων (import-by-query) και μπορούν να λύσουν το πρόβλημα του

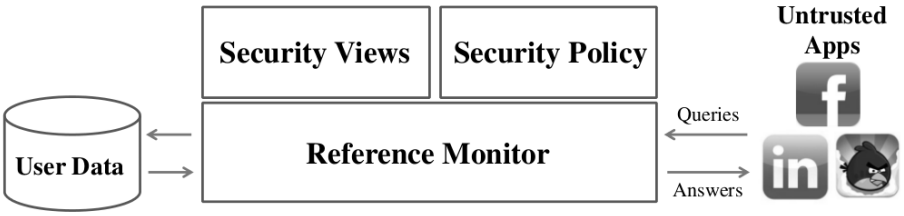
συλλογισμού χωρίς να παραβιάζουν την ιδιωτικότητα. Στο [26], οι συγγραφείς επεκτείνουν τη δουλειά τους. Δείχνουν ότι ορισμένοι περιορισμοί του μοντέλου της προηγούμενης δουλειάς τους είναι πολύ περιοριστικοί ώστε να είναι εφικτός ο συλλογισμός. Οπότε, προτείνουν μια προέκταση για να ξεπεράσουν τους περιορισμούς της εκφραστικότητας του προηγούμενου μοντέλου και προσδιορίζουν νέους αλγορίθμους συλλογισμού.

3.4.2 Διαδραστικά Μοντέλα Ανωνυμίας

Οι πιο αντιπροσωπευτικές εργασίες στον τομέα της ανωνυμοποίησης ημι-δομημένων δεδομένων, υιοθετούν το διαδραστικό μοντέλο ανωνυμίας. Σε αυτό το μοντέλο οι συγγραφείς υποθέτουν [23]:

- μια οντολογία σε OWL.
- ένα σύνολο γεγονότων που πρέπει να παραμείνουν κρυφά (αυτά παίζουν τον ρόλο των SIs).
- έναν αντίπαλο με γνωστικό υπόβαθρο που περιέχει μια σειρά από γεγονότα αλλά και ορισμένες αφηρημένες σχέσεις, περιορισμούς και εξαρτήσεις που ισχύουν για τα μοντελοποιημένα δεδομένα (μέσω μιας οντολογίας).

Έτσι, αν και η αποκάλυψη ενός απλού γεγονότος μπορεί να μην οδηγεί από μόνη της στην παραβίαση της ιδιωτικότητας, ωστόσο, μπορεί να αποκαλύψει ευαίσθητες πληροφορίες αν συνδυαστεί με άλλα γνωστά γεγονότα ή με τις αφηρημένες σχέσεις της οντολογίας που αποτελούν το γνωστικό υπόβαθρο του εισβολέα.



Σχήμα 3.1: Διαδραστικό Μοντέλο Ανωνυμίας

Στο [8], προτείνεται ένα διαδραστικό μοντέλο ανωνυμοποίησης για εφαρμογές. Η διαφορά του συστήματος αυτού είναι ότι προκύπτει από τα δεδομένα και έχει σημασιολογικό νόημα. Υπάρχουν διαφορετικές όψεις, ανάλογα με τα δικαιώματα ασφαλείας. Κάθε ερώτηση στο σύστημα αντιστοιχείται στις όψεις που χρειάζονται για να απαντηθεί και έτσι αποφασίζονται οι πολιτικές ασφαλείας. Το σύστημα αυτό φαίνεται στο σχήμα 3.1. Κάθε εφαρμογή θέτει τα ερωτήματά της στο σύστημα, το οποίο με βάση τις όψεις, τις πολιτικές ασφαλείας και τον Reference Manager αποφασίζει αν θα απαντηθεί το ερώτημα ή όχι.

Στο [7], οι συγγραφείς επεκτείνουν το προηγούμενο μοντέλο ανωνυμοποίησης, περιγράφοντας ένα μοντέλο όπου όλες οι αποφάσεις σχετικά με την ιδιωτικότητα εξηγούνται τυπικά. Αν ένα ερώτημα προς τα δεδομένα γίνεται αποδεκτό ή όχι, το σύστημα επιστρέφει μια σύντομη

αλλά τυπική εξήγηση η οποία επιτρέπει στον χρήστη να προσαρμόσει το ερώτημα του. Η προσέγγιση αυτή βασίζεται σε συσχετίσεις μεταξύ δικτυωμάτων δημοσιοποίησης (disclosure lattices) και άλγεβρας πολιτικών πρόσβασης. Το σύστημα που προτείνουν δέχεται ως είσοδο ερωτήματα και προβολές ασφαλείας σε SQL και στη συνέχεια δημιουργούνται τύποι πολιτικής προσβάσεως και εξηγήσεις με χρήση του αλγορίθμου που προτείνουν.

Στο [5], προτείνεται ένα πλαίσιο για συλλογισμό (reasoning) και παράλληλα διατήρηση της ιδιωτικότητας. Αυτό το σύστημα μπορεί να απαντάει σε ερωτήματα για την γνώση που διαθέτει αλλά και για αυτή τη γνώση που προκύπτει μέσω συλλογισμού χωρίς να αποκαλύπτει τη κρυμμένη γνώση. Η βασική ιδέα είναι ότι ο reasoner για τη μερικώς κρυμμένη KB απαντάει σε ερωτήματα χωρίς να αποκαλύπτει εμμέσως κρυμμένη γνώση. Η κρυφή γνώση παραμένει κρυμμένη αλλά μπορεί να χρησιμοποιηθεί σε συλλογισμό αν δεν παραβιάζεται η ιδιωτικότητα. Αυτός ο reasoner υλοποιείται σαν ένας δεύτερος reasoner ο οποίος ελέγχει τα αποτελέσματα των ήδη υπάρχοντών reasoner.

Ένα διαφορετικό διαδραστικό μοντέλο ανωνυμοποίησης προτείνεται στο [44]. Το μοντέλο αυτό λύνει το πρόβλημα της απάντησης ερωτημάτων από μια KB σε EL με μυστικά χωρίς να τίθενται σε κίνδυνο τα μυστικά. Το μοντέλο αυτό περιλαμβάνει έναν αλγόριθμο πολυωνυμικού χρόνου που απαντάει σε ερωτήματα είτε με *Ναι* είτε με *Άγνωστο*. Ο αλγόριθμος αυτός, επιπρόσθετα από το προηγούμενο μοντέλο, λαμβάνει υπόψιν του τις προηγούμενες απαντήσεις που έχει δώσει. Ακόμη, η προσέγγιση αυτή βασίζεται σε σημασιολογία ανοικτού κόσμου, όπου δεν είναι εφικτό να καταλάβει κανείς αν ο reasoner απαντάει *Άγνωστο* επειδή δεν γνωρίζει την απάντηση ή επειδή προστατεύει κάποιο μυστικό.

Το μοντέλο που παρουσιάζεται στο [24] υποθέτει ότι η οντολογία είναι γνωστή σε όλους τους χρήστες, δηλαδή και στον επιτιθέμενο (το χειρότερο σενάριο για την διασφάλιση της ιδιωτικότητας), ενώ τα δεδομένα θεωρούνται άγνωστα. Αλληλεπίδραση με το σύστημα γίνεται μόνο μέσω ενός διαδραστικού μοντέλου που απαντάει σε συζευκτικά ερωτήματα (conjunctive queries). Το μοντέλο που υιοθετείται είναι μια επέκταση του βασικού σεναρίου Ελεγχόμενων Ερωτημάτων (Controlled Query Evaluation - CQE).

3.5 Μηχανισμοί Ελέγχου Πρόσβασης

Μια άλλη ερευνητική κατεύθυνση σχετικά με την ιδιωτικότητα είναι οι μηχανισμοί ελέγχου πρόσβασης. Έλεγχος πρόσβασης είναι η διαδικασία ελέγχου κάθε αιτήματος για πρόσβαση στα δεδομένα από το σύστημα και ο καθορισμός εάν το αίτημα γίνεται αποδεκτό ή όχι. Στο [11] προτείνεται ένας μηχανισμός ασφαλείας ο οποίος εγγυάται την εμπιστευτικότητα μέσω υποχρεωτικού ελέγχου πρόσβασης για όλους τους χρήστες σε συνδυασμό με έναν μηχανισμό εξαγωγής συμπερασμάτων (Disclosure Inference Engine). Αυτός ο μηχανισμός παράγει όλες τις πληροφορίες που μπορούν να παραχωρηθούν σε έναν χρήστη σύμφωνα με τα προηγούμενα ερωτήματα του χρήστη, καθώς και με τους περιορισμούς της βάσης δεδομένων.

Στο [30] παρουσιάζουν ένα μοντέλο της XACML με χρήση Περιγραφικής Λογικής που υποστηρίζει λογική ανάλυση. Η XACML είναι η επικρατέστερη γλώσσα ελέγχου πρόσβασης στο Διαδίκτυο. Η χρήση της XACML σε συνδυασμό με Περιγραφική Λογική επιτρέπει τη

χρήση reasoners οι οποίοι μπορούν να αναλύσουν πολιτικές πρόσβασης και να τις επαληθεύσουν. Στα [19] [18], ερευνούνε τη χρήση της OWL για τη περιγραφή των δικαιωμάτων και των πολιτικών πρόσβασης. Με αυτό τον τρόπο προσπαθούν να συνδυάσουν τις δυο διαφορετικές προσεγγίσεις στον έλεγχο πρόσβασης. Η μια είναι η δημιουργία νέων ρεαλιστικών μοντέλων ελέγχου πρόσβασης και η άλλη είναι η δημιουργία γλωσσών για τις πολιτικές πρόσβασης.

3.6 Λειτουργικές Εξαρτήσεις

Λειτουργική εξάρτηση είναι ένας περιορισμός που ισχύει μεταξύ συνόλων ιδιοτήτων σε μια σχέση μιας βάσης δεδομένων. Περιορισμοί στα δεδομένα ή κρυμμένες συσχετίσεις μεταξύ ιδιοτήτων μπορούν να χρησιμοποιηθούν από τον επιτιθέμενο χρήστη για να ανακαλύψουν κρυμμένες πληροφορίες. Ένα τέτοιο είδος περιορισμών στα δεδομένα είναι και οι λειτουργικές εξαρτήσεις. Οι λειτουργικές εξαρτήσεις έχουν μεγάλη σημασία στην ανωνυμοποίηση δεδομένων, όπως φαίνεται και στο [28]. Σε αυτή τη δουλειά, παρουσιάζουν ένα μηχανισμό ελέγχου πρόσβασης όταν τα δεδομένα παρουσιάζουν λειτουργικές εξαρτήσεις. Συγκεκριμένα, ερευνούν το πρόβλημα της παράνομης εξαγωγής συμπερασμάτων ως αποτέλεσμα του συνδυασμού σημασιολογικών περιορισμών των δεδομένων με πληροφορίες που έχουν γνώση. Έτσι, φαίνεται ότι αυτού του είδους ο συμπερασμός μπορεί να οδηγήσει σε παραβιάσεις των πολιτικών πρόσβασης των δεδομένων. Τέλος, χρησιμοποιούν έναν αλγόριθμο που βασίζεται σε γράφους για να αποφευχθούν αυτές οι παραβιάσεις.

Ένα άλλο είδος λειτουργικών εξαρτήσεων είναι οι υπό υποθέσεις λειτουργικές εξαρτήσεις (conditional functional dependencies) που παρουσιάζονται στο [9] στα πλαίσια του καθαρισμού δεδομένων. Οι υπό προϋποθέσεις λειτουργικές εξαρτήσεις είναι ικανές να συλλάβουν την έννοια των σωστών δεδομένων σε περιπτώσεις όπου οι απλές λειτουργικές εξαρτήσεις είναι ανεπαρκείς, δηλαδή όταν οι ίδιες οι εξαρτήσεις βασίζονται σε κάποια άλλη προϋπόθεση για να ισχύουν. Σε αυτή τη δουλειά, επεκτείνουν προηγούμενες δουλειές που βασίζονταν σε απλές λειτουργικές εξαρτήσεις, και παρουσιάζουν ένα πλήρες σύστημα συλλογισμού για υπό προϋποθέσεις λειτουργικές εξαρτήσεις.

Οι μετρικές λειτουργικές εξαρτήσεις παρουσιάζονται στο [32]. Ο συνδυασμός δεδομένων από διαφορετικές πηγές, οδηγεί συχνά σε περιπτώσεις με μικρές αποκλίσεις στα δεδομένα παραβιάζοντας τις λειτουργικές απαιτήσεις χωρίς, όμως, να υπάρχει παραβίαση της σημασιολογίας. Μερικά παραδείγματα είναι οι διαφορετικές μορφές γραφής διευθύνσεων ή ημερομηνιών. Σε αυτή τη δουλειά, οι συγγραφείς ορίζουν της μετρικές λειτουργικές εξαρτήσεις, οι οποίες γενικεύουν τις λειτουργικές απαιτήσεις επιτρέποντας μικρές διαφορές στις τιμές των στοιχείου που ισχύουν οι λειτουργικές απαιτήσεις. Οι διαφορές, ελέγχονται από μια μετρική συνάρτηση. Στο άρθρο αυτό παρουσιάζονται αλγόριθμοι για τον αποδοτικό προσδιορισμό παραβιάσεων των μετρικών λειτουργιών και ελέγχονται πειραματικά σε πραγματικά δεδομένα.

Τέλος, στο [22], παρουσιάζεται ένα διαφορετικό είδος εξαρτήσεων, αυτό των διαδοχικών εξαρτήσεων οι οποίες εκφράζουν τη σημασιολογία δεδομένων με διατεταγμένα πεδία. Σε αυτή την περίπτωση βοηθούνε στον προσδιορισμό ποιοτικών προβλημάτων με τα δεδομένα.

3.7 Προέλευση Δεδομένων

Η προέλευση των δεδομένων (provenance) στις βάσεις δεδομένων αναφέρεται στην καταγωγή και την ιστορία των δεδομένων κατά τον κύκλο ζωής τους [14]. Η προέλευση των δεδομένων προσθέτει αξία στα δεδομένα εξηγώντας πως αυτά προέκυψαν. Τα δεδομένα διαρκώς αντιγράφονται, μεταφέρονται και συνδυάζονται, οπότε πληροφορίες για την προέλευση τους βοηθούν τους τελικούς χρήστες να κρίνουν αν τα αποτελέσματα ερωτημάτων είναι αξιόλογα.

Υπάρχουν πολλοί διαφορετικοί ορισμοί της προέλευσης των δεδομένων, ιδιαίτερο ενδιαφέρον για εμάς έχει ένας από τους πρώτους φορμαλιστικούς ορισμούς που ονομάζεται γενεαλογία [16]. Σε αυτή τη δουλειά σχετίζουν κάθε πλειάδα στα αποτελέσματα ενός ερωτήματος με ένα σύνολο πλειάδων που υπάρχουν στην είσοδο. Αυτή τη συσχέτιση την ονομάζουν γενεαλογία (lineage). Διαισθητικά, γενεαλογία μιας πλειάδας t είναι όλα τα δεδομένα εισόδου που συνέβαλαν στην πλειάδα t ή που βοήθησαν στη παραγωγή της.

Επειδή, ο προηγούμενος ορισμός δεν ήταν αρκετά ακριβής, στο [12] ορίσαν τη γιατί-προέλευση (why-provenance, η οποία βασίζεται στην ιδέα του να παρέχονται πληροφορίες σχετικά με τους μάρτυρες ενός ερωτήματος. Μάρτυρας είναι ένα υποσύνολο των εγγραφών της βάσης δεδομένων το οποίο είναι αρκετό για να διασφαλίσει ότι η ζητούμενη εγγραφή θα βρίσκεται στο αποτέλεσμα του ερωτήματος. Στο [27] όρισαν την πως-προέλευση (how-provenance), η οποία περιέχει πληροφορίες για το πως μια πλειάδα στην έξοδο ενός ερωτήματος προήλθε από την είσοδο, πληροφορίες που δεν περιέχονται στην γιατί-προέλευση. Στο [12] ορίστηκε και η που-προέλευση (where-provenance), η οποία περιγράφει από που έχει αντιγραφεί κάθε πληροφορία.

Η έννοια της προέλευσης των δεδομένων σχετίζεται με τον γράφο εξαρτήσεων που παρουσιάζουμε στη συνέχεια αυτής της διπλωματικής. Όπως, ο γράφος εξαρτήσεων δείχνει τις εξαρτήσεις που προκύπτουν για την παραγωγή νέων αξιωμάτων μέσω κανόνων της οντολογίας, έτσι, ιδιαίτερα στην γενεαλογία, σχετίζουν την προέλευση των πλειάδων που παράγονται μέσω ερωτημάτων με πλειάδες από την είσοδο. Διαισθητικά, και στις δυο δουλειές υπάρχει η έννοια της καταγραφής της προέλευσης δεδομένων και πως παράγονται νέα δεδομένα από τα αρχικά δεδομένα.

Κεφάλαιο 4

Ορισμός Προβλήματος

Στο κεφάλαιο αυτό παρουσιάζουμε το πρόβλημα με το οποίο ασχοληθήκαμε σε αυτή τη διπλωματική. Αρχικά, δίνουμε τη χρησιμότητα της επίλυσης αυτού του προβλήματος ανωνυμοποίησης και δίνουμε το κίνητρο μας για ενασχόληση με αυτό το πρόβλημα. Στη συνέχεια, ορίζουμε το πρόβλημα, τη μοντελοποίηση και τις θεωρήσεις που κάναμε για το μοντέλο επίθεσης και δίνουμε ορισμένα παραδείγματα χρησιμότητας.

4.1 Το Κίνητρο

Οντολογίες, όπως η SNOMED CT, αποτελούν ζωτικό μέρος των πληροφοριακών συστημάτων υγείας σε πολλές χώρες. Αυτές οι οντολογίες αποτελούνται από αξιώματα, τα οποία μοντελοποιούν τη γνώση σχετικά με ιατρικές ασθένειες, συμπτώματα, εξετάσεις κτλ. Όταν ιατρικά δεδομένα, δηλαδή οι ηλεκτρονικοί φάκελοι των ασθενών (Electronic Health Records - EHR), μοντελοποιούνται με τέτοιες οντολογίες, τότε κάποιος μπορεί να εξάγει νέες πληροφορίες που δεν υπάρχουν στα αρχικά δεδομένα, δηλαδή πληροφορίες που υπονοούνται, μέσω λογικού συμπερασμού.

Οι παραδοσιακοί μηχανισμοί πρόσβασης των βάσεων δεδομένων και τα ήδη υπάρχοντα μοντέλα ανωνυμοποίησης δεν είναι αρκετά για πληροφοριακά συστήματα που χρησιμοποιούν οντολογίες. Ο απλός περιορισμός της πρόσβασης σε ένα σύνολο από ευαίσθητα δεδομένα που έχουν οριστεί από ειδικούς δεν είναι αρκετό. Ο αντίπαλος μπορεί να χρησιμοποιήσει ένα μέρος της βάσης δεδομένων που γνωρίζει ήδη και στο οποίο έχει πρόσβαση μαζί με την οντολογία ώστε να συμπεράνει νέες πληροφορίες που πιθανώς να είναι ευαίσθητες. Με άλλα λόγια, με χρήση της δημοσιευμένης οντολογίας, ένας αντίπαλος μπορεί να αποκτήσει πρόσβαση στη πληροφορία που θέλουμε να μείνει κρυφή. Το ίδιο ισχύει και με τα ήδη υπάρχοντα μοντέλα ανωνυμοποίησης τα οποία δεν λαμβάνουν υπόψιν τους τις οντολογίες ως γνωστικό υπόβαθρο ενός επιτιθέμενου και επομένως δεν υπολογίζουν ότι αυτές μπορούν να χρησιμοποιηθούν για λογικό συμπερασμό νέας γνώσης. Το πρόβλημα αυτό γίνεται εύκολα κατανοητό μέσα από το ακόλουθο παράδειγμα (σχήμα 2.3):

Έστω ότι ο αντίπαλος γνωρίζει ότι ο Γιάννης έχει εμβολιαστεί το 1994 και ότι έχει μολυνθεί με τον ιό Α. Αυτή η πληροφορία δεν είναι ευαίσθητη, δηλαδή κάθε αντίπαλος μπορεί

να έχει πρόσβαση σε αυτό το μέρος της βάσης δεδομένων. Με χρήση της οντολογίας, ο αντίπαλος γνωρίζει ότι όσοι έλαβαν κάποιο εμβόλιο το 1994, τότε έλαβαν εμβόλιο τύπου *va*. Τα εμβόλια *va*, είναι τύπου X, σύμφωνα με την οντολογία, άρα και ο αντίπαλος συμπεραίνει ότι ο Γιάννης έχει εμβολιαστεί με εμβόλιο τύπου X. Ακόμη, από την οντολογία ο αντίπαλος γνωρίζει ότι όσοι έχουν λάβει εμβόλιο τύπου X, είναι σαν να μην έχουν εμβολιαστεί. Άρα πάλι συμπεραίνει ότι ο Γιάννης δεν έχει εμβολιαστεί. Τέλος, η οντολογία ορίζει ότι όσοι δεν έχουν εμβολιαστεί και έχουν μολυνθεί από τον ιό, είναι άρρωστοι. Ο αντίπαλος συμπεραίνει ότι ο Γιάννης είναι άρρωστος. Αυτή η πληροφορία είναι ευαίσθητη και δεν θα έπρεπε να έχει ο αντίπαλος πρόσβαση σε αυτή.

4.2 Ορισμός του Προβλήματος

Στη συνέχεια εξηγούμε το πρόβλημα και δίνουμε ένα φορμαλιστικό ορισμό. Για να γίνει πιο εύκολα κατανοητό το πρόβλημα, η μοντελοποίηση του που θεωρούμε και η χρησιμότητα του, δίνουμε ορισμένα παραδείγματα.

4.2.1 Η Ιδέα του Προβλήματος

Ότι πληροφορίες έχουμε στη βάση δεδομένων μας το καλούμε *γεγονός*(fact). Κάποιος ειδικός επιλέγει ποια από τα γεγονότα που υπάρχουν στη βάση δεδομένων είναι ευαίσθητα δεδομένα και πρέπει να παραμείνουν κρυφά. Αυτά ονομάζονται *μυστικά* γεγονότα. Θεωρούμε ότι πρόσβαση (άρα και δημοσίευση) γίνεται ανά γεγονός. Σε ορολογία βάσεων δεδομένων, κάθε γεγονός αντιστοιχεί σε ένα κελί ενός σχεσιακού πίνακα μιας βάσης δεδομένων, δηλαδή σε ένα στοιχείο που αντιστοιχεί σε κάποιο συγκεκριμένο άτομο.

Δεδομένου:

- μιας βάσης δεδομένων από γεγονότα,
- ενός υποσυνόλου από αυτά τα γεγονότα που πρέπει να παραμείνουν μυστικά
- και μια δημόσιας οντολογίας που περιέχει ένα σύνολο από αξιώματα που περιγράφουν τις σχέσεις μεταξύ των γεγονότων

έχουμε το πρόβλημα του να βρούμε το ελάχιστο σύνολο γεγονότων της βάσης δεδομένων που πρέπει να παραμείνουν κρυμμένα ώστε ένας επιτιθέμενος με πρόσβαση στο δημόσιο μέρος της βάσης δεδομένων (και φυσικά στην οντολογία η οποία είναι δημόσια διαθέσιμη) να μην μπορεί να συμπεράνει λογικά κάποιο από τα μυστικά γεγονότα.

Η οντολογία του προηγούμενου παραδείγματος, μεταξύ άλλων, περιέχει τα ακόλουθα αξιώματα:

- (E_1) Όσοι έχουν μολυνθεί από τον ιό A, και δεν έχουν εμβολιαστεί είναι άρρωστοι.
- (E_2) Όσοι έλαβαν εμβόλιο τύπου X, θεωρούνται ότι δεν έχουν λάβει κάποιο εμβόλιο.
- (E_3) Το εμβόλιο *va* είναι τύπου X.

- (E_6) Όσοι έλαβαν εμβόλια το 1994, εμβολιάστηκαν με το *να* εμβόλιο.

Ακόμη, η βάση δεδομένων του προηγούμενου παραδείγματος περιέχει τα ακόλουθα γεγονότα:

- (E_4) Ο Γιάννης εμβολιάστηκε το 1994.
- (E_5) Ο Γιάννης έχει μολυνθεί από τον ιό Α.
- (I_1) Όσοι έλαβαν εμβόλια το 1994, εμβολιάστηκαν με εμβόλιο τύπου Χ (υπονοούμενη γνώση).
- (I_2) Ο Γιάννης έλαβε εμβόλιο τύπου Χ (υπονοούμενη γνώση).
- (I_3) Ο Γιάννης δεν έχει εμβολιαστεί (υπονοούμενη γνώση).
- (I_4) Ο Γιάννης είναι άρρωστος (υποννούμενη γνώση).

Το γεγονός I_4 αποτελεί το μυστικό στην περίπτωση μας.

Σε αυτή την περίπτωση χρειάζεται ένα αυτοματοποιημένο σύστημα, το οποίο να μπορεί να χρησιμοποιεί λογικό συμπερασμό και την προηγούμενη πληροφορία που διαθέτει και να βρίσκει το σύνολο των ελάχιστων γεγονότων που πρέπει να κρυφτούν ώστε να προβλεφθεί η αποκάλυψη ευαίσθητης πληροφορίας.

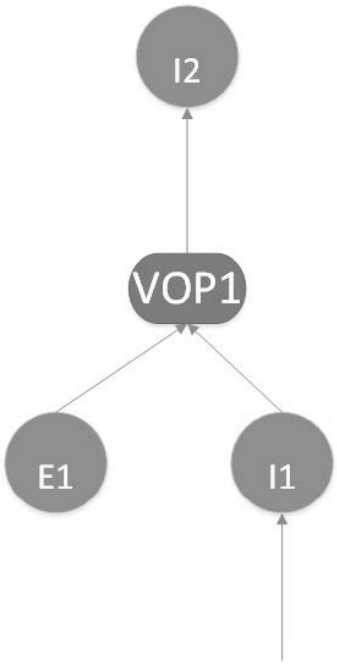
Δεν υπάρχει κάποια φορμαλιστική μέθοδος για τη μοντελοποίηση του προηγούμενου προβλήματος.

4.2.2 Φορμαλιστικός Ορισμός

Για τη μοντελοποίηση του προβλήματος χρησιμοποιούμε μια γραφοθεωρητική προσέγγιση. Θεωρούμε δυο διαφορετικά είδη γεγονότων. Τα *ρητά γεγονότα* (explicit facts) (E_i), τα οποία είναι αυτά που υπάρχουν αποθηκευμένα στη βάση δεδομένων, και τα *υπονοούμενα γεγονότα* (implicit facts) (I_i), τα οποία προκύπτουν με λογικό συμπερασμό (reasoning) με χρήση της οντολογίας και των ρητών ή και άλλων υπονοούμενων γεγονότων.

Θεωρούμε έναν γράφο $G = (V = V_i \cup V_{op}, E)$, όπου:
V_i είναι το σύνολο των κόμβων που καθ'ένας αντιστοιχεί σε ένα E_i ή σε ένα I_i,
V_{op} είναι το σύνολο των κόμβων που αντιστοιχούν σε κάθε λειτουργικό κόμβο, όπως εξηγούνται στη συνέχεια,
E είναι το σύνολο των ακμών που συνδέει κόμβους V_i με V_{op}.

Διαισθητικά, το σύνολο των ακμών δείχνουν τις εξαρτήσεις που υπάρχουν μεταξύ κόμβων. V_{op} είναι το σύνολο των *λειτουργικών κόμβων*, δηλαδή κάθε τέτοιος κόμβος δείχνει ποια ήδη γνωστά γεγονότα χρησιμοποιηθήκαν για να παραχθεί ένα νέο υπονοούμενο γεγονός με λογικό συμπερασμό. Για παράδειγμα, αν το υπονοούμενο γεγονός I_2 προκύπτει από τα γεγονότα E_1 και I_1 λόγω κάποιου αξιώματος της οντολογίας, τότε ορίζουμε τον κόμβο V_{op1} που έχει δυο εισερχόμενες ακμές από του κόμβους E_1 και I_1 και έναν εξερχόμενο κόμβο, τον I_2. Αυτό το απλό παράδειγμα φαίνεται στο σχήμα 4.1.



Σχήμα 4.1: Αναπαράσταση δημιουργίας του γεγονότος I_2 μέσω του λειτουργικού κόμβου V_{op1}

Γενικά, οι V_{op} κόμβοι μπορούν να αναφέρονται σε κανόνες πρωτοβάθμιας λογικής (first-order logic rules), οπότε μπορούν να μοντελοποιούν περιορισμούς ακεραιότητας, πράξεις επιλογής ή προβολής, και ακόμα σύνθετες λειτουργικές εξαρτήσεις [9], μετρικές λειτουργικές εξαρτήσεις [32], ακολουθιακές εξαρτήσεις [22]. Οι εισερχόμενες ακμές κάθε V_{op} κόμβου είναι $n \geq 1$ ενώ υπάρχει πάντα 1 μόνο εξερχόμενη ακμή.

Διαισθητικά, ο γράφος αυτός απεικονίζει όλες τις εξαρτήσεις μεταξύ των ρητών και των υπονοούμενων γεγονότων, και δημιουργείται κατά τον λογικό συμπερασμό(reasoning). Σε αυτή τη δουλειά υποθέτουμε ότι ο λογικός συμπερασμός είναι ντετερμινιστικός, δηλαδή, δεν υπάρχουν διαζεύξεις (είτε το ένα γεγονός είτε το άλλο) στην έξοδο κάθε V_{op} κόμβου. Ακόμη, υποθέτουμε ότι η διαδικασία του λογικού συμπερασμού είναι αναδρομική, δηλαδή ένα υπονοούμενο γεγονός μπορεί να χρησιμοποιηθεί για δημιουργία ενός άλλου υπονοούμενου γεγονότος (όπως το γεγονός I_1 στο σχήμα).

Δεδομένου ενός γράφου εξαρτήσεων G, έστω ότι S_c το σύνολο των κόμβων στο G που πρέπει να παραμείνουν κρυφοί. Έστω ακόμη, $f(N, G)$ μια συνάρτηση που παίρνει ως είσοδο ένα σύνολο κόμβων $N \subseteq E_i \subseteq V_i$ μαζί με τον γράφο G και παράγει ως έξοδο ένα νούμερο. Αυτό το νούμερο αντιστοιχεί στο κόστος αφαίρεσης των κόμβων του N από τον γράφο G. Η συνάρτηση f μπορεί να οριστεί από έναν ειδικό. Το πρόβλημα είναι να οριστεί το σύνολο N, έτσι ώστε:

- η αφαίρεση τους από τον γράφο G να εμποδίζει τον επιτιθέμενο από το να συμπεραίνει οποιοδήποτε από τα μυστικά που ανήκουν στο σύνολο S_c,
- η τιμή $f(N)$ να είναι η ελάχιστη μεταξύ των τιμών όλων των πιθανών συνόλων που ικανοποιούν την προηγούμενη συνθήκη.

Αξίζει να σημειώσουμε ότι το σύνολο των κόμβων N αποτελείται μόνο από κόμβους ρητών γεγονότων, όμως, η αφαίρεση τους οδηγεί και στην αφαίρεση υπονοούμενων κόμβων που εξαρτώνται από αυτούς. Η αφαίρεση ενός υπονοούμενου κόμβου I_i δεν έχει κανένα νόημα χωρίς την αφαίρεση ρητών κόμβων, γιατί ο επιτιθέμενος μπορεί να ξανά αναπαράγει αυτό το γεγονός με συνδυασμό των ρητών κόμβων και της οντολογίας.

Τέλος, πρέπει να επισημάνουμε ότι γεγονότα μεταξύ διαφορετικών οντοτήτων της βάσης δεδομένων, π.χ. ασθενών, μπορεί να αλληλοσχετίζονται. Δηλαδή, να είναι είσοδος σε παρόμοιους V_{op} κόμβους. Για παράδειγμα, έστω ότι στη βάση δεδομένων έχουμε εξής τα γεγονότα:

- Ο Γιάννης βρίσκεται στο Τμήμα Α.
- Ο Γιάννης είναι σε καραντίνα.
- Η Μαρία βρίσκεται στο Τμήμα Α.

Τότε κάποιος μπορεί εύκολα να συμπεράνει ότι η Μαρία είναι σε καραντίνα παρ'ότι δεν υπάρχει κάποιος ρητός κόμβος που να συνδέει τους δυο ασθενείς. *Η σημασία αυτού του παραδείγματος είναι ότι ο συμπερασμός για κάθε ασθενή ξεχωριστά δεν είναι αρκετός.*

4.2.3 Παραδείγματα

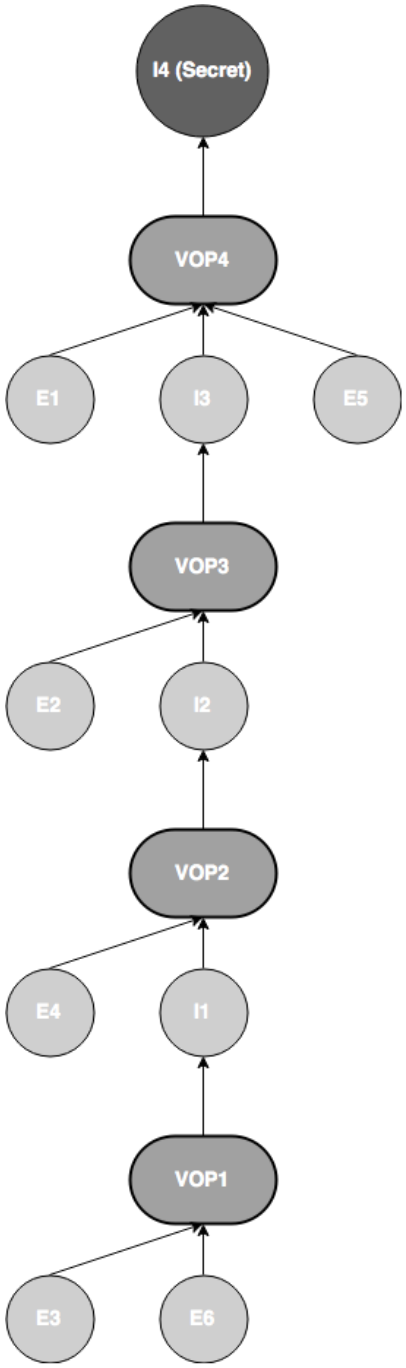
Στη συνέχεια δίνουμε τρία παραδείγματα μαζί με τους αντίστοιχους γράφους και τον υπολογισμό του κόστους της βέλτιστης λύσης. Στο τρίτο παράδειγμα φαίνεται η πολυπλοκότητα του προβλήματος.

Πρώτο Παράδειγμα

Σε αυτό το παράδειγμα, βλέπουμε το γράφο εξαρτήσεων του παραδείγματος που περιγράψαμε στην προηγούμενη ενότητα 4.2.1, και φαίνεται στο σχήμα 4.2. Θέλουμε να κρύψουμε τον κόμβο I_4, το οποίο είναι το υπονοούμενο γεγονός ότι ο Γιάννης είναι άρρωστος. Τα E_i γεγονότα, αντιστοιχούν στα 6 βασικά γεγονότα που προκύπτουν είτε από τη βάση δεδομένων (χωρίς τα υπονοούμενα γεγονότα) είτε από την οντολογία. Τα υπονοούμενα γεγονότα (I_i) είναι τα ακόλουθα. I_1 είναι το γεγονός ότι όσοι έμβολιάστηκαν το 1994, έλαβαν εμβόλιο τύπου Χ. I_2 είναι το γεγονός ότι ο Γιάννης έλαβε εμβόλιο τύπου Χ. I_3 είναι το γεγονός ότι ο Γιάννης δεν έχει λάβει εμβόλιο. Θεωρώντας ως κόστος το συνολικό αριθμό κόμβων που θα χαθούν αν αφαιρέσουμε έναν ρητό κόμβο, είναι εύκολο να δούμε ότι για να ελαχιστοποιήσουμε το κόστος, πρέπει να κρύψουμε είτε το E_1 είτε το E_5, με κόστος 1, τον ίδιο κόμβο μόνο.

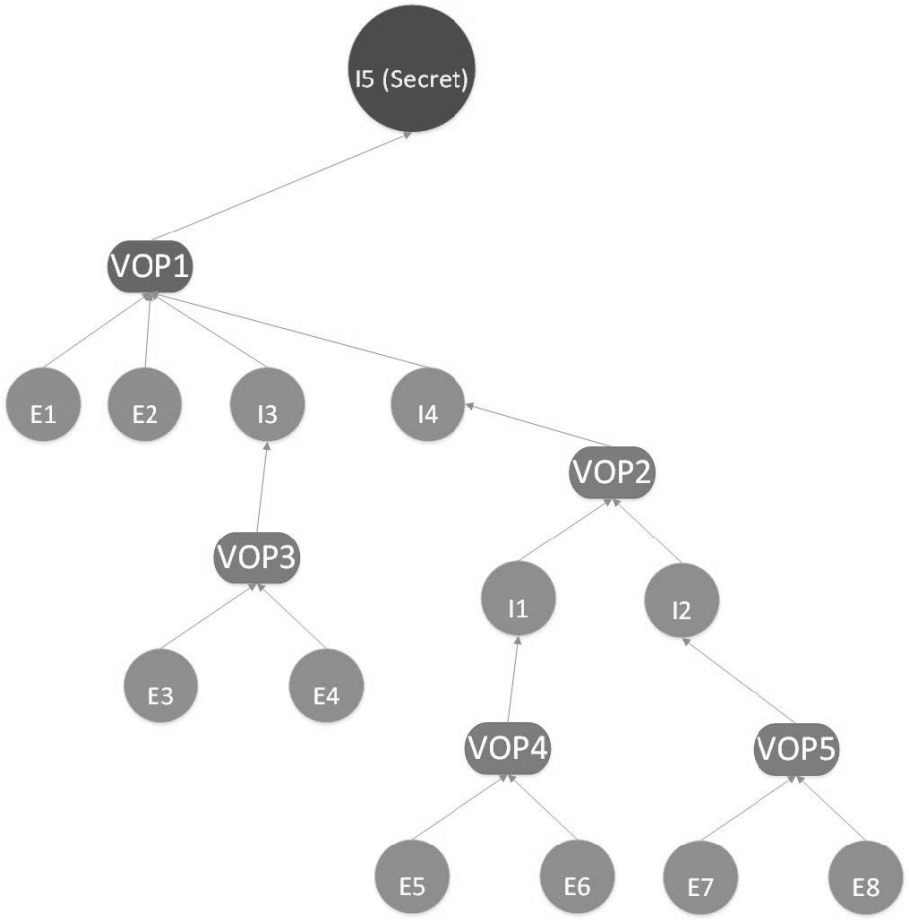
Δεύτερο Παράδειγμα

Σε αυτό το παράδειγμα, έχουμε το γράφο εξαρτήσεων που φαίνεται στο σχήμα 4.3. Θέλουμε να κρύψουμε τον κόμβο I_5 με το μικρότερο δυνατό κόστος. Έστω ότι για συνάρτηση κόστους θεωρούμε το συνολικό αριθμό κόμβων που θα χαθούν αν αφαιρέσουμε έναν ρητό κόμβο. Είναι εύκολα κατανοητό ότι σε αυτό το παράδειγμα το ελάχιστο κόστος το έχουν οι κόμβοι E_1 και E_2, όπου το κόστος της αφαίρεσης τους είναι 1. Αν, για παράδειγμα, κρύβαμε



Σχήμα 4.2: Γράφος εξαρτήσεων του πρώτου παραδείγματος

τον κόμβο E_3, τότε το κόστος θα ήταν 2, γιατί εκτός από αυτόν τον κόμβο, θα κρύβαμε έμμεσα και τον υπονοούμενο κόμβο I_3, μιας και δεν θα μπορούσε να προκύψει κάπως διαφορετικά με συμπερασμό.



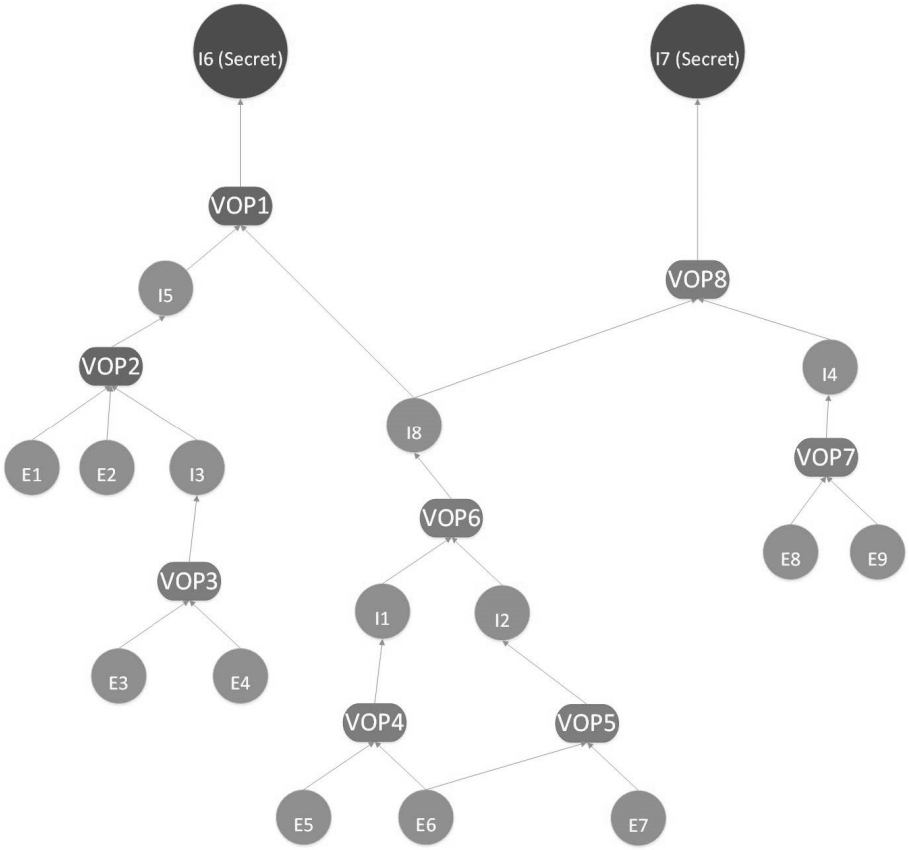
Σχήμα 4.3: Γράφος εξαρτήσεων του δεύτερου παραδείγματος

Τρίτο Παράδειγμα

Σε αυτό το παράδειγμα, έχουμε το γράφο εξαρτήσεων που φαίνεται στο σχήμα 4.4. Τώρα έχουμε πολλαπλά μυστικά και ο υπολογισμός του ελάχιστο κόστους είναι πιο δύσκολος. Θεωρούμε και πάλι την ίδια συνάρτηση κόστους με πριν. Αν είχαμε μόνο το μυστικό I_6, τότε ο κόμβος με το ελάχιστο κόστος για να κρύψουμε θα ήταν είτε ο κόμβος E_1 είτε ο E_2. Αν είχαμε μόνο το μυστικό I_7, τότε ο κόμβος με το ελάχιστο κόστος για να κρύψουμε θα ήταν είτε ο κόμβος E_8 είτε ο E_9. Έτσι, για να κρατήσουμε κρυφά και τα δυο μυστικά, θα μπορούσαμε να κρύψουμε τα E_1, E_8 με συνολικό κόστος $f(\{E_1, E_8\}, G) = |\{E_1, I_5, E_8, I_4\}| = 4$. Όμως, το ελάχιστο κόστος το πετυχαίνουμε αφαιρώντας τον κόμβο E_7, ο οποίος έχει κόστος $f(\{E_7\}, G) = |\{E_7, I_2, I_8\}| = 3$.

4.3 Πολυπλοκότητα του Προβλήματος

Το πρόβλημα που ορίσαμε είναι NP-δύσκολο. Μπορούμε να το δείξουμε αυτό με μια αναγωγή στο πρόβλημα weighted set covering (WSC). Στο WSC μας δίνεται ένα σύνολο U και μια συλλογή S από υποσύνολα του S, κάθε ένα από τα οποία έχει και ένα βάρος



Σχήμα 4.4: Γράφος εξαρτήσεων του τρίτου παραδείγματος

$w_i \geq 0$. Κάλυμμα είναι μια υποσυλλογή $C \subseteq S$ από σύνολα των οποίων η ένωση είναι το U. Το πρόβλημα έγκειται στο να βρεθεί ένα κάλυμμα το οποίο να ελαχιστοποιεί την ποσότητα $\sum_{i=1,\ldots,|C|} W_i$.

Έστω $S_i \in S_c$ ένας μυστικός κόμβος στον γράφο G. Το σύνολο που περιέχει όλους τους κόμβους V_{op} που παράγουν το μυστικό (το έχουν ως έξοδο) συμβολίζεται ως s_i. Κάθε V_{op} κόμβος έχει k εισόδους ($k \geq 1$). Το σύνολο των βασικών αξιωμάτων (E_i) από τα οποία εξαρτώνται οι k είσοδοι συμβολίζονται P_i. Οπότε, κάθε V_{op} σχετίζεται με ένα σύνολο P_i από ρητά γεγονότα. Στην περίπτωσή μας, το σύνολο U είναι η ένωση των V_{op} κόμβων που παράγουν με όλους τους διαφορετικούς τρόπους τα μυστικά που υπάρχουν στο S_c. Το σύνολο S είναι η ένωση όλων των P_i συνόλων για όλους τους V_{op} κόμβους του U. Στόχος είναι να βρεθεί το κάλυμμα $C \subseteq S$, δηλαδή το υποσύνολο όλων των κόμβων που ανήκουν στο S που καλύπτουν όλους τους V_{op} κόμβους στο U, ώστε να ελαχιστοποιείται η συνάρτηση $f(C, G)$. Η συνάρτηση f είναι η συνάρτηση κόστους. Η πολυπλοκότητα του προβλήματος είναι $|U|^{|S|}$.

Η ιδέα πίσω από την αναγωγή είναι η ακόλουθη. Όταν αφαιρούμε έναν κόμβο από το S, σπάμε το μονοπάτι προς τον V_{op} κόμβο με τον οποίο συνδέεται. Με άλλα λόγια, το μυστικό που προκύπτει μέσω αυτού του V_{op} κόμβου δεν μπορεί να προκύψει μέσω λογικού συμπερασμού από αυτό το μονοπάτι. Στόχος είναι να σπάσουμε όλα τα πιθανά μονοπάτια για όλα τα μυστικά που υπάρχουν στο σύνολο S_c.

Κεφάλαιο 5

Περιγραφή Αλγορίθμου

Στην παρούσα διπλωματική εργασία αναπτύσσεται αλγόριθμος ο οποίος επιχειρεί την ανωνυμοποίηση δεδομένων με λειτουργικές εξαρτήσεις που εκφράζονται μέσω οντολογιών πριν τη δημοσίευση τους. Στο κεφάλαιο αυτό παραθέτουμε τον αλγόριθμο που προτείνουμε για το πρόβλημα που περιγράψαμε στην προηγούμενη ενότητα. Πιο συγκεκριμένα, αρχικά περιγράφουμε τον αλγόριθμο για την κατασκευή του γράφου συσχετίσεων από μια οντολογία κατά τον λογικό συμπερασμό και, στη συνέχεια, παραθέτουμε έναν άπληστο αλγόριθμο που λύνει το πρόβλημα της ανωνυμοποίησης των δεδομένων χρησιμοποιώντας το γράφο συσχετίσεων.

Κύριος σκοπός του αλγορίθμου είναι η ποιότητα των αποτελεσμάτων, δηλαδή η ελαχιστοποίηση της ποσότητας $f(C, G)$. Η αποδοτικότητα του αλγόριθμου είναι δευτερεύοντος ενδιαφέροντος διότι η ανωνυμοποίηση γίνεται μια φορά μόνο, πριν την δημοσιοποίηση των δεδομένων. Με άλλα λόγια, ένας αλγόριθμος που τρέχει σε λογικό χρόνο αλλά έχει πολύ μεγάλο χρόνο εκτέλεσης χειρότερης περίπτωσης είναι αποδεκτός.

Η βασική ιδέα του αλγορίθμου είναι σε κάθε βήμα να βρίσκουμε τον βασικό κόμβο που ελαχιστοποιεί τη συνάρτηση $f(C, G)$ από αυτούς που 'κόβουν' τα περισσότερα μονοπάτια στον γράφο συσχετίσεων προς τα μυστικά. Η διαδικασία αυτή επαναλαμβάνεται μέχρι να κρύψουμε όλα τα μυστικά, δηλαδή μέχρι να καλύψουμε το σύνολο U.

Ο αλγόριθμος παίρνει ως είσοδο την οντολογία που χρησιμοποιείται για τη μοντελοποίηση των δεδομένων. Με την οντολογία κατασκευάζουμε το γράφο συσχετίσεων. Στη συνέχεια, κάποιος ειδικός διαλέγει τα μυστικά που θέλουμε να μείνουν κρυφά από τα υπονοούμενα γεγονότα. Ο αλγόριθμος ανωνυμοποίησης μας δίνει ως αποτέλεσμα τα βασικά γεγονότα που πρέπει να μην δημοσιεύσουμε ώστε ο αντίπαλος να μην μπορεί συμπεράνει κάποιο από τα μυστικά.

5.1 Γράφος Συσχετίσεων

Ο γράφος συσχετίσεων, όπως αυτός περιγράφηκε στο τέταρτο κεφάλαιο, χρειάζεται ως είσοδος στον αλγόριθμο ανωνυμοποίησης. Διαισθητικά, ο γράφος αυτός απεικονίζει όλες τις εξαρτήσεις μεταξύ των ρητών και των υπονοούμενων γεγονότων, και δημιουργείται κατά τον λογικό συμπερασμό(reasoning).

Η βασική ιδέα για τη κατασκευή του γράφου συσχετίσεων είναι η εξής: Αρχικά, κανονικοποιούμε τα δεδομένα της οντολογίας ώστε να είναι σε μορφή που να μπορούμε αποδοτικά να εφαρμόσουμε λογικό συμπερασμό. Στη συνέχεια, εφαρμόζουμε το λογικό συμπερασμό αναδρομικά και κρατάμε σε έναν πίνακα τις σχέσεις που χρησιμοποιήθηκαν για την παραγωγή κάθε νέου υπονοούμενου γεγονότος. Αυτές οι σχέσεις μεταξύ των γεγονότων αποτελούν τον γράφο εξαρτήσεων. Στη συνέχεια περιγράφουμε αναλυτικά όλα τα βήματα για τη δημιουργία του γράφου συσχετίσεων, ο οποίος φαίνεται συνοπτικά στον αλγόριθμο 1.

Προετοιμασία Δεδομένων. Πρέπει στην αρχή να φέρουμε τα δεδομένα της οντολογίας στη μορφή που θα χρειαστούμε στη συνέχεια για να εφαρμόσουμε αναδρομικά τους κανόνες του συμπερασμού. Αυτή η μορφή είναι τα τέσσερα διαφορετικά είδη αξιωμάτων που φαίνονται στο σχήμα 5.2. Η οντολογία πρέπει να είναι σε μορφή OWL Functional Syntax. Αν δεν είναι, τότε θα πρέπει να τη μετατρέψουμε.

Δεύτερο βήμα είναι να αφαιρέσουμε αξιώματα τα οποία περιλαμβάνουν συνενώσεις (UnionOf).

Στο Τρίτο βήμα πρέπει να σπάσουμε τα ισοδύναμα αξιώματα σε δυο αξιώματα, γιατί δεν θέλουμε αυτή τη μορφή αξιωμάτων. Κάθε αξίωμα της μορφής EquivalentClasses(A,B) και EquivalentProperties(A,B), θα πρέπει να σπάσει σε δυο ισοδύναμα αξιώματα SubClassOf(A,B), SubClassOf(B,A) και SubPropertyOf(A,B), SubPropertyOf(A,B) αντίστοιχα. Στο τέλος αυτού του βήματος θα πρέπει να έχουμε αξιώματα μόνο των ακόλουθων δυο μορφών:
SubClassOf(A,B), όπου A,B είναι αυθαίρετες κλάσεισ(σύνθετες εκφράσεις με κλάσεις),
SubPropertyOf(R,S), όπου R, S είναι ονόματα ιδιοτήτων (URIs).

Στο επόμενο βήμα πρέπει να εφαρμόσουμε τους κανόνες της κανονικοποίησης που φαίνονται στο σχήμα 5.1, ώστε να απλοποιηθούν οι σύνθετες και φωλιασμένες εκφράσεις και να έχουμε ως αποτέλεσμα τις απλές εκφράσεις που φαίνονται στο σχήμα 5.2. Κάθε ένας από αυτούς τους κανόνες αντικαθιστά το αξίωμα που φαίνεται στο αριστερό μέρος με το σύνολο των αξιωμάτων που φαίνονται στο δεξιό μέρος. Οι κανόνες αυτοί εφαρμόζονται εξαντλητικά, δηλαδή, μέχρι να μην μπορούν να εφαρμοστούν πλέον στα αξιώματα που έχουμε.

NR1: $\widehat{C} \sqsubseteq \widehat{D} \rightarrow \{\widehat{C} \sqsubseteq X, X \sqsubseteq \widehat{D}\}$

NR2: $C \sqsubseteq \top \rightarrow$ **remove the axiom (this knowledge is always true)**

NR3: $\bot \sqsubseteq C \rightarrow$ **remove the axiom (this knowledge is always true)**

NR4: $\widehat{C} \sqcap A \sqsubseteq B \rightarrow \{\widehat{C} \sqsubseteq X, X \sqcap A \sqsubseteq B\}$

NR5: $A \sqsubseteq C \sqcap D \rightarrow \{A \sqsubseteq C, A \sqsubseteq D\}$

NR6: $\exists R.\widehat{C} \sqsubseteq A \rightarrow \{\widehat{C} \sqsubseteq X, \exists R.X \sqsubseteq A\}$

NR7: $A \sqsubseteq \exists R.\widehat{C} \rightarrow \{A \sqsubseteq \exists R.X, X \sqsubseteq \widehat{C}\}$

Σχήμα 5.1: Κανόνες Κανονικοποίησης

$$\mathbf{1}.C_1 \sqsubseteq C_2 \qquad \mathbf{2}.C_1 \sqcap C_2 \sqsubseteq C_3 \qquad \mathbf{3}.C_1 \sqsubseteq \exists R.C_2 \qquad \mathbf{4}.\exists R.C_1 \sqsubseteq C_2$$

Σχήμα 5.2: Επιθυμητή Μορφή Αξιωμάτων

Στο σχήμα ο συμβολισμός $Z \sqsubseteq W$ σημαίνει SubClassOf(Z,W), ενώ ο συμβολισμός $\exists R.W$ συμβολίζει το ObjectSomeValuesFrom(R,W). Ακόμη οι συμβολισμοί $\top$ και $\bot$ συμβολίζουν τα OWLThing και OWLNothing αντίστοιχα, που μπορεί να μην εμφανίζονται σε μια οντολογία. Το σύμβολο $\sqcap$ συμβολίζει την τομή, έτσι, ο συμβολισμός $A \sqcap B \sqsubseteq C$ συμβολίζει το SubClassOf(IntersectionOf(A,B),C). Ακόμη, τα σύμβολα $\widehat{C}$, $\widehat{D}$ συμβολίζουν σύνθετες εκφράσεις κλάσεων. Για παράδειγμα, IntersectionOf(C1,C2,C3) ή ObjectSomeValuesFrom(R,C). Τα σύμβολα A, B αναπαριστούν ονόματα απλών κλάσεων τα οποία υπήρχαν στην αρχική οντολογία ή δημιουργήθηκαν σε προηγούμενα βήματα της κανονικοποίησης. Το σύμβολο X αναπαριστά ένα νέο όνομα κλάσης που δημιουργήθηκε κατά την κανονικοποίηση. Τέλος, τα σύμβολα C, D αναπαριστούν ένα όνομα απλής κλάσης (όπως A, B, X) ή μίας σύνθετης κλάσης (δηλαδή $\widehat{C}, \widehat{D}$). Με άλλα λόγια, όταν χρησιμοποιούνται τα σύμβολα C, D, τότε οι κανόνες εφαρμόζονται ανεξάρτητα από το είδος των κλάσεων.

Στο τέλος της κανονικοποίησης πρέπει να έχουμε αξιώματα μόνο της μορφής που φαίνονται στο σχήμα 5.2 και είναι τύπου 1,2,3 ή 4. R είναι όνομα ιδιότητας και C_1, C_2, C_3 είναι ονόματα απλών κλάσεων, δηλαδή URIs ή ονόματα που παραχθήκαν κατά την κανονικοποίηση.

Λογικός Συμπερασμός. Σε αυτό το βήμα, πρέπει να εκτελέσουμε το λογικό συμπερασμό στα αξιώματα που έχουμε και να βρούμε όλα τα υπονοούμενα γεγονότα που προκύπτουν για τη βάση δεδομένων. Κατά τη διάρκεια του λογικού συμπερασμού, πρέπει να αποθηκεύουμε τις εξαρτήσεις που υπάρχουν στα δεδομένα. Γι'αυτό το λόγο πρέπει να αρχικοποιήσουμε σε αυτό το βήμα τις κατάλληλες δομές δεδομένων που χρειάζονται τόσο για την γρήγορη προσπέλαση των αξιωμάτων της βάσης δεδομένων όσο και για την αποθήκευση των συσχετίσεων που θα βρούμε, δηλαδή για τον γράφο συσχετίσεων που ψάχνουμε να βρούμε.

Για να εφαρμόσουμε τον λογικό συμπερασμό, πρέπει να εφαρμόσουμε τους κανόνες που φαίνονται στο σχήμα 5.3 εξαντλητικά. Το $\wedge$ συμβολίζει το λογικό 'και', το οποίο σε τερμινολογία βάσεων δεδομένων είναι μια ένωση. Με λίγα λόγια, όταν βρίσκονται κάποια αξιώματα στη βάση δεδομένων που ικανοποιούν κάποιο από τους κανόνες του σχήματος 5.3, τότε δημιουργούνται τα αξιώματα που βρίσκονται στο αριστερό μέρος των κανόνων και προστίθενται στην βάση. Όλοι αυτοί οι κανόνες μπορούν να εφαρμοστούν αποδοτικά με τις κατάλληλες

$$\begin{aligned}
&\mathbf{IR1}\ C_1 \sqsubseteq C_3 \leftarrow C_1 \sqsubseteq C_2 \ \wedge\ C_2 \sqsubseteq C_3\\
&\mathbf{IR2}\ C_1 \sqsubseteq C_4 \leftarrow C_1 \sqsubseteq C_2 \ \wedge\ C_1 \sqsubseteq C_3 \ \wedge\ C_2 \sqcap C_3 \sqsubseteq C_4\\
&\mathbf{IR3}\ C_1 \sqsubseteq \exists R.C_3 \leftarrow C_1 \sqsubseteq C_2 \ \wedge\ C_2 \sqsubseteq \exists R.C_3\\
&\mathbf{IR4}\ C_1 \sqsubseteq C_4 \leftarrow C_1 \sqsubseteq \exists R.C_2 \ \wedge\ C_2 \sqsubseteq C_3 \ \wedge\ \exists R.C_3 \sqsubseteq C_4\\
&\mathbf{IR5}\ C_1 \sqsubseteq \exists R_2.C_2 \leftarrow C_1 \sqsubseteq \exists R_1.C_2 \ \wedge\ R_1 \sqsubseteq R_2\\
&\mathbf{IR6}\ C_1 \sqsubseteq \exists R_3.C_3 \leftarrow C_1 \sqsubseteq \exists R_1.C_2 \ \wedge\ C_2 \sqsubseteq \exists R_2.C_3 \ \wedge\ R_1 \circ R_2 \sqsubseteq R_3
\end{aligned}$$

Σχήμα 5.3: Κανόνες Λογικού Συμπερασμού

δομές δεδομένων (π.χ. πίνακες κατακερματισμού για τα αξιώματα).

Οι δομές δεδομένων που χρειάζονται είναι οι ακόλουθοι πίνακες κατακερματισμού:
H αποθηκεύει όλα τα αξιώματα τύπου 1 και 3, με κλειδί το $C1$,
H'αποθηκεύει όλα τα αξιώματα τύπου 1 και 3, με κλειδί το $C2$,
H_2 αποθηκεύει όλα τα αξιώματα τύπου 2, με κλειδί το $C1$,
H_4 αποθηκεύει όλα τα αξιώματα τύπου 4, με κλειδί το $C1$,
H_P αποθηκεύει όλα τα αξιώματα τύπου SubPropertyOf(R,S), με κλειδί το R,
$deltaH$ αποθηκεύει όλα τα αξιώματα τύπου 1 και 3 που παράγονται από τον αλγόριθμο, με κλειδί το $C1$,
$deltaH$'αποθηκεύει όλα τα αξιώματα τύπου 1 και 3 που παράγονται από τον αλγόριθμο, με κλειδί το $C2$,

Πρώτο βήμα του αλγορίθμου συμπερασμού είναι να εφαρμόσουμε την κατάλληλη ένωση ώστε να εφαρμόσουμε τους κανόνες IR1, IR3, IR4, IR6. Η ένωση αυτή γίνεται με βάση τα $C1 = C2$ κλειδιά που πρέπει να είναι ίσα. Η εφαρμογή της ένωσης εξαρτάται από τον τρόπο που έχουμε αποθηκεύσει τα αξιώματα.

Επόμενο βήμα, είναι να εφαρμόσουμε τους υπόλοιπους κανόνες, οι οποίοι δεν χρειάζονται κάποια ένωση. Για κάθε αξίωμα τύπου 2, βρίσκουμε αν υπάρχουν τα κατάλληλα αξιώματα τύπου 1 ώστε να ισχύει ο κανόνας IR2. Για κάθε αξίωμα του τύπου 3, βρίσκουμε αν υπάρχει το κατάλληλο αξίωμα SubPropertyOf που χρειάζεται για να ισχύει ο κανόνας IR5.

Τρίτο βήμα, είναι να εφαρμόσουμε την ένωση του πρώτου ερωτήματος, αλλά σε αυτό το βήμα, χρησιμοποιώντας τα νέα αξιώματα που έχουν παραχθεί για $C1$. Στη συνέχεια, εφαρμόζουμε και πάλι τους κανόνες του βήματος 1, δηλαδή τους κανόνες IR1, IR3, IR4, IR6.

Τέταρτο βήμα είναι να ενώσουμε τα νέα αξιώματα με τα αξιώματα που ήδη έχουμε και να εφαρμόσουμε τους κανόνες IR2, IR5 όπως στο δεύτερο βήμα.

Στο πέμπτο βήμα εφαρμόζουμε και πάλι τους κανόνες του βήματος 1, αλλά αυτή τη φορά χρησιμοποιούμε τα νέα αξιώματα που έχουν παραχθεί για το $C2$.

Στη συνέχεια, ανανεώνουμε όλες τις δομές μας ώστε να έχουν και τα νέα και τα προηγούμενα αξιώματα. Επαναλαμβάνουμε τη διαδικασία από το τέταρτο βήμα και μετά, μέχρι να μην παραχθεί κάποιο νέο αξίωμα σε οποιοδήποτε βήμα της διαδικασίας. Τότε, θα έχουμε βρει εξαντλητικά όλα τα παραγόμενα γεγονότα, και τον γράφο εξαρτήσεων.

Αξίζει να σημειωθεί ότι στα βήματα ένα και δυο, χρειάζονται και ορισμένες άλλες ενώσεις ώστε να εφαρμοστούν αποδοτικά οι κανόνες. Ακόμη, πριν προσθέσουμε τα νέα αξιώματα στους πίνακες $deltaH$, $deltaH$', πρέπει να ελέγχουμε ότι δεν υπάρχουν ήδη ή ότι δεν έχουν παραχθεί σε κάποιο προηγούμενο βήμα από τα ίδια αξιώματα. Αν κάποιο αξίωμα παραχθεί από διαφορετικό κανόνα ή αξιώματα, τότε δημιουργείτε μια νέα εξάρτηση στον γράφο που αναζητούμε, οπότε πρέπει να το αποθηκεύσουμε και στον πίνακα H.

Algorithm 1 Γράφος Συσχετίσεων

Input: Οντολογία

Output: Ο γράφος εξαρτήσεων G

% *Προετοιμασία Δεδομένων*
Μετατροπή της Οντολογίας σε μορφή OWL Functional Syntax
Αφαίρεση UnionOf()
Σπάσιμο ισοδύναμων αξιωμάτων
% *Κανονικοποίηση*
while Υπάρχουν σύνθετα αξιώματα **do**
    Εφάρμοσε κανόνες κανονικοποίησης
end while
% *Αρχικοποίηση Λογικού Συμπερασμού*
Αρχικοποίηση πινάκων κατακερματισμού $H, H_2, H_4, H_p, deltaH, H'$, $deltaH'$
Αποθήκευση αξιώματα τύπου 1 και 3 στον H με κλειδί το C_1
Αποθήκευση αξιώματα τύπου 1 και 3 στον H'με κλειδί το C_2
Αποθήκευση αξιώματα τύπου 2 στον H_2 με κλειδί το C_1
Αποθήκευση αξιώματα τύπου 4 στον H_4 με κλειδί το C_1
Αποθήκευση αξιώματα $SubPropertyOf$ στον H_p με κλειδί το R
% *Λογικός Συμπερασμός*
Ένωση H με H'στα κλειδιά $C_1 = C_2$
Με χρήση της ένωσης, εφαρμογή κανόνων IR1, IR3, IR4, IR6
Αποθήκευση νέων αξιωμάτων και εξαρτήσεων σε $deltaH, deltaH'$
Εφαρμογή κανόνων IR2, IR5
Αποθήκευση νέων αξιωμάτων και εξαρτήσεων σε $deltaH, deltaH'$
repeat
    Ένωση $deltaH$ με H'στα κλειδιά $C_1 = C_2$
    Με χρήση της ένωσης, εφαρμογή κανόνων IR1, IR3, IR4, IR6
    Αποθήκευση νέων αξιωμάτων και εξαρτήσεων σε $deltaH, deltaH'$
    Συγχώνευση $deltaH$ και H
    Εφαρμογή κανόνων IR2, IR5
    Αποθήκευση νέων αξιωμάτων και εξαρτήσεων σε $deltaH, deltaH'$
    Ένωση H με $deltaH'$στα κλειδιά $C_1 = C_2$
    Με χρήση της ένωσης, εφαρμογή κανόνων IR1, IR3, IR4, IR6
    Αποθήκευση νέων αξιωμάτων και εξαρτήσεων σε $deltaH, deltaH'$
    Συγχώνευση $deltaH'$και H'
until Κανένα νέο αξίωμα δεν παράχθηκε

5.2 Αλγόριθμος Ανωνυμοποίησης

Ο αλγόριθμος ανωνυμοποίησης των δεδομένων μας που παρουσιάζουμε εδώ, χρειάζεται ως είσοδο τον γράφο συσχετίσεων που περιγράψαμε προηγουμένως πως παράγεται. Έχοντας τον γράφο συσχετίσεων για την οντολογία που υποθέτουμε ότι έχει ο αντίπαλος, και δεδομένων των μυστικών γεγονότων που θέλουμε να παραμείνουν κρυφά, μπορούμε να εφαρμόσουμε τον αλγόριθμο που περιγράφουμε στη συνέχεια. Ο αλγόριθμος αυτός εξασφαλίζει ότι ο αντίπαλος δεν θα μπορέσει με λογικό συμπερασμό να ανακαλύψει τα μυστικά. Επειδή, όπως ήδη έχουμε αναφέρει και αποδείξει ότι το πρόβλημα είναι $NP-$δύσκολο, ο αλγόριθμος μας χρησιμοποιεί μια άπληστη ευριστική ώστε να διαλέξει τα βασικά γεγονότα που πρέπει να απομακρύνουμε με κάποιον έξυπνο τρόπο ώστε να μπορεί ο αλγόριθμος να τελειώνει σε λογικό χρόνο. Η εύρεση της βέλτιστης λύσης δεν είναι εφικτή στα περισσότερα πραγματικά δεδομένα όπου οι οντολογίες αποτελούνται από εκατομμύρια αξιώματα. Ως αποτέλεσμα, το αποτέλεσμα του αλγορίθμου μπορεί να μην είναι το βέλτιστο, δηλαδή μπορεί να μην πετυχαίνουμε την ελάχιστη απώλεια γνώσης κατά τη δημοσίευση των δεδομένων, αλλά ο αλγόριθμος μας μπορεί να εκτελείται σε πραγματικά δεδομένα.

Ο αλγόριθμος χρειάζεται ως είσοδο τον γράφο συσχετίσεων όπως περιγράψαμε προηγουμένως. Δεδομένου του γράφου, έξοδος του αλγορίθμου είναι το σύνολο των αξιωμάτων που μπορούν να δημοσιευτούν, χωρίς κίνδυνο να παραβιαστεί η ιδιωτικότητα στο σενάριο που ορίζεται από το πρόβλημα. Στη συνέχεια περιγράφουμε αναλυτικά όλα τα βήματα για του αλγορίθμου ανωνυμοποίησης, ο οποίος φαίνεται σε ψευδοκώδικα στον αλγόριθμο 2.

Είσοδος. Αρχικά, ο αλγόριθμος διαβάζει τον γράφο εξαρτήσεων ως είσοδο. Η είσοδος αποτελείται από τις ακμές και τους κόμβους του γράφου, οπότε, στη συνέχεια πρέπει να ανακατασκευάσουμε τον γράφο για να τον έχουμε στην κατάλληλη μορφή για γρήγορη διάσχιση. Ακόμη, αρχικά πρέπει να διαβάσουμε ποιοι κόμβοι θεωρούνται μυστικά. Οι κόμβοι αυτοί είτε επιλέγονται από κάποιον ειδικό και δίνονται ως είσοδος στον αλγόριθμο, είτε διαλέγονται τυχαία για τον πειραματικό έλεγχο.

Υπολογισμός μέγιστου addedVops. Πρώτο βήμα του αλγορίθμου είναι να υπολογίσουμε με πόσους κόμβους V_{op} που θέλουμε να κρύψουμε σχετίζεται κάθε απλός κόμβος, που αντιστοιχεί σε ένα βασικό γεγονός της οντολογίας. Κάθε μυστικό μπορεί να παραχθεί από διαφορετικούς V_{op} κόμβους και αυτοί οι κόμβοι έχουν ως είσοδο άλλα γεγονότα. Σύνολο P_i, είναι το σύνολο βασικών κόμβων που αποτελούν είσοδο αυτών των V_{op} κόμβων. Είναι, δηλαδή, οι βασικοί κόμβοι από τους οποίους εξαρτάται το μυστικό i. Πρέπει να σημειώσουμε ότι οι βασικοί κόμβοι αυτοί μπορούν να βρίσκονται αρκετά επίπεδα μακριά από το μυστικό, συμμετέχοντας εμμέσως σε αυτό μέσω πολλών διαφορετικών επιπέδων υπονοούμενων κόμβων στον γράφο. Η ένωση όλων των P_i συνόλων αποτελούν το S σύνολο, στο οποίο πρέπει να ψάξουμε για να βρούμε ποιος κόμβος συσχετίζεται με περισσότερους V_{op} κόμβους που θέλουμε να κρύψουμε (συνεισφέρει δηλαδή το μέγιστο addedVops). Για να το επιτύχουμε αυτό, διατρέχουμε τα δέντρα που δημιουργούνται αν διαλέξουμε ως ρίζα τα μυστικά του γράφου. Κατά αυτή την ανάγνωση των δέντρων που σχηματίζονται, αν βρούμε κάποιο κόμβο που έχει κρυφτεί σε προηγούμενο βήμα του αλγορίθμου, κουρεύουμε αυτό το μονοπάτι γιατί πλέον

δεν μας απασχολεί. Ψάχνουμε να βρούμε όλους τους βασικούς κόμβους και να βρούμε ποιοι σχετίζονται με ποια $V_{op}s$. Οπότε την πρώτη φορά που βρίσκουμε έναν βασικό κόμβο σε ένα μονοπάτι από κάποιον V_{op} που θέλουμε να κρύψουμε, αυξάνουμε το πεδίο του κόμβου addedVops. Με άλλα λόγια, με αυτόν τον τρόπο μετράμε πόσα μονοπάτια από V_{op} κόμβους που θέλουμε να κρύψουμε καταλήγουν σε κάθε βασικό κόμβο. Διαισθητικά, αν καταλήγουν πολλά μονοπάτια σε έναν βασικό κόμβο, κρύβοντας τον πλησιάζουμε και πιο κοντά στο να κρύψουμε τους V_{op} κόμβους των μυστικών. Πρέπει να προσέξουμε σε αυτό το σημείο να μετράμε μόνο μια φορά κάθε βασικό κόμβο για κάθε V_{op} κόμβο, παρ'ότι μπορεί να τον συναντήσουμε πολλές φορές όταν αποτελεί είσοδο για διαφορετικά $V_{op}s$ σε διαφορετικά μονοπάτια.

Επιλογή κόμβου προς απόκρυψη. Επόμενο βήμα είναι η επιλογή του κόμβου που θα κρύψουμε σε αυτή την επανάληψη του αλγορίθμου. Αν υπάρχει μόνο ένας βασικός κόμβος που έχει το μέγιστο addedVops που υπολογίσαμε προηγουμένως, τότε επιλέγουμε αυτόν. Σε διαφορετική περίπτωση, η επιλογή γίνεται με κριτήριο την ελαχιστοποίηση του κόστους, ανάμεσα στους κόμβους με ίδιο μέγιστο addedVops. Γενικώς, η συνάρτηση κόστους μπορεί να οριστεί με διαφορετικό τρόπο αναλόγως την εφαρμογή του αλγορίθμου. Εμείς, επιλέξαμε έναν απλό τρόπο υπολογισμού του κόστους, το πόσοι κόμβοι συνολικά θα χαθούν με την αφαίρεση του βασικού κόμβου. Το κόστος αυτό μοιάζει με το addedVops, όμως, είναι διαφορετικό για τον ακόλουθο λόγο. Διασχίζουμε τα δέντρα που σχηματίζονται από τους μυστικούς κόμβους με τον ίδιο τρόπο, όμως, πρέπει να λάβουμε υπόψιν μας ότι πολλά υπονοούμενα γεγονότα παράγονται από πολλά διαφορετικά μονοπάτια, οπότε η αφαίρεση ενός μόνο βασικού κόμβου μπορεί να μην οδηγήσει στην απόκρυψη ενός γεγονότος, παρ'ότι το γεγονός αυτό εξαρτάται από βασικό κόμβο που κρύβουμε. Με άλλα λόγια, πρέπει σε κάθε υπονοούμενο κόμβο να ελέγχουμε αν ο βασικός κόμβος που εξετάζουμε είναι κρίσιμος. Ένας βασικός κόμβος είναι κρίσιμος για έναν υπονοούμενο κόμβο, αν αφαίρεση του οδηγεί σίγουρα σε απόκρυψη του υπονοούμενου βρόχου. Αυτό μπορεί να συμβαίνει είτε διότι ο βασικός κόμβος αποτελεί είσοδο στον μοναδικό V_{op} κόμβο που παράγει τον υπονοούμενο κόμβο, είτε διότι ο βασικός κόμβος αποτελεί είσοδο κάποιου άλλου V_{op} απαραίτητο για τον υπονοούμενο κόμβο.

Ενημέρωση κρυμμένων κόμβων. Σε αυτό το βήμα, πρέπει να δούμε ποιοι κόμβοι θα κρυφτούν ως αποτέλεσμα του να κρύψουμε τον βασικό κόμβο που επιλέξαμε στο προηγούμενο βήμα. Για να το κάνουμε αυτό, διατρέχουμε τους γράφους ανάποδα από πριν. Ξεκινάμε από το βασικό κόμβο που έχουμε επιλέξει να κρύψουμε, και διασχίζουμε ανάποδα τον γράφο, προς τους μυστικούς κόμβους. Αν οι κόμβοι στους οποίους συμμετέχει ο βασικός κόμβος, χάνονται, δηλαδή αν ο βασικός κόμβος είναι κρίσιμος για αυτούς, τότε κρύβουμε τους κόμβους αυτούς. Αυτό το σημειώνουμε σε ένα ειδικό πεδίο που έχουμε για κάθε κόμβο για να βλέπουμε αν έχει κρυφτεί σε κάποιο βήμα του αλγορίθμου. Η διαδικασία συνεχίζεται αναδρομικά, μέχρι να βρούμε κάποιο κόμβο που δεν κρύβεται, όποτε και τελειώνει η αναδρομή για το αντίστοιχο μονοπάτι.

Συνθήκη τερματισμού. Τα προηγούμενα τρία βήματα επαναλαμβάνονται μέχρι να κρυφτούν όλοι οι μυστικοί κόμβοι που έχουμε στον γράφο εξαρτήσεων, δηλαδή μέχρι να μην υπάρχει κάποιο μονοπάτι προς τους V_{op} κόμβους των μυστικών. Για να ισχύει η συνθήκη τερματισμού, θα πρέπει να έχει καλυφτεί όλο το σύνολο των μυστικών. Με χρήση του ειδικού

πεδίου που δείχνει αν ένας κόμβος κρύβεται (δηλαδή δεν μπορεί να παραχθεί κάπως με χρήση της οντολογία έχοντας κρύψει τους βασικούς κόμβους που έχουμε επιλέξει), μπορούμε εύκολα να ελέγξουμε αυτή τη συνθήκη. Αρκεί να ελέγξουμε αν όλοι οι V_{op} κόμβοι των μυστικών είναι κρυμμένοι σύμφωνα με αυτό το πεδίο. Τότε, τα μυστικά αξιώματα είναι προστατευμένα αν κρύψουμε όλα τα βασικά γεγονότα που έχουμε επιλέξει και ο αλγόριθμος τελειώνει.

Algorithm 2 Αλγόριθμος Ανωνυμοποίησης

Input: Γράφος εξαρτήσεων G

Output: Αξιώματα προς δημοσίευση

```
1: Ανάγνωση ακμών γράφου εξαρτήσεων
2: Ανακατασκευή γράφου εξαρτήσεων
3: Ανάγνωση μυστικών κόμβων (ή τυχαία επιλογή αυτών)
4: % Όσο όλα τα μυστικά δεν είναι κρυμμένα
5: while !isCovered do
6:    % Εύρεση κόμβων που προσθέτουν περισσότερους V_op κόμβους
7:    while secrets.tree.hasNext() do
8:       if (!node.ishidden() && node.isExplicit() && node.notFoundBefore()) then
9:          node.addedVops++;
10:      end if
11:   end while
12:   max ← findNodeWithMaxAddedVops();
13:   % Επιλογή κόμβου προς κρύψιμο και κρύψιμο αυτού
14:   if (nodesWithSameMax > 1) then
15:      for each node: n do
16:         n.calculateCost();
17:      end for
18:   end if
19:   hide ← nodeWithMinimumCost();
20:   hide.hideThisNode();
21:   graph.updateHiddenNodes();
22: end while
```

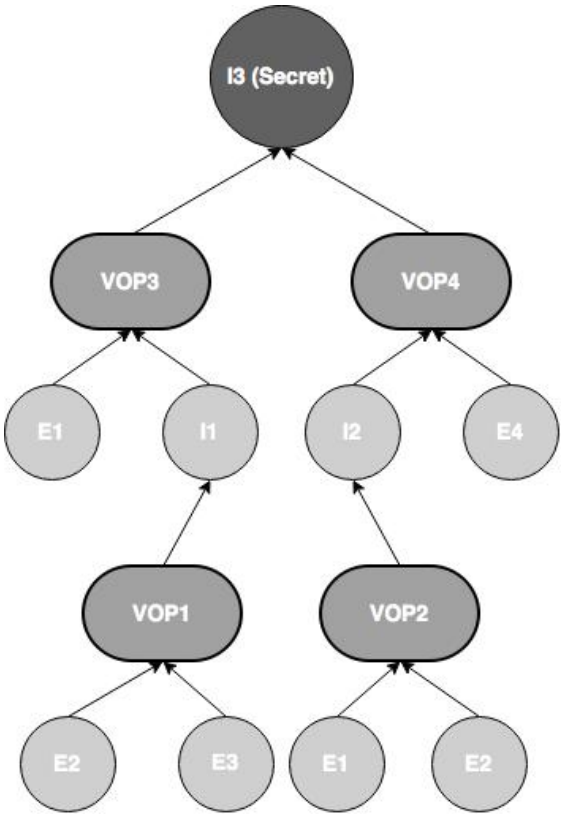
5.2.1 Παραδείγματα Υλοποίησης

Για την καλύτερη κατανόηση του αλγορίθμου, χρησιμοποιούμε το παράδειγμα με γράφο εξαρτήσεων που φαίνεται στο σχήμα 5.4. Μυστικός κόμβος είναι ο κόμβος I_3, τον οποίο θέλουμε να κρύψουμε. Υπάρχουν τέσσερα βασικά αξιώματα τα οποία μπορούμε να κρύψουμε και θέλουμε να βρούμε ποια πρέπει να αφαιρέσουμε ώστε να κρύψουμε το I_3, έχοντας το

μικρότερο κόστος. Θεωρούμε και εδώ ως συνάρτηση κόστους το πλήθος των κόμβων που χάνονται αν αφαιρέσουμε έναν κόμβο (δεν μετράμε τον μυστικό κόμβο ως χαμένο γιατί πάντα θα είναι κρυμμένος στο τελικό αποτέλεσμα).

Αφού διαβάσουμε τον γράφο εξαρτήσεων, πρώτο βήμα του αλγορίθμου είναι να βρούμε τον κόμβο με μέγιστο addedVops. Ξεκινάμε τη διάσχιση των δέντρων που σχηματίζονται από τα V_{op3} και V_{op4}. Τα βασικά αξιώματα E_3, E_4 τα συναντάμε μόνο μια φορά, οπότε συμμετέχουν σε ένα V_{op} το κάθε ένα και έχουν addedVops 1. Τα E_1, E_2 τα συναντάμε σε 2 διαφορετικά μονοπάτια από διαφορετικά V_{op}, οπότε έχουν addedVops 2 και τα δυο. Με άλλα λόγια, τα E_1, E_2 συμμετέχουν σε μονοπάτια για δυο διαφορετικούς V_{op} κόμβους που θέλουμε να κρύψουμε.

Αφού έχουμε πάνω από έναν κόμβο με το ίδιο μέγιστο addedVops, πρέπει να επιλέξουμε ποιον κόμβο θα διαλέξουμε να κρύψουμε. Θα υπολογίσουμε το κόστος του καθενός. Το κόστος της αφαίρεσης του E_1 είναι $cost(E_1) = |E_1, I_2| = 2$, ενώ το κόστος του E_2 είναι $cost(E_2) = |E_2, I_1, I_2| = 3$. Οπότε επιλέγουμε να αφαιρέσουμε τον κόμβο E_1. Αυτό το υπολογίζουμε ξεκινώντας από τον κάθε βασικό κόμβο και βλέποντας, για παράδειγμα, ότι αν αφαιρέσουμε τον E_2 από τα αποτελέσματα, τότε χάνεται ο I_1 γιατί δεν παράγεται παρά μόνο από τον V_{op1}. Ομοίως, χάνεται και ο I_2. Ακόμη, χάνεται και ο V_{op3} και ο V_{op4}, οπότε τελικά χάνεται και ο I_3. Η επιλογή του E_1 κόμβου έχει ως αποτέλεσμα να μην χάσουμε και τον κόμβο I_1 από τα τελικά δεδομένα.



Σχήμα 5.4: Γράφος Συσχετίσεων Παραδείγματος

Τέλος, πρέπει να ελέγξουμε αν κρύψαμε όλα τα μυστικά. Με τον ίδιο τρόπο που ελέγξαμε προηγουμένως το κόστος, σημειώνουμε ποιοι κόμβοι έχουν χαθεί σύμφωνα με την επιλογή που κάναμε προηγουμένως. Τελικώς, κοιτάμε αν έχουμε κρύψει όλους τους V_{op} κόμβους που δείχνουν σε όλους τους μυστικούς κόμβους.

Κεφάλαιο 6

Πειραματική Αξιολόγηση

Στο κεφάλαιο αυτό γίνεται η αξιολόγηση του προτεινόμενου αλγορίθμου ανωνυμοποίησης. Για την αξιολόγηση χρησιμοποιήθηκαν πραγματικά ιατρικά δεδομένα. Στο κεφάλαιο αυτό αναλύουμε όλες τις λεπτομέρειες που αφορούν τις παραμέτρους αξιολόγησης του αλγορίθμου, το σύνολο των δεδομένων που χρησιμοποιήθηκαν, τα πειραματικά αποτελέσματα και τους παράγοντες που επηρεάζουν την απόδοση του.

6.1 Μεθοδολογία Πειραμάτων

Τα πειράματα που εκτελέστηκαν βασίστηκαν στην υλοποίηση του αλγορίθμου όπως αυτή περιγράφεται στο κεφάλαιο 5, με χρήση της γλώσσας προγραμματισμού Java. Τα πειράματα εκτελέστηκαν σε servers με $512GB$ μνήμη.

Η πειραματική αξιολόγηση έγινε με χρήση της οντολογίας SNOMED CT [41]. Η SNOMED CT είναι μια επιστημονική οντολογία που χρησιμοποιείται στη βιοιατρική και η οποία συντηρείται ενεργά και χρησιμοποιείται ευρέως από επαγγελματίες και ερευνητές στους τομείς της βιοιατρικής. Συγκεκριμένα, η SNOMED CT είναι μια συλλογή από ιατρικούς όρους, η οποία είναι υπολογιστικά επεξεργάσιμη, και παρέχει όρους, συνώνυμα, κωδικούς και ορισμούς που χρησιμοποιούνται σε ιατρικά έγγραφα και αναφορές. Ακόμη, αυτή η οντολογία παρέχει τη γενική τερμινολογία που χρησιμοποιείται στους ιατρικούς φακέλους των ασθενών.

Στον πίνακα 6.1, φαίνονται ορισμένα χαρακτηριστικά της έκδοσης της SNOMED CT που χρησιμοποιήσαμε. Τα στοιχεία που φαίνονται αφορούν την οντολογία σε functional μορφή, όπως τη χρειαζόμαστε για να παράξουμε τον γράφο εξαρτήσεων και πριν εφαρμόσουμε την κανονικοποίηση. Αυτό σημαίνει ότι τα περισσότερα αξιώματα είναι αρκετά σύνθετα. Ακόμη,

Σύνολο Δεδομένων	**SNOMED CT**
Κλάσεις	352218
$SubClassOf$ Αξιώματα	265562
$EquivalentClasses$ Αξιώματα	70259

Πίνακας 6.1: Χαρακτηριστικά συνόλου δεδομένων SNOMED CT

από την αρχική μορφή της οντολογίας αγνοούμε τα αξιώματα AnnotationAssertion τα οποία παρέχουν πληροφορίες για κάθε κλάση.

6.2 Αποτελέσματα Πειραμάτων

Στη συνέχεια παρουσιάζουμε τα αποτελέσματα των πειραμάτων που εκτελέσαμε. Αρχικά, δίνουμε ορισμένα χαρακτηριστικά για τον γράφο εξαρτήσεων της SNOMED CT όπως αυτά προέκυψαν. Αυτά τα χαρακτηριστικά, μας βοηθούν να κατανοήσουμε καλύτερα τη μορφή που έχει ο γράφος και, στη συνέχεια, μας βοηθάνε να ερμηνεύσουμε αρκετά από τα αποτελέσματα. Στη συνέχεια, δείχνουμε πως αυξάνεται ο χρόνος εκτέλεσης της ανωνυμοποίησης ανάλογα με το πλήθος των μυστικών και πως επηρεάζεται η απώλεια πληροφοριών ανάλογα με την κατανομή των μυστικών, το πλήθος τους και το μέγεθος του γράφου που 'αγγίζει' ο αλγόριθμος.

6.2.1 Γράφος Εξαρτήσεων SNOMED CT

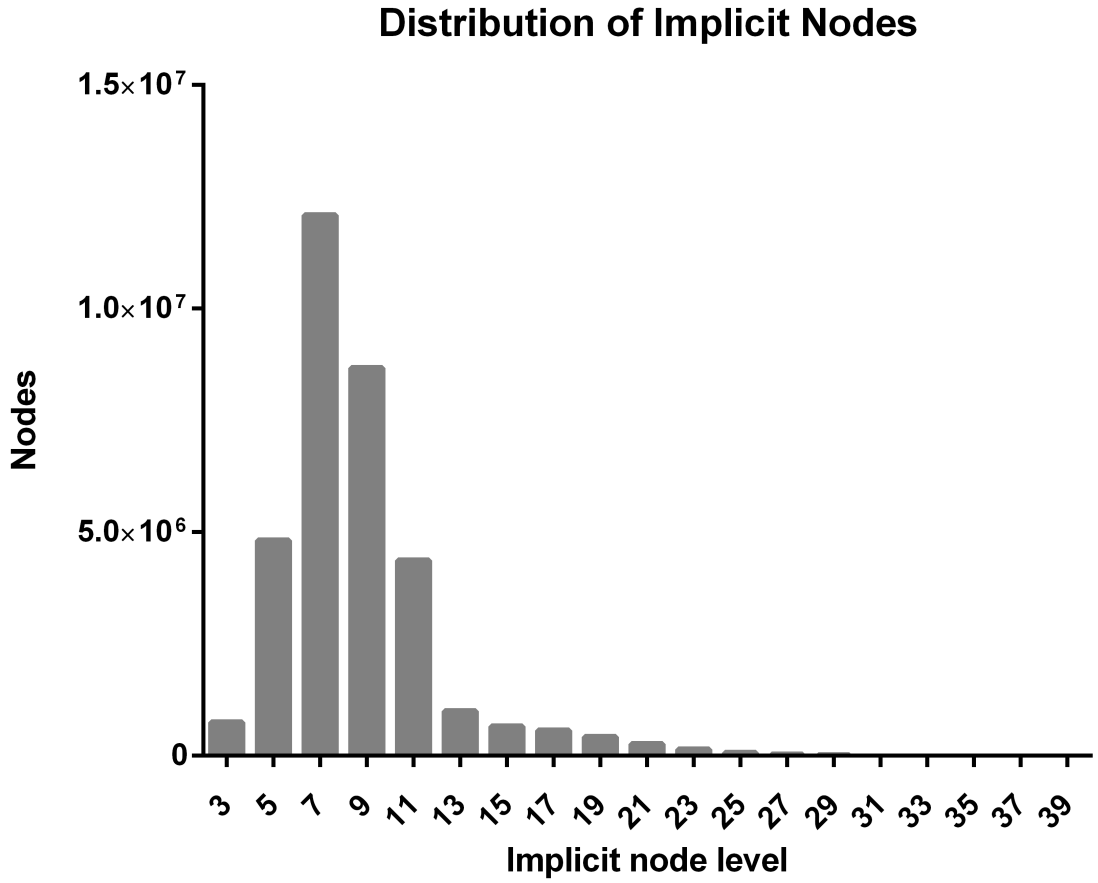
Στον πίνακα 6.2 βλέπουμε ορισμένα στατιστικά στοιχεία για τον γράφο εξαρτήσεων της SNOMED CT. Παρατηρούμε ότι μετά την κανονικοποίηση και την αφαίρεση των $EquivalentClasses$ αξιωμάτων, έχουμε πολλά περισσότερα βασικά αξιώματα. Ιδιαίτερη σημασία έχει ο μέσος όρος των V_{op} κόμβων που αντιστοιχούν σε κάθε υπονοούμενο κόμβο. Με άλλα λόγια, κάθε υπονοούμενος κόμβος παράγεται με 2.415 διαφορετικούς τρόπους, και με τυπική απόκλιση 2.615. Οπότε για να κρύψουμε έναν μυστικό κόμβο, κατά μέσο όρο πρέπει να κρύψουμε 2.415 διαφορετικούς V_{op} κόμβους. Αξιόλογο είναι ότι περίπου 92% των βασικών κόμβων δεν συμμετέχει στη παραγωγή κανενός υπονοούμενου κόμβου. Αυτό σχετίζεται και με το επόμενο στατιστικό, όπου φαίνεται ότι κάθε βασικός κόμβος συμμετέχει κατά μέσο όρο σε 6.648 διαφορετικούς V_{op} κόμβους, έχοντας τυπική απόκλιση 414.8. Έτσι, όπως θα δούμε και στη συνέχεια, αφαίρεση ενός λίγων μόνο βασικών κόμβων, μπορεί να οδηγήσει στην απόκρυψη πολλών υπονοούμενων κόμβων και να έχουμε μεγάλη απώλεια πληροφοριών. Τέλος, αξίζει να παρατηρήσουμε το μέγιστο μονοπάτι που βρίσκουμε στον γράφο, το οποίο είναι 39.

Γράφος Εξαρτήσεων SNOMED CT	
V_{op} Κόμβοι	81719594
Explicit Κόμβοι	213480714
Implicit Κόμβοι	33829104
Μ.Ο. V_{op} κόμβων για κάθε implicit Τυπική Απόκλιση	2.415 2.616
Explicit κόμβοι που δεν συμμετέχουν σε κανέναν V_{op}	198135831
Μ.Ο. V_{op} κόμβων που συμμετέχει κάθε explicit κόμβος Τυπική Απόκλιση	6.648 414.8
Μέγιστο μονοπάτι στον γράφο	39

Πίνακας 6.2: Χαρακτηριστικά γράφου εξαρτήσεων SNOMED CT

Για να κατανοήσουμε σε μεγαλύτερο βάθος τον γράφο εξαρτήσεων της SNOMED CT, παρουσιάζουμε στο σχήμα 6.1 το ιστόγραμμα κατανομής των υπονοούμενων κόμβων του γράφου ανάλογα με το επίπεδο που βρίσκονται. Το επίπεδο, αφού έχουμε κατευθυνόμενο γράφο, δεν είναι εύκολο να προσδιοριστεί. Στη συγκεκριμένη περίπτωση, ως επίπεδο, θεωρήσαμε το μέγιστο μονοπάτι που ξεκινάει από κάθε υπονοούμενο γράφο. Αυτό είναι εύκολο να προσδιοριστεί με μια κατά βάθος διάσχιση του δέντρου που δημιουργείται με ρίζα τον κάθε υπονοούμενο κόμβο. Ακόμη, τα επίπεδα των κόμβων είναι μόνο μονοί αριθμοί στο διάγραμμα λόγω της δομής του γράφου. Κάθε υπονοούμενος κόμβος συμμετέχει σε V_{op} κόμβους, οι οποίοι με την σειρά τους συμμετέχουν σε υπονοούμενους κτλ. Οπότε δυο υπονοούμενοι κόμβοι δεν γίνεται να βρίσκονται σε συνεχόμενα επίπεδα.

Στο σχήμα 6.1 παρατηρούμε την κατανομή των υπονοούμενων κόμβων. Οι περισσότεροι κόμβοι βρίσκονται σε μικρά επίπεδα. Οι πλειοψηφία αυτών βρίσκεται κάτω από το 15 επίπεδο. Αυτό σημαίνει ότι δεν υπάρχουν πολλά σύνθετα μονοπάτια συμπερασμού. Ακόμη, από το επίπεδο 29 και πάνω υπάρχουν υπονοούμενοι κόμβοι, όμως, είναι πολύ λίγοι συγκριτικά με τα μικρότερα επίπεδα και γι'αυτό δεν φαίνονται στο διάγραμμα. Για παράδειγμα, στο επίπεδο 29



Σχήμα 6.1: Ιστόγραμμα κατανομής υπονοούμενων κόμβων στον γράφο, ανάλογα με το επίπεδο που βρίσκονται.

υπάρχουν 4789 κόμβοι ενώ στο τελευταίο επίπεδο, στο επίπεδο 39 υπάρχουν μόλις 42 κόμβοι.

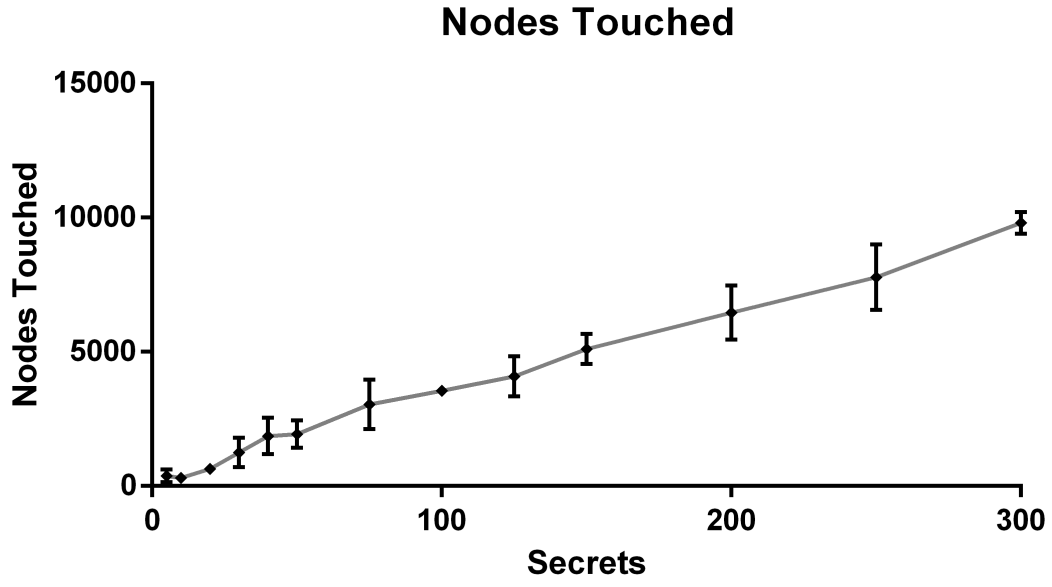
6.2.2 Δομή Γράφου Πειραμάτων

Για καλύτερη κατανόηση του γράφου στα πειράματα που ακολουθούν, αρχικά εξετάσαμε πόσο μεγάλο είναι το δέντρο που σχηματίζεται έχοντας επιλέξει τυχαία τα μυστικά που θέλουμε να κρύψουμε. Με άλλα λόγια, ξεκινώντας από τους κόμβους που έχουν επιλεγεί ως μυστικά, διατρέχουμε το δέντρο και βλέπουμε πόσοι κόμβοι 'αγγίχτηκαν'.

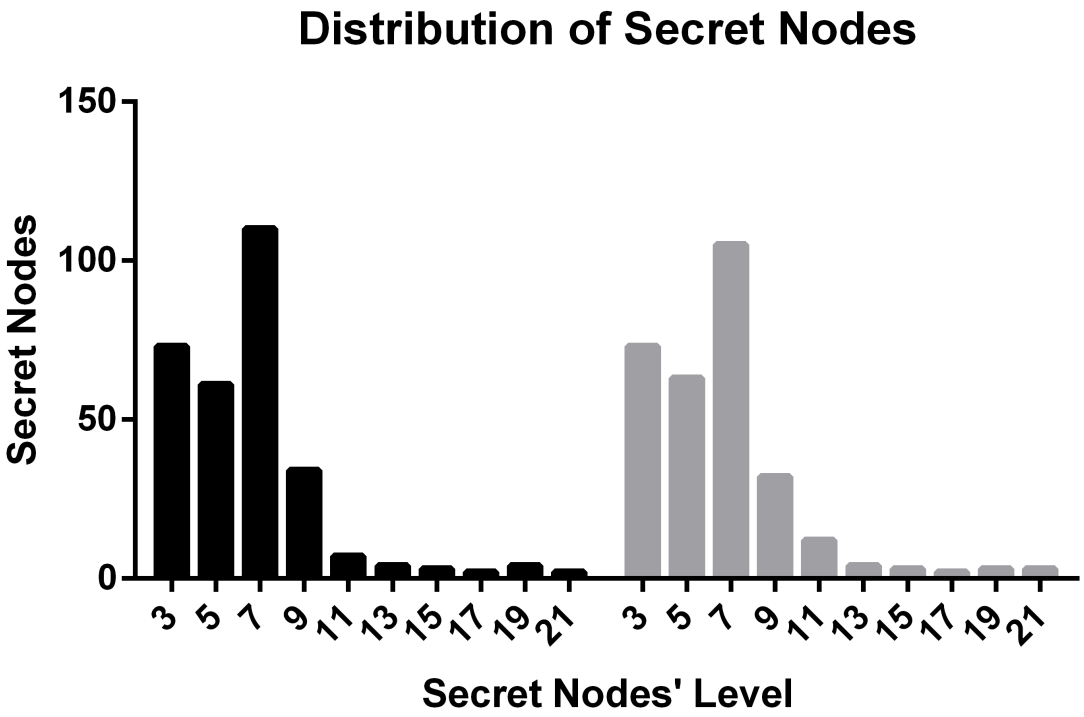
Τα πειράματα που τρέξαμε, και στα οποία βασίζονται όλα τα ακόλουθα αποτελέσματα είναι της ακόλουθης μορφής: Ξεκινήσαμε με 5 μυστικά και σταδιακά αυξάναμε το πλήθος των μυστικών μέχρι 300 μυστικά, όπου η απώλεια πληροφοριών έγινε αρκετά μεγάλη, όπως φαίνεται στη συνέχεια. Για κάθε τιμή, τρέξαμε τουλάχιστον 3 φορές το κάθε πήραμε ώστε να έχουμε μια μέση τιμή, η οποία φαίνεται και σε όλα τα ακόλουθα διαγράμματα μαζί με την τυπική απόκλιση. Ακόμη, η επιλογή αυτών των τιμών είναι αρκετή ώστε να σκιαγραφήσει την απόδοση και την αποτελεσματικότητα του αλγορίθμου, όπως φαίνεται και στη συνέχεια.

Στο διάγραμμα 6.2, βλέπουμε πως αυξάνεται το πλήθος των κόμβων(βασικοί κόμβοι, υπονοούμενοι κόμβοι και V_{op}) που σχηματίζουν τα δέντρα των μυστικών που επιλέγονται καθώς αυξάνεται και το πλήθος των μυστικών που επιλέγονται. Η αύξηση φαίνεται να είναι γραμμική. Για 300 μυστικά έχουμε περίπου 10000 κόμβους, και επιλέγονται 240-270 για να αφαιρεθούν.

Στο διάγραμμα 6.2, πρέπει να προσέξουμε τι ακριβώς είναι οι κόμβοι του δέντρου. Το μέγεθος του δέντρου λαμβάνεται υπόψιν στο πρώτο βήμα του αλγορίθμου, όπου πρέπει να διαλέξουμε από τους υποψήφιους βασικούς κόμβους που επηρεάζουν έναν μυστικό κόμβο που θέλουμε να κρύψουμε. Όμως, στη συνέχεια για να υπολογίσουμε το κόστος των υποψήφιων κόμβων αλλά και αφού επιλέξουμε έναν για να κρύψουμε τους κόμβους που χάνονται, διασχί-



Σχήμα 6.2: Σύνολο κόμβων που χρησιμοποιήθηκαν.



Σχήμα 6.3: Κατανομή μυστικών κόμβων, με 300 μυστικά.

ζουμε ανάποδα τον γράφο. Οπότε, τότε σχηματίζονται διαφορετικά δέντρα. Δεν μετρήσαμε αυτό το μέγεθος διότι αλλάζει κάθε φορά, ανάλογα με τον βασικό κόμβο που εξετάζουμε. Όμως, εκείνο το μέγεθος θα ήταν πολύ μεγαλύτερο όπως μπορούμε να συμπεράνουμε αν παρατηρήσουμε ότι κάθε βασικός κόμβος συμμετέχει περίπου σε 6.5 V_{op} κόμβους ενώ κάθε υπονοούμενος κόμβος σχηματίζεται από περίπου 2.5 κόμβους V_{op}.

Στο σχήμα 6.3 βλέπουμε την κατανομή των μυστικών κόμβων ανά επίπεδο σε 2 πειράματα με 300 μυστικά, τα οποία επιλέχτηκαν τυχαία. Παρατηρούμε ότι τα μυστικά αυτά ακολουθούν την κατανομή και όλων των υπονοούμενων κόμβων, κάτι που είναι λογικό μιας και επιλέχτηκαν τυχαία. Παρόμοια κατανομή έχουν και οι υπόλοιπες κατανομές στα υπόλοιπα πειράματα. Αυτό που μπορούμε να συμπεράνουμε από τις δυο αυτές κατανομές είναι ότι παρ'ότι μοιάζουν αρκετά, έχουν αρκετά διαφορετικά αποτελέσματα. Στις δυο αυτές περιπτώσεις που φαίνονται, υπάρχει μια διαφορά 20% στην απώλεια πληροφοριών. Από αυτό μπορούμε να συμπεράνουμε ότι η κατανομή των μυστικών κόμβων ανά επίπεδο από μόνη της δεν επηρεάζει το αποτέλεσμα της απώλειας πληροφοριών. Άρα, πρέπει να ερευνήσουμε άλλους παράγοντες που μπορεί να επηρεάζουν αυτό το αποτέλεσμα, όπως το πλήθος των V_{op} κόμβων που έχουν οι μυστικοί κόμβοι.

6.2.3 Χρόνος Εκτέλεσης

Μια παράμετρος αξιολόγησης που χρησιμοποιήθηκε στα πειραματικά αποτελέσματα σχετικά με την αποδοτικότητα και την ικανότητα χρήσης του αλγορίθμου με πραγματικά δεδομένα, είναι ο χρόνος εκτέλεσης του. Ο χρόνος εκτέλεσης αποτελείται από δυο συνιστώσες. Αρχικά μετρήσαμε μια φορά τον χρόνο ανάγνωσης των τριών αρχείων εισόδου που είχαν μέγεθος πολλών GB και τα αρχεία αυτά ήταν ίδια για όλα τα πειράματα, οπότε και ο αρχικός χρόνος ανάγνωσης των δεδομένων και αποθήκευση αυτών στις κατάλληλες δομές στην μνήμη του υπολογιστή. Ο χρόνος αυτός φαίνεται στον πίνακα 6.3.

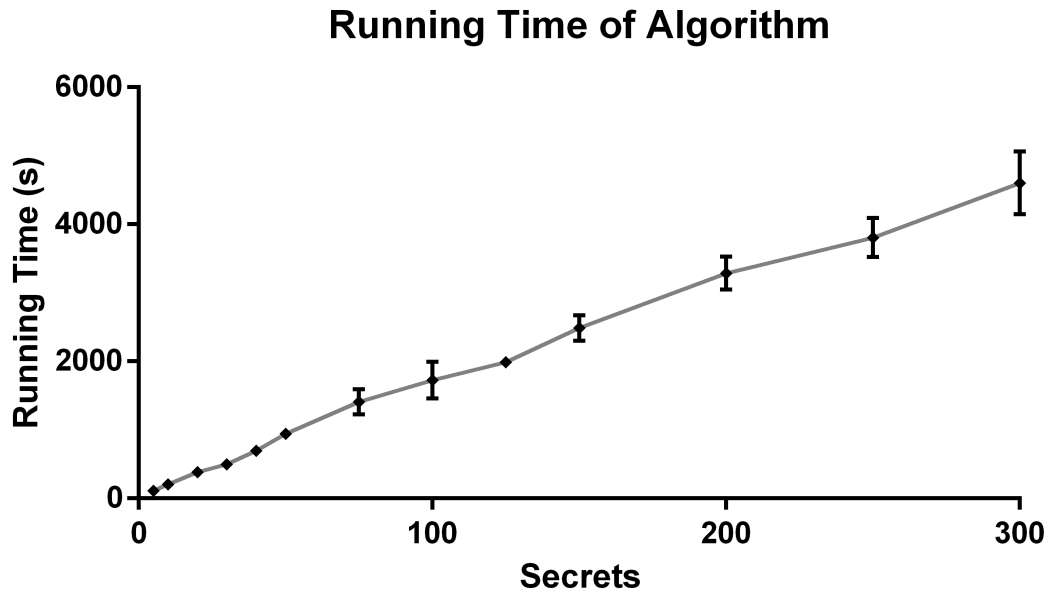
Στη συνέχεια, στο σχήμα 6.4 βλέπουμε τον χρόνο εκτέλεσης σε δευτερόλεπτα για το κυρίως μέρος του αλγορίθμου. Παρατηρούμε ότι ο χρόνος αυξάνεται γραμμικά ως προς το πλήθος των μυστικών οπότε ο αλγόριθμος είναι πρακτικά υλοποιήσιμος. Το ότι δεν είναι πολύ γρήγορος δεν μας απασχολεί ιδιαίτερα λόγω της φύσης της χρησιμότητας του αλγορίθμου που έχει ήδη αναφερθεί. Ο αλγόριθμος εκτελείται μια φορά, πριν τη δημοσίευση των δεδομένων.

Χρόνος ανάγνωσης αρχείων εισόδου (s)	4369

Πίνακας 6.3: Χρόνος ανάγνωσης αρχείων εισόδου

6.2.4 Απώλεια Πληροφοριών

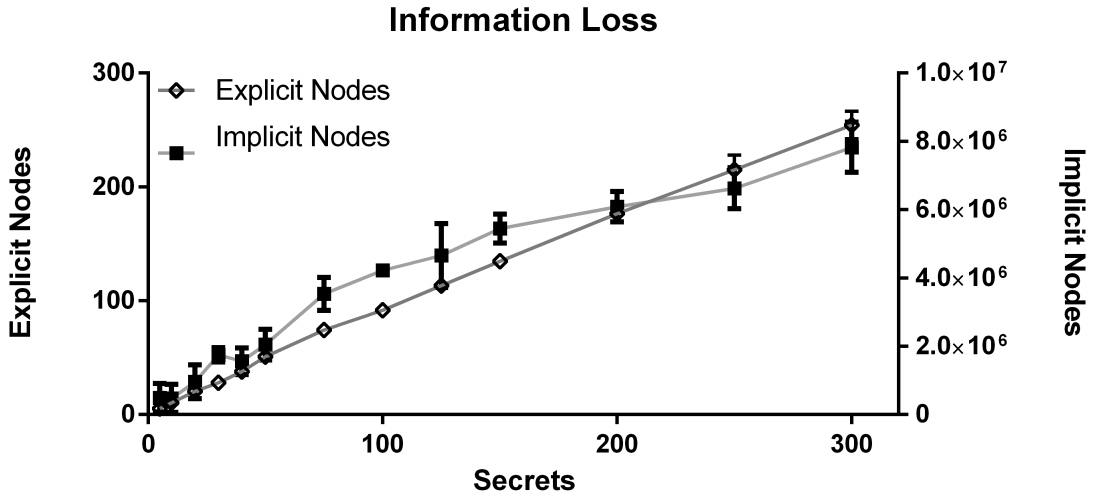
Η πιο σημαντική παράμετρος αξιολόγησης του αλγορίθμου είναι η απώλεια πληροφοριών που επιφέρει ο αλγόριθμος. Τα αποτελέσματα για την απώλεια πληροφοριών που είχαμε στα πειράματα που εκτελέσαμε φαίνονται στο σχήμα 6.5.



Σχήμα 6.4: Χρόνος εκτέλεσης του αλγορίθμου.

Αρχικά, παρατηρούμε ότι το πόσοι βασικοί κόμβοι χρειάζονται για να κρυφτούν όλα τα μυστικά είναι γραμμικό ως προς το πλήθος των κόμβων που κρύβουμε, έχει όμως πτωτική τάση. Για 300 μυστικά χρειαζόμαστε 250 βασικούς κόμβους ενώ για 100 μυστικά σχεδόν 100 βασικούς κόμβους. Αυτό φανερώνει ότι καθώς έχουμε περισσότερα μυστικά, η αφαίρεση ενός βασικού κόμβου βοηθάει στην απόκρυψη περισσότερων του ενός υπονοούμενου κόμβου-μυστικού.

Μια δεύτερη παρατήρηση είναι ότι οι υπονοούμενοι κόμβοι εμφανίζουν και αυτοί αυξητική σχέση σε σχέση με τα μυστικά που πρέπει να κρύψουμε. Για 300 μυστικά, χάνουμε περίπου 23% των υπονοούμενων κόμβων. Αυτό οφείλεται στο ότι κάθε απλός και υπονοούμενος κόμβος συμμετέχει σε πολλούς διαφορετικούς V_{op} κόμβους. Έτσι κάθε αφαίρεση ενός βασικού κόμβου οδηγεί σε μεγάλη απώλεια πληροφοριών. Τέλος, παρατηρούμε ότι η απώλεια των υπονοούμενων κόμβων έχει πτωτική τάση από τα 150 μυστικά και πάνω. Αυτό μπορεί να οφείλεται στα επικαλυπτόμενα μονοπάτια συμπερασμού που έχουν οι βασικοί κόμβοι που επιλέγονται για να αφαιρεθούν.



Σχήμα 6.5: Απώλεια Πληροφοριών ανάλογα με τα μυστικά.

Κεφάλαιο 7

Τεχνικές Λεπτομέρειες

Στο κεφάλαιο αυτό παρουσιάζονται όλες οι τεχνικές λεπτομέρειες σχετικά με την υλοποίηση του προτεινόμενου αλγορίθμου όπως αυτός παρουσιάστηκε στις προηγούμενες ενότητες. Αναλύονται όλες οι δομές που χρησιμοποιήθηκαν, καθώς και όλες οι κλάσεις και οι αντίστοιχες μέθοδοι που αναπτύχθηκαν κατά τη διάρκεια της υλοποίησης τόσο για την παραγωγή του γράφου εξαρτήσεων όσο και για τον αλγόριθμο ανωνυμοποίησης.

7.1 Λεπτομέρειες Υλοποίησης

Για την ανάπτυξη του αλγορίθμου χρησιμοποιήθηκε η αντικειμενοστραφής γλώσσα προγραμματισμού Java. Κατά τη διάρκεια της υλοποίησης του αλγορίθμου αλλά και για την διεξαγωγή ελέγχου χρησιμοποιήθηκε το ολοκληρωμένο περιβάλλον ανάπτυξης Eclipse.

7.1.1 Μορφή Δεδομένων Εισόδου-Εξόδου

Η Οντολογία, η οποία είναι τα αρχικά δεδομένα, βρίσκεται στη μορφή κειμένου, όπου κάθε γραμμή της εισόδου αποτελεί ένα αξίωμα της οντολογίας. Όμως, για την αποτελεσματική υπολογιστική ανάλυση και χρήση των δεδομένων χρειάζεται ένας συντακτικός αναλυτής της οντολογίας για να μπορούμε να έχουμε τα δεδομένα στην κατάλληλη μορφή. Η χρήση αυτού αναλύεται στην επόμενη ενότητα.

Έξοδος είναι ο γράφος εξαρτήσεων ο οποίος βρίσκεται σε 3 διαφορετικά αρχεία κειμένου. Τα 3 αυτά έγγραφα, τα οποία αποτελούν και την είσοδο του αλγορίθμου ανωνυμοποίησης, είναι τα ακόλουθα:

1. *Graph.txt*: Το αρχείο αυτό αποτελείται από γραμμές της μορφής <ID1,ID2> που αντιστοιχούν στις ακμές του γράφου. Κάθε τέτοιο ζεύγος δηλώνει μια ακμή από τον κόμβο ID1 στον κόμβο ID1. Κάθε ID είναι μοναδικό για κάθε κόμβο.

2. *Axioms.txt*: Το αρχείο αυτό αποτελείται από γραμμές της μορφής <Axiom_ID, Type, R_ID, C1_ID, C2_ID> που αντιστοιχούν στο μοναδικό ID του αξιώματος, στον τύπο του, και στα ID των κλάσεων εισόδου του.

3. *Operations.txt*: Το αρχείο αυτό αποτελείται από γραμμές της μορφής <Vop_ID, RULE_ID, {INPUT AXIOM IDs}>, όπου το Vop_ID είναι το μοναδικό ID των V_{op} κόμβων, RULE_ID ποιος κανόνας λογικού συμπερασμού χρησιμοποιήθηκε και INPUT AXIOM IDs αντιστοιχεί στα IDs των αξιωμάτων που χρησιμοποιήθηκαν ως είσοδο του κανόνα για να παραχθεί ο υπονοούμενος κόμβος.

Επιπρόσθετα, ως είσοδο του αλγορίθμου ανωνυμοποίησης χρειάζεται να εισάγουμε το πλήθος των μυστικών που θέλουμε να έχουμε στο γράφο. Τα μυστικά επιλέγονται τυχαία για την πειραματική αξιολόγηση. Στην περίπτωση, όπου ένας ειδικός επίλεγε τα μυστικά ή τα μυστικά αντιστοιχούσαν βάση του ηλεκτρονικό φάκελο ασθενών, τότε αυτή η παράμετρος δεν θα χρειαζόταν.

Έξοδος του αλγορίθμου ανωνυμοποίησης είναι το πλήθος των κόμβων που χάθηκαν κατά την ανωνυμοποίηση μαζί με κάποια στατιστικά, καθώς και το κομμάτι της οντολογίας που απομένει και μπορεί να δημοσιευτεί χωρίς κίνδυνο της ιδιωτικότητας.

7.1.2 Συντακτική Ανάλυση Οντολογίας

Η οντολογία στην είσοδο του αλγορίθμου είναι σε μορφή κειμένου. Στη μορφή αυτή είναι δύσκολο να αναλυθεί αποδοτικά η οντολογία. Ιδιαίτερα δύσκολη είναι η ανάλυση σύνθετων αξιωμάτων που έχουν πολλά φωλιασμένα αξιώματα. Γι'αυτόν τον λόγο χρειάστηκε η χρήση ενός προγράμματος που πραγματοποιεί συντακτική ανάλυση στην οντολογία. Ένας τέτοιος συντακτικός αναλυτής μας χρειάστηκε διότι: υλοποιεί τη συντακτική ανάλυση της οντολογίας, δέχεται ως είσοδο την οντολογία με τη μορφή κειμένου και παράγει το συντακτικό δένδρο που αντιστοιχεί στην ακολουθία εισόδου.

Το πρόγραμμα που χρησιμοποιήσαμε είναι το ANTLR [1], το οποίο είναι ένα πρόγραμμα που παράγει αναλυτές που μπορούν να χρησιμοποιηθούν για να διαβάσουν, να εκτελέσουν ή να μεταφράσουν δομημένο κείμενο. Το εργαλείο αυτό χρησιμοποιείται ευρέως στην ακαδημαϊκή κοινότητα και στην βιομηχανία για να κατασκευάσει όλων των ειδών γλώσσες, εργαλεία και συστήματα. Για παράδειγμα, το ANTLR χρησιμοποιείται από το Twitter για να ανάλυση ερωτημάτων.

Το ANTLR από μια τυπική περιγραφή γλώσσας, που ονομάζεται γραμματική, παράγει έναν αναλυτή για αυτή τη γλώσσα που μπορεί αυτόματα να κατασκευάζει δένδρα συντακτικής ανάλυσης, τα οποία είναι δομές δεδομένων που απεικονίζουν πως μια γραμματική ταιριάζει με την είσοδο. Ακόμη, το ANTLR αυτόματα κατασκευάζει περιπατητές δέντρων που χρησιμοποιούνται για διάσχιση του παραγόμενου δένδρου.

Εμείς, για τον αλγόριθμο, χρειάστηκε να γράψουμε μια γραμματική για οντολογίες σε functional μορφή. Το εργαλείο ANTLR με αυτή την γραμματική κατασκεύασε έναν lexer και έναν parser. Με χρήση του lexer παράχθηκαν τα tokens που αντιστοιχούν στην είσοδο, δηλαδή στην οντολογία. Αυτά τα tokens έγιναν είσοδος του parser, ο οποίος έλεγξε ότι αντιστοιχούν στην γλώσσα που περιγράφεται στην γραμματική που φτιάξαμε και παράγει το δένδρο συντακτικής ανάλυσης. Τέλος, το ANTLR παράγει και έναν walker του δένδρου

αυτού, τον οποίο χρησιμοποιούμε στη συνέχεια για να διασχίσουμε την οντολογία και να βρούμε τη μορφή των αξιωμάτων της.

7.1.3 Δομές Δεδομένων για την Παραγωγή Γράφου Εξαρτήσεων

Κατά την ανάπτυξη του αλγορίθμου για την παραγωγή του γράφου εξαρτήσεων χρησιμοποιήθηκαν οι κατάλληλες δομές δεδομένων ώστε να γίνει πιο αποδοτική η υλοποίηση. Οι δομές που χρησιμοποιήθηκαν περιγράφονται στη συνέχεια.

- **Δεντρική δομή myTree:** Οργανώνει τα δεδομένα της οντολογίας σε μορφή δένδρου. Το ANTLR παράγει ένα δένδρο συντακτικής ανάλυσης, το οποίο όμως δεν επιτρέπει την επεξεργασία του. Έτσι, χρειάζεται αυτή η δομή δεδομένων ώστε να επεξεργαζόμαστε κατάλληλα το δένδρο κατά τα βήματα της αφαίρεσης των Equivalent κλάσεων και κατά το βήμα της κανονικοποίησης των αξιωμάτων. Κάθε κόμβος συνδέεται με άλλους κόμβους του δένδρου είτε στο πιο κάτω επίπεδο (κόμβοι παιδιά) είτε στο πιο πάνω επίπεδο (γονικοί κόμβοι). Δεν υπάρχει περιορισμός στο πόσα παιδιά μπορεί να έχει ένας κόμβος. Η πρόσβαση στους κόμβους γίνεται μέσω δεικτών.

- **Στοίβα stack:** Χρησιμοποιείται για την κατά βάθος διάσχιση της προηγούμενης δεντρικής δομής.

- **Πίνακες κατακερματισμού HashMap<String, Integer> uniqueIDs:** Χρησιμοποιούνται για τη δημιουργία μοναδικών χαρακτηριστικών IDs για τις κλάσεις ή τα αξιώματα. Όταν βρίσκουμε ένα όνομα κλάσης ή ένα αξίωμα, το ψάχνουμε στον πίνακα κατακερματισμού, και αν δεν υπάρχει τότε δημιουργούμε το μοναδικό του χαρακτηριστικό αυξάνοντας έναν μοναδικό μετρητή.

- **Πίνακες κατακερματισμού HashMap<Integer, ArrayList<Tuple> > H, H1, H2, H_P, deltaH:** Χρησιμοποιούνται για την αποθήκευση και επεξεργασία των αξιωμάτων. Κλειδί είναι το μοναδικό χαρακτηριστικό ID, ανάλογα με το είδος του αξιώματος. Η λίστα έχει δείκτες σε όλα τα Tuples που αντιστοιχούν στα αξιώματα και έχουν το ίδιο χαρακτηριστικό ως ID.

- **Λίστα ArrayList<Tuple>:** Αυτές οι λίστες χρησιμοποιούνται για την αποθήκευση και διάσχιση του συνόλου tuples που χρησιμοποιούνται στους πίνακες κατακερματισμού H1, deltaH κτλ.

- **tuple:** Κλάση που χρησιμοποιήθηκε για την αποθήκευση κάθε αξιώματος. Αποτελείται από τα πεδία: *AxiomID, Type, X, Y, Z, dependencies.*

  - AxiomID: το μοναδικό χαρακτηριστικό κάθε αξιώματος.
  - Type: ο τύπος του αξιώματος που είναι ένας αριθμός μεταξύ του 1 και του 4.
  - X, Y, Z: τα 3 αυτά πεδία αποθηκεύουν, αναλόγως με το αξίωμα, τα R_IDs, C_IDs.

– dependencies: δομή για την αποθήκευση του υπο-γράφου για το πως έχει προκύψει κάθε αξίωμα. Αποθηκεύει ποια άλλα αξιώματα χρησιμοποιήθηκαν για την παραγωγή αυτού του κόμβου και με ποιες εισόδους.

7.1.4 Δομές Δεδομένων Αλγορίθμου Ανωνυμοποίησης

Κατά την ανάπτυξη του αλγορίθμου ανωνυμοποίησης χρησιμοποιήθηκαν οι κατάλληλες δομές δεδομένων ώστε να γίνει πιο αποδοτική η υλοποίηση του. Οι δομές που χρησιμοποιήθηκαν περιγράφονται στη συνέχεια.

- **Λίστα ArrayList<Node>:** Αυτές οι λίστες χρησιμοποιούνται για την αποθήκευση και διάσχιση κόμβων που ανήκουν σε μία συγκεκριμένη ομάδα, όπως η λίστα με όλους του implicit κόμβους που πρέπει να εξεταστούν σε κάποιο βήμα του αλγορίθμου.

- **Πίνακας Δεικτών στους κόμβους nodes_array[i]:** Για την υλοποίηση του αλγορίθμου, χρησιμοποιήσαμε έναν πίνακα από δείκτες στους κόμβους του δέντρου. Κάθε κόμβος του δέντρου είναι μια δομή δεδομένων που έχουμε κατασκευάσει ώστε να αποθηκεύονται όλες οι πληροφορίες που χρειάζονται για κάθε κόμβο.

- **Δομή κόμβου node:** η δομή δεδομένων ενός κόμβου αποτελείται από τα εξής πεδία:

  – data: στο πεδίο αυτό αποθηκεύεται το μοναδικό χαρακτηριστικό node ID του κάθε κόμβου. Ο ακέραιος αυτός αντιστοιχεί και στον δείκτη(index) για τη θέση του πίνακα όπου υπάρχουν αποθηκευμένοι οι δείκτες στους κόμβους. Έτσι, έχοντας μόνο τον δείκτη κάποιου συνόλου κόμβων (πχ. τους μυστικούς (secret) κόμβους) μπορούμε γρήγορα να έχουμε πρόσβαση σε όλα τα στοιχεία αυτών.

  – children: αυτό το πεδίο είναι μια λίστα με δείκτες σε όλα τα παιδιά του κάθε κόμβου.

  – parents: αντίστοιχα με το προηγούμενο πεδίο, σε αυτό αποθηκεύονται δείκτες σε όλους τους γονείς των κόμβων από τη δεντρική αναπαράσταση των δεδομένων. Έτσι, είναι εύκολη η διάσχιση και από τους explicit κόμβους (φύλλα στο δένδρο) προς τους μυστικούς κόμβους αλλά και το ανάποδο.

  – addedVops: σε αυτό το πεδίο αποθηκεύεται το πόσους V_{op} κόμβους προσθέτει κάθε βασικός κόμβος. Αυτό το πεδίο χρησιμεύει στο πρώτο βήμα του αλγορίθμου για να βρούμε τους πιθανούς βασικούς κόμβους που μπορεί να κρύψουμε. Το κόστος αυτό υπολογίζεται εκ νέου σε κάθε βήμα του αλγορίθμου με τον τρόπου που έχουμε περιγράψει σε προηγούμενη ενότητα.

  – cost: σε αυτό το πεδίο αποθηκεύεται το κόστος του να αφαιρεθεί ένας βασικός κόμβος. Το κόστος είναι ο αριθμός των υπονοούμενων κόμβων που θα χαθούν με την αφαίρεση του συγκεκριμένου βασικού κόμβου. Το κόστος αυτό υπολογίζεται αναδρομικά όταν χρειάζεται.

– hidden: αυτή είναι μια λογική μεταβλητή όπου δείχνει αν ένας κόμβος είναι κρυμμένος λόγω κάποιας αφαίρεσης explicit κόμβου μέχρι αυτή τη στιγμή στον αλγόριθμο.

7.2 Ανάλυση Κλάσεων για τη Δημιουργία του Γράφου Εξαρτήσεων

Κατά την υλοποίηση του αλγορίθμου για τη δημιουργία του γράφου εξαρτήσεων αναπτύχθηκαν διάφορες κλάσεις προκειμένου να υπάρχει η σωστή αλληλεπίδραση μεταξύ των δομών και των τύπων του προγράμματος. Στη συνέχεια, παρουσιάζονται οι κλάσεις αυτές, τα πεδία αυτών και οι βασικές μέθοδοι τους.

7.2.1 public class Node

Η κλάση αυτή υλοποιεί ως αντικείμενο έναν κόμβο της δομής δεδομένων myTree που χρησιμοποιούμε για να αποθηκεύσουμε και να επεξεργαστούμε το δένδρο συντακτικής ανάλυσης. Η κλάση περιλαμβάνει τα ακόλουθα πεδία και συναρτήσεις.

Πεδία

String data

Συμβολοσειρά που αποθηκεύει τα δεδομένα του κόμβου.

ArrayList<Node> children

Λίστα από δείκτες σε όλους τους κόμβουσ-παιδιά του κόμβου.

Node parent

Δείκτης στον κόμβο του γονιού του κάθε κόμβου.

boolean leaf

Λογική μεταβλητή που δηλώνει αν ο κόμβος είναι φύλλο στο δένδρο ή όχι.

Μέθοδοι

Μέθοδος Node

Κατασκευαστής ενός νέου αντικειμένου. Χρειάζεται ως παραμέτρους τα αρχικά δεδομένα του κόμβου και τον γονικό κόμβο, αν υπάρχει.

Μέθοδος addChild

Προσθέτει τον κόμβο που δέχεται ως όρισμα ως παιδί του κόμβου που την καλεί. Προαιρετικά μπορεί να δεχτεί και ως παράμετρο συγκεκριμένη θέση που θα προσθέσει το παιδί.

Μέθοδος deleteNode

Διαγράφει τον κόμβο, αφαιρώντας τον από τα παιδιά του γονέα του και αφαιρώντας τον

δείκτη προς τον γονέα του.

Μέθοδος updateLeaves

Διασχίζει το δένδρο που σχηματίζεται με ρίζα τον κόμβο που την καλεί και σημειώνει τα φύλλα μέσω της λογικής μεταβλητής leaf. Η μέθοδος αυτή χρειάζεται για τη διόρθωση του δένδρου που επιστρέφει ο συντακτικός αναλυτής.

Μέθοδος isComplex

Ελέγχει αν ένα αξίωμα είναι σύνθετο, δηλαδή IntersectionOf ή SomeValuesFrom.

Μέθοδος getLeftChild

Επιστρέφει το αριστερό παιδί ενός κόμβου.

Μέθοδος print

Τυπώνει το δένδρο με ρίζα τον κόμβο που την κάλεσε.

7.2.2 public class Tuple

Η κλάση αυτή υλοποιεί μια πλειάδα από τα δεδομένα που χρειάζονται για την αποθήκευση ενός αξιώματος. Η κλάση περιλαμβάνει τα ακόλουθα πεδία.

Πεδία

int AxiomID

Ακέραιος αριθμός που είναι το μοναδικό χαρακτηριστικό κάθε αξιώματος.

int Type

Ακέραιος αριθμός που δηλώνει το είδος του αξιώματος. Παίρνει τιμές από 1 ως 4.

int X, int Y, int Z

Ακέραιοι αριθμοί που αντιστοιχούν στα δυο ή τρία πεδία του αξιώματος (R_ID, C_ID). Αν το αξίωμα έχει μόνο δυο πεδία, τότε το Z έχει τιμή -1.

ArrayList<ArrayList<ArrayList<Integer> > > dependencies

Το πεδίο αυτό είναι τρεις φωλιασμένες λίστες που δείχνουν όλες τις εξαρτήσεις κατά την παραγωγή του συγκεκριμένου αξιώματος. Η πρώτη λίστα αποτελείται από 7 λίστες, κάθε μια από τις οποίες αντιστοιχεί σε έναν από τους 7 κανόνες συμπερασμού. Η θέση στη πρώτη λίστα δηλώνει ποιος κανόνας συμπερασμού χρησιμοποιήθηκε για την παραγωγή του αξιώματος. Η κάθε μια από τις επόμενες λίστες είναι το σύνολο των αξιωμάτων που χρησιμοποιήθηκαν για την παραγωγή του νέου υπονοούμενου αξιώματος. Τέλος, η τρίτη λίστα είναι το σύνολο των axiomIDs που χρησιμοποιήθηκαν στον συγκεκριμένο κανόνα συμπερασμού για την παραγωγή του υπονοούμενου αξιώματος. Το σύνολο των *dependencies* όλων των αξιωμάτων αποτελεί το γράφο εξαρτήσεων.
Για παράδειγμα, $dependencies = [[[2,3],[7,3]],[],[],[],[],[],[]]$, σημαίνει ότι το συγκεκριμένο αξίωμα παράχθηκε με 2 διαφορετικούς τρόπους με χρήση του πρώτου κανόνα συμπερασμού.

Στην μια περίπτωση τα αξιώματα 2, 3 αποτέλεσαν είσοδο του πρώτου κανόνα συμπερασμού και στην δεύτερη περίπτωση τα αξιώματα 7, 3. Και στις δυο περιπτώσεις το αποτέλεσμα του κανόνα συμπερασμού ήταν το ίδιο, το υπονοούμενο αξίωμα του tuple αυτού.

7.2.3 Κύρια Συνάρτηση

Η βασική συνάρτηση main() υλοποιεί το κύριο μέρος του αλγορίθμου, συνδέοντας και χρησιμοποιώντας τις επιμέρους συναρτήσεις μεταξύ τους. Οι κύριες λειτουργίες της συνάρτησης είναι:

- Καλεί τον συντακτικό αναλυτή ο οποίος επιστρέφει το δένδρο συντακτικής ανάλυσης για την οντολογία εισόδου. Στη συνέχεια, μεταφέρουμε τα δεδομένα στη δικιά μας δομή δεδομένων myTree, κλαδεύουμε αχρείαστα φύλλα που υπάρχουν από τα επιπλέον βήματα που έχει εκτελέσει ο συντακτικός αναλυτής (αναλύει ένα ένα βήμα που υπήρχε στην γραμματική, ενώ εμείς θέλουμε μόνο τα τελικά tokens που βρίσκει).

- Αφαιρεί τα αξιώματα EquivalentClass και EquivalentProperty και κανονικοποιεί τα υπάρχοντα αξιώματα στο δένδρο, δημιουργώντας και προσθέτοντας κατάλληλα τους κόμβους που χρειάζονται.

- Μεταφέρει τα αξιώματα από τη δεντρική μορφή στους κατάλληλους πίνακες κατακερματισμού, ανάλογα με τον τύπο του αξιώματος. Κατά τη διάρκεια αυτής της μεταφοράς, ορίζει και τα μοναδικά χαρακτηριστικά ID των αξιωμάτων.

- Εκτελεί τον κύριο αλγόριθμο, όπως αυτός έχει περιγραφεί σε προηγούμενη ενότητα. Ιδιαίτερο ενδιαφέρον από άποψη υλοποίησης έχει η προσθήκη των νέων αξιωμάτων στους κατάλληλους πίνακες κατακερματισμού. Χρειάζεται να ελέγξουμε αρχικά αν το ID του νέου αξιώματος υπάρχει ήδη. Αν δεν υπάρχει, τότε απλώς προσθέτουμε το αξίωμα στους πίνακες deltaH, deltaH'. Αν υπάρχει, τότε πρέπει να βρούμε αν το αξίωμα υπάρχει ήδη στον πίνακα deltaH ή στον H, και να ανανεώσουμε τον πίνακα με της εξαρτήσεις, dependencies. Για να βρούμε σε ποιον πίνακα είναι, πρέπει να ψάξουμε με σε όλες τις τούπλες που έχουν το ίδιο C_ID.

- Τέλος, τυπώνει στα κατάλληλα αρχεία τον γράφο εξαρτήσεων.

7.3 Ανάλυση Κλάσεων Αλγορίθμου Ανωνυμοποίησης

Κατά την υλοποίηση του αλγορίθμου ανωνυμοποίησης αναπτύχθηκαν διάφορες κλάσεις για τη σωστή αλληλεπίδραση μεταξύ των δεδομένων του προγράμματος. Στη συνέχεια, παρουσιάζονται οι κλάσεις αυτές, τα πεδία αυτών και οι βασικές μέθοδοι τους.

7.3.1 public class Node

Η κλάση αυτή υλοποιεί ως αντικείμενο έναν κόμβο του γράφου εξαρτήσεων. Η κλάση περιλαμβάνει τα ακόλουθα πεδία και συναρτήσεις.

Πεδία

int data

Ακέραιος αριθμός που αποθηκεύει το χαρακτηριστικό του κόμβου.

ArrayList<Node> children

Λίστα από δείκτες σε όλους τους κόμβουσ-παιδιά του κόμβου.

ArrayList<Node> children

Λίστα από δείκτες σε όλους τους κόμβουσ-γονείς του κόμβου.

int type

Ακέραιος αριθμός που δείχνει το είδος του κόμβου. Τιμή 1 σημαίνει ότι ο κόμβος είναι explicit, τιμή 2 ότι είναι implicit, τιμή 3 ότι είναι V_{op} και τιμή 5 ότι είναι υπονοούμενος και μυστικός, οπότε πρέπει να τον κρύψουμε.

int cost

Ακέραιος αριθμός που δηλώνει το κόστος αφαίρεσης του ενός βασικού κόμβου.

int addedVops

Ακέραιος αριθμός που δηλώνει το πόσους V_{op} κόμβους προσθέτει κάθε βασικός κόμβος.

boolean hidden

Λογική μεταβλητή που δηλώνει αν ο κόμβος έχει κρυφτεί λόγω των βασικών κόμβων που έχουμε αφαιρέσει.

int level

Ακέραιος αριθμός που δηλώνει το επίπεδο που βρίσκεται ο κόμβος στον γράφο. Χρησιμεύει για την εξαγωγή στατιστικών της κατανομής των μυστικών.

Μέθοδοι

Μέθοδος isExplicit

Επιστρέφει θετική αληθοτιμή αν ο κόμβος είναι βασικός (explicit).

Μέθοδος isImplicit

Επιστρέφει θετική αληθοτιμή αν ο κόμβος είναι υπονοούμενος (implicit).

Μέθοδος isVop

Επιστρέφει θετική αληθοτιμή αν ο κόμβος είναι V_{op}.

Μέθοδος isHidden

Επιστρέφει θετική αληθοτιμή αν ο κόμβος έχει κρυφτεί σε κάποιο προηγούμενο βήμα.

Μέθοδος addChild

Προσθέτει τον κόμβο που δέχεται ως όρισμα ως παιδί του κόμβου που την καλεί.

Μέθοδος calculateCost

Υπολογίζει το κόστος αφαίρεσης του βασικού κόμβου. Το κόστος αυτό υπολογίζεται αναδρομικά με τον εξής τρόπο: Το κόστος των βασικών κόμβων ισούται με 1 (αφαίρεση του ίδιου κόμβου) συν το κόστος των V_{op} κόμβων που συμμετέχει.
Οι υπονοούμενοι δεν έχουν κάποιο κόστος, αλλά το κόστος τους βρίσκεται στους V_{op} κόμβους του. Έτσι, το κόστος ενός υπονοούμενου κόμβου ισούται με το κόστος των V_{op} στους οποίους συμμετέχει (παρόμοια λογική με το κόστος των βασικών κόμβων).
Το κόστος των V_{op} κόμβων είναι πιο δύσκολο να υπολογιστεί, γιατί πρέπει να δούμε αν ο κόμβος αυτός είναι κρίσιμος για τον υπονοούμενο στον οποίον συμμετέχει. Δηλαδή, αν ο υπονοούμενος κόμβος δεν παράγεται από κάποιον άλλο V_{op}. Τότε με αφαίρεση του V_{op}, θα αφαιρεθεί και ο υπονοούμενος κόμβος, οπότε το κόστος του V_{op} είναι 1 (για αφαίρεση του υπονοούμενου κόμβου) συν το κόστος του ίδιου (το οποίο είναι το κόστος όλων των V_{op} που συμμετέχει ο ίδιος).

Μέθοδος isCritical

Ελέγχει αν ο V_{op} κόμβος είναι κρίσιμος για τον υπονοούμενο κόμβο που παράγει, δηλαδή κοιτάει αν υπάρχει άλλος V_{op} κόμβος που να παράγει τον ίδιο υπονοούμενο κόμβο και να μην είναι κρυμμένος.

Μέθοδος printNode

Τυπώνει όλα τα στοιχεία του κόμβου.

7.3.2 Κύρια Συνάρτηση

Η βασική συνάρτηση main() υλοποιεί το κύριο μέρος του αλγορίθμου ανωνυμοποίησης, συνδέοντας και χρησιμοποιώντας τις επιμέρους συναρτήσεις μεταξύ τους. Οι κύριες λειτουργίες της συνάρτησης είναι:

- Διαβάζει το γράφο εξαρτήσεων από τα αρχεία εισόδου και ανακατασκευάζει το γράφο εξαρτήσεων σημειώνοντας τι μορφής είναι οι κόμβοι και εξάγοντας διάφορα στατιστικά για τον γράφο.
- Διαβάζει την επιλογή του χρήστη για το πλήθος των μυστικών και επιλέγει τυχαία τα μυστικά (στην περίπτωση των τυχαίων πειραμάτων).
- Εκτελεί τον αλγόριθμο ανωνυμοποίησης, όπως αυτός έχει περιγραφεί σε προηγούμενη ενότητα.
- Τέλος, τυπώνει το πλήθος των κόμβων που χάθηκαν κατά την ανωνυμοποίηση και το σύνολο των δεδομένων που μπορεί να δημοσιευτεί.

Κεφάλαιο 8

Επίλογος

8.1 Σύνοψη και Συμπεράσματα

Ο κατακλυσμός ψηφιακών δεδομένων έχει δημιουργήσει την ανάγκη για δημοσίευση αυτών. Όμως η δημοσίευση δεδομένων όλων των μορφών φέρει και τον κίνδυνο της παραβίασης της ιδιωτικότητας των ατόμων. Σε αυτή την εργασία ακολουθήσαμε μια πρωτότυπη και διαφορετική από τις υπόλοιπες προσέγγισης επίλυσης του προβλήματος της ανωνυμοποίησης δεδομένων με λειτουργικές εξαρτήσεις.

Αρχικά, μελετήσαμε το θεωρητικό υπόβαθρο που ήταν απαραίτητο ώστε να εμβαθύνουμε στη περιοχή της ανωνυμοποίησης δεδομένων. Μελετήσαμε σχετικά με την ιδιωτικότητα, τις βασικές μεθόδους ανωνυμοποίηση δεδομένων και παρουσιάσαμε πως σχετίζονται οι οντολογίες και η συλλογική ανάλυση με αυτή την εργασία. Στη συνέχεια, εμβαθύναμε στους διαφορετικούς τρόπους ανωνυμοποίησης που υπάρχουν στη βιβλιογραφία. Παρουσιάσαμε συνοπτικά τους αλγορίθμους για ανωνυμοποίηση σχεσιακών δεδομένων, δεδομένων συναλλαγών, δεδομένων στον ιστό και ημι-δομημένων δεδομένων και δείξαμε πως αυτές οι δουλειές διαφέρουν από τη προσέγγιση μας.

Στη συνέχεια δώσαμε το κίνητρο που είχαμε για να ασχοληθούμε με το συγκεκριμένο πρόβλημα. Δείχνουμε ότι οι παραδοσιακοί μηχανισμοί πρόσβασης στις βάσεις δεδομένων υπολείπονται ενός αυτόματου τρόπου ανωνυμοποίησης δεδομένων με λειτουργικές εξαρτήσεις και δώσαμε ένα παράδειγμα χρήσης με ιατρικά δεδομένα όπου φαίνεται η επιτακτικότητα επίλυσης του προβλήματος. Με βάση αυτό το παράδειγμα, ορίσαμε το πρόβλημα με έναν φορμαλιστικό τρόπο ώστε να βρούμε στη συνέχεια έναν αποδοτικό τρόπο επίλυσης του.

Με βάση τον φορμαλιστικό ορισμό που δώσαμε, αναπτύξαμε έναν ευριστικό αλγόριθμο που επιτυγχάνει την ανωνυμοποίηση δεδομένων με λειτουργικές εξαρτήσεις που εκφράζονται μέσω μιας οντολογίας. Ο αλγόριθμος βασίζεται στον γράφο εξαρτήσεων για την οντολογία, έναν γράφο που δηλώνει τις διαδρομές συμπερασμού που μπορούν να οδηγήσουν σε διαρροή πληροφοριών. Αναλύσαμε σε βάθος την τεχνική υλοποίηση του αλγορίθμου.

Τέλος, αξιολογήσαμε σε βάθος τον αλγόριθμο μέσω πειραμάτων βασισμένα σε αληθινά ιατρικά δεδομένα. Τα πειράματα αυτά έδειξαν ότι ο αλγόριθμος είναι αποδοτικός χρονικά και ότι εγγυάται την προστασία της ιδιωτικότητας έχοντας μια μικρή απώλεια πληροφοριών.

Συνοψίζοντας, μέσω αυτής της εργασίας στο θέμα της ανωνυμοποίησης κατά τη δημοσίευση δεδομένων με εξαρτήσεις, συμπεραίνουμε τα εξής:

1. Οι ήδη υπάρχοντες μέθοδοι ανωνυμοποίησης δεδομένων που υπάρχουν στα παραδοσιακά πληροφοριακά συστήματα δεν είναι επαρκείς για να ανταποκριθούν σε επιθέσεις που βασίζονται στην εξαγωγή υπονοούμενων πληροφοριών με χρήση των λειτουργικών εξαρτήσεων των δεδομένα. Ακόμη, οι υπάρχοντες τρόποι ανωνυμοποίησης στα πληροφοριακά συστήματα υπολείπονται ενός αυτόματου συστήματος ανωνυμοποίησης.

2. Προτείνουμε μια πρωτότυπη προσέγγιση στην προστασία της ιδιωτικότητας. Ορίζουμε το πρόβλημα με έναν τρόπο που δεν υπάρχει στη βιβλιογραφία, και προτείνουμε έναν ευριστικό αλγόριθμο που επιλύει το πρόβλημα.

3. Ο αλγόριθμος που προτείνουμε, δεν είναι βέλτιστος διότι το πρόβλημα είναι NP-δύσκολο, όμως, όπως δείξαμε μέσω της πειραματικής αξιολόγησης έχει περιορισμένη απώλεια πληροφοριών ώστε να είναι αρκετά αποδοτικός και να έχει πρακτική χρησιμότητα.

8.2 Μελλοντικές Επεκτάσεις

Κατά την εκπόνηση της παρούσας διπλωματικής, αναγνωρίσαμε αρκετά ενδιαφέροντα θέματα για περαιτέρω επέκταση. Μερικά από αυτά είναι τα εξής:

- Εύρεση διαφορετικών ευριστικών μεθόδων και σύγκριση των αποτελεσμάτων και χρόνων εκτέλεσης των διαφορετικών αλγορίθμων. Σε διαφορετικά δεδομένα, πιθανώς να είναι καλύτερες διαφορετικές ευριστικές μέθοδοι.

- Εύρεση του λόγου προσέγγισης του άπληστου αλγορίθμου ανωνυμοποίησης. Μέσω θεωρητικής ανάλυσης της ευριστικής θα μπορούσαμε να βρούμε στη χειρότερη περίπτωση πόσο χειρότερο είναι το αποτελέσματα που πετυχαίνει ο αλγόριθμος από το βέλτιστο.

- Ο αλγόριθμος ανωνυμοποίησης μπορεί να επεκταθεί και σε διαφορετικά μοντέλα επιθέσεων. Μια χρήσιμη επέκταση αφορά τη μελέτη επιθέσεων όπου ο επιτιθέμενος έχει μερική γνώση και της βάσης δεδομένων εκτός από τη δημόσια οντολογία. Σε αυτή την περίπτωση, δεν είναι δυνατή η αφαίρεση ορισμένων βασικών γεγονότων, γιατί θεωρούμε ότι τα γνωρίζει ήδη ο επιτιθέμενος.

- Χρήση του συνόλου δεδομένων MIMIC II, που περιγράφουν πραγματικούς ιατρικούς φακέλους ασθενών. Μπορούμε να βρίσκουμε ποιες κλάσεις της SNOMED CT βρίσκονται στους φακέλους του ασθενή και αντί να διαλέγουμε τα μυστικά που θέλουμε να κρύψουμε τυχαία, να διαλέγαμε με βάση τους ιατρικούς φακέλους των ασθενών. Αυτή η προσέγγιση δείχνει ακόμα παραπάνω την πρακτική χρησιμότητα του αλγορίθμου.

- Πειραματικό έλεγχο του αλγορίθμου σε διαφορετικά πραγματικά ή συνθετικά δεδομένα. Πέρα από τα πραγματικά δεδομένα που χρησιμοποιήσαμε, θα μπορούσαμε να βρούμε και

να δοκιμάσουμε διαφορετικά δεδομένα, τα οποία θα έχουν διαφορετική δομή. Η διαφορετική δομή του γράφου εξαρτήσεων που θα προκύψει, θα οδηγήσει και σε διαφορετικά αποτελέσματα σχετικά με την απώλεια πληροφοριών.

Βιβλιογραφία

[1] Antlr tool. `http://www.antlr.org/`, 2015.

[2] Maurizio Atzori, Francesco Bonchi, Fosca Giannotti και Dino Pedreschi. Anonymity preserving pattern discovery. *The VLDB Journal-The International Journal on Very Large Data Bases*, 17(4):703–727, 2008.

[3] Franz Baader. *The description logic handbook: theory, implementation, and applications*. Cambridge university press, 2003.

[4] Franz Baader και Ulrike Sattler. An overview of tableau algorithms for description logics. *Studia Logica*, 69(1):5–40, 2001.

[5] Jie Bao, Giora Slutzki και Vasant Honavar. Privacy-preserving reasoning on the semanticweb. Στο *Web Intelligence, IEEE/WIC/ACM International Conference on*, σελίδες 791–797. IEEE, 2007.

[6] Michael Barbaro, Tom Zeller και Saul Hansell. A face is exposed for aol searcher no. 4417749. *New York Times*, 9(2008):8Φορ, 2006.

[7] Gabriel Bender, Lucja Kot και Johannes Gehrke. Explainable security for relational databases. Στο *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, σελίδες 1411–1422. ACM, 2014.

[8] Gabriel M Bender, Lucja Kot, Johannes Gehrke και Christoph Koch. Fine-grained disclosure control for app ecosystems. Στο *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*, σελίδες 869–880. ACM, 2013.

[9] Philip Bohannon, Wenfei Fan, Floris Geerts, Xibei Jia και Anastasios Kementsietsidis. Conditional functional dependencies for data cleaning. Στο *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, σελίδες 746–755. IEEE, 2007.

[10] Piero A Bonatti και Luigi Sauro. A confidentiality model for ontologies. Στο *The Semantic Web–ISWC 2013*, σελίδες 17–32. Springer, 2013.

[11] Alexander Brodsky, Csilla Farkas και Sushil Jajodia. Secure databases: Constraints, inference channels, and monitoring disclosures. *Knowledge and Data Engineering, IEEE Transactions on*, 12(6):900–919, 2000.

[12] Peter Buneman, Sanjeev Khanna και Tan Wang-Chiew. Why and where: A characterization of data provenance. Στο *Database Theory•ICDT 2001*, σελίδες 316–330. Springer, 2001.

[13] Jianneng Cao, Panagiotis Karras, Chedy Raïssi και Kian Lee Tan. ρ-uncertainty: inference-proof transaction anonymization. *Proceedings of the VLDB Endowment*, 3(1-2):1033–1044, 2010.

[14] James Cheney, Laura Chiticariu και Wang Chiew Tan. *Provenance in databases: Why, how, and where*. Now Publishers Inc, 2009.

[15] Bernardo Cuenca Grau. Privacy in ontology-based information systems: A pending matter. *Semantic Web*, 1(1):137–141, 2010.

[16] Yingwei Cui, Jennifer Widom και Janet L Wiener. Tracing the lineage of view data in a warehousing environment. *ACM Transactions on Database Systems (TODS)*, 25(2):179–227, 2000.

[17] Cynthia Dwork. Differential privacy. Στο *Encyclopedia of Cryptography and Security*, σελίδες 338–340. Springer, 2011.

[18] Rodolfo Ferrini και Elisa Bertino. Supporting rbac with xacml+ owl. Στο *Proceedings of the 14th ACM symposium on Access control models and technologies*, σελίδες 145–154. ACM, 2009.

[19] Tim Finin, Anupam Joshi, Lalana Kagal, Jianwei Niu, Ravi Sandhu, William Winsborough και Bhavani Thuraisingham. R owl bac: representing role based access control in owl. Στο *Proceedings of the 13th ACM symposium on Access control models and technologies*, σελίδες 73–82. ACM, 2008.

[20] Benjamin Fung, Ke Wang, Rui Chen και Philip S Yu. Privacy-preserving data publishing: A survey of recent developments. *ACM Computing Surveys (CSUR)*, 42(4):14, 2010.

[21] Gabriel Ghinita, Yufei Tao και Panos Kalnis. On the anonymization of sparse high-dimensional data. Στο *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on*, σελίδες 715–724. IEEE, 2008.

[22] Lukasz Golab, Howard Karloff, Flip Korn, Avishek Saha και Divesh Srivastava. Sequential dependencies. *Proceedings of the VLDB Endowment*, 2(1):574–585, 2009.

[23] Bernardo Cuenca Grau και Ian Horrocks. Privacy-preserving query answering in logic-based information systems. Στο *ECAI*, τόμος 178, σελίδες 40–44, 2008.

[24] Bernardo Cuenca Grau, Evgeny Kharlamov, Egor V Kostylev και Dmitriy Zheleznyakov. Controlled query evaluation over owl 2 rl ontologies. Στο *The Semantic Web–ISWC 2013*, σελίδες 49–65. Springer, 2013.

[25] Bernardo Cuenca Grau και Boris Motik. Importing ontologies with hidden content. *Description Logics*, 477, 2009.

[26] Bernardo Cuenca Grau και Boris Motik. Pushing the limits of reasoning over ontologies with hidden content. Στο *KR*, 2010.

[27] Todd J Green, Grigoris Karvounarakis και Val Tannen. Provenance semirings. Στο *Proceedings of the twenty-sixth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, σελίδες 31–40. ACM, 2007.

[28] Mehdi Haddad, Jovan Stevovic, Annamaria Chiasera, Yannis Velegrakis και Mohand Saïd Hacid. Access control for data integration in presence of data dependencies. Στο *Database Systems for Advanced Applications*, σελίδες 203–217. Springer, 2014.

[29] Daniel Kifer και Ashwin Machanavajjhala. No free lunch in data privacy. Στο *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*, σελίδες 193–204. ACM, 2011.

[30] Vladimir Kolovski, James Hendler και Bijan Parsia. Formalizing xacml using defeasible description logics. *University of Maryland, USA, Tech. Rep. TR-233-11*, 2006.

[31] Aleksandra Korolova, Krishnaram Kenthapadi, Nina Mishra και Alexandros Ntoulas. Releasing search queries and clicks privately. Στο *Proceedings of the 18th international conference on World wide web*, σελίδες 171–180. ACM, 2009.

[32] Nick Koudas, Avishek Saha, Divesh Srivastava και Suresh Venkatasubramanian. Metric functional dependencies. Στο *Data Engineering, 2009. ICDE'09. IEEE 25th International Conference on*, σελίδες 1275–1278. IEEE, 2009.

[33] Kristen LeFevre, David J DeWitt και Raghu Ramakrishnan. Incognito: Efficient full-domain k-anonymity. Στο *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, σελίδες 49–60. ACM, 2005.

[34] Kristen LeFevre, David J DeWitt και Raghu Ramakrishnan. Mondrian multidimensional k-anonymity. Στο *Data Engineering, 2006. ICDE'06. Proceedings of the 22nd International Conference on*, σελίδες 25–25. IEEE, 2006.

[35] Ninghui Li, Tiancheng Li και Suresh Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. Στο *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, σελίδες 106–115. IEEE, 2007.

[36] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke και Muthuramakrishnan Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):3, 2007.

[37] Adam Meyerson και Ryan Williams. On the complexity of optimal k-anonymity. Στο *Proceedings of the twenty-third ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, σελίδες 223–228. ACM, 2004.

[38] Mehmet Ercan Nergiz, Chris Clifton και Ahmet Erhan Nergiz. Multirelational k-anonymity. *Knowledge and Data Engineering, IEEE Transactions on*, 21(8):1104–1117, 2009.

[39] HweeHwa Pang, Xuhua Ding και Xiaokui Xiao. Embellishing text search queries to protect user privacy. *Proceedings of the VLDB Endowment*, 3(1-2):598–607, 2010.

[40] Hyoungmin Park και Kyuseok Shim. Approximate algorithms for k-anonymity. Στο *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, σελίδες 67–78. ACM, 2007.

[41] Michael Q Stearns, Colin Price, Kent A Spackman και Amy Y Wang. Snomed clinical terms: overview of the development process and project status. Στο *Proceedings of the AMIA Symposium*, σελίδα 662. American Medical Informatics Association, 2001.

[42] Latanya Sweeney. Uniqueness of simple demographics in the us population. Τεχνική Αναφορά υπ. αρίθμ., Technical report, Carnegie Mellon University, 2000.

[43] Latanya Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.

[44] Jia Tao, Giora Slutzki και Vasant Honavar. Secrecy-preserving query answering for instance checking in\ mathcal {EL}. Στο *Web Reasoning and Rule Systems*, σελίδες 195–203. Springer, 2010.

[45] Manolis Terrovitis, Nikos Mamoulis και Panos Kalnis. Privacy-preserving anonymization of set-valued data. *Proceedings of the VLDB Endowment*, 1(1):115–125, 2008.

[46] Manolis Terrovitis, Nikos Mamoulis και Panos Kalnis. Privacy preservation in the publication of sparse multidimensional data. *Privacy-Aware Knowledge Discovery: Novel Applications and New Techniques*, σελίδα 35, 2010.

[47] Vassilios S Verykios, Ahmed K Elmagarmid, Elisa Bertino, Yücel Saygin και Elena Dasseni. Association rule hiding. *Knowledge and Data Engineering, IEEE Transactions on*, 16(4):434–447, 2004.

[48] L Willenborg και Ton De Waal. Statistical disclosure control in practice. *Lecture Notes in Statistics, vol.¡ b*, 111, 2001.

[49] Xiaokui Xiao και Yufei Tao. Anatomy: Simple and effective privacy preservation. Στο *Proceedings of the 32nd international conference on Very large data bases*, σελίδες 139–150. VLDB Endowment, 2006.

[50] Yabo Xu, Ke Wang, Ada Wai Chee Fu και Philip S Yu. Anonymizing transaction databases for publication. Στο *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, σελίδες 767–775. ACM, 2008.