tkl-Score: A Misuseability Score for Deciding How To Share Data

by

Kalvin Eng

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computing Science

University of Alberta

Abstract

As more and more data is collected, individuals and organizations are beginning to share their collected data to gain valuable insights. In doing so, these data stakeholders must be aware of the kind of impact that releasing data will have. Therefore, the misuseability scores M-Score and L-Severity have been developed to provide a measure of the potential damage to individuals and organizations when sensitive information from a dataset is released. This thesis introduces tkl-Score and its derivative tkl-Score$_{\max}$ which augments M-Score and L-Severity measures by increasing record scores when records are more identifiable in a source table with l-Distinguishing Factor, and also increasing record scores when sensitive attributes are less granular in a source table with l-Distinguishing Factor and t-Distinguishing Factor. In contrast, M-Score and L-Severity account for only record identifiability in a source table with k-Distinguishing Factor. tkl-Score and tkl-Score$_{\max}$ are shown to better characterize the risk of releasing records compared to M-Score and L-Severity due to accounting for sensitive attribute granularity.

Preface

This thesis is an original work by Kalvin Eng. Segments from this thesis have been published in the following literature:

- K. Eng, D. Serrano, E. Stroulia, *et al.*, "(Semi)Automatic Construction of Access-Controlled Web Data Services," in *Proceedings of the 28th Annual International Conference on Computer Science and Software Engineering*, IBM Corp., 2018, pp. 72–80

Acknowledgements

A huge thank you to my supervisor Eleni Stroulia.

I appreciate Jacob Jaremko's interest in data sharing privacy concerns which prompted this research.

A special thanks to Omid Ardakanian and Nidhi Hegde for serving on my defence committee.

And finally, a big thanks to the Service Systems Research Group for providing a collaborative atmosphere.

Detailed acknowledgements can be found here: `https://docs.google.com/document/d/1x1ZATnjeUluMFYkt2CgQ01uDfvx7w4qW-7J369JVjtE/`

Contents

List of Tables

List of Figures

List of Definitions

Chapter 1

Introduction

As data-collection software using systems such as the internet of things (IoT) becomes more ubiquitous, the privacy and sensitivity of collected data is becoming an increasing concern to data stakeholders. Gartner [13] estimates that in 2021 at least 25 billion IoT devices will be deployed. This ever-increasing amount of devices leads to a plethora of data being collected — possibly underutilized due to factors such as the implications of policies governing data collection and management.

Thus, organizations who wish to share data with different organizations to gain more valuable insights need to be more aware of the consequences that can affect the different stakeholders of the data being shared. Unnecessary disclosures of sensitive information by organizations can lead to severe consequences for subjects of the data and the organizations themselves. For example, the personal information of over 50 million Facebook users were unintentionally exposed to Cambridge Analytica and used for political gain leading to backlash against Facebook [15].

Within the healthcare space, interconnected data systems allow for better level of service and treatment as providers can collaborate over decisions and gain additional insight over richer data. For example, a radiologist may come to a series of conclusions on hip dysplasia based on MRI imaging and would like to tell other clinicians about treatment outcomes and effectiveness of treatment. At the same time, in order to have a full report that includes all factors, specific patient attributes may be shared such as gender, birthdate, specific

medical conditions, and historical treatment history. In order to enable these collaborations, strict protocols must be followed to ensure that patient rights and privacy are respected.

Most organizations are wary of distributing data that may infringe on the rights of the subjects that they have collected data from. As a result, organizations such as Microsoft are developing teams to help manage the rights of their users [11]. Furthermore, policies and procedures are also developed and enforced to deal with the potential harm of releasing private or sensitive data.

For example, the Government of Canada has created checklists to ensure data integrity and security for government teams to become more open about the software they develop and data they collect. These checklists, currently known as the "Digital Playbook"[1], strive to create a transparent, digital government.

Moreover, organizations are also developing policies to classify and handle their information. For example, the North Atlantic Treaty Organization (NATO) has developed a classification standard[2] for sharing sensitive information (from most secret to least): cosmic top secret, NATO secret, NATO confidential, NATO restricted.

For many organizations, classification and approval are manual — a delegated team or person must approve the data request before any data is released or accessed. Consequently, there is a need to design systems that can differentiate the sensitivity of data to improve the data sharing process.

Therefore, the question this thesis wishes to address is: How can we quantify the sensitivity of data to guide decision making about sharing data?

In this thesis, a scoring function, known as a misuseability score, is designed to provide a measure of the potential damage to an organization when sensitive information from a dataset is released. The misuseability score takes into account: the sensitivity of values, the identifying information that may be inferred through the release of the data set, and the amount of data. Thus,

[1] https://canada-ca.github.io/digital-playbook-guide-numerique/

[2] https://www.act.nato.int/images/stories/structure/reserve/hqrescomp/nato-security-brief.pdf

the focus of this thesis is threefold:

1. We determine the potential needs of data sharing with regards to privacy and sensitivity in an ecosystem of organizations and people wishing to share their data.

2. This thesis augments the current state-of-the-art misuseability scoring measures by including more factors that can identify an individual, as well as a more stringent process for deriving the sensitivity of data values for score calculation. This new score, *tkl*-Score, can help determine whether or not data may be too sensitive for being released to certain parties by providing a relative measure of misuseability. A derivative of *tkl*-Score, *tkl*-$\text{Score}_{\text{max}}$, is also designed to account for the worst case severity of releasing any one maximum record score from a set of published records.

3. We establish the validity and usability of *tkl*-Score and *tkl*-$\text{Score}_{\text{max}}$ by comparing it against M-Score and L-Severity.

The rest of this thesis is organized as follows:

- In Chapter 2, the concept of a "data ecosystem" is introduced to illustrate how it can fit into the privacy/sensitivity aware paradigm for data access and sharing.

- In Chapter 3, an illustrative example is introduced to demonstrate current anonymity measures in privacy-preserving data publishing (PPDP), and also state-of-the-art misuseability scoring.

- In Chapter 4, a new misuseability score, *tkl*-Score, is introduced using the illustrative example from Chapter 3. Also defined is a derivative of *tkl*-Score, *tkl*-$\text{Score}_{\text{max}}$.

- In Chapter 5, *tkl*-Score and *tkl*-$\text{Score}_{\text{max}}$ is compared against M-Score and L-Severity using cases created from the illustrative example dataset introduced in Chapter 3.

- In Chapter 6, we reflect on the contributions and provide different avenues for using *tkl*-Score.

Chapter 2

Background

In order to understand the broad context of data sensitivity, it is useful to consider the whole "data ecosystem" which includes the relevant *actors* (i.e., all stakeholders with an interest in the data in question), their *roles* (i.e., their purpose and duties), and the actual *resources* (i.e., the data itself). Also included are the *relationships* among *actors*, *roles*, and *resources* defined by how data is collected, used, and shared [28]. As well, the data ecosystem consists of data misuse quantification methodologies to deal with data sharing concerns.

2.1 Actors' Perceptions of Data Sensitivity

Actors, consisting of enterprises, institutions, and individuals, can have varying attitudes and beliefs towards data sharing. Enterprises are organizations such as service providers who provide the necessary infrastructure for collecting and sharing data. Institutions can be organizations that act as data intermediaries or data owners, while individuals can be data owners or data subjects.

Numerous studies have been conducted to understand organizational and individual attitudes and beliefs towards privacy.

Nget et al. [26] perform an online survey as part of personal data market platform and conclude that people are more willing to sell their data when privacy protections are in place and they know how the data will be used and how sensitive it is.

An empirical study performed by Hadar et al. [16] on software developers'

mindsets towards privacy by design finds that developers often have a limited understanding of privacy and believe it is a low priority. These attitudes may be due partly to the fact that organizations do not consider privacy as a top priority. This study suggests that a mindset of privacy has to be perpetuated across all aspects of software development from developers to management in order to design privacy preserving software systems that can protect data privacy.

Another study of software developers on designing privacy minded software systems by Senarath and Arachchilage [36] also comes to a similar conclusion that better guidelines and education on what privacy encompasses is needed.

In an online survey conducted by Bilogrevic and Ortlieb [3], it was found that users' comfort with sharing data was influenced by three contextual factors such as the type of service the data was being shared with, the type of data being shared, and the pre-existing relationships a user has with the third party that the service may be sharing to. Bilogrevic and Ortlieb conclude that to increase user acceptance of data sharing, there needs to be clear communication and transparency of how data is used, more concrete value propositions about the data collected need to be provided, and more controls over data sharing with third parties need to be implemented.

Moreover, a study commissioned by a telecommunications company, Orange, finds that customers recognize that there is value to their data. Because of this recognition, organizations should make clear how customer data is used, allow customers to control what data can be used, and educate the benefits of using the data [22], [23].

From these studies, it is a common theme that education and awareness is a big factor towards user willingness for sharing data. As such, a misuseability score to quantify the severity of potential damage when data is released can help users become more aware of the privacy implications.

2.2 Regulatory Frameworks Around Roles

Roles are defined as "a function played by an *actor* in a data ecosystem" [28]. For example, an enterprise *actor* such as Amazon Web Services[1] has a *role* that provides infrastructure *resources* for hosting and collecting data. While software developers, who are also enterprise *actors*, would have *roles* to create and maintain data collection software that is hosted on an infrastructure. These enterprise *actors* work with institution *actors* who have the *role* of using the software and infrastructure to collect and share data. Individual *actors* are also involved by having *roles* such as being the subjects of data, and having rights to the data that is collected about them.

There are many factors that can govern a *role*. For example, in the case of commodifying data, there has been legal discussions for the implications of sharing and selling personal information based on laws. Bankruptcy and debt can lead to the leak of information as companies are obligated to sell off their assets such as customer information to recuperate and pay off their debts and there are no strict governmental regulations for the distribution of customer data in bankruptcy and debt law [8].

Moreover, Crain [6] argues that more regulation is needed against the commodification of data in order to preserve privacy. Bishop [4] also supports this idea by suggesting that more regulatory action is needed to address the privacy of IoT data collection.

Laws such as the European Union's General Data Protection Regulation (GDPR)[2] are good first steps in protecting consumer privacy as it regulates how data is to be maintained and collected for all citizens of the European Union and is generalized to encompass all data.

In contrast, laxer regulations in the United States have led to unintended data exposure. For example, despite the Health Insurance Portability and Accountability Act (HIPAA) that governs the protection of health informa-

[1] `https://aws.amazon.com/`

[2] `https://ec.europa.eu/commission/priorities/justice-and-fundamental-rights/data-protection/2018-reform-eu-data-protection-rules/eu-data-protection-rules_en`

tion, Google was still able to obtain the medical data of 50 million Americans without asking their permission while allegedly conforming with HIPAA [31]. Therefore, the varying laws surrounding data management influence what *actors* can and cannot do with data.

Consequently, technologies to share data must be developed to help respect these laws, and help *actors* uphold their *roles* that are governed by regulations.

2.3 Technologies for Collecting Resources

Resources are defined as datasets, data-based software, and infrastructure [28]. Data infrastructures work in tandem with data collection systems (often data-based software) to collect and create datasets for distribution. Government and scientific organizations have invested heavily in infrastructure development for open data. CKAN is widely used by governments such as the Canadian Government[3], US government[4], the UK government[5], and Australian government[6] for sharing open data. Similarly, Dataverse[7] is widely used by academic institutions such as Harvard and the University of Alberta to share research data. There are also many other popular platforms to disseminate academic research and results such as SciCrunch[8], Dryad[9], and Figshare[10]. These platforms share a common goal of streamlining collection of data, increasing discoverability of data, and encouraging the reproducibility of data.

Government and public institutions, in order to enhance their transparency and improve society, have been adopting open data plans and policies to release data that can be freely used and repurposed. Moreover, for-profit companies would also like to release their consumer data to third parties such as marketers in order to enhance their earnings. These two purposes have allowed for third parties to provide services based on integrating third-party data while also

[3] `https://open.canada.ca/en`
[4] `http://Data.gov`
[5] `http://data.gov.uk`
[6] `http://data.gov.au`
[7] `https://dataverse.org/`
[8] `https://scicrunch.org/`
[9] `https://datadryad.org/`
[10] `https://figshare.com/`

raising privacy concerns about what data is exposed.

Cities and companies are looking to adopt digital strategies that consider the acquisition of IoT data in order to deliver better services to their inhabitants and clients.[11] Hence, data owners can capitalize on the data produced by their IoT devices by sharing the data collected in a data marketplace that controls payments and the exchange of data between parties. This has motivated the creation of industry approaches, such as the IOTA marketplace[12] and Streamr[13]. In these services, data is collected and delivered to consumers for a fee over a central exchange. However, these data marketplaces lack a uniform security layer that limits the exposure of their information and minimizes the leakage of sensitive data.

For many organizations, it becomes onerous to orchestrate different access control policies, based on the level of trust with users they are sharing with, to maintain the privacy of restricted information. Many datasets contain proprietary and personal information, which fundamentally limits the capability of businesses to publicly distribute data due to the risk of releasing sensitive information to the public, losing consumer trust in the process [9]. Furthermore, the datasets they collect have a variety of structural schemas, syntactic formats, and semantic meanings leading to difficulty when integrating data [28].

Hence, a misuseability score can help by quantifying the sensitivity of information in a dataset and help support decisions for accessing sensitive information.

2.4 Data Usage Relationships

Relationships are formed between *actors* in the data ecosystem based on a common interest or related to an actor's *role* in the data ecosystem [28]. These *relationships* help facilitate the sharing of *resources*. For example, a company may for a relationship with another company to sell data that they have col-

[11] `https://www.networkworld.com/article/3257664/internet-of-things/iot-sensor-as-a-service-run-by-blockchain-is-coming.html`

[12] `https://data.iota.org/`

[13] `https://streamr.network/`

lected. It is important to understand how these *relationships* are formed, and what may be the trade-offs in a relationship.

Acquisti et al. [1] reviews the state of decision-making in privacy and security in the context of "nudges" from a multidisciplinary perspective. Nudges are the aspects that can influence a person's decision to form *relationships* with different *actors* in the data ecosystem. From a privacy and security perspective, understanding nudges are important as improper *relationships* can be formed because of improper nudges that can misinform or mislead an *actor* to believe that having their security and privacy being exposed is a fair trade-off. For instance, consider a user who exposes their birthday publicly online so that they can receive birthday wishes. While a benefit may be receiving attention, there are also potential risks such as a malicious *actor* using this information to gain access to other personal accounts that require a birthdate to confirm an identity.

The influence of forming *relationships* is a complex issue that is multidisciplinary in nature. Economically, there is a trade-off between costs and utility that influences a decision. Legally, certain laws or regulations can also influence how *relationships* are formed. From a behavioral perspective, certain nudges can influence a decision. These nudges can be supported through technical means such as algorithms, heuristics, and persuasive technologies. All these influences can be important factors for *actors* to make decisions when forming *relationships*.

Therefore, a misuseability score for quantifying the sensitivity of data can help nudge *actors* who form *relationships* by providing them a basis for their decisions.

2.5 Quantifying and Mitigating Data Misuse

Scoring data concerns occurs in many different areas of software systems. A brief overview of some scoring systems that can be related to the misuse of data is introduced in this section to illustrate the need for a misuseability score to help make decisions for sharing data.

Data Loss Prevention Scoring

Data loss prevention (DLP) systems monitor network traffic and databases to see how data is accessed and provide a score for administrators to assess the criticality of incidents. DLP systems help identify users that are risky by classifying the behavior of users with a risk score and helps control and manage the flow of information within an organization. They are designed to detect and prevent sensitive and private information from leaking.

Companies such as Digital Guardian[14], ForcePoint[15], Microsoft[16], SecureTrust[17], and Symantec[18] have developed DLP solutions that can produce a risk score based on the activity performed by users on a network or system. For example, when a sensitive file is attempted to be transmitted outside the organization via methods such as email or file upload, the action can be blocked and logged. When a user attempts repeated actions to move sensitive files outside of the organization, a high risk score is assigned to the user. To recognize the sensitive files, the DLP system has policies created by a system administrator to help identify and classify sensitive and private information within files. DLP systems are useful to manage the internal access of sensitive information, but they do not manage how data should be systematically shared external to an organization. Most systems to date consider only scoring after data is accessed or leaked — they do not score based on the potential of leakage and misuseability when giving access – which highlights the need for a misuseability score.

Risk Adaptive Access Control

Scoring mechanisms are a key component in risk adaptive access control models. Users requesting access to certain information are evaluated by a cost function. If they lie under a given threshold, they are granted access. This

[14] https://digitalguardian.com/

[15] https://www.forcepoint.com/product/dlp-data-loss-prevention

[16] https://docs.microsoft.com/en-us/microsoft-365/compliance/data-loss-prevention-policies

[17] https://securetrust.com/solutions/compliance-technologies/data-loss-prevention/

[18] https://www.symantec.com/products/data-loss-prevention

differs from traditional access control models that have predefined policies set for granting access and can be more permissive. Chen and Crampton [5] propose cost functions for risk that deal with: user trustworthiness, the degree of competence of a user with respect to a particular user-role assignment, and the degree of appropriateness of a permission-role assignment for a given role. Bijon et al. [2] propose a risk based access control framework that incorporates quantified risk for granting access involving thresholds that can be calculated by factors such as attributes, purpose, and situational factors. Most risk adaptive access control systems grant access to resources using decisions made automatically in real-time. This can lead to access that is too permissive allowing users to use resources that were never intended to be accessed. Nevertheless, the use of a cost function in access control models draws similar parallels to misuseability scoring for deciding on how data should be shared. These parallels can be seen in Harel et al.'s [17] proposed access control model mechanism using the misuseability score, M-Score, to regulate user access to sensitive data in relational databases.

Vulnerability Scoring

The Common Vulnerability Scoring System (CVSS) helps to characterize a software vulnerability and create a numerical score to quantify its severity.[19] CVSS relies on a base metric group that determines how a vulnerability can be exploited and the impacts of the vulnerability [10]. Moreover, this base metric can also be impacted by temporal, and environmental metrics. Temporal metrics consider how a vulnerability may change over time (e.g. if a security patch is released, score is reduced), whereas environmental metrics consider where the vulnerability is located (e.g. the vulnerability requires root access before executing and therefore risk is reduced). The CVSS can be translated to a qualitative scale (none, low, medium, high, critical) in order to better understand the severity of a vulnerability. This also helps organizations to make decisions on how to prioritize and fix vulnerabilities. The use of CVSS to quantify the severity of software vulnerabilities draws similar parallels to mis-

[19] `https://www.first.org/cvss/`

useability scoring and highlights the need for a misuseability score to quantify the severity of data being shared.

2.6 Summary

In this chapter, we establish the need for a misuseability score to help share data within a "data ecosystem", and also present a brief overview of scoring methodologies related with data sharing concerns. In the next chapter, we explore anonymity measures and their incorporation into misuseability scores.

Chapter 3

Analyzing Data Sensitivity

The value and protection of data has been the subject of interdisciplinary research and several solutions have been proposed to manage data privacy and security. Specifically in the computing sciences, substantial research efforts are dedicated to the anonymization of data to prevent the leakage of private or sensitive information when data is released, in what is known as privacy-preserving data publishing (PPDP).

This chapter reviews the PPDP concepts of k-anonymity which provides a measure of re-identifiability, and l-diversity, and t-closeness which provide a measure attribute similarity. Also reviewed are the current state-of-the-art misuseability scores M-Score, and L-Severity, both of which use a modified notion of k-anonymity to estimate the potential of misuse when portions of a dataset are released.

3.1 General Definitions and Illustrative Example

To illustrate data sensitivity concepts Table 3.1, an example dataset, is introduced below. This table is similar to the example used by Vavilis et al. [39] to demonstrate how M-Score and L-Severity is calculated for an electronic health records database. To create more robust scenarios, additional columns "Age" and "Initial Diagnosis" are added to the table demonstrate how continuous values and repeated attributes are handled, and the values in the table are changed.

Table 3.1: An illustrative example source table created to model a table of a patient health record database.

id	Job	City	Gender	Disease	Medication	Age	Initial Diagnosis
0	Lawyer	Calgary	Male	H1N1	Tamiflu	27	Flu
1	Lawyer	Calgary	Female	H1N1	Antibiotics	19	Flu
2	Lawyer	Calgary	Female	Flu	Antibiotics	23	Migraine
3	Lawyer	Edmonton	Male	HIV	ARV	49	HIV
4	Lawyer	Edmonton	Male	Hypertension	Statin	70	Hypertension
5	Lawyer	Edmonton	Female	Flu	Paracetamol	29	Migraine
6	Lawyer	Edmonton	Female	Flu	Paracetamol	29	Migraine

Table 3.1 represents the medical data of an EHR (electronic health records) system where each row in the table represents a patient record. In this table, there are no "identifier" attributes, i.e., attributes which can directly link the record of a patient to their real-world identity, such as their full name for example.

The attributes of a table can be defined as one of the following:

Definition 1 (Identifier Attribute)**.** An *identifier attribute* is information which can directly link an individual to their identity (e.g. full name).

Definition 2 (Quasi-identifier Attribute)**.** A *quasi-identifier attribute* is information that is not a unique identifier, but when combined with other information partially reveals an individual's identity. For example when "city of residence", and "gender" attributes are common among two different datasets, the datasets can be merged on these quasi-identifier attributes to reveal identifying information such as an individual's name present only in one of the datasets.

Definition 3 (Sensitive Attribute)**.** A *sensitive attribute* is information that should not be exposed publicly (e.g. health condition). The release of the information has the potential to harm an individual such as damaging their reputation.

Definition 4 (Insensitive Attribute)**.** An *insensitive attribute* is information that is considered insignificant and can be ignored (e.g. number used to signify row number of database).

In addition, the two types of tables to analyze data sensitivity can be defined as follows:

Definition 5 (Source Table)**.** A *source table*, also referred to as a dataset, is a collection of records with one or more kinds of attributes (identifier, quasi-identifier, sensitive, or insensitive).

Definition 6 (Published Table)**.** A *published table* is a subset of a source table.

3.1.1 Scenarios

Consider the following scenarios, where there are many stakeholders, such as doctors and clinical researchers, who wish to access Table 3.1 but require different amounts and parts of the data.

1. **Scenario 1:** Before data is released to an outside provider, a doctor would like to know how misuseable the data may be. They wish to share the results of their diagnoses for patients in records 3, 4, 5, and 6 in Table 3.1 to discuss patient treatment strategies with another doctor practicing in a different clinic.

2. **Scenario 2:** A researcher wishes to access all records of the health records system in order to better understand the epidemiology of diseases in various regions. Before the data is released, the data controllers of Table 3.1 would like to know the misuseability potential if all rows of the table were to be released.

In both scenarios, the stakeholders must consider how the data can be misused as a factor to decide whether the data should be shared. Scenario 1 is intended to demonstrate how some records can be more sensitive than others based on their attributes when only releasing a subset of the data. Scenario 2 is intended to illustrate that the sensitivity of the scores is also associated with the amount of data released in misuseability scores.

3.2 Privacy-Preserving Data Publishing

Fung et al. [12] define privacy-preserving data publishing (PPDP) as "methods and tools for publishing useful information while preserving data privacy". Two of the most common attacks against privacy by an adversary are identity disclosure, and attribute disclosure.

1. **Identity disclosure** occurs when an individual can be identified from a group of records based on a set of attributes. For example, if we group together Table 3.1 using the quasi-identifier attributes *Job*, *City*, and *Gender* as seen in Table 3.2, the identity of record 0 can be easily identified in the dataset since it is the only record in the quasi-identifier grouping. Thus, we can identify that a 27-year-old male lawyer living in Calgary has H1N1 and is taking Tamiflu. When record 0 is linked with another external dataset, the identity of the individual could be further revealed if identifying attributes such as name are in the external dataset.

2. **Attribute disclosure** occurs when an individual's sensitive attributes become known because they are shared among the released table records even though they cannot be specifically identified in a group of records. For instance, in records 5 and 6 of Table 3.1 all attributes are the same. As a result, an adversary can know for certain that all female lawyers in Edmonton were initially diagnosed as having a migraine but were later found to have the flu and were treated with paracetamol. This example makes it clear that, although an individual's identity may not be distinguished between two people, there is no guarantee that other information will not be inferred.

To estimate the risk of identity disclosure on a dataset, k-anonymity is commonly used to determine how "unique" individual records are in a table. To estimate the risk of attribute disclosure attacks, l-diversity and t-closeness are commonly used as measures of how "identifying" sensitive attributes are in a table.

3.2.1 *k*-anonymity

k-anonymity is a metric of how distinguishable an individual record is in a dataset, based on common quasi-identifier attribute groupings. These groupings can be defined as an equivalence class:

Definition 7 (Equivalence Class)**.** An equivalence class is a set of records that have the same values for quasi-identifiers attributes [21].

If an equivalence class is small, it can easily be combined with a different dataset containing the same quasi-identifiers to uniquely identify an individual. Therefore, the k-anonymity measure is developed to quantify the maximal risk of linking outside records to identify an individual, and is defined as follows:

Definition 8 (k-anonymity)**.** A dataset is k-anonymous if every equivalence class in the dataset has at least k records that are indistinguishable. In other words, the information for each record of a table cannot be distinguished from at least $k-1$ other records [37].

We can see how k-anonymity works in Table 3.2, where equivalence classes are formed based on the quasi-identifier attributes: *Job*, *City*, and *Gender*. The table is 1-anonymous, as the record 0 has an equivalence class of size one. However, if record 0 is removed, then the table becomes 2-anonymous as an individual must be discerned from at least an equivalence class of size two.

With a table having $k = 1$, an identity can be positively identified as there contains at least one equivalence class of size one that has quasi-identifiers that can be linked to a specific identity. However, If $k > 1$, then it becomes more difficult to distinguish a single identity as there are more possibilities of how the quasi-identifiers in an equivalence class can be linked to an identity.

It should be noted that k-anonymity does not consider attribute disclosure attacks. Although an individual's identity may not be distinguished from an equivalence class, k-anonymity cannot guarantee that other sensitive information will not be inferred as evidenced by the similar sensitive attribute values of records 5 and 6 in Table 3.2. To account for this shortcoming, the measures l-diversity and t-closeness were developed.

Table 3.2: Equivalence classes of Table 3.1 grouped based on rows with the common values of the quasi-identifier attributes *Job*, *City*, and *Gender*

id	Job	City	Gender	Disease	Medication	Age	Initial Diagnosis
0	Lawyer	Calgary	Male	H1N1	Tamiflu	27	Flu
1	Lawyer	Calgary	Female	H1N1	Antibiotics	19	Flu
2	Lawyer	Calgary	Female	Flu	Antibiotics	23	Migraine
3	Lawyer	Edmonton	Male	HIV	ARV	49	HIV
4	Lawyer	Edmonton	Male	Hypertension	Statin	70	Hypertension
5	Lawyer	Edmonton	Female	Flu	Paracetamol	29	Migraine
6	Lawyer	Edmonton	Female	Flu	Paracetamol	29	Migraine

3.2.2 *l*-diversity

l-diversity is a measure focused on how "well-represented" sensitive attributes are in a dataset, and can be used in conjunction with k-anonymity to account for both identity and attribute disclosure.

Definition 9 (l-diversity)**.** A table is l-diverse if every equivalence class in the table contains at least l distinct values for a sensitive attribute. [24].

The definition of "well-represented" could have many different interpretations including distinctness of sensitive attribute values, entropy of equivalence classes, and frequency of appearance of sensitive attribute values. This thesis considers "well-represented" to be how distinct sensitive attribute values are.

For a basic example of l-diversity, consider the equivalence classes in Table 3.3. The l-diversity of the equivalence class containing the records with id 1 and 2 would be 2-diverse since there are two unique sensitive attribute values: "H1N1", and "Flu". The equivalence class containing the records with id 3 and 4 would also be 2-diverse, while the remaining two equivalence classes would be 1-diverse as they only contain one unique sensitive attribute value. We can then say that Table 3.3 is 1-diverse as that is the smallest l-diversity among all equivalence classes.

Table 3.3: Subset of Table 3.2 and with *Disease* as the only sensitive attribute.

id	Job	City	Gender	Disease
0	Lawyer	Calgary	Male	H1N1
1	Lawyer	Calgary	Female	H1N1
2	Lawyer	Calgary	Female	Flu
3	Lawyer	Edmonton	Male	HIV
4	Lawyer	Edmonton	Male	Hypertension
5	Lawyer	Edmonton	Female	Flu
6	Lawyer	Edmonton	Female	Flu

It is important to note that the above example only considers a single sensitive attribute of an equivalence class. To account for multiple sensitive attributes in a table, Machanavajjhala et al. [24] introduce the notion of multi-attribute l-diversity for the case when multiple sensitive attributes are present in a table:

Definition 10 (Multi-attribute l-diversity)**.** Let T be a table with nonsensitive attributes $Q_1, ..., Q_{m_1}$ and sensitive attributes $S_1, ..., S_{m_2}$. If for all iterations $i = 1...m_2$, the table T is l-diverse when S_i is treated as the sole sensitive attribute and $\{Q_1, ..., Q_{m_1}, S_1, ..., S_{i-1}, S_{i+1}, ..., S_{m_2}\}$ is treated as the "quasi-identifiers" [24].

To illustrate multi-attribute l-diversity, a smaller Table 3.4 of records 1 and 2 from Table 3.1 is used. In Table 3.4, we consider *Job*, *City*, and *Gender* to be the quasi-identifier attributes, while *Disease* and *Medication* are the sensitive attributes. For multi-attribute l-diversity we will have to consider two different groupings to form "equivalence classes": *(Job, City, Gender, Disease)* with *Medication* as the sensitive attribute, and *(Job, City, Gender, Medication)* with *Disease* as the sensitive attribute. The *(Job, City, Gender, Disease)* grouping is considered 1-diverse as the rows are separate groups, while the *(Job, City, Gender, Medication)* is considered 2-diverse as the *Disease* attribute contains the unique values "H1N1" and "Flu". Therefore, only 1-diversity holds for Table 3.4.

A limitation of the l-diversity metric is that it only accounts for the frequency of attribute values within an equivalence class and not throughout the

Table 3.4: Subset of illustrative example Table 3.1 to illustrate multi-attribute l-diversity using the quasi-identifiers *Job*, *City*, *Gender*, and the sensitive attributes *Disease*, and *Medication*.

id	Job	City	Gender	Disease	Medication
1	Lawyer	Calgary	Female	H1N1	Antibiotics
2	Lawyer	Calgary	Female	Flu	Antibiotics

complete table. This shortcoming motivated the development of t-closeness, which is a measure designed to account for the distribution of values within equivalence classes and the whole table.

3.2.3 *t*-closeness

t-closeness measures the frequency of attributes within equivalence classes and throughout the whole table by comparing their distributions for similarity using the distance between them.

Definition 11 (t-closeness)**.** An equivalence class is said to have t-closeness if the distance between the distribution of sensitive attribute values in this equivalence class and the distribution of the sensitive attribute values in the whole table is no more than a threshold t. A table is said to have t-closeness if all equivalence classes have t-closeness [21].

To compare the distance between distributions, a number of methods can be used such as variational distance, Kullback-Leibler distance [20], and Earth mover's distance (Wasserstein metric) [14]. Li et al. [21] suggest using earth mover's distance for distributions of continuous attributes, and variational distance for distributions of discrete attributes.

Earth mover's distance provides a calculation of the distance between two distributions that determines the minimal cost of transforming one distribution into the other. The earth mover's distance equation is the following where P and Q denote the distribution:

$$\text{Earth Mover's Distance} = E(P,Q) = \frac{1}{m-1}\sum_{i=1}^{m}\left|\sum_{j=1}^{i}(p_j - q_j)\right| \tag{3.1}$$

Variational distance is a simplification of earth mover's distance that assumes the distance between any two values in a distribution is 1. The variational distance equation is the following where P and Q denote the distribution:

$$\text{Variational Distance} = E'(P,Q) = \frac{1}{2}\sum_{i=1}^{m}|p_i - q_i| \tag{3.2}$$

It should be noted that t is bounded to be in the domain of $[0,1]$ based on the assumption that all distances in the earth mover's distance are in the domain of $[0,1]$. If $t = 1$, it means that the attribute distribution between the equivalence class and global table distributions are not similar and an attribute disclosure attack is very likely. As opposed to $t = 0$, which means that the distributions are the same and the risk of an attribute disclosure is low. Therefore, it is ideal to achieve a lower t in a table.

To illustrate how t-closeness is calculated, let us refer back to Table 3.2 and assume the sensitive attribute columns to be *Disease*, *Medication*, *Age*, and *Initial Diagnosis*. The t-closeness then needs to be found for each sensitive attribute independently.

To compute the t-closeness of sensitive attribute *Disease*, the global distribution will be $Q = \{H1N1, Flu, HIV, Hypertension\}$ containing the frequencies of the attribute values globally. Four other distributions will also be formed based on the frequencies of attribute values within the four equivalence classes found in Table 3.2: $P_0 = \{H1N1\}$, $P_1 = \{H1N1, Flu\}$, $P_2 = \{HIV, Hypertension\}$, and $P_3 = \{Flu\}$.

The distributions are then determined to be: $Q = \{\frac{2}{7}, \frac{3}{7}, \frac{1}{7}, \frac{1}{7}\}$, $P_0 = \{\frac{1}{1}, \frac{0}{1}, \frac{0}{1}, \frac{0}{1}\}$, $P_1 = \{\frac{1}{2}, \frac{1}{2}, \frac{0}{2}, \frac{0}{2}\}$, $P_2 = \{\frac{0}{2}, \frac{0}{2}, \frac{1}{2}, \frac{1}{2}\}$, and $P_3 = \{\frac{0}{2}, \frac{2}{2}, \frac{0}{2}, \frac{0}{2}\}$.

Next, the distribution distance between all the equivalence classes and the global distribution can be calculated using Equation 3.2: $E'(Q, P_0) = 0.7143$, $E'(Q, P_1) = 0.2857$, $E'(Q, P_2) = 0.7143$, and $E'(Q, P_3) = 0.5714$. From the calculations, the t value threshold for Disease is therefore determined be 0.7143 as that is the largest distance.

If the process for calculating t-closeness is continued for the other sensitive attributes, the following t values are obtained: *Medication* as 0.8571, *Age* as

0.2380, and *Initial Diagnosis* as 0.7143. It should be noted that *Age* is calculated using earth mover's distance instead of variational distance since it is continuous. Given all the threshold values obtained for each sensitive attribute in the table, we use the maximum as the final t value of the table, which is 0.8571. Ideally, if the table had the sensitive attribute values distributed similarly among the equivalence class and whole table, the t value would be lower and closer to 0.

t-closeness should be considered in conjunction with k-anonymity and l-diversity, since a low t-closeness does not necessarily guarantee a reasonable k-anonymity or l-diversity measure. t-closeness only accounts for the distribution of attribute values in a table but not consider how identifiable an individual is based on quasi-identifiers, and also how identifiable sensitive attributes may be within an equivalence class.

3.3 Data Misuseability Scores

All the above PPDP measures consider the distribution of sensitive attribute values and identifiability of individuals to determine anonymity. Furthermore, they assume that all the different sensitive attribute values are equally sensitive, and they do not take into account the quantity of data. To address these issues, M-Score and L-Severity are developed as misuse metrics to quantify the severity of a dataset release based on the amount of data, the differing sensitivities of sensitive attribute values, and the identifiability of records.

There is a need for misuse measures in addition to PPDP measures since some records in a table may have the same k-anonymity value, but the sensitive values may make some the records more valuable than others. For example, in Table 3.5 when forming equivalence classes based on the same *Job*, *City*, and *Gender* quasi-identifier attribute values, they are found to have 2-anonymity. However, it can be argued that releasing information about "HIV" may be more severe than releasing information about "Flu" or "Migraine". Therefore, M-Score and L-Severity are developed to account for the sensitivities of data attributes in a table.

Furthermore, PPDP measures also do not account for the amount of data in a table. Intuitively, if more data is released from the table, the more severe a data release should be as more information is accessed.

Table 3.5: A subset of Table 3.1 containing rows 3,4,5,6 with two equivalence classes to illustrate how one class may contain more sensitive attributes than the other.

id	Job	City	Gender	Disease	Medication	Age	Initial Diagnosis
3	Lawyer	Edmonton	Male	HIV	ARV	49	HIV
4	Lawyer	Edmonton	Male	Hypertension	Statin	70	Hypertension
5	Lawyer	Edmonton	Female	Flu	Paracetamol	29	Migraine
6	Lawyer	Edmonton	Female	Flu	Paracetamol	29	Migraine

3.3.1 *M*-Score

M-Score, proposed by Harel et al. [18], is the first metric designed specifically for identifying the impacts of a data release. It is a score for tabular data that quantifies the ability of a user to maliciously exploit exposed data, taking into account the anonymity of individuals in a dataset and the sensitivity of the data attribute values.

The process of calculating the *M*-Score for an exposed dataset, known as a published table, consists of two steps: (1) eliciting weights for sensitive attribute values, and (2) calculating the *M*-Score for the published table.

(1) Eliciting Weights of Sensitive Attribute Values There are several methods for eliciting sensitive value weights from domain experts, but the authors of *M*-Score argue that the Analytic Hierarchy Process (AHP) elicits the best results for discretized data.

AHP involves a pairwise comparison between different sensitive attributes in order to determine their sensitivity weights relative to each other. This helps a domain expert systematically account for different contexts of sensitive attributes within a table by deriving weights in the domain of $[0, 1]$ where a higher weight implies a more sensitive value.

A three-level tree model is defined to perform pairwise comparisons. In the top-level, it consists of a root node with a local weight of 1. In the second

level there are children of the root node with sensitive attributes as each node. While the third level contains children of the sensitive attribute nodes with discretized values of the sensitive attributes as nodes. For example, the three-level tree model of Table 3.1 can be seen in Figure 3.1.

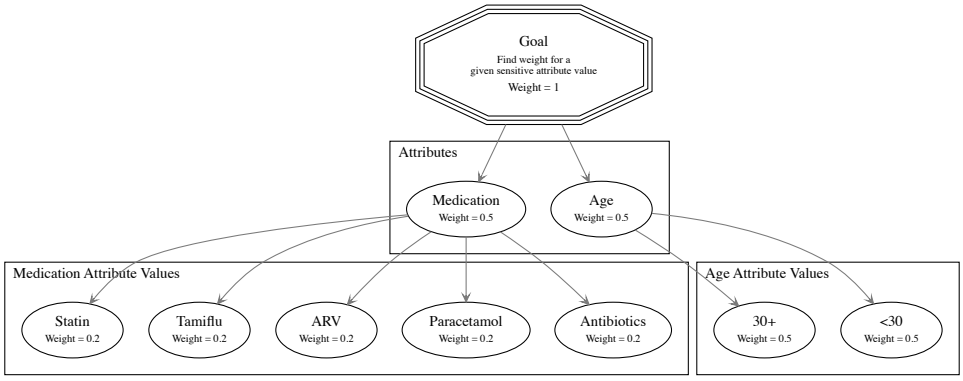


Figure 3.1: Three-level tree model of Table 3.1 assuming that "Medication", and "Age" are the sensitive attributes used for AHP.

With a three-level tree model, pairwise comparisons are done by a domain expert following the process outlined in Appendix A.1 to assign local weights (known as priorities) to each node in the second and third levels. The final weights of the sensitive attribute values can then be calculated by multiplying the priorities of each node in the pathway to the sensitive attribute value node.

For example, to calculate the weight for the sensitive attribute value "Statin", each node in the pathway to "Statin" are multiplied as follows: priority(Statin)$\times$ priority(Medication) $\times$ priority(Goal) $= 0.2 \times 0.5 \times 1 = 0.1$.

(2) M-Score of the Published Table Given the weights of the sensitive attribute values, a published table, and a source table, M-Score can be calculated. To begin, each record of a published table is given a record score as follows:

$$RS_{M_r} = \frac{\min(1, \sum_{A_{S_i} \in r} \text{weight}(A_{S_i}))}{DF_{k_r}} \tag{3.3}$$

A record score RS_{M_r} of the rth record of a published table is the sum of each ith sensitive attribute value weight of the record minimized to 1 divided by the k-Distinguishing Factor DF_{k_r} of the rth record.

The k-Distinguishing Factor is a measure dependent on comparing the published table to the source table. It provides a quantifiable value of how easily an individual can be identified based on the uniqueness of records in a "lookup table". The "lookup table" is collection of records related to a population that can identify an individual. Since such a collection is not easily acquired, the "lookup table" is approximated to be the source table. It is based on the k-anonymity measure (Definition 8), and is defined as follows:

Definition 12 (k-Distinguishing Factor)**.** The k-Distinguishing Factor of a record in a published table is the size of the equivalence class in the source table that contains the record in the published table. If there are no quasi-identifiers to form an equivalence class, then the k-Distinguishing Factor of a record is the size of the published table.

k-Distinguishing Factor is meant to account for how distinguishable an exposed record is when it is published from the source table, and helps to differentiate the records by their identifiability — when the k-Distinguishing Factor is smaller, the record is more identifiable and the record score therefore becomes larger.

To calculate the M-Score of a published table, the maximal record score is taken and multiplied with the number of records in a published table.

Definition 13 (M-Score)**.** Given a table with n records, the table's M-Score is then: $M\text{-Score} = n^{\frac{1}{x}} \times \max_{0 \leq r \leq n} \left(\frac{\min(1, \sum_{A_{S_i} \in \text{ weight}(A_{S_i}))}}{DF_{k_r}} \right)$ where $x \geq 1$ is a parameter for the importance of the amount of records, A_{S_i} is the ith sensitive attribute value of a record r, and DF_{k_r} is the k-Distinguishing Factor of a record r.

Using the definition above, the M-Score of the published table can be computed by multiplying the highest individual record score among the n records weighted with a power $\frac{1}{x}$. The x of $n^{\frac{1}{x}}$ is a parameter for specifying

the importance of the quantity of records in a published table. If $x = 1$ then the amount of records is given more importance compared to the sensitivity of data. If $x \to \infty$, then $n^{\frac{1}{\infty}} \approx 1$ which means that we would like to know the highest individual record score of M-Score. The parameter x can be assigned any value where $x \geq 1$ with a trade-off between the importance of the highest individual record score to the importance of the amount of records.

3.3.2 M-Score of the Illustrative Example

To illustrate how M-Score is calculated, we use the published table of Scenario 1 described in Section 3.1 shown in Table 3.6 below, and the sensitive attribute value weights from Table 3.7.

Table 3.6: The "published table" of Scenario 1 described in Section 3.1. It is the subset of source Table 3.1 that is released.

id	Job	City	Gender	Disease	Medication	Age	Initial Diagnosis
3	Lawyer	Edmonton	Male	HIV	ARV	49	HIV
4	Lawyer	Edmonton	Male	Hypertension	Statin	70	Hypertension
5	Lawyer	Edmonton	Female	Flu	Paracetamol	29	Migraine
6	Lawyer	Edmonton	Female	Flu	Paracetamol	29	Migraine

We begin by calculating each individual record score, to find the maximal record score needed for M-Score.

For example, to calculate the record score of the row with id 3 in the Table 3.6, we begin by obtaining the k-Distinguishing Factor by finding the size of the equivalence class in the source table containing the row with id 3. Assuming that *Age*, *City*, and *Gender* are the quasi-identifiers, the relevant equivalence class will be of size 2; thus, $DF_{k_{id_3}} = 2$. This is because, there are two records with the same quasi-identifiers $\{Lawyer, Edmonton, Male\}$ (rows with id 3, and id 4) as seen in the equivalence classes in Table 3.2. Once the k-Distinguishing Factor is found for the row with id 3, the record score can be

calculated using Equation 3.3 and the weights from Table 3.7 as follows:

$$\begin{aligned} RS_{M_{id_3}} &= \frac{\min\left(1, \begin{array}{c}\text{weight}(HIV_{\text{Disease}}) + \text{weight}(ARV_{\text{Medication}}) \\ + \text{weight}(30+_{\text{Age}}) + \text{weight}(HIV_{\text{Initial Diagnosis}})\end{array}\right)}{DF_{k_{id_3}}} \\ &= \frac{\min(1, 1+1+0.1+0.4)}{2} = \frac{1}{2} \\ &= 0.5 \end{aligned}$$

When all the record scores of a published table are calculated, the maximal record score is found to be 0.5 and can be used to calculate the M-Score of the published table as:

$$M\text{-Score} = 4^{\frac{1}{x}} \times 0.5$$

where x is a parameter that can be set to how much importance should be given to the number of records released. For instance, if $x = 1$ then the M-Score is 2 as the number of records are considered to be very important, compared to when $x \to \infty$ where M-Score becomes 0.5 and the number of records is considered to be not important as we get the maximum score record only. Notably, x can also be set to a number such as 10 to get 0.574, but it is difficult to decide the appropriate x parameter for the trade-off between the importance of the number of records or the maximum record score in a published table.

If the process is repeated for Scenario 2 of Section 3.1, where the complete source table is published, the M-Score obtained is:7 when $x = 1$, and 1 when $x \to \infty$.

If we compare Scenario 2 to Scenario 1, we can see that the M-Score ($x = 1$) for Scenario 2 is greater. This increase is explained by the greater amount of data being released. Furthermore, we can also determine that the row with id 0 of the published table in Scenario 2 has a record score of 1 which is greater than any other record score in Scenario 1. Therefore, the M-Score ($x \to \infty$) in Scenario 2 is also greater than Scenario 1.

3.3.3 Drawbacks of M-Score

M-Score has many drawbacks. First, the pairwise comparison for determining sensitive attribute weights can be time-consuming when there is a large amount values. Second, M-Score is an approximation score which may over quantify the misuseability of a published table because it takes only the maximum record score. As well, the x parameter of M-Score may be difficult to decide on. Finally, M-Score only accounts for identity disclosure attacks, but not attribute disclosure attacks.

The AHP pairwise comparison method can become time-consuming when there are a lot of nodes in the tree model. As a model for a dataset becomes larger due to an increasing number of sensitive attributes or sensitive attribute values, it increases the number of pairwise comparisons quadratically [38].

M-Score is also approximative in nature as it takes the maximum record score of a published table for its score calculation. For example, consider a source table where 99 records of 100 records have a record score of 0.0001 with the remaining record having a record score of 1. If we release a published table from the source table where nine records are 0.0001 and one record is 1, the one record with a score of 1 will be used to calculate M-Score. This leads to a score of 10 (if $x = 1$) which is misleading as only a single record has a score of 1 in the published table.

The approximation in M-Score leads to issues when attempting to identify the "percentage of severity" a published table takes from the source table score. If we consider the example from the previous paragraph, where the published table has a score of 10, and the source table has a score of 100, we can say that the published table makes up 10 percent of the severity of the source table: $10/100 = 0.1$. However, this is not the case when we compare the sum of the actual values of the record scores in each of the tables: $1.0009/1.0099 = 0.991$.

Next, as observed when calculating the M-Score for Scenario 1 in Section 3.3.2, it can be difficult to decide on the parameter x when deciding on the trade-off between the importance of the number of records or the maximum record score in a published table. If we would like to approximate the

severity based on the amount of records, then x can be set as 1. If we assume that releasing any single maximum record of a published table is the maximum severity, then x can be set as $x \to \infty$. However, to decide on a value between 1 and $x \to \infty$ that represents the trade-off between these two factors, an arbitrary decision would need to be made.

The last drawback to note is that the k-Distinguishing Factor in M-Score only accounts for identity disclosure attacks. The authors of M-Score suggest that measures such as l-diversity can be used to account for attribute disclosure attacks, but provide no method to do so.

3.3.4 *L*-Severity

Building on the work of M-Score, Vavilis et al. [39] design a misuseability score named L-Severity. L-Severity aims to address the approximative nature of the M-Score calculation, and also the time-consuming task of determining sensitive attribute value weights.

To address the approximative nature of M-Score, L-Severity does not get maximum record score and multiply it by the number of records like M-Score. Instead, it sums all the record scores calculated. This makes L-Severity more decimal sensitive compared to M-Score and can help illustrate the potential severity of misuse more clearly.

To save time when eliciting weights, L-Severity proposes a new propagation mechanism based on a hierarchical data model. The data model is designed so that sensitive attributes and their values can be categorized under a single classification, and weights assigned to the single classification can be propagated to the attributes. The purpose of propagation is so that every possible sensitive attribute value weight does not need to be separately examined, as is the case with M-Score.

The process of calculating L-Severity can be divided into three steps: (1) developing a data model of the sensitive attributes in a source table, (2) eliciting weights for sensitive attribute values from the data model, and (3) calculating the overall L-Severity for the published table.

(1) Developing the Data Model The data model in *L*-Severity is designed to represent the hierarchy of concepts surrounding the sensitive attributes in a source table.

The data model contains two different types of nodes: "data types" and "data instances". "Data types" are either attributes or a categorization of attributes (i.e. the column names of a table or a generalization of other nodes). "Data instances" are values of attributes (i.e. the values of a column in a table). This is similar to the three-level tree model for AHP described in Section 3.3.1, but allows for a more flexible representation of hierarchical relationships with the "data type" node that can generalize other nodes.

Each node can be assigned a "sensitivity value" by a domain expert, but it is not necessary to assign values to each node individually as the values can be propagated top-down in the hierarchy. It should be noted that because of the top-down propagation, there should be a parent node with an assigned value for any children node without a value.

Special single directed edges can be formed between "data instance" nodes to create an "inference relationship". For each inference relationship, an "inference value" is assigned in the range of $[0, 1]$ to signify how much information can be inferred by linking the source node to the sink node. Inference relationships can only be established for "data instance" nodes. These "inference relationships", and "inference values" in the data model are created by a domain expert.

Figure 3.2 illustrates the *L*-Severity data model of Table 3.1, modeled to closely resemble the domain expert decisions of inference relationships and weights in Vavilis et al.'s [39] data model.

(2) Eliciting Sensitive Attribute Value Weights using the Data Model Once all "sensitivity values" and "inference values" are assigned by a domain expert in the model, they can be used to calculate the sensitive attribute value weights needed for *L*-Severity.

First, for any nodes that do not have an assigned sensitivity value, they are propagated a value by finding the first parent that has a value. For example,

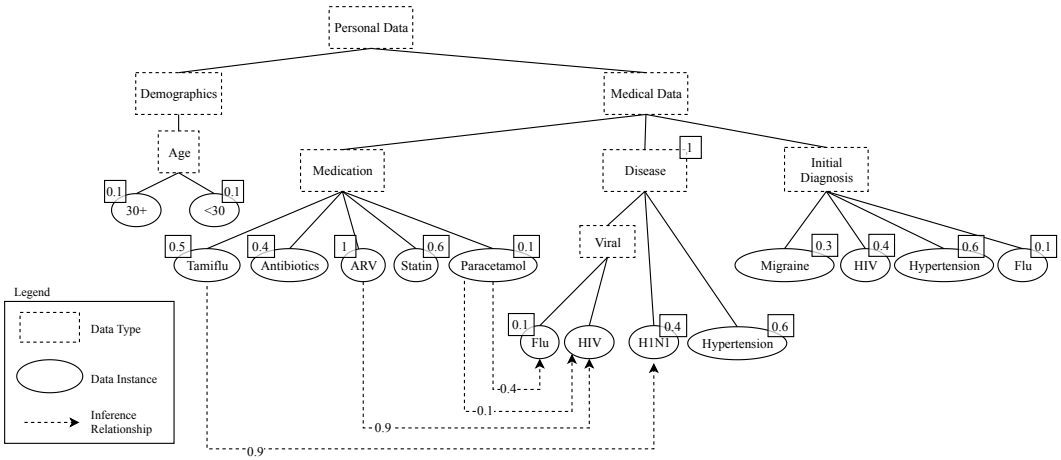


Figure 3.2: The *L*-Severity data model of Table 3.1 in the illustrative example. The dotted rectangular shapes are data types corresponding to the sensitive attribute table columns and their generalizations. The elliptical shapes are the sensitive attribute values present in Table 3.1. The inference relationships are denoted as dashed directed edges that indicate a relationship between sensitive attribute values. The numbers on the inference relationships indicate "how related" two attribute values are. The small rectangular boxes containing numbers in the top left or right of data types and instances are sensitivity values assigned by a domain expert. It should be noted that quasi-identifiers are omitted in this figure for succinctness, but are included in the data model defined Vavilis et al. [39].

the sensitivity value of "HIV" in Figure 3.2 is 1, from the propagation of: "Disease" $\rightarrow$ "Viral" $\rightarrow$ "HIV".

Next, the "sensitive attribute value weight" can be calculated by taking the sensitivity value assigned to a "data instance" node, and any inference relationship where the "data instance" node is the source.

For instance, to calculate the sensitive attribute value weight of the medication "Paracetamol", we begin by iterating through and calculating the value for its inference relationships. The inference relationship from "Paracetamol" to "Flu" has an inference value of 0.4, which is taken and multiplied by the sensitive attribute value weight of "Flu" resulting in: $0.4 \times 0.1 = 0.04$. This process is repeated for the other inference relationship from "Paracetamol" to "HIV" yielding $0.1 \times 1 = 0.1$. Once all values for inference relationships are calculated, they are summed to be: $0.04 + 0.1 = 0.14$. This sum is compared against the sensitivity value given to "Paraceta-

mol" and the max is taken. Hence, the weight for "Paracetamol" will be: $\max(\text{sum}(\text{InferenceRelationshipValues}), \text{SensitivityValue}(\text{Paracetamol})) = \max(0.14, 0.1) = 0.14$. This method is outlined in Algorithm 1.

Algorithm 1: Algorithm for Calculating Sensitive Attribute Value Weight

Input: let y be a data instance node with a sensitivity value
Output: Sensitive Attribute Value Weight for y

```
let relValues = 0;
for each Inference Value IV(y, z) of y do
    relValues = relValues +
                (IV(y, z)) × SensAttValWeight(z));
end
return max(relValues, SensitivityValue(y));
```

(3) *L*-Severity of the Published Table Once the sensitive attribute value weights for the data instances are calculated, L-Severity can be calculated. To determine the L-Severity of a published table, the record scores are summed together. A record score can be calculated as follows:

$$RS_{L_r} = \frac{\sum_{A_{S_i} \in r} \text{weight}(A_{S_i})}{DF_{k_r}} \tag{3.4}$$

A record score RS_{L_r} of the rth record of a published table is the sum of each ith sensitive attribute value weight of the record divided by the k-Distinguishing Factor DF_{k_r} of the rth record.

Using the sum of the record scores in a published table, L-Severity is:

Definition 14 (L-Severity)**.** Given a published table T, the L-Severity of the table is: $L\text{-Severity} = \sum_{r \in T} \left(\frac{\sum_{A_{s_i} \in r} \text{weight}(A_{s_i})}{DF_{k_r}} \right)$ where r is each record of the published table, A_{S_i} is the ith sensitive attribute value of a record r, and DF_{k_r} is the k-Distinguishing Factor of a record r.

3.3.5 *L*-Severity of the Illustrative Example

To illustrate how L-Severity is calculated, we refer back to Scenario 1 in Section 3.1 where Table 3.6 is to be released and a misuseability score needs to be calculated for it.

Before referring to the published and source table for calculating L-Severity, we refer back to the data model in Figure 3.2 to calculate all the sensitive attribute value weights following step 2 of Section 3.3.4. First, we propagate all sensitivity values to any of the data instance nodes such as "HIV" that are missing sensitivity values. Next, the sensitive attribute value weights are calculated using Algorithm 1, and the results are shown in Table 3.7.

Table 3.7: Sensitive attribute value weights elicited from the data model in Figure 3.2 using L-Severity's methodology.

Disease	**Medication**	**Age**	**Initial Diagnosis**
W(Flu) = 0.1	W(Paracetamol) = 0.14	W(30+) = 0.1	W(Flu) = 0.1
W(H1N1) = 0.4	W(Antibiotics) = 0.4	W(<30) = 0.1	W(Migraine) = 0.3
W(Hypertension) = 0.6	W(Tamiflu) = 0.5		W(HIV) = 0.4
W(HIV) = 1	W(Statin) = 0.6		W(Hypertension) = 0.6
	W(ARV) = 1		

Now referring back to Table 3.6, we can calculate L-Severity of Scenario 1 using the sensitive value weights in Table 3.7. As a reminder, we also need to use Table 3.1 as the source table to calculate the k-Distinguishing Factor.

To begin, we iterate through each row of Table 3.6 and using Equation 3.4 to calculate a record score. For example, for the row with id 3 the record score will be:

$$\begin{aligned} RS_{L_{id_3}} &= \frac{W(HIV_{Disease}) + W(ARV) + W(30+) + W(HIV_{Initial})}{DF_{id_3}} \\ &= \frac{(1 + 1 + 0.1 + 0.4)}{2} \\ &= 1.25 \end{aligned}$$

Repeating the process for each record we get the following scores: row id 4 as 1.9, row id 5 as 0.64, and row id 6 as 0.64. Summing up all the scores, we get the L-Severity of Scenario 1 to be:

$$L\text{-Severity} = 1.25 + 1.9 + 0.64 + 0.64 = 4.43$$

If this process is repeated for Scenario 2 of Section 3.1 where the whole table is released, we get L-Severity = 1.1+1+0.9+1.25+1.9+0.64+0.64 = 7.43 by iterating from row 0 to row 6 of Table 3.1.

Comparing Scenario 2 to Scenario 1, the score of Scenario 2 is much larger signifying that releasing the whole table which will be much more severe.

3.3.6 Drawbacks of *L*-Severity

The major drawback of *L*-Severity is the assignment of weights to the model, and the inference relationships in the data model. It is difficult to justify the value of inference relationships between two data instance nodes. For example, how can one decide in a quantifiable value that the inference relationship from "Tamiflu" to "H1N1" is 0.9 compared to the relationship from "Paracetamol" to "Flu" being 0.4. The authors of *L*-Severity suggest that domain experts would make decisions on inference values, but provide no method on how to do so and coming to a consensus among experts may be difficult. The assignment of sensitivity values to nodes is also of concern as there is no systematic methodology to assign values like *M*-Score which uses AHP pairwise comparison.

It should also be noted that the method to derive the sensitive attribute value weights from the *L*-Severity data model makes the weights unbounded. Therefore, the sum of the sensitive attribute value weights for a record score is not bounded to a maximum sum. In contrast, *M*-Score bounds the sum of sensitive attribute value weights by taking the minimum between 1 and the sum of weights. Without any maximum bounds on the sum, it becomes difficult to know the extent of severity for the record score calculation as there is no theoretical maximum score to normalize against. Without bounding the weights, we cannot normalize against a theoretical maximum is discussed in Section 5.4.

As well, *L*-Severity cannot account for the case that assumes releasing the maximum record score of a published table is the maximum severity. Recall that *M*-Score accounts for this case, by allowing the x parameter to be set to $x \to \infty$. *L*-Severity, instead, only accounts for the case that calculates severity based on the amount of records.

Another issue of *L*-Severity is the structure of inference relationships defined in the data model. The authors of *L*-Severity indicate in their example

data model [39] that inference relationships can be bidirectional between nodes. However, when calculating the sensitive attribute value weight (Algorithm 1) a cycle of dependencies is formed. For instance, consider two nodes that have inference relationships in both directions to each node, and each node has an assigned sensitivity value. It then becomes impossible to calculate the sensitive attribute value weight of both nodes as they are dependent on each other to calculate their sensitive attribute value weight.

Finally, L-Severity has the same drawback as M-Score, where additional anonymity measures like l-diversity and t-closeness are not included to account for attribute disclosure. The authors of L-Severity suggest that these measures can be integrated into misuseability scoring, but provide no method to do so.

3.4 Summary

In this chapter, the PPDP measures k-anonymity, l-diversity, and t-closeness are introduced. k-anonymity quantifies the likelihood of an identity disclosure attack, while l-diversity, and t-closeness quantify the likelihood of attribute disclosure attacks.

Since PPDP measures do not account for the quantity of data, and the differing sensitivities of values in data, the misuseability scores M-Score and L-Severity were developed to address these factors in addition to integrating k-anonymity to account for identity disclosure attacks. However, neither M-Score nor L-Severity account for attribute disclosure attacks.

Furthermore, M-Score always assumes the worst case when it calculates its misuseability score, while L-Severity fails to systematically assign weights and has structural issues in its data model. These drawbacks are addressed in Chapter 4 where tkl-Score is introduced.

Chapter 4

The *tkl*-Score

In this chapter, a new misuseability score *tkl*-Score is designed to: (a) capture both risks of identity disclosure and attribute disclosure attacks, and (b) reflect the different sensitivities of sensitive attribute values using derived weights from a domain specific model. *tkl*-Score addresses the drawback of M-Score and L-Severity where attribute-disclosure attacks are not accounted for. It also addresses the non-systematic method of eliciting weights for sensitive attribute values in L-Severity.

The calculation of *tkl*-Score consists of three steps similar to M-Score and L-Severity: (1) the *tkl*-Data Model, a domain specific model of sensitive attributes and sensitive attribute values, is developed; (2) weights of sensitive attributes are derived from the developed *tkl*-Data Model; and (3) the *tkl*-Score of a published table using the derived weights.

4.1 Developing the *tkl*-Data Model

The *tkl*-Data Model is a well-defined semantic model designed to represent the underlying hierarchical relationships of sensitive attributes, and the sensitive attribute values in a source table. It is used to derive weights for sensitive attribute values.

To construct a *tkl*-Data Model, a sensitive attribute domain taxonomy is aligned with a model of sensitive attributes in a source table. To model the sensitive attributes in a source table, a novel data sensitivity ontology (DSO) is used to represent the sensitive attributes of the source table.

4.1.1 Sensitive Attribute Domain Taxonomy

The domain taxonomy is developed using expert knowledge and is a reflection of the hierarchical relationships between sensitive attributes found in a source table. It provides an understanding of the underlying relationships not present in a table structure.

For example, in the illustrative example Table 3.1 we can see that it contains data about demographics and disease diagnoses. There are many relationships in the table that may appear to be obvious that should be accounted for when calculating the misuseability score. Take for instance, the relationship between medication and disease, it is clear that they are related to medical health. However, this relationship is not accounted for in M-Score. In L-Severity, this relationship is modeled in a data model like Figure 3.2 and used to derive weights for attribute values.

Considering Preexisting Models

To develop a domain taxonomy, domain expert knowledge is needed to model the concepts and relationships between them. This process is often time-consuming, as it requires a consensus among experts and there may be a significant amount of concepts and relationships. Furthermore, these efforts may produce duplicate representations for similar tables, as well as discourage interoperability. Therefore, in this thesis, the use of preexisting ontology, vocabulary, and taxonomy models to develop the domain taxonomy are explored.

Before explaining the process of using preexisting models, we must clarify the use of the terms: "ontology", "vocabulary", and "taxonomy". "Ontology" is considered to be a collection of concepts and relationships with defined assertions known as axioms that can be used for inference. Whereas "vocabulary" is a subset of "ontology" that has no assertions of concepts and relationships [29]. "Taxonomy" is a subset of "vocabulary" that contains a hierarchical definition of concepts and relationships. Therefore, "ontology" is used as a generalized term to describe all three terms.

Discovering and choosing the right existing ontologies to create a domain

taxonomy can be difficult due to the lack of a reliable strategy and sources. The W3C community has likened this to the process of "dowsing" where resources are attempted to be found in the ground without any scientific apparatus [30]. They recommend using a combination of indexes, search engines, repositories, and online communities to help find relevant ontologies.

We should also note that not all ontologies are created equal. Ontologies are often prone to design errors, and can vary in quality and correctness [27]. As well, ontologies can be defined using a combination of a variety of specifications such as the resource description framework (RDF), RDF Schema (RDFS), and Web Ontology Language (OWL). Each specification allows for different ways to represent concepts, relationships, and axioms.

As an example, consider finding a relevant ontology model for the sensitive attributes of Table 3.1. The sensitive attributes contain values relating to occupation, medical relationships, location, and demographics. Using the Google search engine, some of the following relevant ontologies can be found: the ResumeRDF ontology[1] to describe occupation, the CPR ontology[2] to describe medical relationships, the vCard ontology[3] to describe location, and the FOAF ontology[4] to describe demographics. For these ontologies, extracting the relevant concepts and relationships to build a domain taxonomy prove to be difficult as they vary in specifications for expression. In order to build a taxonomy from the ontologies, the varying expressions of concepts and relationships must be aligned together which is left for future work.

For Table 3.1 in this thesis, the data privacy vocabulary (DPV)[5] from another resulting Google search is found to be the most relevant. This is because DPV is already designed as a taxonomy that encompasses values relating to occupation, medical relationships, location, and demographics and uses RDFS to define concepts and relationships.

[1] `http://rdfs.org/resume-rdf/`
[2] `https://www.w3.org/wiki/images/3/3a/CPR-W3C-Presentation.pdf`
[3] `https://www.w3.org/TR/vcard-rdf/`
[4] `http://xmlns.com/foaf/spec/`
[5] `https://www.w3.org/ns/dpv`

Extracting Concepts and Relationships from Preexisting Models

Only relevant concepts and relationships should be extracted from existing domain ontologies. In the case of DPV, "class" is used to define concepts, and the "subclass of" property is used to define the hierarchical relationships of classes. By extracting the relevant "classes", and "subclass of" properties in DPV, a smaller hierarchical structure is created.

To illustrate how to extract the relevant concepts and relationships, we refer back to the illustrative example Table 3.1. This table is a medical dataset containing patient information with the following sensitive attribute columns: *Disease*, *Medication*, *Age*, and *Initial Diagnosis*.

The sensitive attribute columns of Table 3.1 can then be used to find the relevant DPV class as follows:

- Disease $\mapsto$ DPV:Health
- Medication $\mapsto$ DPV:Prescription
- Age $\mapsto$ DPV:Age
- Initial Diagnosis $\mapsto$ DPV:HealthRecord

These mappings consider the sensitive attribute columns of Table 3.1 to be "directly related" to the DPV class. The decision of finding the relevant DPV classes comes from referencing the description of classes which can be seen in Table A.4.

Now to build the hierarchy, a search of DPV is needed to find all the weakly connected components containing the mapped DPV classes. Weakly connected components are a subgraph where all nodes are connected by an edge regardless of edge direction. A weakly connected component forms multiple pathways to a common root node as seen in Figure 4.1 where there are multiple pathways to "PersonalDataCategory".

The process for finding weakly connected components of an ontology, i.e., the DPV in our example, begins with initializing a set of concept nodes corresponding to the sensitive attributes of the source table. Next, a breadth-first

search process adds to the current node set, by adding the parent classes of the nodes in the current set. The process ends as soon as no new parent classes can be found, which results in one or more weakly connected components.

The extracted weakly connected component from DPV in Figure 4.1 is a close to ideal extraction as minimal pruning and no alignment is needed to form the appropriate domain taxonomy structure. There is a single component so no alignment between other components is needed. As well, there is only a single edge that needs to be pruned to form the mono-hierarchical tree structure.

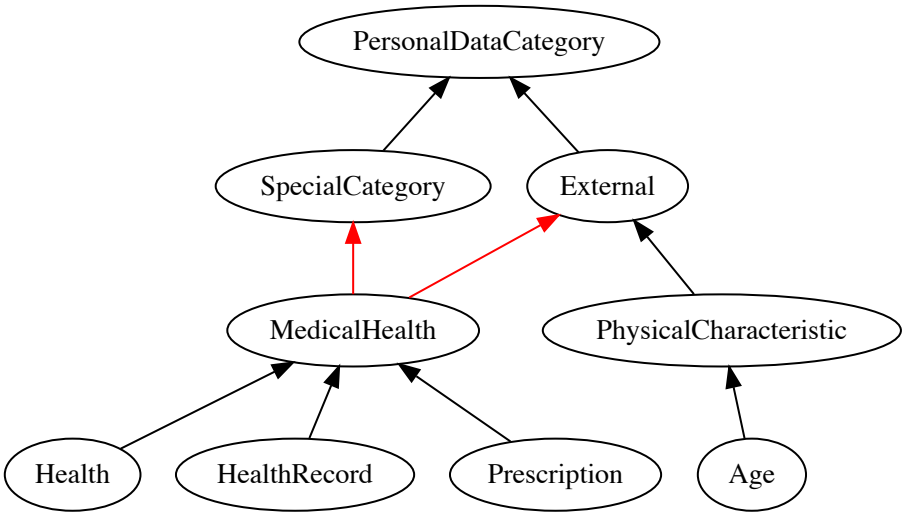


Figure 4.1: Connected component of DPV forming the relationships between sensitive attribute concepts. It contains multiple pathways to the root node "PersonalDataCategory". Of note are the two edges (highlighted in red) from "MedicalHealth" which need to be pruned to form the appropriate domain taxonomy structure

Domain Taxonomy Structure

The domain taxonomy models the generalization/specialization relationships of sensitive attributes in a mono-hierarchical tree structure. A mono-hierarchical tree structure means that each node has only single parent, as opposed to a poly-hierarchical structure where a node can have multiple parents.

Each node of the domain taxonomy is an instance of a RDFS class, denoted by "RDFS:Class". The relationships between nodes are edges, denoted by "RDFS:subClassOf". The resulting domain taxonomy is defined by multiple RDF triples in the form of "RDFS:Class → RDFS:subClassOf → RDFS:Class" where the second element of the triple denotes a directed edge from the first element to the third element.

The purpose of the domain taxonomy structure is so that pairwise comparisons can be done between groupings of sub-classes to derive relative weights. Thus, the tree structure needed to be obtained is as follows:

- A single node with no parent, denoted as the "root node"
- Any other node only has a single parent
- All leaf nodes are directly related to the sensitive attribute columns of a table, i.e. the starting nodes for finding the connected components
- There are no repeated edges between nodes
- Edges all follow a single direction top-down or bottom-up
- Each level of the tree are related to the same generalization (e.g. H1N1 should not be in the same level as Disease since it is a specific value)

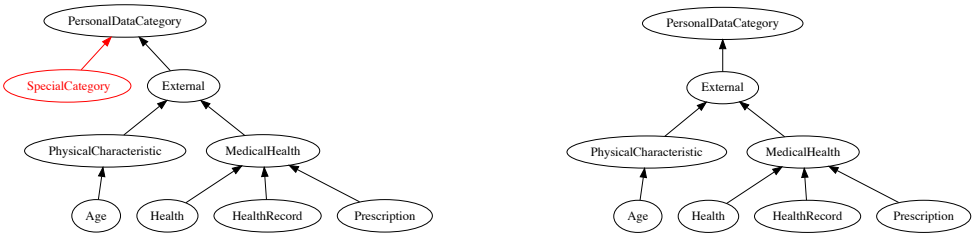
To achieve the appropriate domain taxonomy structure from the extracted components of an ontology, the pruning of extra edges, pruning of leaf nodes, and alignment of components may be needed as outlined below.

Pruning Extra Edges Notably, in Figure 4.1, the connected component needs to be pruned as "MedicalHealth" has the two parent classes: "External", and "SpecialCategoryPersonalData". Therefore, one of the edges must be removed to form a mono-hierarchical tree structure.

For this thesis, we arbitrarily remove the edge between "MedicalHealth" and "SpecialCategoryPersonalData" assuming that the edges are of equal importance as seen in Figure 4.2a. If one edge is considered more important than another, then the edge that should be kept should be decided among a

consensus of domain experts. This consensus is to ensure that the domain taxonomy still models an accurate representation of the hierarchical concepts and relationships of sensitive attributes.

Pruning Leaf Nodes The edge pruning of Figure 4.1 results in a leaf node, highlighted in red in Figure 4.2a, that is not directly related to the sensitive attribute columns of Table 3.1 since it is not a starting node. Thus, the "SpecialCategory" node and its edge is removed resulting in Figure 4.2b.

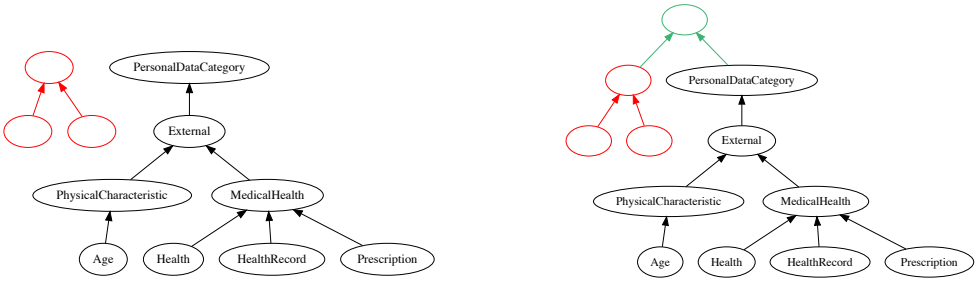


(a) Result of pruning the edge between "MedicalHealth" and "SpecialCategoryPersonalData". Note that there is now a leaf node (highlighted in red) that also needs to be pruned.

(b) Domain taxonomy with the appropriate structure.

Figure 4.2: Pruning an extra edge and leaf node (a) and its resulting structure (b).

Aligning Connected Components Consider Figure 4.3a which contains an extra unlabeled component and the component in Figure 4.1. To align the components a root node is created to take the separated components as its children resulting in an appropriate domain taxonomy structure as seen in Figure 4.3b. A consensus of domain experts should be used to verify that the alignment is correct, and make adjustments to the generalizations of connected components as needed.



(a) Two weakly components differentiated by color.

(b) Alignment of two weakly connected components by creating and connecting to a common root node (outlined in green).

Figure 4.3: Weakly connected components (a) and their alignment (b).

4.1.2 Data Sensitivity Ontology (DSO)

The data sensitivity ontology (DSO) is created to manage the representation of sensitive attributes in a source table for calculating *tkl*-Score, and to manage the metadata needed for weight derivation using the *tkl*-Data Model. It is defined using the RDF/RDFS model.[6]

There are 4 main classes described as follows:

- **ContinuousAttributes** are the sensitive attributes of a source table that contain continuous values. They are initiated as an instance to represent the sensitive attribute column used to map to a domain taxonomy leaf node. The "value" nodes of its children, are the binned values of the continuous attribute decided under a consensus of domain experts.

- **DiscreteAttributes** are the sensitive attributes of a source table that contain discrete values. They are also initiated as an instance to represent the sensitive attribute column used to map to a domain taxonomy leaf node.

- **Elements** are the generalization of values. They are subclasses of attributes or other elements.

[6] `https://www.w3.org/TR/rdf-schema/`

- **Values** are the values of discrete attributes, or the ordinal values of continuous attributes. They can either be a subclass of an attribute or element, and are leaf nodes.

These classes are used to instantiate a model of sensitive attributes as shown in Section 4.1.3.

DSO also defines properties that are used in the process for deriving weights of sensitive attribute values. They consist of "hasPriority" which is used to assign priorities to all nodes of the *tkl*-Data Model, and "hasWeight" which is used to assign weights derived from priorities to instances of DSO Value classes:

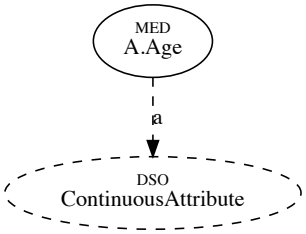
- **hasPriority** is used to assign a weight value, known as a "priority", to an instance of a DSO class. It can be used as a predicate in a triple as follows: "DPV:MedicalHealth $\rightarrow$ *hasPriority* $\rightarrow$ literal" where "literal" is a decimal value such as 1.0.

- **hasWeight** is used to assign the weights calculated based on the priorities in a tree. It is also used as a predicate in a triple with the subject being a sensitive attribute value and object being a literal as such: "Sensitive Attribute Value $\rightarrow$ *hasWeight* $\rightarrow$ literal" where "literal" is a decimal value such as 0.2.

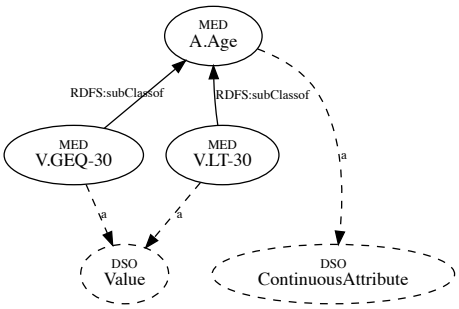
4.1.3 Sensitive Attribute Source Table Model

The modeling of sensitive attributes in a source table involves the use of DSO to structure its representation.

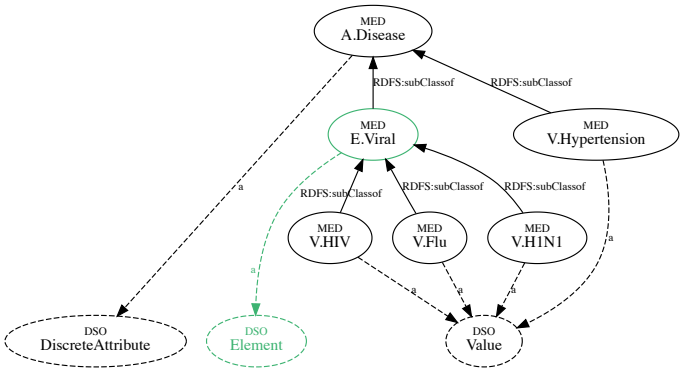
For example, to model the sensitive attribute *Age* of Table 3.1 using DSO, we first begin by creating the namespace to represent the table denoted as "MED". Next, we create a node to represent the sensitive attribute denoted as "MED:A.Age". "A." is prepended to the attribute name *Age* to indicate the node is an attribute. "MED:A.Age" can then be defined as a "ContinuousAttribute" as seen in Figure 4.4a.



(a) Defining *Age* of the source table as a continuous attribute.



(b) Defining the attribute values of the *Age* attribute.



(c) Categorizing some of the attribute values of *Disease* to be under a "Viral" element.

Figure 4.4: Creating a source table model using DSO. Dashed lines indicate the use of DSO.

To model values of the sensitive attribute *Age*, two additional value nodes are created: "MED:V.GEQ-30" to represent values greater or equal to 30, and "MED:V.LT-30" to represent values less than 30. The names are prepended with "V." to indicate that the node is an attribute value. Each of the nodes are then made an instance of "Value" and subclass of "MED:A.Age" as seen in Figure 4.4b. It should be noted that choosing the appropriate binning for the value nodes of continuous attributes is decided among domain experts.

Many of the times, using a consensus of domain experts, sensitive attribute values can be categorized to create a hierarchy of values. This hierarchy can later be used to propagate weights. For instance, consider the sensitive attribute *Disease* in Figure 4.4c where an instance of the "Element" class

"E.Viral" is created. "E.Viral" creates a hierarchy of values as it is the parent class for the attribute values: "V.H1N1", "V.HIV", and "V.Flu".

Duplicate Node Names When creating the source table model with DSO, domain experts should be aware of DSO node instances with the same names as they can have different contexts and should not be considered the same sensitivity. To illustrate, consider Figure 4.5a where the attributes *Age* and *Disease* are given the "Duplicate" value. This should not occur if the "Duplicate" value under *Age* and *Disease* do not have the same meaning. Thus, a naming scheme such as using "V-0" and "V-1" in Figure 4.5b must be created to differentiate the "V.Duplicate" node.

(a) The "V.Duplicate" node has two parent nodes (edges highlighted in red) even though it may not be related to both parents.

(b) The "V.Duplicate" recreated with a naming scheme to signify that values are not similar.

Figure 4.5: Handling duplicate node names of a DSO defined model.

The resulting source table model of Table 3.1 created using DSO can be seen in Figure 4.6.

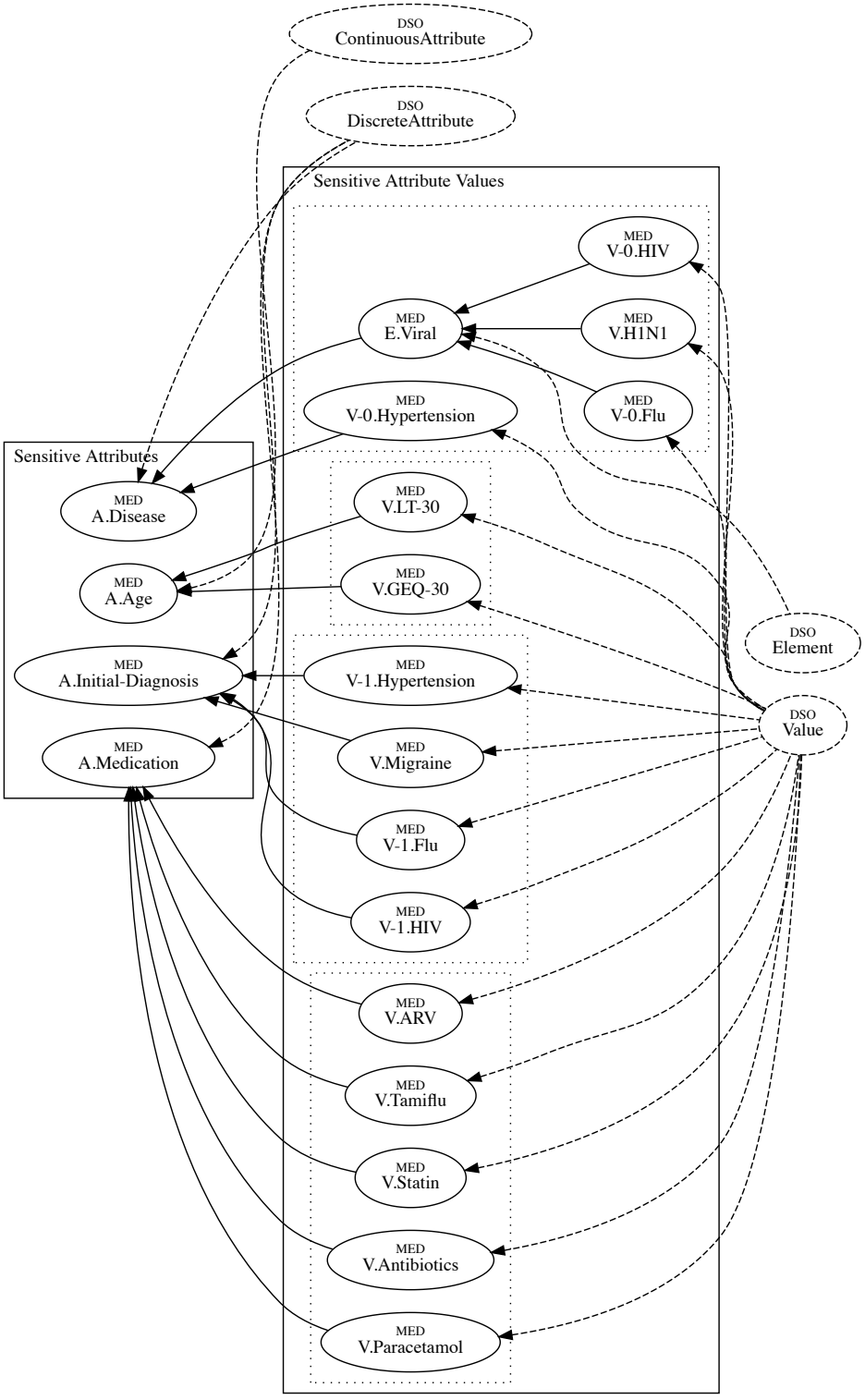


Figure 4.6: This figure is the sensitive attribute source table model of Table 3.1 created using DSO. The dashed lines indicate the instances of DSO classes used to define the sensitive attributes and their values in a table. This model is aligned with the domain taxonomy to create the *tkl*-Data Model.

4.1.4 Alignment of the Table Model with the Domain Taxonomy

Once a model of a source table has been created with DSO, it can be aligned with the sensitive attribute domain taxonomy to create the *tkl*-Data Model. The purpose of the alignment to support assigning weights, and propagation of values to nodes that are missing them.

To align the source table model, the relevant sensitive attribute column is made a subclass of the relevant class in the domain taxonomy. For example, recall the relevant DPV classes that were found to extract concepts and relationships of sensitive attributes in Section 4.1.1, and the attribute nodes created to represent the sensitive attributes in Section 4.1.3. Using the representation of the sensitive attributes, the following alignments are made in Figure 4.7 between the domain ontology extracted from DPV in Figure 4.3b and the source table model in Figure 4.6:

- MED:A.Disease $\mapsto$ DPV:Health
- MED:A.Medication $\mapsto$ DPV:Prescription
- MED:A.Age $\mapsto$ DPV:Age
- MED:A.Initial-Diagnosis $\mapsto$ DPV:HealthRecord

Figure 4.7 demonstrates a complete mapping of Table 3.1's sensitive attributes to the domain taxonomy, and is now ready to be used for deriving weights of sensitive attribute values.

It should be noted that in Figure 4.7, DSO classes and properties are omitted. This is because, while the metadata information provided is useful to identify its type corresponding to the source table, it is unnecessary to visually appear for deriving weights.

Similarity with *L*-Severity Data Model The *tkl*-Data Model is similar to the *L*-Severity data model, but removes the notion of ad hoc "inference relationships" that increase complexity and are often difficult to justify. As

well, the *tkl*-Data Model does not model quasi-identifier attributes, while the *L*-Severity data model does. Lastly, the *tkl*-Data Model is used to derive sensitive attribute weights in a more systematic manner compared to the *L*-Severity data model using the Analytic Hierarchy Process (AHP) to determine weights as shown in Section 4.2 below.

Similarity with *M*-Score Three-Level Tree Model The *tkl*-Data Model is a well-defined model that includes the hierarchical concepts and relationships among sensitive attributes. On the other hand, the *M*-Score three-level tree model is a simple model that places sensitive attributes in the second level, and sensitive attribute values in the third level. Therefore, the underlying relationships between sensitive attributes are not modeled in the three-level tree model. As well, *tkl*-Data Model allows for the propagation of weights because of its structure as seen in Section 4.2.3. *M*-Score is only limited to AHP for determining weights which can be time-consuming when a lot of nodes in the tree model need to be compared pairwise.

4.2 Deriving Weights with the *tkl*-Data Model

To derive weights for sensitive attribute values, two methods are used on the *tkl*-Data Model: the Analytic Hierarchy Process (AHP), and domain expert assignment. These methods assign all nodes of the *tkl*-Data Model with "priorities". "Priorities" are a numeric value that represent a relative weight for a grouping of nodes in a level of a hierarchical structure. From these priorities, a "weight" for sensitive attribute values can be calculated.

The weight derivation process involves the elicitation of priorities with the following steps: (1) the "domain taxonomy" elements and the "sensitive attribute" elements are compared using their AHP criteria groupings to establish their relative priorities, (2) a domain expert directly assigns priorities to select "sensitive attribute value" elements, (3) if not all "sensitive attribute value" elements are assigned priorities, a propagation mechanism is used to propagate priorities to "sensitive attribute value" elements missing them, (4) once all pri-

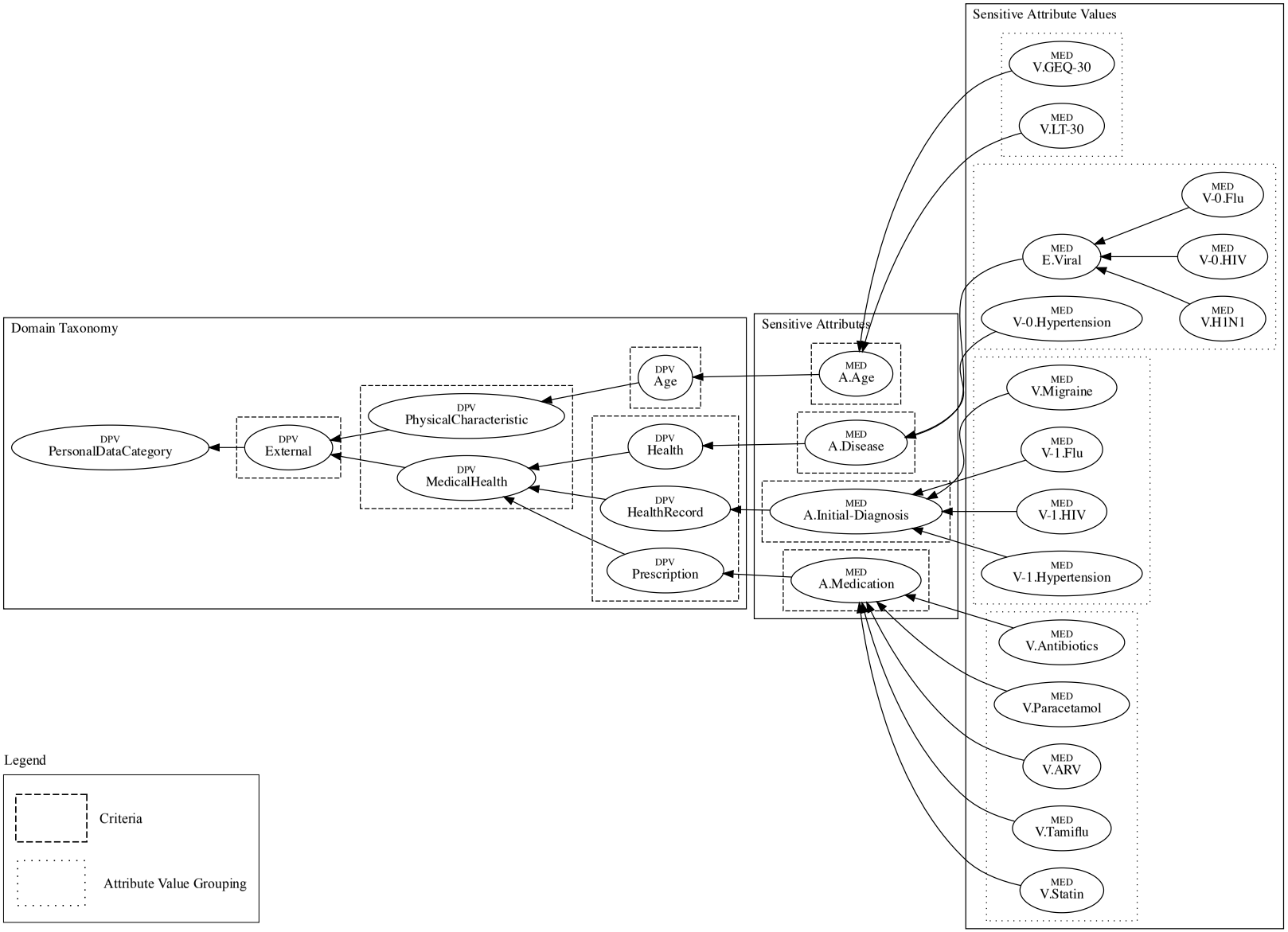


Figure 4.7: This figure is the *tkl*-Data Model of Table 3.1 and is used to derive weights of sensitive attribute vales for Table 3.1. It should be noted that DSO relations are used programmatically to derive weights, but are omitted from this figure for succinctness. Using this figure, four methods are needed to derive weights: (1) AHP pairwise comparison on criteria to assign priorities, (2) direct assignment of priorities on attribute groupings, (3) propagation of priorities within attribute value groupings if attribute values are not all directly assigned weights, and (4) the calculation of weights from the priorities assigned.

orities have been assigned to every node of the *tkl*-Data Model, the weights of "sensitive attribute values" can be calculated from following pathways to the "sensitive attribute value" elements.

4.2.1 AHP on the Domain Taxonomy and Attributes

The Analytic Hierarchy Process (AHP) is a decision-making technique that can be used to prioritize criteria for a decision to achieve a common goal. By comparing the criteria pairwise, it helps systematically decide on the importance of each criteria relative to others in order to make a decision. It has been used for a wide variety applications including assessing risk in operating pipelines [7] and quantifying the overall quality of software systems [25].

Using the tree structure formed by the domain taxonomy and sensitive attributes of Figure 4.7, we illustrate how AHP can be used to assign different priorities to criteria. The criteria in the figure, are the nodes grouped with dashed boxes. The root node is not a criteria, because it is does not need any comparisons.

To begin, the root node of the domain taxonomy is assigned a priority of 1 since there are no other comparisons. For each of the criteria under the root node, a reciprocal matrix of the pairwise comparisons is created by following the process outlined in Appendix A.1. As an example, consider the pairwise comparison of the criteria grouping under the node *MedicalHealth*: "Health", "HealthRecord", and "Prescription" as seen in Table 4.1a. For each combination pair of criteria, a domain expert indicates which criteria are more important by assigning a rating based on the Saaty scale seen in Table A.2.

For instance, in Table 4.1a "Prescription" is given rating that it is considered slightly more important than "HealthRecord". When all pairwise combinations of criteria are compared, a reciprocal matrix can be from the ratings as illustrated in Table 4.1b.

From Table 4.1b a priority vector of weights can be calculated as $\{\text{Health} = 0.54, \text{Health Record} = 0.297, \text{Prescription} = 0.163\}$ with a consistency ratio of 0.01. Since the consistency ratio is less than 0.10, the ratings of the reciprocal matrix are considered to be consistent as explained in Appendix A.1.3.

Criteria		More Important	Rating
A	B		
Health	HealthRecord	A	2
Health	Prescription	A	3
HealthRecord	Prescription	B	1/2

(a) Ratings given by a domain expert.

	Health	HealthRecord	Prescription
Health	1	2	3
HealthRecord	1/2	1	2
Prescription	1/3	1/2	1

(b) Reciprocal matrix with consistency ratio of 0.01.

Table 4.1: Ratings (a) and reciprocal matrix (b) of the criteria: "Health", "HealthRecord", and "Prescription".

The reciprocal matrices and consistency ratios for all criteria groupings of Figure 4.7 can be found in Appendix A.3, and their assignment to the domain taxonomy and sensitive attribute nodes can be seen in Figure 4.8.

4.2.2 Direct Assignment on the Table Model

In the *tkl*-Data Model, the "sensitive attribute value" nodes are directly assigned a "qualitative priority". The reason is that AHP does not accurately elicit the appropriate priorities among attribute values. To understand why this happens, we can frame the attribute values to be the nodes that "take a proportion of the weight of a sensitive attribute".

For example, if we wanted the sensitive attribute values of *Medication* to have the same weight in Figure 4.7, it would not be possible doing pairwise comparison. This is because pairwise comparison will equally divide the "priorities" among the five attribute value nodes of *Medication* to be 0.2, meaning that the nodes will only "take 0.2 of the weight of *Medication*". Instead, if we keep a constant "priority" such as 1 among all the attribute value nodes, then the nodes will "take all the weight of *Medication*". Thus, with direct assignment, attribute value priorities can be assigned to take the perceived "appropriate proportion of the weight of a sensitive attribute".

A "qualitative priority" is a label from a classification system that is ordinal. For example, consider the HL7 confidentiality classification labels[7]: unrestricted, low, moderate, normal, restricted, very restricted. We can see that the labels are ordinal in nature from low to very restricted.

[7] `https://www.hl7.org/fhir/v3/Confidentiality/cs.html`

To translate a qualitative priority to a quantifiable value for calculating weights, numerical values are assigned to the labels of a classification system monotonically increasing according to their order. This has also been done by the authors [39] of *L*-Severity whom translate the HL7 confidentiality classification labels to be: unrestricted = 0, low $\geq$ 20, moderate $\geq$ 40, normal $\geq$ 60, restricted $\geq$ 80, and very restricted $\geq$ 100.

In this thesis, a similar translation of the HL7 confidentiality classification labels is done where: unrestricted = 0, low = 0.2, moderate = 0.4, normal = 0.6, restricted = 0.8, and very restricted = 1. Using these labels, qualitative priorities are assigned to the attribute values in Figure 4.8. It should be noted that a priority is not assigned to "V.H1N1" because the priority can be propagated as demonstrated below in Section 4.2.3.

The qualitative priorities must be assigned to all first level attribute value nodes. This is important to ensure that every attribute value is assigned an appropriate priority when propagating. Referring back to Figure 4.8, all nodes of the first level of attribute values are assigned a qualitative priority using the HL7 confidentiality classification labels.

The result of direct assignment of priorities to the "sensitive attribute" and "sensitive attribute value" nodes for the *tkl*-Data Model in Figure 4.7 can be seen in Figure 4.8.

4.2.3 Propagation on the Table Model

Once all qualitative priorities have been assigned to at least the first level of attribute value nodes, any other attribute value nodes missing priorities can be propagated to. This strategy is similar *L*-Severity's propagation mechanism.

The algorithm for propagating weights can be seen in Algorithm 2. It is based on depth first search and a non-recursive implementation is described.

To propagate sensitive attribute value priorities, the sensitive attribute value nodes that are not currently assigned a priority are found. For the nodes that have no priority, it is assigned the priority of the parent node that has a priority.

For example, the propagation mechanism can be used in Figure 4.8 as

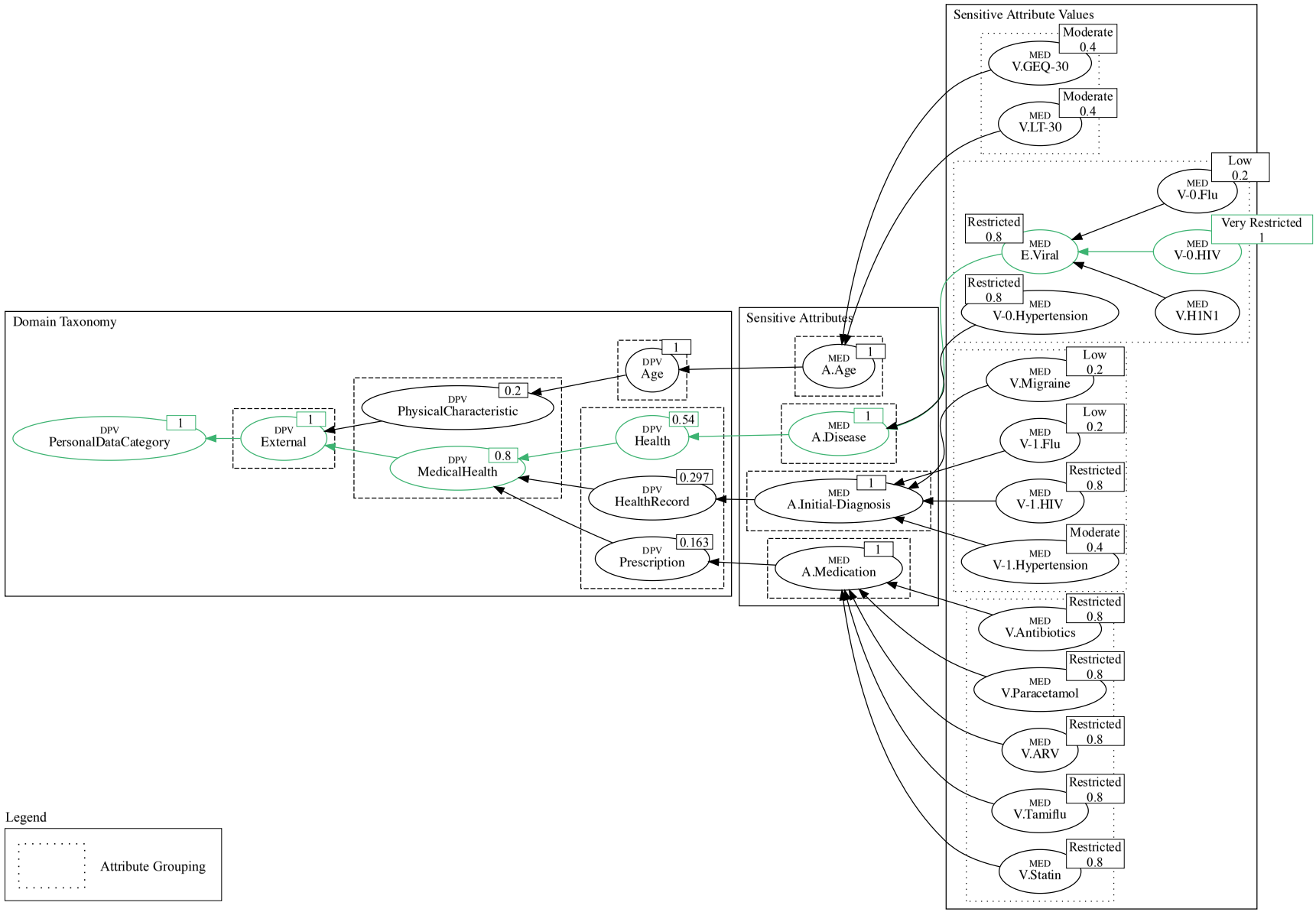


Figure 4.8: This is a domain ontology with priorities for attributes elicited from AHP, and priorities for attribute values assigned directly using HL7 confidentiality classification labels. It should be noted that a priority is not assigned to "V-0.H1N1" because it can be propagated from "E.Viral" to be "Restricted". The path of green nodes are used to calculate a weight for "V-0.HIV".

Algorithm 2: Algorithm to Propagate Attribute Value Priorities

Input: let C be the *tkl*-Data Model, let V be the first level of sensitive attribute value nodes of C

Result: *tkl*-Data Model with all sensitive attribute value priorities propagated

```
1  let S be a stack;
2  let Visited be a set;
3  S.push(V);
4  while S is not empty do
5  |  node = S.pop();
6  |  if node is not in Visited then
7  |  |  Visited.add(node);
8  |  |  if node.Priority DNE then
9  |  |  |  node.Priority = Priority of ParentNode containing priority;
10 |  |  end
11 |  |  for child of Children Nodes of node do
12 |  |  |  S.push(child);
13 |  |  end
14 |  end
15 end
```

"V.H1N1" is missing a priority. Using Algorithm 2, the priority can be propagated from "E.Viral" to "V.H1N1" to assign a restricted priority of 0.8.

4.2.4 Calculating Sensitive Attribute Value Weights

When all nodes have been assigned a priority in the *tkl*-Data Model, the weights for the sensitive attribute values can be calculated.

To calculate the sensitive attribute value weights, we follow the path from the attribute value node to the root node of the *tkl*-Data Model. Using the pathway the priority of the single attribute value node is multiplied with any proceeding attribute nodes (note the exclusion of attribute value nodes), and the root node.

For example, consider the pathway from "V-0.HIV" to "PersonalDataCategory" highlighted green in Figure 4.8. The weight for "V-0.HIV" is then calculated by multiplying the priority of "V-0.HIV", priorities of all attribute nodes in the pathway, and the priority of the root node. It should be noted that the priority for the root node is considered to be 1,

and any other attribute value nodes other than the one where a weight is being calculated for is ignored. The resulting calculation for "V-0.HIV" is as follows: priority(PersonalDataCategory) $\times$ priority(External) $\times$ priority(MedicalHealth) $\times$ priority(Health) $\times$ priority(A.Disease) $\times$ priority(HIV) $= 1 \times 1 \times 0.8 \times 0.54 \times 1 \times 1 = 0.432$.

The resulting weights for all the sensitive attribute values can be found below in Table 4.2. An interesting property to note is that when summing the maximum weight from each sensitive attribute, the sum will never be greater than 1. This makes it easy to know the theoretical maximum for normalizing any misuseability score calculation. Also of note is how "Medication" contains the same weights because the attribute value nodes were assigned the same "restricted" priority.

We can also observe that there are similar attribute values such as "Flu" in the *Disease* and *Initial Diagnosis* columns, but different weights are calculated for them. The sensitive attribute value weight for *Disease* has a higher sensitive weight because it is worse for people to know an actual diagnosis, as opposed to the knowing the potential diagnosis under the *Initial Diagnosis* attribute. The *tkl*-Data Model manages to capture this intuition by modeling these two attributes to be generalized under two different concepts: "HealthRecord" and "Health" when developing the model in Section 4.1. This differentiation between the concepts allows for different weights to be decided with pairwise comparison as determined in Section 4.2.1.

Table 4.2: Weights calculated from priorities of the *tkl*-Data Model in Figure 4.8 using the *tkl*-Score methodology to derive scores.

Disease	**Medication**	**Age**	**Initial Diagnosis**
$W(Flu) = 0.0864$	$W(Antibiotics) = 0.10432$	$W(30+) = 0.08$	$W(Migraine) = 0.05472$
$W(H1N1) = 0.3456$	$W(Paracetamol) = 0.10432$	$W(< 30) = 0.08$	$W(Flu) = 0.05472$
$W(Hypertension) = 0.3456$	$W(ARV) = 0.10432$		$W(Hypertension) = 0.10944$
$W(HIV) = 0.432$	$W(Tamiflu) = 0.10432$		$W(HIV) = 0.21888$
	$W(Statin) = 0.10432$		

4.3 Calculating *tkl*-Score

tkl-Score incorporates l-diversity as l-Distinguishing Factor, and t-closeness as t-Distinguishing Factor to account for attribute disclosure attacks in addition to identity disclosure attacks. It is important to also account for attribute disclosure as it may be difficult to be certain of an individual's identity, but attributes relating to an individual can still be disclosed when similar information about identities are grouped together.

The *tkl*-Score record score is defined as:

$$RS_{tkl_r} = \frac{DF_{t_r} + \sum_{A_{S_i} \in r} \text{weight}(A_{S_i})}{DF_{l_r}} \tag{4.1}$$

In the equation above, r is a record, DF_{l_r} is the l-Distinguishing Factor of a record, DF_{t_r} is the t-Distinguishing Factor of a record, and weight(A_{S_i}) is the weight of the ith sensitive attribute value of a record.

Section 4.3.1 introduces the intuition behind l-Distinguishing Factor that replaces the k-Distinguishing Factor of M-Score and L-Severity. Section 4.3.2 introduces the intuition behind t-Distinguishing Factor, which is added to the sum of sensitive attribute value weights in a record score.

It should be noted that the l-Distinguishing Factor and t-Distinguishing Factor, like the k-Distinguishing Factor, aim to quantify the identity and attribute uniqueness of records in a "lookup table" that contains all records related to a population. However, since it is difficult to obtain all records related to a population, the "lookup table" is approximated to be the source table of a published table.

4.3.1 *l*-Distinguishing Factor

Recall that l-diversity (Definition 9) is the measure used to determine how distinguishable an individual is based on attribute frequency in an equivalence class (a group of records with common quasi-identifier attribute values). An interesting property to note is that for the l-diversity of the smallest equivalence class in k-anonymity, k will never be less than l. This is proven below.

Proof. Let k be the k-anonymity of a dataset which is the smallest equivalence class with size k of rows. Assume that the l-diversity of the smallest equivalence class has $k < l$. Based on the assumption, let $l = k + 1$. Then based on the definition of l-diversity, there must be $k + 1$ unique attribute values. However, this is a contradiction as the size of the equivalence class must be $k + 1$ to have $k + 1$ unique attribute values. Now assume that the dataset has multiple sensitive attributes and therefore creates a combination of groupings using Definition 10 to find the multi-attribute l-diversity. The largest equivalence class of the combinations will still be at most be the size of the original equivalence class that is matched with only quasi-identifiers. The reason is that as more attributes need to be matched to form a grouping of records, the size of the groupings either remains the same as the attribute values are all the same, or the size of the grouping becomes smaller when the attribute values are different. Therefore, $l \leq k$. □

Recall that k-Distinguishing Factor is used to determine how distinguishable a record is in a source table for a misuseability score calculation, and is a divisor in the score equation of M-Score and L-Severity. If the k-Distinguishing Factor is large it signifies that there is less risk of identifiability as there are more records need to distinguish an identity from. The inverse, is that with a smaller k-Distinguishing Factor there is an increase to severity as an individual becomes more identifiable. Hence, we wish to capture the maximal severity by minimizing the k-Distinguishing Factor.

Since $l \leq k$, if we take the l-diversity of the multi-attribute l-diversity equivalence class containing the published record in the source table, instead of the k-anonymity of the equivalence class containing the published record in a misuseability score calculation, the k-anonymity metric for identity disclosure attacks of k-Distinguishing Factor will also be accounted for in addition to accounting for attribute disclosure attacks. Therefore in *tkl*-Score, l-Distinguishing Factor replaces the k-Distinguishing Factor factor used in M-Score and L-Severity.

Definition 15 (l-Distinguishing Factor)**.** The l-Distinguishing Factor of a

record in a published table is the minimal multi-attribute l-diversity (Definition 10) "equivalence class" in the source table that contains the record in the published table. If there are no quasi-identifiers to form an equivalence class in the source table, then the minimal multi-attribute l-diversity of the published table (forming equivalence classes based on the sensitive attributes only) is the l-Distinguishing Factor of a record.

For example, recall the illustrative example source Table 3.1. To find the l-Distinguishing Factor of record 5, we begin by getting the "equivalence classes" of the multi-attribute l-diversity on the source Table 3.1. Assuming that the quasi-identifiers are *Age*, *City*, and *Gender* while the sensitive attributes are *Disease*, *Medication*, *Age*, and *Initial Diagnosis*, the multi-attribute l-diversity "equivalence class" groupings containing record 5 paired with their unique sensitive attribute values are:

(Job, City, Gender, Medication, Age, Initial Diagnosis) : "Flu"

(Job, City, Gender, Disease, Age, Initial Diagnosis) : "Paracetamol"

(Job, City, Gender, Medication, Age, Initial Diagnosis) : "Flu"

(Job, City, Gender, Disease, Medication, Age) : "Migraine"

Since there is only one unique sensitive attribute per "equivalence class", the smallest multi-attribute l-diversity is 1, and therefore the l-Distinguishing Factor will be 1.

If we were to take the k-Distinguishing Factor of record 5, the equivalence class (grouped based on the quasi-identifiers: *Age*, *City*, and *Gender*) containing record 5 would contain: record 5, and record 6. Thus, the k-Distinguishing Factor would be 2 since that is the size of the equivalence class containing record 5. As a result, the misuseability score of record 5 calculated with l-Distinguishing Factor would be greater and therefore more distinguishable than a misuseability score of record 5 calculated with k-Distinguishing Factor.

4.3.2 *t*-Distinguishing Factor

Recall that t-closeness (Definition 11) provides a measurement for the similarity between the attribute value distribution of an equivalence class, and the attribute value distribution of an entire table. The similarity of distributions for sensitive attribute values helps to determine the true diversity of sensitive attributes globally. In comparison, l-diversity only considers the diversity of sensitive attributes within an equivalence class, which means sensitive attributes may still be prone to attribute disclosure if all the unique attributes are in a single equivalence class. As a result, t-closeness and l-diversity are two different anonymity measures that should be used together as a way to measure the anonymity of sensitive attributes in table.

The objective of integrating t-closeness into *tkl*-Score is to increase the relative severity of a exposure as the value of the t-closeness of the released records increase. This is because the higher the t-closeness, the higher the likelihood of an attribute disclosure attack as the distributions are less similar and sensitive attributes are easier to discern. Therefore, adding t to the record score is chosen as the best option.

The intuition behind adding t to the record score, comes from considering three options to increase the record sensitivity score keeping in mind that $t \in [0, 1]$. The three options considered for maximizing the record score using t are: adding t to the record score, subtracting t from the denominator of the record score, and dividing the record score by t. For each case, we assume to have a record score function $\frac{S}{DF}$ modeled after Equation 3.4 where $1 \geq S \geq 0$ and represents the sum of sensitive attribute value weights of a record, and $DF \geq 1$ represents the l-Distinguishing Factor.

By adding t to the numerator of the function to calculate a record score, it produces a linear translation of the score by $\frac{t}{DF}$ on the y-axis when visualized on a plot. For instance, if we consider the slope $y = \frac{S}{DF}$, then it follows that when we add t to the numerator $y = \frac{S+t}{DF} = \frac{S}{DF} + \frac{t}{DF}$. The y-intercept (by the definition of slope) is translated from 0 to $\frac{t}{DF}$ meaning that slope is also translated by $\frac{t}{DF}$. In the best case the record score is minimally reduced when

$t \approx 0$ (the attribute distributions are completely unique), and in the worst case when $t = 1$ (all the attributes are the same) the score increases by $\frac{t}{DF}$.

If we subtract t from the denominator $\frac{S}{DF-t}$ of the record score function, we encounter the issue of division by 0 if $DF = t$ as the function is hyperbolic by nature. Thus, subtracting from the denominator is not an ideal choice.

We can also divide the record score function by t as such: $\frac{S}{DF \cdot t}$. However, there is also the possibility of encountering division by 0 if $t = 0$.

Therefore, to avoid division by 0, the best option is adding t to the numerator of the record score function. As a result, the record store will either be maintained or increased by a factor t to indicate the severity of a record release more finely.

Definition 16 (t-Distinguishing Factor)**.** The t-Distinguishing Factor of a record in a published table is the maximal t-closeness of the equivalence class in the source table that contains the record in the published table. If there are no quasi-identifiers to form an equivalence class in the source table, then the t-Distinguishing Factor of a record is 0 as there are no equivalence classes to compare the distribution of attribute values.

For example, to get the t-Distinguishing Factor for row 0 of the illustrative example Table 3.1 assuming that *Job*, *City*, and *Gender* are the quasi-identifiers, and *Disease*, *Medication*, *Age*, and *Initial Diagnosis* are the sensitive attributes. First, determine the equivalence class that row 0 is in, which is an equivalence class with only itself in it.

Next, we get the global frequencies for each of the sensitive attribute values:

$$Q_{Disease} = \{\text{H1N1}, \text{Flu}, \text{HIV}, \text{Hypertension}\} = \{\tfrac{2}{7}, \tfrac{3}{7}, \tfrac{1}{7}, \tfrac{1}{7}\}$$

$$Q_{Medication} = \{\text{Paracetamol}, \text{Antibiotics}, \text{Tamiflu}, \text{Statin}, \text{ARV}\} = \{\frac{2}{7}, \frac{2}{7}, \frac{1}{7}, \frac{1}{7}, \frac{1}{7}\}$$

$$Q_{Age} = \{19, 23, 27, 29, 49, 70\} = \{\tfrac{1}{7}, \tfrac{1}{7}, \tfrac{1}{7}, \tfrac{2}{7}, \tfrac{1}{7}, \tfrac{1}{7}\}$$

$$Q_{InitialDiagnosis} = \{\text{Migraine}, \text{Flu}, \text{HIV}, \text{Hypertension}\} = \{\tfrac{3}{7}, \tfrac{2}{7}, \tfrac{1}{7}, \tfrac{1}{7}\}$$

And the frequencies of the sensitive attribute values for the equivalence class that contains row 0:

$$P_{Disease} = \{\text{H1N1}\} = \{\tfrac{1}{7}, 0, 0, 0\}$$

$$P_{Medication} = \{\text{Tamiflu}\} = \{0, 0, \tfrac{1}{7}, 0, 0\}$$

$$P_{Age} = \{27\} = \{0, 0, \tfrac{1}{7}, 0, 0, 0\}$$

$$P_{InitialDiagnosis} = \{\text{Flu}\} = \{0, \tfrac{1}{7}, 0, 0\}$$

Finally, the distances between the P and Q distributions are calculated using Equation 3.1 for the continuous attribute *Age*, and Equation 3.2 for the discrete attributes *Disease*, *Medication*, and *Initial Diagnosis*. The maximum of the distances calculated is the greatest t-closeness of the equivalence classes containing row 0.

The distances of the distributions are as follows: $E(P_{Age}, Q_{Age}) = 0.2381$; $E'(Q_{Disease}, P_{Disease}) = 0.7143$; $E'(Q_{Medication}, P_{Medication}) = 0.8571$; $E'(Q_{InitialDiagnosis}, P_{InitialDiagnosis}) = 0.7143$.

Therefore, the t-Distinguishing Factor is determined to be 0.8571.

4.3.3 Published Table Score

To calculate the *tkl*-Score of a published table, every record in the table is scored and summed together using sensitive attribute value weights derived from the *tkl*-Data Model.

Definition 17 (*tkl*-Score)**.** Given a published table T, the *tkl*-Score is calculated as: *tkl*-Score $= \sum_{r \in T} \left(\frac{DF_{t_r} + \sum_{A_{S_i} \in r} \text{weight}(A_{S_i})}{DF_{l_r}} \right)$ where for each record r, the total sum of each ith sensitive attribute value weight A_{S_i} (derived from the *tkl*-Data Model) is summed with the t-Distinguishing Factor DF_{t_r} of r and divided with the l-Distinguishing Factor DF_{l_r} of r. The total sum of each ith sensitive attribute weight, $\sum_{A_{S_i} \in r} \text{weight}(A_{S_i})$, is bounded to $[0, 1]$.

For instance, to calculate the published Table 3.6 *tkl*-Score of Scenario 1 described in Section 3.1, using the sensitive attribute value weights from Table 4.2, we first need to calculate the record scores for each row of Table 3.6.

As an example of a record score calculation, the row with id 3 can be calculated using Equation 4.1 as follows:

$$\begin{aligned} RS_{tkl_{id_3}} &= \frac{DF_{t_{id_3}} + (W(HIV_{Disease}) + W(ARV) + W(30+) + W(HIV_{Initial}))}{DF_{l_{id_3}}} \\ &= \frac{0.7143 + (0.432 + 0.10432 + 0.08 + 0.21888)}{1} \\ &= 1.5495 \end{aligned}$$

Repeating the process for each record we get the following record scores: row id 4 as 1.3536, row id 5 as 1.0397, and row id 6 as 1.0397. Summing up all the record scores, we get the *tkl*-Score of Scenario 1 to be:

$$tkl\text{-Score} = 1.5495 + 1.3536 + 1.0397 + 1.0397 = 4.983$$

When this process is repeated for Scenario 2 of the illustrative example, where the whole source table is released, we get $tkl\text{-Score} = 1.4418 + 1.2989 + 1.0397 + 1.5495 + 1.3536 + 1.0397 + 1.0397 = 8.763$ by iterating from row 0 to row 6 of Table 3.1. The score of Scenario 2 is much larger than Scenario 1 which indicates that releasing a bigger table has the potential to cause more severe consequences.

Similarity to *L*-Severity *tkl*-Score is similar to *L*-Severity as it computes the score of a published table by summing the corresponding scores of each released row. However, there are three key differences between the two scores.

First, *tkl*-Score uses a combination of AHP and domain expert assignment to derive sensitive attribute value weights as explained in Section 4.2 on the *tkl*-Data Model. The *tkl*-Data Model also does not have a dependency problem as explained in Section 3.3.6, because no "inference relationships" are defined. This is in contrast to *L*-Severity which includes "inference relationships" that have hard to define quantifiable values. In addition, *L*-Severity also only uses domain expert assignment in the process of assigning sensitivity values, as opposed to *tkl*-Score which uses AHP to systematically evaluate attribute sensitivity.

Next, since *tkl*-Score uses AHP to determine attribute sensitivity, the sum of sensitive attribute value weights for a record is bounded to $[0, 1]$. In comparison to *L*-Severity which does not have any bound for the sum of sensitive attribute value weights of a record.

Finally, *tkl*-Score accounts for attribute disclosure in addition to identity disclosure by incorporating l-Distinguishing Factor to increase the record score when sensitive attribute values are similar within an equivalence class of a source table containing the record; and t-Distinguishing Factor to increase the record score when the distribution of sensitive attribute values in an equivalence class of a source table containing the record are different from the distribution of sensitive attribute values in the source table. In contrast, *L*-Severity only accounts for identity disclosure with k-Distinguishing Factor which increases the record score when the set of records in a table are more identifiable.

Similarity to *M*-Score *tkl*-Score is similar to *M*-Score as the sum of sensitive attribute value weights are bounded to a maximum of 1 for each record score. However, the reasons for the maximal bound of the sum are different in both scores. The sum in *M*-Score is bounded to a maximum because it minimizes the sum of its sensitive attribute value weights for its record score to 1 as seen in its record score Equation 3.3. In *tkl*-Score, the sum is bounded to a maximum because the weights are derived using AHP on the *tkl*-Data Model.

Like *M*-Score, *tkl*-Score also derives attribute sensitivity in a systematic manner using AHP. However, *tkl*-Score uses the *tkl*-Data Model to derive sensitive attribute value weights while *M*-Score uses a three-level tree model. With the *tkl*-Data Model the number of pairwise comparisons can be reduced since AHP is only performed on the "domain taxonomy" and "sensitive attributes", while direct assignment and propagation are used on "sensitive attribute values". The use of propagation can reduce the amount of effort needed to assign a weight for each sensitive attribute value. In contrast, all "sensitive attribute values" of the three-level three model will need to be compared pairwise increasing the amount of time needed for deriving sensitive attribute

value weights.

Unlike M-Score, *tkl*-Score is non-approximative as it sums the record scores of a published table instead of multiplying the maximum record score of a published by the number of records in a published table. Thus, *tkl*-Score provides a more fine-grained severity of records in a published table instead of an estimated severity of records in a published table.

Also, similar to the *tkl*-Score comparison with L-Severity, *tkl*-Score incorporates l-Distinguishing Factor and t-Distinguishing Factor to account for attribute disclosure in addition to identity disclosure. In comparison, M-Score only accounts for identity disclosure with k-Distinguishing Factor.

Lastly, *tkl*-Score cannot account for the case where releasing any single maximum record score of the table is the maximum severity as there is no x parameter to set to $x \to \infty$ like M-Score. Instead, *tkl*-Score only accounts for the case that calculates the severity based on the number of records. Therefore, *tkl*-Score$_{\max}$ is introduced in the section below to account for the case where releasing any single maximum record score is the maximum severity.

4.3.4 Maximum Record Severity

tkl-Score$_{\max}$ is a score that is modeled after M-Score ($x \to \infty$) to signify that releasing any one maximum record score is the maximum severity. The difference of *tkl*-Score$_{\max}$ is that it also accounts for attribute disclosure because it incorporates l-Distinguishing Factor and t-Distinguishing Factor, instead of k-Distinguishing Factor like M-Score. An example of how *tkl*-Score$_{\max}$ differs from *tkl*-Score can be seen in Section 5.3 where the presence of a maximum record in two different published table sizes produce the same *tkl*-Score$_{\max}$ but different *tkl*-Scores.

Definition 18 (*tkl*-Score$_{\max}$)**.** Given a table with n records, the table's *tkl*-Score$_{\max}$ is then: *tkl*-Score$_{\max} = \max_{0 \le r \le n} \left(\frac{DF_{t_r} + \sum_{A_{S_i} \in r} \text{weight}(A_{S_i})}{DF_{l_r}} \right)$ where for each record r, the total sum of each ith sensitive attribute value weight A_{S_i} (derived from the *tkl*-Data Model) is summed with the t-Distinguishing Factor DF_{t_r} of r and divided with the l-Distinguishing Factor DF_{l_r} of r. The total

sum of each ith sensitive attribute weight, $\sum_{A_{S_i} \in r} \text{weight}(A_{S_i})$, is bounded to $[0, 1]$.

4.3.5 Considerations for Dynamic Data

tkl-Score is designed to be calculated for a source table that does not change when data is published. However, in the real-world the state of data is dynamic and may change with the addition of new rows or columns to a source table. Thus, the implications of data changes to the source table must be considered.

The run-time cost of *tkl*-Score depends on four factors: (1) the derivation of attribute value weights, (2) the record score of each record in a source table, (3) the distinguishing factors, and (4) the number of records in the published table.

The derivation of attribute value weights is by far the most time-consuming process as it requires human intervention to perform pairwise comparisons on the criteria of the *tkl*-Data Model which grows quadratically in the worst case [38]. If there are c criteria and the maximum amount of nodes in the criteria are m nodes, then the worst case of needed pairwise comparisons is $c \times m(m-1)/2$ comparisons which is $O(m^2)$ running time. As well, the direct assignment of qualitative priorities must be done on the attribute value nodes of the *tkl*-Data Model which is constant time as each node just needs to be assigned a weight which is linear and inconsequential to the pairwise comparisons of criteria. Finally, the attribute value weights are calculated from the priorities of the criteria and the qualitative priorities. This operation is also linear and inconsequential. Therefore, the overall running time of the derivation of weights is $O(m^2)$.

The record score of each record in a source table a constant $O(1)$ running time as it sums the attribute value weights for the attribute values of a record with the t-Distinguishing Factor of a record and divides the sum by the l-Distinguishing Factor.

The calculation of the distinguishing factors are dependent on two variables: s is defined as the number of sensitive attribute columns, and n is defined as the number of records in the source table. To calculate the

l-Distinguishing Factor it requires at most $s + 1$ groupings and n iterations of each record leading to $O((s + 1) \times n)$ running time. The s groupings comes from the multi-attribute l-diversity groupings, and the 1 comes from the k-anonymity equivalence class calculation. The t-Distinguishing Factor requires the same equivalence calculation as k-anonymity and n iterations of each record to get the attribute value frequency counts leading to $O(n)$ running time. Since these calculations can occur in parallel, the worst case running time is linear at $O((s + 1) \times n)$ as the number of sensitive attribute columns s remains constant unless a new sensitive attribute column is added.

Finally, to calculate *tkl*-Score the n record scores of a published table are summed together which is $O(n)$ running time.

This run-time analysis is similar to the analysis of M-Score by Harel et al. [18] which come to a conclusion that the worst case is linear when distinguishing factors need to be determined. It should be noted that analysis of M-Score does not consider the calculation of sensitive attribute value weights and assumes that they are predetermined and remain constant.

Modifying the Number of Columns The addition of new sensitive attribute columns to the source table will greatly affect the performance of *tkl*-Score as new weights will need to be derived again for the sensitive attribute columns, and new distinguishing factors will need to be calculated. This leads to a quadratic and linear operation that is unavoidable. If only additional quasi-identifier columns are added to the source table, the performance will not be as greatly affected as only new distinguishing factors and record scores will need to be calculated which is a linear operation.

If the number of columns are reduced in the source table, no modifications need to be made to the derived weights if the column removed was not sensitive or assume that releasing the other attribute values would be more severe. Otherwise, new sensitive attribute weights need to be calculated. In any case that the number of columns are reduced, the distinguishing factors will need to be recalculated with a linear operation.

Modifying the Number of Rows The addition or removal of rows from the source table requires the recalculation of distinguishing factors, and possibly require the calculation of new sensitive attribute value weights if there are new sensitive attribute values in the rows. The distinguishing factors will also have to be recalculated as well. The worst case overall is therefore still linear time as the calculation of new sensitive attribute value weights is constant time with direct assignment, and the calculation of the distinguishing factor is a linear operation.

4.4 Summary

In this chapter, the rationale behind *tkl*-Score and its methodology is explained. *tkl*-Score accounts for both identity and attribute disclosure attacks for by incorporating the PPDP measures t-closeness and l-diversity. Furthermore, it uses the underlying relationships of the data, modeled in the *tkl*-Data Model, to derive weights for sensitive attribute values. A derivative of *tkl*-Score, *tkl*-Score$_{\max}$, is also introduced to account for the case where releasing any one maximum record score is the maximum severity. In the next chapter, *tkl*-Score and *tkl*-Score$_{\max}$ are compared against M-Score and L-Severity.

Chapter 5

Comparison of Misuseability Scores

To compare *tkl*-Score and its derivative *tkl*-Score$_{\max}$ against its predecessors M-Score and L-Severity, three assumptions are considered: (1) the maximum severity is when a complete source table is released, (2) the maximum severity is when any one record with the maximum record score in the source table is released, (3) the maximum severity is when a score reaches the theoretical maximum score. With assumption (1) the severity is related to the amount of records that are released, and therefore *tkl*-Score, L-Severity, and M-Score ($x = 1$) are compared. With assumption (2) the severity is related to the maximum record of a source table, and therefore *tkl*-Score$_{\max}$ and M-Score ($x \to \infty$) are compared. For assumption (3) any misuseability score can be used, but it is best to be used with *tkl*-Score$_{\max}$ and M-Score ($x \to \infty$) as illustrated in Section 5.4.

Using assumption (1), a case is made for needing t-Distinguishing Factor in Section 5.1, and a case is also made for needing l-Distinguishing Factor in Section 5.2.

In Section 5.3, we refer back to the illustrative example of Section 3.1 to see how an increasing amount of records with the same subset of records in a published table contributes to a greater or equal misuseability score for assumption (1) and (2).

In Section 5.4, a case is made for why it can be beneficial to normalize against the theoretical maximum to use as an indicator of severity using as-

sumption (3).

Normalizing Against the Source Table Score Harel et al. [18] suggest that M-Score can be normalized by taking the M-Score of the published table and dividing by the M-Score of the source table. This form of normalization assumes that releasing a complete source table is the maximum severity — when a subset of a source table is published, it will take a percentage of the source table score.

Therefore, one method of normalization that would be appropriate for comparing the *tkl*-Score against its predecessors is by dividing the misuseability score of a published table by the score of the complete source table. The normalized score obtained will be in the domain of $[0, 1]$.

It should be noted that scores normalized against one source table should not be compared to a score normalized against a different source table. The reason is that source tables can have different sensitivities of attributes, can vary in size, and can have different attribute values meaning that the records may have different distinguishing factors.

5.1 A Case for *t*-Distinguishing Factor

The t-Distinguishing Factor is used to determine a record's distinguishability based on the similarity of the distribution of sensitive attributes in the equivalence class of the source table containing the record and the distribution of sensitive attributes in the whole table. Recall that the higher the t value, the higher the severity a record would be as the distribution of the values of the sensitive attributes and the whole table are more different and therefore easier to differentiate.

To illustrate how t-Distinguishing Factor helps to distinguish records, the record scores for the source Table 5.1a are calculated in Table 5.1c. From the normalized scores in Table 5.1b, we can see that: (i) *tkl*-Score is reduced from row 5 to row 1, while M-Score ($x = 1$) and L-Severity maintains consistency for rows 0, 1, and 5; and (ii) the *tkl*-Score is reduced from row 2 to 4, while

M-Score ($x = 1$) and L-Severity are increased. These observations can be visualized in Figure 5.1.

It should be noted that the M-Score (regardless of x parameter) and L-Severity calculated for each record seen in Table 5.1c are the same because there is only a single record for the misuseability score calculation. However, if we were to calculate the misuseability scores for a larger subset of published records, M-Score ($x = 1$), M-Score ($x \to \infty$), and L-Severity will produce different results.

id	Job	City	Gender	Initial Diagnosis
0	Lawyer	Calgary	Female	Flu
1	Lawyer	Calgary	Female	Migraine
2	Lawyer	Edmonton	Male	HIV
3	Lawyer	Edmonton	Male	Hypertension
4	Lawyer	Edmonton	Female	HIV
5	Lawyer	Edmonton	Female	Migraine

(a) Source table where the sensitive attribute is *Initial Diagnosis*, and the quasi-identifiers are *Job*, *City*, and *Gender*.

Row	***tkl*-Score**	***M*-Score** ($x = 1$)	***L*-Severity**
0	0.1642	0.0417	0.0769
1	0.1642	0.0417	0.0769
5	0.1149	0.0417	0.0769
3	0.1804	0.0833	0.1538
2	0.2128	0.1667	0.3077
4	0.1635	0.1667	0.3077

(b) Misuseability scores for each row of Table 5.1a normalized against the source table score rounded to four decimal places.

Row	***tkl*-Score**	***M*-Score**	***L*-Severity**	DF_t	DF_k	DF_l	Weights
0	0.27736	0.02736	0.02736	0.50000	2	2	0.05472
1	0.27736	0.02736	0.02736	0.50000	2	2	0.05472
5	0.19403	0.02736	0.02736	0.33333	2	2	0.05472
3	0.30472	0.05472	0.05472	0.50000	2	2	0.10944
2	0.35944	0.10944	0.10944	0.50000	2	2	0.21888
4	0.27611	0.10944	0.10944	0.33333	2	2	0.21888

(c) For each row of Table 5.1a: the raw misuseability scores, distinguishing factors, and sum of sensitive attribute value weights (Table 4.2) rounded to five decimal places.

Table 5.1: A case for using t-Distinguishing Factor.

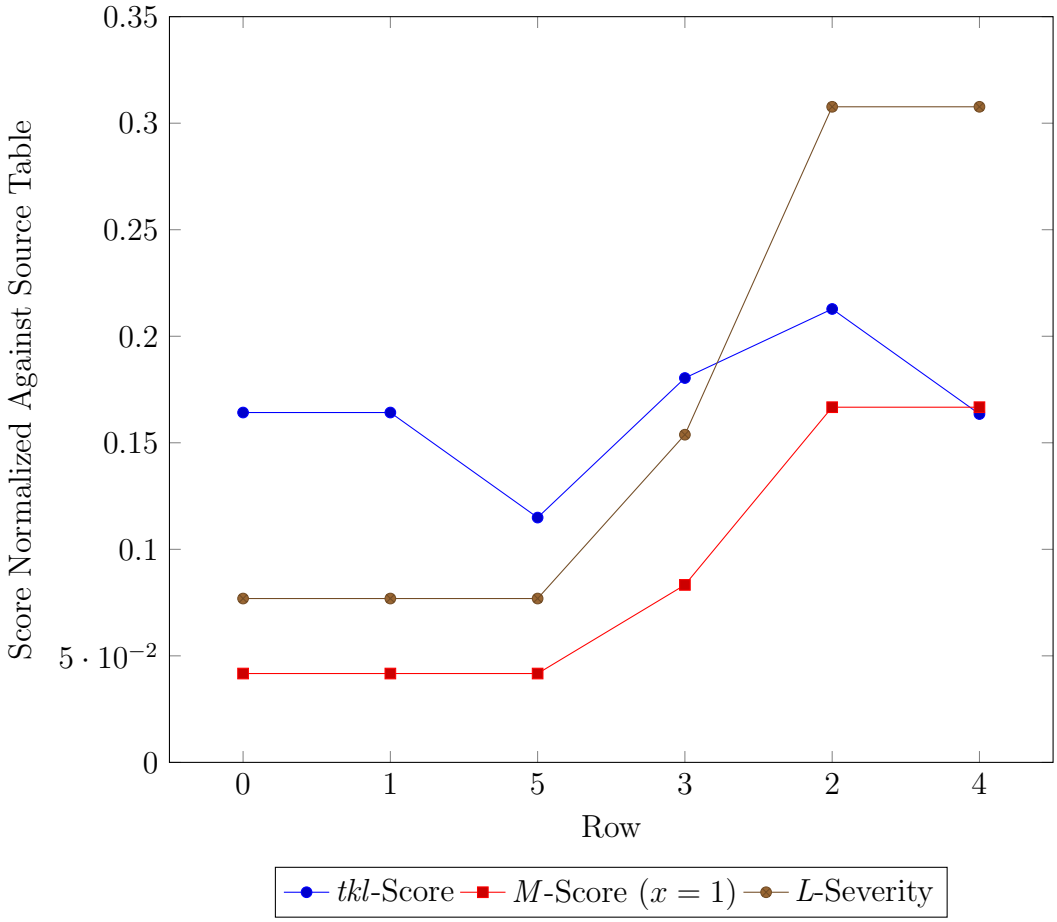


Figure 5.1: Line chart of per row scores normalized against the source table score for *tkl*-Score, M-Score ($x = 1$), and L-Severity of the records in Table 5.1a. It is sorted ascending by L-Severity.

The differences of (i) and (ii) stem from the equivalence class consisting of rows 4 and 5 in Table 5.1a. The distribution of the *Initial Diagnosis* values in the equivalence consisting of rows 4 and 5 has a lower t value compared to the other equivalence classes of the table as "HIV" and "Migraine" occur elsewhere in the table. In contrast, the equivalence class with rows 0 and 1, and the equivalence class with rows 2 and 3 have a unique value that do not appear in any other equivalence class leading to a higher t value. Therefore, rows 4 and 5 have a lower t-Distinguishing Factor than the other records, and as a result rows 4 and 5 considered less severe by *tkl*-Score than the other records in the table that have the same l-Distinguishing Factor and sum of sensitive attribute value weights as seen in Table 5.1c.

Intuitively, if we were to observe the attribute values of Table 5.1a, we can see that either a female lawyer from Edmonton or a male lawyer from Edmonton has "HIV". Likewise, either a female lawyer from Edmonton or a female lawyer from Calgary has a "Migraine". Since "HIV" and "Migraine" must be distinguished from two distinct equivalence classes, it is less likely that these sensitive attributes will be inferred as opposed to "Flu" and "Hypertension" which occur only in a single equivalence class.

5.2 A Case for *l*-Distinguishing Factor

The l-Distinguishing Factor is used to determine a record's distinguishability based on the size of the equivalence class that contains the record in the source table and also the values of sensitive attributes in the equivalence class that contains the record in the source table. Recall that as the l-Distinguishing Factor is part of the denominator of a record score (Equation 4.1), and therefore the smaller the l-Distinguishing Factor, the more the potential implications of releasing a record. So if there are more unique sensitive attribute values in an equivalence class, the harder it will be to link specific sensitive attributes to the identities of an equivalence class.

To illustrate how t-Distinguishing Factor can distinguish records, the record scores for the source Table 5.2a are calculated in Table 5.2c. From the normalized scores in Table 5.2b, we can see that M-Score and L-Severity maintain consistent scores for rows 0, 2, 4, and 5 while *tkl*-Score has greater scores in rows 4 and 5 compared to rows 0 and 2. This observation can be visualized in Figure 5.2.

We can account for the discrepancies between rows 4, 5 and rows 0, 2 in *tkl*-Score by observing its l-Distinguishing Factor and t-Distinguishing Factor values. In Table 5.2c, we see that the t-Distinguishing Factor values of rows 4, 5 are twice as much as the t-Distinguishing Factor values of rows 0, 2. This increases *tkl*-Score additively, but does not explain why the *tkl*-Score of rows 4,5 are more than double rows 0, 2. To explain this multiplicative doubling effect, we can look at the l-Distinguishing Factor values of the rows.

id	Job	City	Gender	Initial Diagnosis
0	Lawyer	Calgary	Female	HIV
1	Lawyer	Calgary	Female	Flu
2	Lawyer	Edmonton	Male	HIV
3	Lawyer	Edmonton	Male	Flu
4	Lawyer	Edmonton	Female	HIV
5	Lawyer	Edmonton	Female	HIV

(a) Source table where the sensitive attribute is *Initial Diagnosis*, and the quasi-identifiers are *Job*, *City*, and *Gender*.

Row	***tkl*-Score**	***M*-Score** ($x = 1$)	***L*-Severity**
1	0.0647	0.0417	0.0556
3	0.0647	0.0417	0.0556
0	0.1126	0.1667	0.2222
2	0.1126	0.1667	0.2222
4	0.3227	0.1667	0.2222
5	0.3227	0.1667	0.2222

(b) Misuseability scores for each row of Table 5.2a normalized against the source table rounded to four decimal places.

Row	***tkl*-Score**	***M*-Score**	***L*-Severity**	$\boldsymbol{DF_t}$	$\boldsymbol{DF_k}$	$\boldsymbol{DF_l}$	Weights
1	0.11069	0.02736	0.02736	0.16667	2	2	0.05472
3	0.11069	0.02736	0.02736	0.16667	2	2	0.05472
0	0.19277	0.10944	0.10944	0.16667	2	2	0.21888
2	0.19277	0.10944	0.10944	0.16667	2	2	0.21888
4	0.55221	0.10944	0.10944	0.33333	2	1	0.21888
5	0.55221	0.10944	0.10944	0.33333	2	1	0.21888

(c) For each row of Table 5.2a: the raw misuseability scores, distinguishing factors, and sum of sensitive attribute value weights (Table 4.2) rounded to five decimal places.

Table 5.2: A case for using l-Distinguishing Factor

From the equivalence class containing rows 4,5 as seen in Table 5.2a, it can be observed that "HIV" is the only unique sensitive attribute in the equivalence class. In every other equivalence class, there are two unique values. This means that if we knew a female lawyer was from Edmonton, we can deduce that they have HIV if the table were to be released, as opposed to having to distinguish between two different sensitive attribute values with the other equivalence class groupings. Because the equivalence class containing rows 4,5 have only "HIV" as the sensitive attribute, the l-Distinguishing Factor for rows 4,5 is 1. As a

result, the record scores of rows 4,5 are not reduced by a factor of 2 like rows 0, 2.

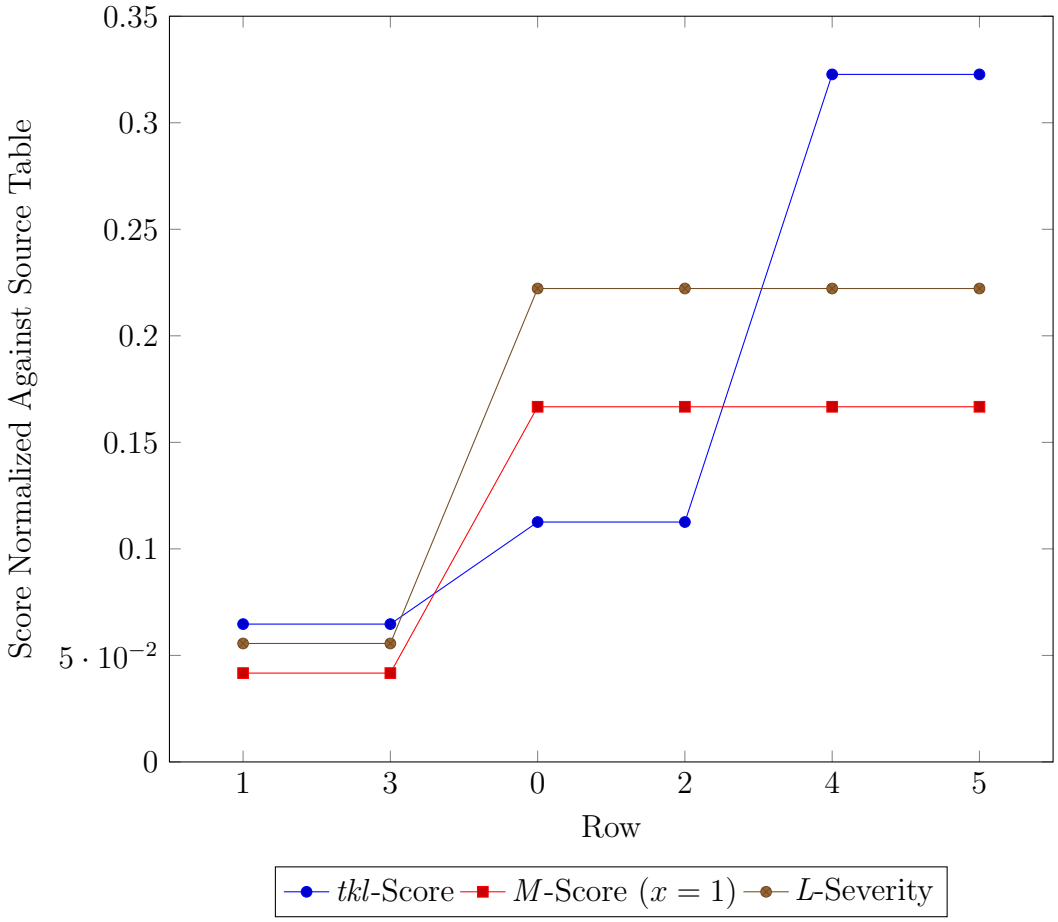


Figure 5.2: Line chart of per row scores normalized against the source table score for *tkl*-Score, *M*-Score ($x = 1$), and *L*-Severity of the records in Table 5.2a. It is sorted ascending by *L*-Severity.

5.3 Scoring the Illustrative Example

Recall the illustrative example in Section 3.1 involving the source Table 3.1 where the rows with ids 3, 4, 5, and 6 are published in Scenario 1, and the complete source table is published in Scenario 2. The misuseability scores are calculated for the published table of Scenario 1 in Table 5.3a, and for the published source table of Scenario 2 in Table 5.3b.

id	Job	City	Gender	Disease	Medication	Age	Initial Diagnosis
3	Lawyer	Edmonton	Male	HIV	ARV	49	HIV
4	Lawyer	Edmonton	Male	Hypertension	Statin	70	Hypertension
5	Lawyer	Edmonton	Female	Flu	Paracetamol	29	Migraine
6	Lawyer	Edmonton	Female	Flu	Paracetamol	29	Migraine

Normalized Against Source Table	Raw Misuseability Scores
tkl-Score = 0.5686	tkl-Score = 4.98258
tkl-Score$_{\max}$ = 1.0000	tkl-Score$_{\max}$ = 1.54949
M-Score ($x = 1$) = 0.4082	M-Score ($x = 1$) = 1.67040
M-Score ($x \to \infty$) = 0.7143	M-Score ($x \to \infty$) = 0.41760
L-Severity = 0.5055	L-Severity = 1.06272

(a) Published subset of Table 3.1 consisting of the rows with ids 3, 4, 5, and 6.

id	Job	City	Gender	Disease	Medication	Age	Initial Diagnosis
0	Lawyer	Calgary	Male	H1N1	Tamiflu	27	Flu
1	Lawyer	Calgary	Female	H1N1	Antibiotics	19	Flu
2	Lawyer	Calgary	Female	Flu	Antibiotics	23	Migraine
3	Lawyer	Edmonton	Male	HIV	ARV	49	HIV
4	Lawyer	Edmonton	Male	Hypertension	Statin	70	Hypertension
5	Lawyer	Edmonton	Female	Flu	Paracetamol	29	Migraine
6	Lawyer	Edmonton	Female	Flu	Paracetamol	29	Migraine

Normalized Against Source Table	Raw Misuseability Scores
tkl-Score = 1.0000	tkl-Score = 8.76302
tkl-Score$_{\max}$ = 1.0000	tkl-Score$_{\max}$ = 1.54949
M-Score ($x = 1$) = 1.0000	M-Score ($x = 1$) = 4.09248
M-Score ($x \to \infty$) = 1.0000	M-Score ($x \to \infty$) = 0.58464
L-Severity = 1.0000	L-Severity = 2.10240

(b) Published source Table 3.1.

Row	tkl-Score	tkl-Score$_{\max}$	M-Score	L-Severity	DF_t	DF_k	DF_l	Weights
2	1.03973	1.03973	0.16272	0.16272	0.71429	2	1	0.32544
5	1.03973	1.03973	0.16272	0.16272	0.71429	2	1	0.32544
6	1.03973	1.03973	0.16272	0.16272	0.71429	2	1	0.32544
1	1.29893	1.29893	0.29232	0.29232	0.71429	2	1	0.58464
4	1.35365	1.35365	0.31968	0.31968	0.71429	2	1	0.63936
3	1.54949	1.54949	0.41760	0.41760	0.71429	2	1	0.83520
0	1.44178	1.44178	0.58464	0.58464	0.85714	1	1	0.58464

(c) For each row of Table 3.1: the raw misuseability scores, distinguishing factors, and sum of sensitive attribute weights rounded to five decimal places.

Table 5.3: The misuseability scores for: (a) Scenario 1 of the illustrative example, (b) Scenario 2 of the illustrative example, and (c) the records of the illustrative example in Section 3.1. The scores are calculated using the sensitive attribute value weights of Table 4.2. The sensitive attributes of the source table are *Disease*, *Medication*, *Age*, and *Initial Diagnosis*, and the quasi-identifiers are *Job*, *City*, and *Gender*.

The scores of these scenarios illustrate that the misuseability score of a

superset of records is either greater than or equal to the subset of records. For example, by comparing the misuseability scores normalized against the source table score in Table 5.3a to the raw misuseability scores in Table 5.3c, we can see that the individual raw misuseability scores of rows 3, 4, 5, and 6 are smaller than the scores of the published table containing the rows together. We can also see that the scores of Table 5.3a (a subset of Table 5.3b) are smaller than Table 5.3b. It should also be noted that *tkl*-Score and *tkl*-Score$_{\max}$ are the same in Table 5.3c because there is only a single record for the misuseability score calculation; likewise for M-Score (regardless of x parameter) and L-Severity.

We can also see that *tkl*-Score$_{\max}$ is the maximum severity in both scenarios as its score normalized against the source table score is 1. This does not occur for any other misuseability score. Recall that *tkl*-Score$_{\max}$ and M-Score ($x \to \infty$) account for assumption (2) where releasing any single maximum record score of the source table is the maximum severity. Therefore, *tkl*-Score$_{\max}$ and M-Score ($x \to \infty$) report the maximum severity in different published subsets as long as the published subset has at least one maximum record of the source table. On the other hand, *tkl*-Score, L-Severity, and M-Score ($x = 1$) — which account for assumption (1) — will never reach maximum severity when normalized against the source table until all records of the source table are released in the published table since these misuseability scores rely on table size as part of their calculations.

The reason that *tkl*-Score$_{\max}$ has the maximum severity in both scenarios can also be explained by looking at the record score of row 3 which is considered to be more severe than row 0 as seen in Table 5.3c. When looking at the sensitive attribute values of the rows in Table 5.3b, it can be seen that row 3 has more sensitive attribute values such as "HIV" compared to row which has attribute values such as "Flu". This higher sensitivity is reflected in the sum of weights as seen in Table 5.3c. However, for M-Score and L-Severity it gets reduced multiplicatively by the k-Distinguishing Factor that is 2, compared to the l-Distinguishing Factor of *tkl*-Score and *tkl*-Score$_{\max}$ that is 1.

5.4 Indicating Severity

Recall that the previous sections of this chapter only normalizes the misuseability score against the source table score. We can also normalize against the theoretical maximum as described below. However, to compare any of the scores normalized against the theoretical maximum, we must assume that the source table used to calculate the misuseability score is the best possible approximation of a "lookup table" and the sensitive attribute value weights of similar values reflect a similar importance.

Normalizing Against the Theoretical Maximum A misuseability score can be normalized against the theoretical maximum by taking the misuseability score for a published table and dividing by the theoretical maximum score for the number of records in the published table. The normalized score obtained will be in the domain of $[0, 1]$. To calculate the theoretical maximum score of a table, the theoretical maximum score for an individual record is multiplied by the number of records in the table.

The theoretical maximum scores for an individual record can be determined when all the upper and lower bounds needed to maximize the record score are known. Thus, if the sensitive attribute value weights are derived using AHP as part of its process in Section 4.2, the sum of elicited weights can be bounded to maximum of 1. As well, the t-Distinguishing Factor is in the domain of $[0, 1]$ and therefore it is also bounded to a maximum of 1. The lower bounds of l-Distinguishing Factor and k-Distinguishing Factor is at least 1 since that is the size of the smallest equivalence class containing a single record in the source table. Therefore, the theoretical maximum scores for an individual record in each of the misuseability scores are:

- M-Score$_{\text{record}}$ (regardless of x) $= \frac{\min(1,\max(weights))}{\min(DF_k)} = \frac{1}{1} = 1$
- L-Severity$_{\text{record}} = \frac{\max(weights)}{\min(DF_k)} = \frac{1}{1} = 1$
- tkl-Score$_{\text{record}} = tkl$-Score$_{\max_{\text{record}}} = \frac{\max(DF_t)+\max(weights)}{\min(DF_l)} = \frac{1+1}{1} = 2$

The normalization against the theoretical maximization, also "rescales" misuseability scores as it is modeled after min-max normalization where: $\frac{score-\min(score)}{\max(score)-\min(score)} = \frac{score-0}{\max(score)-0} = \frac{score}{\max(score)}$.

It should be noted that the normalization against a theoretical maximum using *tkl*-Score, M-Score ($x = 1$), and L-Severity is difficult to comprehend as the tables may have the same number of records, but different scores for the records. For example, consider the two different published tables with four records where the theoretical maximum score is 1:

$$\text{Table}_1 = \frac{1 + 0.5 + 0.\overline{3} + 0.\overline{3} + 0.\overline{3}}{1 \times 5} = \frac{1}{2}$$
$$\text{Table}_2 = \frac{0.5 + 0.5 + 0.5 + 0.5 + 0.5}{1 \times 5} = \frac{1}{2}$$

If we normalize against the theoretical maximum the differences, the tables both produce the same normalized score. However, it should be noted that the differences between the record scores are ignored even though one table may release more severe records than the other.

Therefore, if we were to rely on the normalization against the theoretical maximum as a metric of severity for assumption (3) using *tkl*-Score, L-Severity, or M-Score ($x = 1$), we cannot say that one set of records is more severe than the other. However, it can be a good indicator of severity. For instance, if the normalization is closer to 1, the set of records in the published table are likely to have larger record scores as opposed to a normalization closer to 0 where the set of records are likely to have smaller record scores.

If we use *tkl*-$\text{Score}_{\max}$ or M-Score ($x \to \infty$) with assumption (3), the normalization against the theoretical maximum allows for the comparison between different tables as the "size" of the published table will always be 1. For example, we can say M-Score ($x \to \infty$) in Table 5.3b is more severe than the M-Score ($x \to \infty$) of record 3 calculated in Table 5.1c: $0.58464 > 0.05472$.

However, there is a caveat to the comparison as the t-Distinguishing Factor, k-Distinguishing Factor, and l-Distinguishing Factor of each record score are approximated with the source table of a subset of records. This is because it is difficult to obtain a complete database with records related to a population for determining the distinguishability of a record. Therefore, we must assume

that record scores calculated for each of the tables compared have been calculated with the best possible "lookup table". As well, we have to assume that the sensitive attribute value weights of similar values reflect a similar importance if the tables being compared had their sensitive attribute value weights derived independently. With these assumptions, we can also compare across misuseability scores as we are also "rescaling" the misuseability scores to the domain of $[0, 1]$. For example, if we refer back to Table 5.3b we can say that the *tkl*-Score$_{\max}$ considers the records to be more severe than M-Score $(x \to \infty)$ as: $\frac{1.54949}{2} > \frac{0.41760}{1}$.

5.5 Summary

In this chapter, the maximum severity when a complete source table is release is considered by using *tkl*-Score, L-Severity, and M-Score $(x = 1)$. The maximum severity when any one maximum record score in a source table is released is considered by using *tkl*-Score$_{\max}$ and M-Score. Lastly, the severity relative to a theoretical maximum is also considered and can be used as an indicator when the scores are calculated with *tkl*-Score, L-Severity, and M-Score $(x = 1)$. It can also be used with *tkl*-Score$_{\max}$ or M-Score $(x \to \infty)$ to compare misuseability scores from different tables under the assumptions that the best possible "lookup tables" were used to find the distinguishability of records, and the sensitive attribute value weights of similar values from different tables reflect a similar importance.

Two cases are presented to demonstrate how t-Distinguishing Factor and l-Distinguishing Factor can better distinguish records than k-Distinguishing Factor and make record scores more granular. Because of these new distinguishing factors, *tkl*-Score and *tkl*-Score$_{\max}$ are better at characterizing the severity of records compared to L-Severity and M-Score. It is demonstrated that when more records are added to a published table, the misuseability score becomes greater or equal to the score of the previous published table.

Therefore, *tkl*-Score is a promising misuseability score that can be used in-

place of L-Severity or M-Score ($x = 1$) to determine the severity of a releasing a published table based on the amount of records in it. tkl-Score$_{\max}$ is a promising misuseability score that can be used in-place of M-Score ($x \to \infty$) to decide the severity of releasing a published table by assuming that releasing any one maximum record score is the maximum severity.

Chapter 6

Conclusion

This thesis explores the privacy/sensitivity aware paradigm for data access and sharing to motivate the need for misuseability scoring. Misuseability scoring enables comparisons between different datasets concerning the severity of a release by quantifying the sensitivity of data. *tkl*-Score is presented as an improvement to existing misuseability scoring by accounting for attribute disclosure attacks in addition to identity disclosure attacks. Furthermore, *tkl*-Score introduces a new systematic procedure to elicit weights for sensitive attribute values using the *tkl*-Data Model. This procedure uses the underlying concepts and relationships of sensitive attributes in a dataset to decide on the appropriate weights.

The contributions of this thesis can be summarized follows:

A new misuseability score ***tkl*-Score** is designed (Chapter 4) to better characterize the severity of records, in a more fine-grained manner, by accounting for attribute and identity disclosure using l-Distinguishing Factor (Section 4.3.1) and t-Distinguishing Factor (Section 4.3.2). A derivative ***tkl*-Score$_{\mathbf{max}}$** is designed to account for the worst case severity of releasing any one maximum record score from a set of published records (Section 4.3.4).

A **Data Sensitivity Ontology (DSO)** is created to manage the metadata of sensitive attributes of a source table (Section 4.1.2).

A new **systematic procedure to elicit sensitive attribute value weights** is demonstrated using the *tkl*-Data Model (Section 4.2). As well, a **propagation mechanism** is introduced to reduce the amount of human

intervention needed to elicit weights (Section 4.2.3).

Future work includes investigating applications of *tkl*-Score and *tkl*-Score$_{\max}$ such as being incorporated into access control frameworks for data privacy compliance when delegating access. Another application is using *tkl*-Score and *tkl*-Score$_{\max}$ to quantify the sensitivity of records leaked in a data breach and indicate the extent of a breach. Furthermore, *tkl*-Score and *tkl*-Score$_{\max}$ could be integrated with data loss prevention systems to monitor for any anomalies in user behaviour when accessing database data by calculating a score each time data is accessed by a user and identifying and extreme scores. *tkl*-Score and *tkl*-Score$_{\max}$ could also be used as part of a risk assessment process to determine, based on a score, which sensitive records could cause issues when released.

As well, other aspects to explore include improving the sensitive attribute value weight elicitation process by extending the *tkl*-Data Model to include more metadata such as data handling and governance policies to provide additional context for decisions during weight elicitation. Moreover, the effectiveness of automated ontology alignment techniques should be investigated to help align preexisting ontologies for the domain taxonomy of the *tkl*-Data Model.

References

[1] A. Acquisti, I. Adjerid, R. Balebako, L. Brandimarte, L. F. Cranor, S. Komanduri, P. G. Leon, N. Sadeh, F. Schaub, M. Sleeper, *et al.*, "Nudges for Privacy and Security: Understanding and Assisting Users' Choices Online," *ACM Computing Surveys (CSUR)*, vol. 50, no. 3, p. 44, 2017. 10

[2] K. Z. Bijon, R. Krishnan, and R. Sandhu, "A Framework for Risk-Aware Role Based Access Control," in *2013 IEEE Conference on Communications and Network Security (CNS)*, IEEE, 2013, pp. 462–469. 12

[3] I. Bilogrevic and M. Ortlieb, "'If You Put All The Pieces Together...': Attitudes Towards Data Combination and Sharing Across Services and Companies," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, ACM, 2016, pp. 5215–5227. 6

[4] S. Bishop, "The Internet of Things: Implications for Consumer Privacy Security," in *2019 IEEE 12th International Conference on Global Security, Safety and Sustainability (ICGS3)*, Jan. 2019, pp. 1–9. DOI: 10.1109/ICGS3.2019.8688024. 7

[5] L. Chen and J. Crampton, "Risk-Aware Role-Based Access Control," in *International Workshop on Security and Trust Management*, Springer, 2011, pp. 140–156. 12

[6] M. Crain, "The limits of transparency: Data brokers and commodification," *New Media & Society*, vol. 20, no. 1, pp. 88–104, 2018. DOI: 10.1177/1461444816657096. 7

[7] P. K. Dey, "Analytic Hierarchy Process Analyzes Risk of Operating Cross-Country Petroleum Pipelines in India," *Natural Hazards Review*, vol. 4, no. 4, pp. 213–221, 2003. 52

[8] S.-A. Elvy, "Commodifying Consumer Data in the Era of the Internet of Things," *BCL Rev.*, vol. 59, p. 423, 2018. 7

[9] K. Eng, D. Serrano, E. Stroulia, and J. Jaremko, "(Semi)Automatic Construction of Access-Controlled Web Data Services," in *Proceedings of the 28th Annual International Conference on Computer Science and Software Engineering*, IBM Corp., 2018, pp. 72–80. iii, 9

[10] FIRST, *CVSS v3.0 Specification Document*. [Online]. Available: https://www.first.org/cvss/v3.0/specification-document (visited on 08/07/2019). 12

[11] M. J. Foley, "Microsoft's new 'Data Dignity' team could help users control their personal data," *ZDNet*, Sep. 23, 2019. [Online]. Available: https://www.zdnet.com/article/microsofts-new-data-dignity-team-could-help-users-control-their-personal-data/ (visited on 11/07/2019). 2

[12] B. C. M. Fung, K. Wang, R. Chen, and P. S. Yu, "Privacy-preserving Data Publishing: A Survey of Recent Developments," *ACM Comput. Surv.*, vol. 42, no. 4, 14:1–14:53, Jun. 2010. DOI: 10.1145/1749603.1749605. 17

[13] Gartner, "Forecast: Internet of Things - Endpoints and Associated Services, Worldwide, 2017," Dec. 21, 2017. [Online]. Available: https://www.gartner.com/en/documents/3840665/forecast-internet-of-things-endpoints-and-associated-ser (visited on 08/07/2019). 1

[14] C. R. Givens, R. M. Shortt, *et al.*, "A class of Wasserstein metrics for probability distributions.," *The Michigan Mathematical Journal*, vol. 31, no. 2, pp. 231–240, 1984. 21

[15] E. Graham-Harrison and C. Cadwalladr, "Revealed: 50 million Facebook profiles harvested for Cambridge Analytica in major data breach," *The Guardian*, Mar. 17, 2018. [Online]. Available: https://www.theguardian.com/news/2018/mar/17/cambridge-analytica-facebook-influence-us-election (visited on 08/07/2019). 1

[16] I. Hadar, T. Hasson, O. Ayalon, E. Toch, M. Birnhack, S. Sherman, and A. Balissa, "Privacy by designers: software developers' privacy mindset," *Empirical Software Engineering*, vol. 23, no. 1, pp. 259–289, Feb. 2018. DOI: 10.1007/s10664-017-9517-1. 5

[17] A. Harel, A. Shabtai, L. Rokach, and Y. Elovici, "Dynamic Sensitivity-based Access Control," in *Proceedings of 2011 IEEE International Conference on Intelligence and Security Informatics*, IEEE, 2011, pp. 201–203. 12

[18] A. Harel, A. Shabtai, L. Rokach, and Y. Elovici, "M-score: A Misuseability Weight Measure," *IEEE Transactions on Dependable and Secure Computing*, vol. 9, no. 3, pp. 414–428, 2012. 24, 68, 71

[19] D. Ho, G. Newell, and A. Walker, "The importance of property-specific attributes in assessing CBD office building quality," *Journal of Property Investment & Finance*, vol. 23, no. 5, pp. 424–444, 2005. 92

[20] S. Kullback and R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951. 21

[21] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness: Privacy beyond k-anonymity and l-diversity," in *2007 IEEE 23rd International Conference on Data Engineering*, IEEE, 2007, pp. 106–115. 18, 21

[22] Loudhouse, *A European study on the nature of consumer trust and personal data*, Feb. 2014. [Online]. Available: https://www.orange.com/en/content/download/21358/412063/version/5/file/Orange%20Future%20of%20Digital%20Trust%20Report.pdf (visited on 08/07/2019). 6

[23] Loudhouse, *A European study on the nature of consumer trust and personal data*, Sep. 2014. [Online]. Available: https://www.orange.com/content/download/25973/582245/version/2/file/Report%20-%20My%20Data%20Value%20-%20Orange%20Future%20of%20Digital%20Trust%20-%20FINAL.pdf (visited on 08/07/2019). 6

[24] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 1, no. 1, 3–es, 2007. 19, 20

[25] J. McCaffrey, "Test Run: The Analytic Hierarchy Process," *MSDN Magazine*, Jun. 2005. [Online]. Available: https://docs.microsoft.com/en-us/archive/msdn-magazine/2005/june/test-run-the-analytic-hierarchy-process (visited on 10/18/2019). 52

[26] R. Nget, Y. Cao, and M. Yoshikawa, "How to balance privacy and money through pricing mechanism in personal data market," in *Proceedings of the SIGIR 2017 Workshop On eCommerce, eCOM@SIGIR 2017, Tokyo, Japan, August 11, 2017*, ser. CEUR Workshop Proceedings, vol. 2311, CEUR-WS.org, 2017. [Online]. Available: http://ceur-ws.org/Vol-2311/paper%5C_15.pdf. 5

[27] L. Obrst, W. Ceusters, I. Mani, S. Ray, and B. Smith, "The Evaluation of Ontologies," in *Semantic Web*, Springer, 2007, pp. 139–158. 39

[28] M. I. S. Oliveira and B. F. Lóscio, "What is a Data Ecosystem?" In *Proceedings of the 19th Annual International Conference on Digital Government Research: Governance in the Data Age*, ACM, 2018, p. 74. 5, 7–9

[29] "Ontologies," *W3C*, [Online]. Available: https://www.w3.org/standards/semanticweb/ontology.html (visited on 08/07/2019). 38

[30] "Ontology Dowsing," *W3C Wiki*, [Online]. Available: https://www.w3.org/wiki/Ontology_Dowsing (visited on 08/07/2019). 39

[31] E. Pilkington, "Google's secret cache of medical data includes names and full details of millions – whistleblower," *The Guardian*, Nov. 19, 2019. [Online]. Available: https://www.theguardian.com/technology/2019/nov/12/google-medical-data-project-nightingale-secret-transfer-us-health-information (visited on 11/21/2019). 8

[32] T. L. Saaty, "A Scaling Method for Priorities in Hierarchical Structures," *Journal of Mathematical Psychology*, vol. 15, no. 3, pp. 234–281, 1977. 89

[33] T. L. Saaty, "What is the Analytic Hierarchy Process?" In *Mathematical Models for Decision Support*, Springer, 1988, pp. 109–121. 92

[34] T. L. Saaty, "The Analytic Hierarchy Process," *European Journal of Operational Research*, vol. 48, pp. 9–26, 1990. 90

[35] T. L. Saaty and L. G. Vargas, *Models, Methods, Concepts & Applications of the Analytic Hierarchy Process*. Springer Science & Business Media, 2012, vol. 175. 92

[36] A. Senarath and N. A. Arachchilage, "Why developers cannot embed privacy into software systems?: An empirical investigation," in *Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018*, ACM, 2018, pp. 211–216. 6

[37] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002. 18

[38] E. Triantaphyllou, "Reduction of Pairwise Comparisons Via a Duality Approach," *Journal of Multicriteria Decision Analysis*, vol. 8, no. 6, p. 299, 1999. 29, 67

[39] S. Vavilis, M. Petković, and N. Zannone, "A Severity-based Quantification of Data Leakages in Database Systems," *Journal of Computer Security*, vol. 24, no. 3, pp. 321–345, 2016. 14, 30–32, 36, 54

Appendix A

Background Material

A.1 Analytic Hierarchy Process

The analytic hierarchy process (AHP) performs pairwise comparisons on groups of criteria in a hierarchy to determine the relative weights of criteria in the groups. This section follows the process outlined by Saaty [32], and begins by illustrating what groups of criteria look like in Figure A.1.

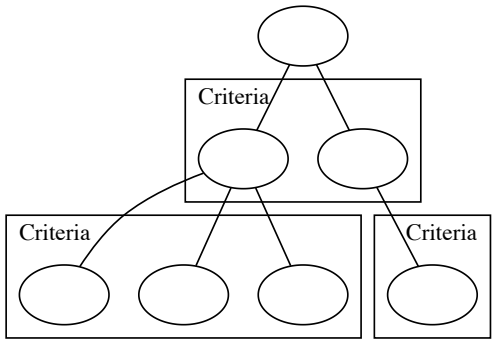


Figure A.1: Criterion of AHP tree.

Using the criteria of a group, reciprocal matrices can be formed by doing pairwise comparisons as explained in Section A.1.1. From the pairwise comparisons a priority vector can be computed as seen in Section A.1.2, and verified for consistency as described in Section A.1.3.

A.1.1 Pairwise Comparisons

To perform pairwise comparisons, an evaluator compares each of the criteria of a criteria grouping pairwise. For example consider a criteria grouping with

3 criteria: X, Y, and Z. Then a reciprocal matrix of size 3×3 as illustrated in Table A.1 can be made to facilitate the pairwise comparisons.

Table A.1: Reciprocal matrix of the criteria grouping: X, Y, and Z.

	X	Y	Z
X	1	2	6
Y	—	1	3
Z	—	—	1

Table A.1 is a reciprocal matrix because, the entries of the lower triangular matrix of the table are the inverse of the upper triangular matrix. Thus, if entry $T_{X,Y} = 2$, then entry $T_{Y,X} = \frac{1}{2}$. Because of this property, only an upper or lower triangular matrix needs to be compared in order to infer the other triangular matrix.

Each comparison is assigned a value in based on the Saaty scale described in Table A.2. For example, the comparison $T_{Y,Z} = 3$ means that "Y is of slightly more important compared to Z". The reciprocal comparison $T_{Z,Y} = \frac{1}{3}$ means that "Z is slightly less important than Y".

Table A.2: The Saaty rating scale taken from [34] and used to compare two elements relative to each other in terms of their importance. It is in the range of 1–9. Note that the reciprocal can also be used to compare elements in the other direction.

Rating	**Definition**
1	Equal importance
2	Equal to weak importance of one over another
3	Weak importance
4	Weak to essential importance
5	Essential or strong importance
6	Essential to very strong importance
7	Very strong importance
8	Very strong to extreme importance
9	Extreme importance

A.1.2 Calculating Priority Vector

The priority vector is a vector of priorities for each criteria of a criteria grouping. It is calculated by using the eigenvector $\mathbf{w}$ of the largest unique real

eigenvalue λ_{max} of a reciprocal matrix A. The relationship between the dominant eigenvalue, reciprocal matrix, and the corresponding eigenvector $\mathbf{w}$ is described as follows:

$$A\mathbf{w} = \lambda_{max}\mathbf{w} \tag{A.1}$$

$\mathbf{w}$ and λ_{max} can be computed using software such as the NumPy[1] package in Python.

Using $\mathbf{w}$, the priority vector can be calculated by converting $\mathbf{w}$ into a stochastic vector $\bar{\boldsymbol{w}}$. The stochastic vector is a vector consisting of non-negative entries that sum to one and can be defined as follows:

$$\text{Priority Vector} = \bar{\boldsymbol{w}} = \frac{1}{\sum_{i=1}^{n} \mathbf{w}_i} \cdot \mathbf{w} \tag{A.2}$$

For example, the eigenvector of Table A.1 is computed to be: $\{X, Y, Z\} = \{0.88465174, 0.44232587, 0.44232587\}$ which can then be converted to the priority vector $\{0.6, 0.3, 0.1\}$ using Equation A.2. This priority vector's sum of entries $0.6 + 0.3 + 0.1$ is 1 which satisfies the stochastic vector property.

A.1.3 Calculating Consistency Ratio

Once comparisons have been completed for a criteria grouping, the consistency of judgements in the reciprocal matrix must be evaluated.

The consistency index (**CI**) can then be calculated from λ_{max} and size n of the reciprocal matrix as follows:

$$\mathbf{CI} = \frac{\lambda_{max} - n}{n - 1} \tag{A.3}$$

Using **CI**, it can be used to finally compute the consistency ratio (**CR**) as follows:

$$\mathbf{CR} = \frac{\text{CI}}{\text{RI}} \tag{A.4}$$

RI is the random index, computed by averaging the consistency index of reciprocal matrices filled with random values on the Saaty scale. Table A.3 provides some precomputed random indices of different sized reciprocal matrices.

[1] `https://numpy.org/doc/stable/reference/generated/numpy.linalg.eig.html`

Table A.3: Random indices to calculate consistency ratio from [35].

Matrix Size	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Random Index	0	0	0.52	0.89	1.11	1.25	1.35	1.40	1.45	1.49

For each reciprocal matrix, a reasonable consistency ratio must also be maintained. This ratio is suggested to be less than or equal 10% (0.1) up to a maximum of 15% (0.15) for a single individual who makes comparisons [33]. In a group setting where multiple individuals are making comparisons that will be averaged, the suitable ratio maximum is increased to 20% (0.20) [19].

For instance, the consistency index of Table A.1 is calculated to be 0 using Equation A.3 where $\lambda_{max} = 3$ and $n = 3$. Then it follows that the consistency ratio will be 0 using Equation A.3 and the values CI = 0 and RI = 0.52. This means that the comparisons in Table A.1 are consistent as it is less than the suggested threshold 0.1.

A.2 DPV Concepts Related to Table 3.1

The data privacy vocabulary (DPV) is a collection of terms with hierarchical relationships meant to manage and classify data to be legally compliant with laws and regulations. DPV is used in this thesis to extract a domain model that can represent the sensitive attributes of Table 3.1. The definitions of these concepts can be found in Table A.4.

Table A.4: Concepts of DPV[2] that are relevant to the sensitive attributes *Disease*, *Medication*, *Age*, *Initial Diagnosis* of Table 3.1.

Concept	**Definition**
PersonalDataCategory	A category of personal data (as defined by GDPR article 4.1) from the personal data categories taxonomy, i.e. for instance denoting the category of an object/field or data item that is used for processing
SpecialCategoryPersonalData	Special category or personal data as per GDPR Art. 9 (1)
External	Personal Data that can be observed by another person i.e. has external characteristics that make it visible
MedicalHealth	Information that describes an individual's health, medical conditions or health care
PhysicalCharacteristic	Information that describes an individual's physical characteristics
Health	Information about an individual's health
HealthRecord	Information about an individual's health record
Prescription	Information about prescriptions made for an individual
Age	Information about an individual's age

[2] `https://www.w3.org/ns/dpv`

A.3 Reciprocal Matrices of Figure 4.7

The reciprocal matrices are comparisons made for the criteria groupings outlined in Figure 4.7. It should be noted that criteria groupings of size 1 are omitted because the matrix will only have a single cell containing the value 1.

Criteria		More Important	Rating
A	B		
PhysicalCharacteristic	MedicalHealth	B	4

(a) Ratings given by a domain expert.

	PhysicalCharacteristic	MedicalHealth
PhysicalCharacteristic	1	1/4
MedicalHealth	4	1

(b) Reciprocal matrix with a consistency ratio of 0.

Table A.5: Ratings (a) and reciprocal matrix (b) for the criteria: "PhysicalCharacteristic", and "MedicalHealth".