Semantic Attack on Transaction Data Anonymised by Set-Based Generalisation

A thesis submitted in partial fulfilment

of the requirement for the degree of Doctor of Philosophy

Hoang N. Ong

January 2015

School of Computer Science & Informatics
Cardiff University

Declaration

This work has not previously been accepted in substance for any degree and is not concurrently submitted in candidature for any degree.

Signed ..... (candidate)

Date .....

Statement 1

This thesis is being submitted in partial fulfillment of the requirements for the degree of PhD.

Signed ..... (candidate)

Date .....

Statement 2

This thesis is the result of my own independent work/investigation, except where otherwise stated. Other sources are acknowledged by explicit references.

Signed ..... (candidate)

Date .....

Statement 3

I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed ..... (candidate)

Date .....

Abstract

Publishing data that contains information about individuals may lead to privacy breaches. However, data publishing is useful to support research and analysis. Therefore, privacy protection in data publishing becomes important and has received much recent attention.

To improve privacy protection, many researchers have investigated how secure the published data is by designing de-anonymisation methods to attack anonymised data. Most of the de-anonymisation methods consider anonymised data in a syntactic manner. That is, items in a dataset are considered to be contextless or even meaningless literals, and they have not considered the semantics of these data items.

In this thesis, we investigate how secure the anonymised data is under attacks that use semantic information. More specifically, we propose a de-anonymisation method to attack transaction data anonymised by set-based generalisation. Set-based generalisation protects data by replacing one item by a set of items, so that the identity of an individual can be hidden. Our goal is to identify those items that are added to a transaction during generalisation. Our attacking method has two components: *scoring* and *elimination*. *Scoring* measures semantic relationship between items in a transaction, and *elimination* removes items that are deemed not to be in the original transaction. Our experiments on both real and synthetic data show that set-based generalisation may not provide adequate protection for transaction data, and about 70% of the items added to the transactions during generalisation can be detected by our method with a precision

greater than 85%.

Acknowledgements

I would like to express my special appreciation and thanks to my advisor, Dr Jianhua Shao, who has been a tremendous mentor to me. I would like to thank him for encouraging my research and for allowing me to grow as a research scientist. I am also thankful to Dr Irena Spasic, Dr Andrew Jones and Professor Ralph R. Martin for their valuable feedback that has served to improve my work. Also, I would like to express special thanks to my family and my friends for their continuous support and encouragement.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

Data about individuals is being increasingly collected, analysed and disseminated. For example, when a patient visits a hospital, their diagnostic data may be recorded and then used in medical studies, and when a customer shops online, their browsing activities may be retained by the vendor to help recommend products to other customers. Such data is valuable to organisations and society as a whole, as it can help, for example improve business intelligence.

However, the collected data may contain personal and sensitive information. Releasing such data directly could pose a privacy threat. For example, it has been shown that 87% of the population in the United States can be uniquely identified based on a combination of their 5-digit zip code, gender and date of birth [46], and 84% of Netflix (a movie rental service) subscribers could easily be identified by an adversary [81] who knows 6 ratings of individuals. This has led to the development of technologies for anonymising data before its release.

Many privacy models for protecting data privacy have been proposed in recent years. However, a key question still remains "do these privacy models provide enough protection for the data?" In this thesis, we study specifically how secure transaction data anonymised by set-based generalisation is.

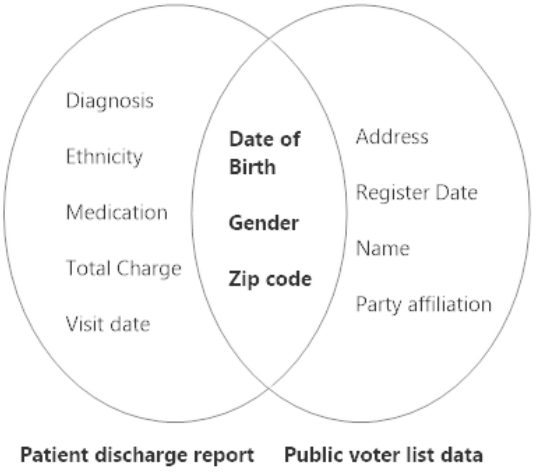


Figure 1.1: Identifying individuals, using a combination of non-identifying attributes [46].

1.1 Data Privacy and Its Protection

Releasing data by simply removing identifying information such as names and IDs can still pose a serious privacy risk [99]. Combinations of some attributes could be used to identify an individual. For example, Figure 1.1 shows that an adversary may use date of birth, gender and zip code, which are available from some public sources, to identify an individual from patient discharge report data that contains no identifiers.

The challenge in data privacy protection is to allow data sharing while prohibiting individuals from being identified. In recent years, privacy, security and statistical database research communities have responded to the challenge by proposing various privacy protection models and methods. A common approach is to distort the original data to prevent a unique combination of some attributes that may be used to link the data to an individual. For example, Figure 1.2 shows a typical anonymisation method where the original data in Figure 1.2 (a) is transformed to a more general form to create an anonymised version in Figure 1.2 (b) for release.

In Figure 1.2, TID is not a part of the released data, but is simply used to help with illustration. Basically, an adversary can use background knowledge (e.g. zip code

TID	ZIP Code	Age	Disease
1	47677	29	HIV
2	47602	29	Cough
3	48548	37	Cold
4	48541	34	Cancer

(a) Original dataset

TID	ZIP Code	Age	Disease
1	476**	29	HIV
2	476**	29	Cough
3	475**	[30-39]	Cold
4	475**	[30-39]	Cancer

(b) Anonymised dataset

Figure 1.2: An example of data anonymisation

and the age of an individual) to identify an individual's record in Figure 1.2 (a), and then infer the disease information about the individual. To prevent that, each record is generalised. That is, each unique combination of zip code and age is linked to more than one record in Figure 1.2 (b). As such, an individual's record cannot be uniquely identified. For instance, knowing that Mary lives at zip code 47677 and is 29 years old, an adversary cannot infer precisely that Mary's record is linked to TID 1 or 2 in the released data. On the other hand, the utility of the original data is still retained. For instance, a researcher can estimate a range of age of individuals with a specific disease.

Different privacy models are designed to protect different types of data (e.g. relation [99], transaction [100], text [51], graph [29] or trajectory [25] data). In our work, we consider *transaction data*.

1.2 Anonymising Transaction Data

Transaction data is records that contain items about individuals. For example, Figure 1.3 shows a set of 4 transactions, each recording a set of medical terms associated with a patient. TID is a transaction identifier which is included here for reference only; it will not be part of the released data. Releasing data in Figure 1.3 may violate individual privacy. For example, knowing that Mary has a *blood pressure* problem and she is in the dataset, an adversary can infer that transaction 1 belongs to Mary and find out

other information about her.

TID	Items
1	heart disease, blood pressure, icd, weakness, dizziness
2	anesthesia, icd, pain, diabetes
3	gangrene, limbs, injury
4	knee, injury

Figure 1.3: An Example of Transaction Data

Unlike relational data, transaction data has some unique properties that make its anonymisation more difficult. One important property is that transaction data is often *high dimensional* compared to relational data. That is, if we consider each item as an attribute, a set of transactions will have a high number of attributes compared to a typical relational dataset. For example, if we convert Figure 1.3 into a relation by turning each item into an attribute, we have Figure 1.4, where 1 indicates that the transaction contains the corresponding item. It has been shown in [100] that k-anonymisation (a popular technique for anonymising relational data) is not useful in high dimensional data because it can significantly destroy data utility; that is because, with high dimensional data, there is a low chance for records to share attribute values, hence more generalisation (or distortion) needs to be applied to the data.

TID	heart disease	blood pressure	icd	weakness	dizziness	anesthesia	pain	diabetes	gangrene	limbs	injury	knee
1	1	1	1	1	1	0	0	0	0	0	0	0
2	0	0	1	0	0	1	1	1	0	0	0	0
3	0	0	0	0	0	0	0	0	1	1	1	0
4	0	0	0	0	0	0	0	0	0	0	1	1

Figure 1.4: A relational form of transaction data in Figure 1.3

A key issue in anonymising transaction data is therefore to ensure that only necessary items are protected, as protecting all combinations of items would result in too much data utility loss, rendering the anonymised data useless in applications. One such method is COAT [77] which allows the data publisher to specify which items

should be protected and how items may be generalised. For example, applying COAT to Figure 1.3 will result in Figure 1.5, where items in brackets are generalised items. Here, we require *blood pressure, icd, limbs* and *injury* to be protected by not being able to be distinguished in at least 4 transactions, and we allow any combination of *blood pressure, icd, weakness, dizziness, pain, diabetes, limbs* and *injury* to be used to achieve the protection. As a result, knowing that Mary has *blood pressure* will now no longer be enough to determine if Mary is the owner of T1 in Figure 1.5 with a probability greater than $1/4$.

TID	Items
1	heart disease, (*blood pressure, icd, limbs, injury*), weakness, dizziness
2	anesthesia, (*blood pressure, icd, limbs, injury*), pain, diabetes
3	gangrene, (*blood pressure, icd, limbs, injury*)
4	knee, (*blood pressure, icd, limbs, injury*)

Figure 1.5: Set-Based Generalisation

COAT uses set-based generalisation to anonymise a set of transactions. It is well established that set-based generalisation is more flexible in satisfying privacy constraints and produces anonymous data that has better utility (i.e. with less distortion and information loss). In this thesis, we investigate whether set-based generalisation will provide sufficient protection for transaction data.

1.3 De-anonymising Transaction Data

Set-based generalisation rests on the assumption that items are contextless or even meaningless literals, and it does not consider the transaction as a whole when forming a set to replace (or generalise) an item. The only requirement is that it should make some combinations of items appear frequently enough within the released dataset and that it should result in minimum distortion of the data.

We argue that when semantic relationships among the items are considered, such protection may not be sufficient. For example, consider the generalised items in Figure 1.5 again. Although (*blood pressure, icd, limbs, injury*) in T4 suggests that Mary could have *blood pressure*, *icd*, *limbs*, *injury* or any combination of them, the presence of $knee$ in the transaction suggests that it is more likely to be $injury$, rather than any of the others. This type of semantic analysis will allow an adversary to reduce a generalised item to its original form.

1.4 Research Hypothesis and Contributions

Our hypothesis is: set-based generalisation may not provide adequate protection for transaction data. Non-original items in set-based generalised transactions can be eliminated with high precision and recall, which may lead to original items being revealed and privacy breached.

To the best of our knowledge, it is the first time that semantic relationships have been used to de-anonymise transaction data. The main contribution of the thesis includes:

- We propose de-anonymisation methods that aim to reconstruct original transaction data from its set-generalised version by analysing semantic relationships that exist among the items. This is in contrast to other studies on quantifying privacy risk involved in publishing transaction data [81, 32, 44] where attackers are assumed to have some auxiliary information about the individuals; we require no such information and rely on the released data only. Thus, our de-anonymisation method is independent of the background knowledge (e.g. knowing information about individuals in the released data) that an attacker is assumed to have, and represents a realistic assessment of privacy risk associated with set-based generalisation.

- To determine the semantic relationship among data items, we build our work on

a measure called Normalised Google Distance [27]. This measure establishes semantic relationship between two terms by querying the Google repository of WWW pages: the more pages in which the two terms appear together, the more related they are considered to be. This eliminates the need to construct a single dictionary or a corpus for testing term relationships and ensures that our approach is generic and realistic.

Our experiments on both real and synthetic data show that set-based generalisation may not provide adequate protection for transaction data, and about 70% of the items added to transactions during generalisation can be detected by our method, with a precision greater than 85%. Note that our de-anonymisation approach uses information that is readily available from the released data and Google, thus the identified privacy risk is realistic.

1.5 Assumptions

We make the following assumptions in our work:

- We assume that published transaction data retain some semantic content. That is, items of a transaction retain some natural meanings. This is often the case, for example, when a transaction (vector) is extracted from a text (for document analysis). However, we recoganise that there are applications where datasets have their semantics removed, for example, the actual items are replaced by a system of codes before being published. Our methods cannot be used to attack such published datasets.

- Our attacking strategy is based on the assumption that given a generalised item, we have different contexts (i.e. items that are not generalised) in different transactions associated with it. For example, (*blood pressure, icd, limbs, injury*) in

Figure 1.5 has four different contexts, one from each published transaction. We rely on such differences to distinguish non-original from original items.

- Our attacking strategy is to eliminate non-original items from set-generalised items. Therefore, it is necessary to know which items in a transaction are generalised items. We assume that this information is available to an adversary from the published transactions. While it is possible to consider how to identify generalised items in a transaction where a generalised items is not marked, we do not consider this issue in detail in this thesis.

It is useful to note that these assumptions are not restrictive and are practical. They are satisfied by most of the datasets we have seen, including those used in our experiments.

1.6 Thesis Organisation

Chapter 2 discusses the background of privacy protection for relational, transaction and text data in general. We also review de-anonymisation techniques, which are classified into: link attack, probabilistic attack and context-awareness attack.

In Chapter 3, we introduce a framework for *semantic attack* which includes two main components: *Scoring* and *Elimination*. We cover the scoring component in this chapter in detail which is based on Normalised Google Distance.

In Chapter 4, we focus on the elimination component of the proposed framework. The algorithms take scores from the scoring component as inputs to decide whether an item is in the original transaction or not.

Chapter 5 begins with a description of the datasets used in the experiments and the methodologies adopted in evaluation. Then, we empirically evaluate and compare the methods that we have proposed.

Chapter 6 concludes and summarises the thesis and discusses future work.

Chapter 2

Background and Literature Review

While our work focuses on de-anonymisation of released data, understanding anonymisation techniques may help us to see how released data can be de-anonymised. In this chapter, we review the relevant privacy protection techniques as well as attacking methods.

Research in data privacy is primarily about three issues [22, 37, 66]: privacy model, sanitisation methods and optimisation criteria. A model defines the constraints of privacy protection. For example, k-anonymity [99] is a privacy model which requires that each individual in the released data has no more than $1/k$ chance of being identified by using quasi-identifiers. For instance, the data in Figure 1.2 (b) is 2-anonymised, and each individual has no more than 1/2 chance of being identified, using age and zip code.

A sanitisation method is used to change data in order to satisfy a privacy model. There are several common approaches: generalisation [98, 106, 39, 56], suppression [98, 56], bucketisation [111, 49] and perturbation [60, 23, 61]. Generalisation transforms a value to make it less specific (see Figure 1.2 (b)). Suppression removes values or records from the released data. Bucketisation splits released data into multiple relations. Perturbation allows data items to be swapped or noise added.

Optimisation criteria are needed to balance between data privacy and data utility. For example, removing all the data would achieve the highest level of privacy but it makes the data useless. Thus, some trade-off [88, 73] between utility and privacy of the

released data is necessary.

In the following sections, we concentrate on discussing relevant privacy models and their protection methods. We will examine the protection for different types of data including relational, transaction, statistical and text data. For each data type, we review how it may be anonymised and how it may be attacked, and discuss how our work is unique, compared with the works reviewed here.

2.1 Privacy in Relational Data

This section discusses anonymisation and de-anonymisation techniques for relational data. We first review typical methods for anonymising relational data, including generalisation, suppression, bucketisation and perturbation. We then discuss how anonymised data may be attacked.

2.1.1 Anonymisation Methods

Generalisation

One of the common methods for anonymising relational data is generalisation, which attempts to transform specific data items into more general ones. For example, an age of 29 may be generalised to a range [25-30], and as such, the exact age is hidden.

Generalisation can be performed on any attributes of a relational dataset, but not all attributes need to be generalised. Attributes are classified into: identifier attributes, quasi-identifier attributes and sensitive attributes. Identifier attributes (ID) contain unique information about individuals such as names or phone numbers, and they are removed before releasing the data. Sensitive attributes (SA) are needed in analysis, therefore they will be kept intact in the released data. Quasi-identifier attributes (QID) contain information about individuals such as gender or age. While each attribute is

not unique enough to identify an individual, a combination of them may do so and because of that, generalisation is often performed on QIDs (see Figure 1.2 (b)).

One way of generalising data is by organising a hierarchy over the domain of an attribute [98, 39, 106, 101]. For example, Figure 2.1 shows a hierarchy over zip code. Value 0124* is a generalisation of 02141 and 02142.

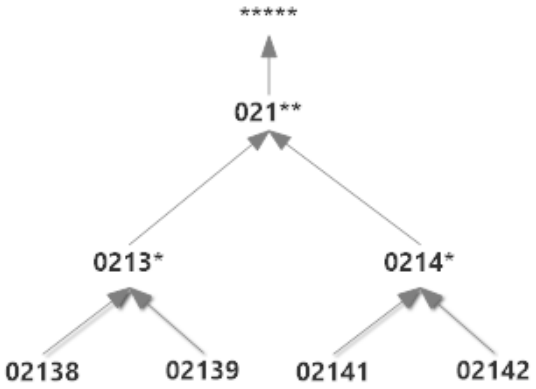


Figure 2.1: A hierarchy of a zip code's domain

Hierarchy-based generalisation is one of the popular approaches to anonymising data [99, 78, 71]. However, Loukides et al. [77] argued that hierarchy-based generalisation can cause too much *information loss*. For example, for a 5-digit zip code, 021** could represent any one of 100 possible zip codes which does not help data analysis.

In contrast, set-based generalisation is more flexible and preserves data utility better [77].

Definition 1 (Set-Based Generalisation)**.** *A set-based generalisation is a partition $\tilde{\mathcal{I}}$ of $\mathcal{I}$ in which each item $i \in \mathcal{I}$ is replaced by the partition to which it belongs. Each partition is called a generalised item, and each i is mapped to its generalised version $\tilde{i}$ using a generalisation function $\Phi : \mathcal{I} \rightarrow \tilde{\mathcal{I}}$. When an item is generalised to itself, we say that the item is trivially generalised.*

For example, Figure 2.2 demonstrates the mapping of items from zip code (Z) to generalised zip code ($\tilde{Z}$) where 02138 and 02141 are grouped to create a generalised item

(02138, 02141).

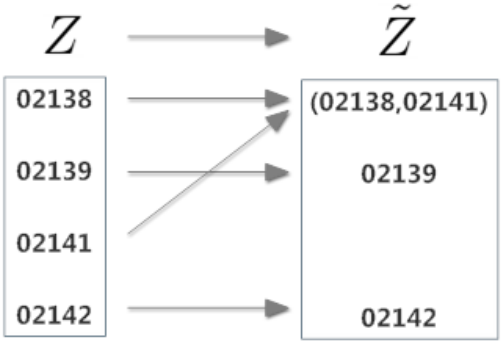


Figure 2.2: An example of set-based anonymisation

The problem with hierarchy-based generalisation is that it forces all siblings of the original value to be mapped to a parent node in the hierarchy when this value is generalised. As a result, hierarchy-based generalisation is restricted to representing a much smaller number of possible generalisations than set-based generalisation can. Set-based generalisation can be used to represent all possible mapping of values from $\mathcal{I}$ to $\tilde{\mathcal{I}}$, effectively allowing a larger anonymisation space to be explored. This is important because it offers the potential for finding anonymisations of "low" information loss.

Suppression

Protecting privacy by suppression is simply removing items from released data. For example, Figure 2.3 shows the anonymous data of Figure 1.2 (a) by generalising zip codes and suppressing some age values to prevent unique identification of an individual, by using a combination of zip code and age values.

With relational data, suppression can be regarded as a special form of generalisation where an item is generalised to the most general value and it stands for any possible value of a domain.

	ZIP Code	Age	Disease
1	476**	29	HIV
2	476**	29	Cough
3	475**	-	Cold
4	475**	-	Cancer

Figure 2.3: Using suppression to anonymise data

Bucketisation

Bucketisation [111, 43] produces non-overlapping groups (or buckets) and then, for each group, releases its projection on the QIDs and also its projection on SAs. Figure 2.4 is a bucketisation of the data in Figure 1.2 (a). QIDs and SAs are not modified; instead, Group-IDs are added to link two tables. As such, it does not precisely link a record to a specific sensitive value. Count field relates to the counting of the number of occurrences of sensitive value in each group.

Zip Code	Age	Group-ID
47677	29	1
47602	29	1
48548	37	2
48541	34	2

(a) A bucket with quasi-identifier

Group-ID	Disease	Count
1	HIV	1
1	Cough	1
2	Cold	1
2	Cancer	1

(b) A bucket with sensitive values

Figure 2.4: An example of bucketisation of the data in Figure 1.2 (a)

Comparing this method to generalisation, they both offer equivalent protection for data in that an adversary cannot link an individual to a specific sensitive value. The main difference lies in the fact that bucketisation does not generalise QID values. That is, the method does not stop identity disclosure, however, sensitive values still remain private.

Perturbation

Data perturbation is a procedure that changes data (e.g. by adding noises) to protect privacy. This method is popularly used to protect summary data in statistical analysis, but it is not commonly used in protecting relational data. We will explain this method further in Section 2.3, where we deal with statistical data.

2.1.2 Record Linkage Attack and k-anonymity Model

Attack

In considering the release of individuals' data as a relation, each tuple contains information about an individual, for example {1, 47677, 29, HIV} in the table of Figure 1.2 (a), it is about one individual. Although ID attributes are not present in the released data, an adversary can still attack the data by using combinations of QIDs. For example, if an adversary knows that Tom is a 29-year-old living at an address with a zip code of 47677 and is in the released data of Figure 1.2 (a), then they can infer that Tom is the owner of the first record, and learn other information about Tom.

Both record linkage attack and our attack are related to identity disclosure. However, the main difference is that record linkage attack uses some prior knowledge about individuals to narrow down and identify the individuals in the dataset, while our work only uses the available information in the dataset, without any assumption about the knowledge of an adversary.

Protection

To prevent this type of attack, Sweeney [99] proposed a fundamental privacy model called k-anonymity:

Definition 2 (k-anonymity)**.** *Let $\tilde{T} = \{a_1, ..., a_d\}$ be a table. Without loss of generality, we assume that the first m attributes (a_i, $i = 1, ..., m$) are QIDs, and the remaining attributes (a_i, $i = m+1, ..., d$) are SAs. $\tilde{T}$ satisfies k-anonymity if it is partitioned into groups $g_1, ..., g_h$ s.t. $|g_j| \geq k$, $1 \leq j \leq h$, where $|g_j|$ denotes the size of g_j, and tuples in each g_j are made identical w.r.t. QIDs.*

Informally, k-anonymity defines a constraint for releasing a relational dataset in which there are at least k (with $k > 1$) records that have identical QID attributed values for each *equivalent group* (i.e. a group that has identical QID values) in the released data. By satisfying this principle, an adversary who has knowledge about some individuals, cannot distinguish this one from at least $k - 1$ other individuals. For example, the data in the Figure 1.2 (b) is 2-anonymous with {Zip Code, Age} as QIDs and {Disease} as SA. That is, if an adversary knows someone, who is a 29-year-old, living at an address with a zip code of 47677 and is in the released dataset, they still cannot identify the record of the individual among the first two rows.

2.1.3 Attribute Linkage Attack and ℓ-diversity Model

Attack

k-anonymity has grown in popularity due to algorithmic advances in creating an anonymous version of data [98, 85, 76, 16, 82, 110]. However, Machanavajjhala et al. [78] observed that k-anonymity has several limitations which can be exploited, in terms of breaching privacy. For example, if Alice knows that Tom's record is in Table 2.5, his zip code is 14852 and his age is 38, then without identifying which record is Tom's, Alice can still infer that Tom has *Cancer*.

This is because groups created by k-anonymity lack diversity in the sensitive attribute. The adversary does not need to link an individual to a specific record, but can still determine the sensitive value associated with the individual. This type of attack is called *attribute linkage attack*.

TID	Zip code	Age	Condition
1	130***	< 30	Heart disease
2	130***	< 30	Viral infection
3	130***	< 30	Viral infection
4	148***	[30-40]	Cancer
5	148***	[30-40]	Cancer
6	148***	[30-40]	Cancer

Figure 2.5: 3-anonymous inpatient data

The difference between attribute linkage attack and record linkage attack is the way in which privacy is violated. In attribute linkage attack, privacy is violated in that an adversary can get to know sensitive information about individuals despite the specific record of the individual remaining private, while record linkage attack specifically targets an individual's record. Compared with our attacking method, both of these linkage attacks rely on prior knowledge about individuals before the attack, while we do not.

Protection

Machanavajjhala et al. introduced a stronger notion of privacy, called ℓ-diversity [78], to defend against attribute linkage attack:

Definition 3 (ℓ-diversity)**.** *Given an equivalent group g, g is ℓ-diverse if g contains at least ℓ "well-presented" values in the sensitive attribute. A table is ℓ-diverse if every group g is ℓ-diverse.*

Based on this definition, *entropy ℓ-diversity* is proposed to ensure that the probability of linking sensitive value to an individual in each and every equivalent group is not more than $1/\ell$.

2.1.4 Probabilistic Attack and t-closeness

Attack

While the ℓ-diversity principle represents an important step beyond k-anonymity in protecting sensitive attributes, Li et al. [71] pointed out that the privacy notion in k-anonymity and ℓ-diversity is still not sufficient. This is demonstrated in the following example. Suppose that the original data has a sensitive attribute with two values of HIV disease, positive and negative. Let us consider the case where the data contains 10,000 records, with 99% of individuals in the data being negative, and the other 1% being positive (i.e. 9,900 negative and 100 positive). Suppose that the released dataset needs to satisfy ℓ-diversity in order to prevent this sensitive information from being disclosed. As there are two possible sensitive values, the dataset should satisfy 2-diversity, which means that there should be two different sensitive values in each equivalent group. Furthermore, the dataset has 100 positive values, therefore, at most, 100 equivalent groups can be established, each containing 1 positive value and from 1 to 9,801 negative values. Many different ways to distribute negative values into these groups are possible, we consider one extreme situations: An equivalent group contains 1 negative and 1 positive. Although it satisfies the ℓ-diversity principle, it poses another serious privacy threat for individuals in this group as they have a 50% chance of being HIV positive, which is significantly higher than the 1% in the original dataset.

In the above example, although an adversary does not know the record or sensitive value of an individual for certain, the difference in distribution of sensitive values in the original dataset and the released dataset suggests that some individuals may now be identified more likely to have HIV. Because of that, both k-anonymity and ℓ-diversity principles are deemed to be not sufficient to protect privacy. This attacking technique is called *probabilistic attack*.

Protection

Before seeing the released data, the adversary has some prior belief about the sensitive attribute value of an individual. After seeing the released data, the adversary has a posterior belief. "Information gain" can be represented as the difference between the posterior and prior beliefs. The higher the information gain is, the higher the threat to privacy breach is. Li et al. [71] proposed a t-closeness privacy principle, based on the idea that "privacy is measured by the information gain of an adversary":

Definition 4 (t-closeness)**.** *An equivalent group is said to have t-closeness if the distance between the distribution of a sensitive attribute in this group and the distribution of the attribute in the whole table is no more than threshold t. A table is said to have t-closeness if all groups have t-closeness.*

ℓ-diversity and t-closeness syntactically protect sensitive information from disclosure, however, they may not if considering the meaning of values. For example, Figure 2.6 shows a 3-diverse dataset, with distinct sensitive values. However, diseases of individuals in the first three records are analogous, therefore, an adversary can infer that individuals in this group have a "stomach" problem. Increasing ℓ and t to add more distinct sensitive values into the equivalent group can help protection in this case, however, the protection is ad-hoc. Although our work does not deal with this privacy problem, it shows that considering the meaning of values while protecting privacy is important.

2.1.5 Minimality Attack

Common attacking techniques consider situations where an adversary only possesses some knowledge about individuals in the released data. Wong et al. [109] and Zhang et al. [116] showed that if an adversary also has the knowledge of the algorithm used to anonymise the data, they may use that knowledge to break the anonymity.

Zip code	Age	Disease
130***	< 30	gastric ulcer
130***	< 30	gastritis
130***	< 30	stomach cancer
148***	3*	flu
148***	3*	bronchitis
148***	3*	Cancer

Figure 2.6: 3-diverse data may still cause a privacy issue by considering the meaning of values.

For example, given the original data in Figure 2.7 (a) and its 2-diverse data in Figure 2.7 (b), and assuming that an adversary has the knowledge of Figure 2.7 (c), they know that each individual has a QID of either q_1 or q_2. After q_1 and q_2 are generalised to a general value Q, the adversary cannot link any individual in Figure 2.7 (c) to a specific disease in Figure 2.7 (b). However, if the adversary also knows the algorithm used to generalise the data (i.e. the data is to be 2-diverse), they may reason as follows: because there are 4 individuals who have QID=q_2 and 5 distinct sensitive values associated with 6 individuals in the dataset, the equivalent group of q_2 can satisfy 2-diversity without generalisation. Therefore, the need for generalising in this dataset came from the equivalent group of q_1. This implies that the q_1 group is not 2-diverse, hence must have the HIV value. That is, the individuals with IDs 1 and 2 have HIV.

The issue with the above protection is that it is based on a minimum requirement to anonymise data (i.e. it only generalised the data when l-diversity is not satisfied). This allows an adversary to infer the original data that needs to be modified in order to satisfy the requirement. This type of attack is called *minimality attack* [69], based on the minimality principle:

Definition 5 (Minimality Principle)**.** *Suppose $\mathcal{K}$ is an anonymisation algorithm for a privacy principle $\mathcal{P}$. Let $\tilde{T}$ be a table generated by $\mathcal{K}$ and $\tilde{T}$ satisfies $\mathcal{P}$. $\tilde{T}$ is minimality if there is no specialisation of $\tilde{T}$ (reverse of generalisation), which results in another*

QID	Disease
q1	HIV
q1	HIV
q2	Cough
q2	Red Eyes
q2	Flu
q2	Cold

(a) Original data

QID	Disease
Q	HIV
Q	HIV
Q	Cough
Q	Red Eyes
Q	Flu
Q	Cold

(b) Anonymous data which is protected by 2-diversity

ID	QID
1	q1
2	q1
3	q2
4	q2
5	q2
6	q2

(c) Public data of individual in Figure 2.7 (a)

Figure 2.7: Example of Minimality attack

table $\tilde{T}'$, which also satisfies $\mathcal{P}$.

As has been discussed, data which is regarded as having minimality could be attacked by minimality attack. Therefore, it is desirable to define some notions of minimality in terms of the privacy principle. For example, a k-anonymisation should not generalise, suppress or distort the data more than is necessary to achieve k-anonymity. This principle is correct not only for k-anonymity but also for its extensions such as ℓ-diversity [78], t-closeness [71], confidence bounding [105], (α, k)-anonymity [110], (k, e)-anonymity [117] and (c, k)-safety [79]. Therefore, minimality attack can be used to attack most of these privacy models.

2.1.6 Inference Attack and Knowledge Hiding

Inference Attack

Data mining enables us to discover information that is hidden in data. While some discovered patterns are useful for improving an organisation's business strategy, others can be used to violate an individual's privacy [28]. Using data mining [3] to infer

private information from data is often called *inference attack*. An example of such an attack is given below:

Considering anonymous data in Figure 2.8 that is protected by the bucketisation method (i.e. QID and SA are released into separate tables). *Tuple ID* is not released and is used here for discussion only. *Smoke?* is a QID with two values yes (y) or no (n) and the *Disease* is an SA which indicates the disease that an individual has. The released table satisfies 2-diversity. The attacker can find out that the individual, who has TID of 11, is more likely to have *Cancer* by the following reasoning: by observing "smoke?" and "disease" attributes, the adversary can see that any QID group, which does not have any individual smoker, does not have "Cancer" linked to the group either (groups 2 and 5); and any group that has at least one individual who smokes, always contains at least one Cancer in sensitive value in the group (groups 1, 3, 4 and 6). From that, the adversary can infer a pattern whereby an individual who is in the smoking group is more likely to have cancer. Therefore, the adversary can infer that the individual, who has TID of 11, is more likely to have Cancer (similar reasoning can be made for the individual who has TID of 6).

Based on this observation, Kifer [63] proposed an attacking method by using a machine learning approach. The main idea is to model the correlations between sensitive and non-QID attributes, using Naive Bayes classifier. More specifically, the problem is a classifier problem where an anonymised dataset is used as trained data, prior knowledge (e.g. *smoke?* is either yes or no) can be classified into a sensitive value (e.g. whether *cancer* or not). Similar models have also been proposed in [36, 89].

Knowledge Hiding

Knowledge hiding is another method of preserving privacy [28, 7, 59, 86, 8, 54, 104, 72, 45]. The aim is to prevent an adversary from inferring sensitive knowledge patterns from the released data. A pattern is denoted as $X \rightarrow Y$, where X is the information that

Tuple ID	Smoke?	Group ID
1	y	1
2	y	1
3	n	2
4	n	2
5	y	3
6	n	3
7	y	4
8	y	4
9	n	5
10	n	5
11	y	6
12	n	6

(a) Quasi-identifier Table

Group ID	Disease
1	Cancer
1	Flu
2	Flu
2	None
3	Cancer
3	None
4	Cancer
4	None
5	Flu
5	None
6	Cancer
6	None

(b) Sensitive Table

Figure 2.8: An anonymous data protected by bucketisation

the adversary may know about individuals and Y is the sensitive information. Privacy is compromised when a sensitive pattern is established with sufficient confidence [107].

A common approach to reducing the confidence of sensitive patterns is to increase or decrease the supports of some sets of items in the released dataset. Various methods have been studied, including synthetically generating part of the dataset [45], suppressing items [59, 86] or perturbing (i.e. adding noise to) the data [104]. Regarding the question of utility, the protection also needs to ensure that the data is still useful for analysis, in terms of the patterns that can still be mined, and to avoid generating new patterns that do not exist in the original data.

2.1.7 Privacy Issues in Releasing Multiple Relations

Much research has been done to extend the k-anonymity principle. Those algorithms assume that each individual is stored as one row in a table. In practical applications, data publishing is more complicated and may involve several releases or data has to be stored in multiple relations. To address these issues, existing definitions and algorithms are insufficient. In this section, we briefly summarise the privacy issues for other types of relational data releasing.

The recent works [83, 114, 9, 15, 17, 38, 87, 108] have investigated the problem and analysed it under two scenarios:

- **Multiple Views Release:** In this scenario [83], several releases (views) of the same underlying table are published at the same time or at different times. This publishing may cause several privacy issues that have not been addressed in previous privacy models (i.e. k-anonymity or ℓ-diversity) such as, each view may be generalised separately, enabling an adversary to join views to identify an individual. There are no constraints on releasing multiple views to prevent this type of attack.

- **Sequential Release:** In this scenario, after a dataset is released, new information could be available for releasing (e.g. adding new attributes to support more studies [15], or adjusting (i.e. adding, modifying or removing) records of the previous release [38, 87, 108]). As an adversary may have access to all released versions, the anonymisation of later releases should guarantee that an adversary cannot use previous anonymised data to breach privacy. For example, for the same record, it is not necessary for it to be generalised in the first release as the data has already satisfied a privacy constraint. However, in the next release, to satisfy such a constraint, the record is generalised. The difference in two releases will allow an adversary to infer additional information to identify individuals.

2.2 Privacy in Transaction Data

Similar to publishing relational datasets, publishing transactions could unveil the identity of a person associated with a particular transaction if an adversary has some partial knowledge about that person. However, as we have explained in Chapter 1, transactions have variable lengths and high dimensionality, and protecting their privacy by using typical approaches designed for relational data (e.g. k-anonymity, ℓ-diversity) will not be effective. In this section, we review related attacking and protecting methods for transaction data.

2.2.1 Anonymisation Methods

Like relational data, generalisation and suppression are popularly used in anonymising transaction data. They work in exactly the same way as we reviewed in Section 2.1.1 for relational data. One important difference is that suppression in transaction anonymisation is no longer a special case of generalisation. In transaction data, suppressing an item means that the item is removed from the data.

2.2.2 k^m-anonymity

Anonymising transaction data is quite different from well-studied k-anonymisation of relational data because the data has no well-defined set of quasi-identifiers and sensitive values. Any subset of items in a transaction could play the role of quasi-identifiers for the remaining (sensitive) ones. Another fundamental difference is that transactions have variable lengths and high dimensionality. To protect the privacy of transaction data in such conditions, Terrovitis et al. [100] proposed the k^m-anonymity privacy model:

Definition 6 (k^m-anonymity)**.** *Given a set of transactions T, no adversary who has background knowledge of up to m items of a transaction can use these items to identify*

less than k transactions from T.

Intuitively, Definition 6 gives a constraint for protecting privacy of transaction data T by ensuring that there is no set of m items (or any of its subsets) that are supported by less than k transactions in T. For example, given transactions in Figure 2.9 (a), where values in *TID* are used for illustration and not included in published data, *shopping items* are bought by four individuals. Assuming that an adversary knows a maximum of 2 items bought by an individual, they can easily identify transactions associated with some individuals. For example, knowing someone has purchased {skim milk, hard cheese}, an adversary can link t_1 to this individual. Figure 2.9 (b) shows 2^2-anonymous data where *skim milk* and *choco milk* are generalised to *milk* following an ontology given in Figure 2.10. Now, there are at least two transactions that contain a combination of any two items, and an adversary cannot identify anyone in the released data by knowing a combination of any two items.

TID	Shopping items
t_1	{skim milk, hard cheese, soft cheese}
t_2	{choco milk, hard cheese}
t_3	{choco milk, hard cheese, soft cheese}
t_4	{skim milk, choco milk, soft cheese}

(a) Original transaction

TID	Shopping items
$\tilde{t_1}$	{**milk**, hard cheese, soft cheese}
$\tilde{t_2}$	{**milk**, hard cheese}
$\tilde{t_3}$	{**milk**, hard cheese, soft cheese}
$\tilde{t_4}$	{**milk**, soft cheese}

(b) k^m-anonymised transaction with $k = 2$ and $m = 2$

Figure 2.9: 2^2-anonymity based on an ontology in Figure 2.10

However, k^m-anonymity [100] and other related works [113, 50, 18, 75, 81] have two main limitations:

- Approaches do not support detailed privacy requirements enforcement. For example, in k^m-anonymity, all possible combinations of m items are required to be protected. In real applications, not all items need to be protected. Over-

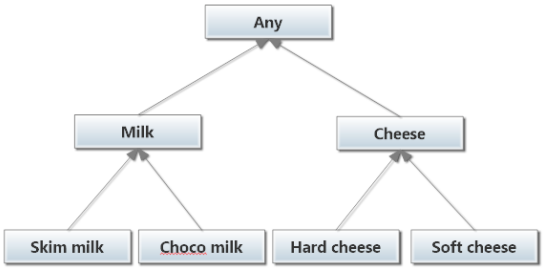


Figure 2.10: Ontology used to generalise transactions in Figure 2.9 (a)

protection could lead to unnecessary loss of data utility. It is desirable that a data publisher can specify, in detail, how data is to be protected.

- Generalisation is dependent on a hierarchy [50, 106], which is not flexible enough as a generalised item has to be a parent node of items that need to be protected. For example, if we want to generate a generalised item that contains both *skim milk* and *hard cheese*, all items in Figure 2.9 (a) have to be generalised to *any*.

In the following section, we will explain how these issues are addressed.

2.2.3 Constraint-based Anonymisation of Transactions

To ensure that transactions are not over-generalised, Loukides et al. [77] proposed a constraint-based model for transaction protection. This allows a data publisher to specify two sets of constraints. In the following section, we first explain how information loss is an important issue in protecting privacy and then explain how constraint-based anonymisation can be used to address this problem.

Information Loss in Anonymising Data

The term *information loss* is used to indicate how much information is lost during the generalisation and suppression of an anonymisation process. How information loss

may be measured is a challenge [52, 42, 6, 90, 103, 14, 95, 14]. Broadly, the measurement should be independent from any anonymisation method used to sanitise the data because the data publisher may not know how the released data may be analysed by the recipient. Askari et al. [6] argued that the utility of a dataset is a correlation between its attributes. As a result, the data distribution is a good indicator of its usefulness. For example, if an attribute of an original dataset has a uniform distribution, then the anonymised data is considered to have less utility if it has a non-uniform distribution. Based on this observation, Askari et al. proposed a utility measurement based on entropy and argued that information loss is the amount of increase of entropy before and after anonymisation.

COAT

Loukides et al. [77] proposed a constraint-based anonymisation method (COAT) to reduce information loss that may have occurred in k^m-anonymisation. To protect a set of transactions, COAT allows a data publisher to specify *privacy constraints* and *utility constraints*. A privacy constraint defines a set of items that need to be protected. With a parameter $k \geq 2$, any subset of a privacy constraint is required to have at least k support or no support in the released data, for the constraint to be deemed satisfied. Utility constraints define how an item may be generalised. For example, given an original dataset in Figure 2.11 (a), Figure 2.11 (c) shows the result of anonymisation by COAT, where the privacy constraints are {{a,b},{e,f}} and the utility constraints are {{b,d,g},{c,e,f}}. Comparing this result with an anonymised form, using k^m-anonymity in Figure 2.11 (b), COAT made fewer modifications, because it focused on specific items for protection, and therefore more utility is preserved after anonymisation (i.e. there is less information loss).

COAT generalises items by set-based generalisation which replaces an item with a set of items. Because COAT assumes that items are isolated literals, Figure 2.11 (c) is assumed to be protected. However, in practice, an adversary may use a non-generalised

TID	items
t_1	{a,d}
t_2	{a,b,g}
t_3	{c,f}
t_4	{e,h}

(a) Original transactions

TID	items
t_1	{(a,b),(c,d)}
t_2	{(a,b),(g,h)}
t_3	{(d,c),(e,f)}
t_4	{(e,f),(g,h)}

(b) 2^2-anonymised transactions

TID	items
t_1	{a,(b,d)}
t_2	{a,(b,d),g}
t_3	{c,(e,f)}
t_4	{(e,f),h}

(c) Anonymised data using COAT with $k = 2$

Figure 2.11: Comparison between k^m-anonymity and COAT

item (e.g. a in t_1 or a, g in t_2) to attack the generalised item (e.g. identifying if b, d or both are original).

2.2.4 Attacking Sparse Data

Sparsity is one of the properties of transaction data (i.e. records do not often share the same attribute values). Narayanan and Shmatikov proposed an attacking method [81] based on this property (the work is further analysed in [32]). The attacking mechanism is based on an observation that when data is sparse, an adversary, who has some random knowledge about an individual, has a high chance of matching this knowledge with an individual in the dataset. Their attacking framework has three components: scoring, matching and selection.

Scoring measures the similarity between an adversary's knowledge and a transaction, in an anonymous dataset. Adversary knowledge is modelled as a vector v, where its items are the known information about an individual. A released transaction is also modelled as a vector r, where its items are anonymised items in the transaction. The similarity between the two vectors is then calculated as follows:

$$Sim(v, r) = \frac{\sum Sim(v_i, r_i)}{|supp(v) \cup supp(r)|} \tag{2.1}$$

where $Sim(v_i, r_i)$ is the similarity between i^{th} attribute in vector v and r (0 if they

are deemed to be similar and 1 if they are not), $supp(v)$ and $supp(r)$ are supports of attributes in v and r (i.e. attributes have a value of 1).

Matching criterion is what the adversary will apply to the set of scores given by scoring function to determine if there is a match between an adversary's knowledge and the anonymised data. Given a set of anonymised transactions $\tilde{\mathcal{T}}$, the adversary computes the matching set $M = \{r \in \tilde{\mathcal{T}} : Sim(v, r) > \alpha\}$, where α is a predefined threshold, to determine if there is a match.

Record selection selects the "best" transaction. The result of this step identifies a transaction which is most similar to the adversary's knowledge. One simple approach is to select a transaction $r' \in M$ with the highest score.

Our work has a similar attacking framework which has two components: scoring and matching (which we call elimination in our framework). However, each component of our framework operates differently. Our scoring measures semantic relationships among items in the released data, whereas Narayanan's framework measures the similarity between an adversary's knowledge and the anonymised data.

2.3 Privacy in Statistical Data

In the previous sections, we considered one scenario in data publishing where data containing personal information is to be published. In this section, we consider another scenario where aggregated information about individuals is to be published. Such data is often called *statistical data* [2]. Applying privacy protection methods, discussed in the previous sections, to this type of data often leads to considerable information loss, unnecessarily. For example, the following query returns 1 in the original dataset $\mathcal{T}$ in Figure 2.12 (a):
SELECT COUNT(*) FROM $\mathcal{T}$ WHERE Disease="pneumonia" AND Age $\leq$ 30 AND Zip code BETWEEN 10001 AND 20000
However, it returns 2 in anonymised form in Figure 2.12 (b).

Age	Sex	Zip code	Disease
23	M	11000	pneumonia
27	M	13000	dyspepsia
25	M	59000	dyspepsia
59	M	12000	pneumonia
61	F	54000	flu
65	F	25000	gastritis
65	F	25000	flu
70	F	30000	bronchitis

(a) Original data with QIDs are {Age,Sex,Zip code} and SA is {Disease}

Age	Sex	Zip code	Disease
[21,60]	M	[10001,60000]	pneumonia
[21,60]	M	[10001,60000]	dyspepsia
[21,60]	M	[10001,60000]	dyspepsia
[21,60]	M	[10001,60000]	pneumonia
[61,70]	F	[10001,60000]	flu
[61,70]	F	[10001,60000]	gastritis
[61,70]	F	[10001,60000]	flu
[61,70]	F	[10001,60000]	bronchitis

(b) 2-diverse version of data in Figure 2.12 (a)

Figure 2.12: An example of perturbed data

Compared with the methods discussed in the previous sections, this type of data publishing requires very different privacy methods and principles to ensure privacy of individuals, while preserving data utility for analysis. In the following sections, we discuss typical attacking methods and privacy models in statistical data.

2.3.1 Perturbation Methods

Perturbation represents one common approach to privacy preserving data publishing for statistical data. The approach is to protect privacy and, at the same time, preserve statistical information (e.g. counting, mean, and standard deviation) by adding noise, using various methods, into the released data: additive noise [1], multiplicative noise [65], data swapping [31], data rotating [24], re-sampling [89], data shuffling [117], etc. Compared with other anonymisation operations discussed earlier, one limitation of the perturbation methods is that the published records are “synthetic”, in that they do not correspond to the real-world entities represented by the original data. Below, we discuss several commonly used perturbation methods that include additive noise and

multiplicative noise.

Additive noise [1, 13, 4] is commonly used in perturbing statistical data. The general idea is to replace original sensitive values x by:

$$y = x + r \tag{2.2}$$

where r is a random value drawn from a distribution. For example, Figure 2.13 shows anonymised data that is protected by additive noise, where random values are added into Age and Zip code values using two zero mean random vectors R_{age} and $R_{zipcode}$

$R_{age} = \{2, -3, 1, 2, -2, 1, 2, -3\}$

$R_{zipcode} = \{1000, -1000, 2000, -3000, 1000, 1000, -3000, 2000\}$

That is, each item in vector R_{age} and $R_{zipcode}$ is added to each value of attribute Age and Zip code, respectively. Now an adversary who knows QIDs about individuals, cannot precisely query the disease of those individuals, yet statistical information of the data is preserved and some aggregation queries can be answered accurately using the perturbed data. For example, both the original in Figure 2.12 (a) and the anonymised data in Figure 2.13 return 1 for the above query.

Age	Sex	Zip code	Disease
25	M	12000	pneumonia
24	M	12000	dyspepsia
26	M	61000	dyspepsia
61	M	9000	pneumonia
59	F	55000	flu
66	F	26000	gastritis
67	F	22000	flu
67	F	32000	bronchitis

Figure 2.13: An example of perturbed data that is protected by additive noise from original data in Figure 2.12 (a).

Although additive noise can be used to protect privacy in statistical data, this method

does not preserve the relationship between attributes in the released data, leading to poor accuracy in terms of the distance-based mining methods [67]. *Multiplicative data perturbation* [65] is commonly used to address this issue. It replaces a value x by

$$y = m \times x \tag{2.3}$$

where m is a random value that is drawn from a distribution. Let M be a matrix that contains noise values for perturbing the data, X is a matrix that contains the original data which is to be perturbed by M (i.e. $Y = MX$). M is chosen with a certain property such as if M is an orthogonal matrix, then the perturbation preserves Euclidean distance between X and Y, exactly. That is, for any attribute values x_1, x_2 in X, their corresponding attribute values y_1, y_2 in Y, the Euclidean distance between x_1,x_2 and y_1,y_2, does not change, such as $||x_1 - x_2|| = ||y_1 - y_2||$.

As Euclidean distance between attribute values is important for data mining, the perturbation that preserves this property allows many important data mining algorithms (e.g. k-means [48]) to be applied to the perturbed data, with the results being similar to, or exactly the same as, those produced using the original data.

2.3.2 Privacy Models

(c, t)-isolation

Isolation attack is a common attacking method on statistical data, where an adversary has auxiliary information about an individual and uses this information to query data in order to isolate the individual, even though the adversary does not have the exact information about an individual. To prevent this attack, it is necessary to ensure that any individual is difficult to isolate from others. Chawla et al. [21] proposed a privacy principle, (c, t)-isolation to prevent an isolating attack that models each individual's information as a data point in a space and ensures that in a small space area (specified by parameter c), there are at least t data points. That is, for each individual's information,

at least $t-1$ other individuals have "similar" information, making it is more difficult to isolate one of them.

Definition 7 ((c,t)-isolation)**.** *Given a dataset $\mathcal{T}$ where items can be represented as data points in a multidimensional space, let $y \in \mathcal{T}$, and any point q such as their distance be $\delta = ||y-q||$. y is said to be (c,t)-isolated by q if there are less than t data points ($t \geq 2$) in an area that is specified by a "ball" of radius $c\delta$ ($c \geq 2$) and centre point q.*

Intuitively, (c,t)-isolation considers the privacy of an individual, who has a data point of q, as protected if there are at least $t-1$ other individuals who have a data point near to q with a distance no more than $c\delta$, where δ is the distance between q with a given data point in $\mathcal{T}$ and c is a constraint to specify the radius of the "ball" (note that $c=1$ can always isolate one point and $c<1$ cannot isolate any point).

ϵ-differential Privacy

An organisation may not release the whole data to a third party, but they allow third parties to access their data through some *restricted queries* (e.g. aggregation query over some fields in a database). In such a situation, Dwork claimed that it is impossible to absolutely prevent disclosure due to the impossibility of adversary background knowledge prediction [64, 72].

For example, let $Q_1(x)$ be a query over data in Figure 2.12 (a) to count the number of *females* who have *flu* and aged over x. Also, let $Q_2(x)$ be a query over data in Figure 2.12 (a) to count the number of *females* who are aged over x. An adversary, who has the knowledge that Mary is the youngest and is also present in the data, can identify Mary's disease by using Q_1 and Q_2 as follows:

- $Q_2(0) = 4$, $Q_2(60) = 4$ and $Q_2(61) = 3$. With answers to three Q_2s, the adversary infers that the youngest female is 61 years old. Therefore, they learn that Mary is 60 years old.

- $Q_1(60) = 2$, $Q_1(61) = 1$. With answers to two Q_1s, the adversary infers that there is one female who is 60 years old and has flu. Therefore, they infer that Mary has flu.

The key point of the above attack is that an adversary can try to use a set of queries to infer information about individuals. This problem is very different from those discussed earlier. Here, privacy leakage is not from the data but from the *releasing mechanism* (i.e. Q_1 and Q_2 functions). The privacy concern in this example is that if an adversary knows Mary's personal information, removing Mary's record from the dataset would not affect the output significantly. To achieve that, Dwork proposed a robust privacy model [34], called *differential privacy*. Differential privacy ensures that the removal or addition of a single record in a dataset does not "substantially" affect the outcome of the dataset. The model is formally defined as follows:

Definition 8 (ϵ-differential privacy)**.** *A randomised function $\mathcal{K}$ gives ϵ-differential privacy if, for all data D_1 and D_2 differing in at most one element, and a set of possible outputs S in the range of $\mathcal{K}$:*

$$Pr[\mathcal{K}(D_1) \in S] \leq e^{\epsilon} \times Pr[\mathcal{K}(D_2) \in S]$$

where ϵ is a constant to ensure a small change of probability of outputs that could be made by two datasets D_1 and D_2, which are different at most in one record (e.g. removal or addition of one record).

2.4 Privacy in Text

Like other types of data, text data may also contain identifier and sensitive information. Therefore, releasing text data (e.g. a patient discharge report or a legal document) could violate privacy. Protecting privacy in text is more challenging because there are semantic relationships among items which an adversary can use to infer additional information.

In this section, we discuss common privacy issues in text, including identity disclosure and sensitive disclosure, and how semantic relationships among items affect the privacy protection of text.

2.4.1 Scrubbing

One common assumption about relational and transaction data is that the released data does not contain identifier information. This assumption is reasonable for such types of data because they are structured or semi-structured (i.e. there are descriptors for values of data), therefore, it is easy to automatically pinpoint identifier information and remove it before releasing the data. However, this assumption is not reasonable in text data as text is unstructured data (i.e. there is no descriptor for values) and identifier information may exist among items in different forms, which are difficult to identify and remove.

Scrubbing is used to locate and then replace identifier or sensitive information in text. To specify some information that is to be protected, most scrubbing methods use a predefined set of terms (e.g. HIPPA [19] defines 18 types of information that need to be removed).

Sweeney [97] and Douglass et al. [33] used parsers with a predefined template to replace terms such as name, address, and phone number. For example, Figure 2.14 gives an example of a template for parsing phone numbers where the right column contains templates and the left column contains the terms deemed to be phone numbers, according to the template. The limitation is that some information may not fit into any template, thus it may not be possible to parse, such as disease name.

In the medical application domain, as medical terms are important for studying, Berman [11] proposed a scrubbing method that is used to replace all personal information by parsing the text, while leaving terms which are in the Unified Medical Language

Phone numbers	Templates
255 - 1423	ddd - dddd
(304) 255 1423	(ddd) ddd - dddd
304 / 255 - 1432	ddd / ddd - dddd
255 - 1000 ext 1423	ddd - dddd ext* d*
extension 1423	ext* d*
phone: 255 - 1423	{tel*, ph*} ddd - dddd

Figure 2.14: Sample template for parsing phone numbers [97]

System (UMLS)[1].

As much work in the medical field is rule-based, several issues remain unsolved by this approach such as word ambiguity (i.e. finding the meaning of a term) or considering the semantics between terms in a text. In other words, it is necessary to consider the meaning of terms in protecting privacy. To solve this problem, "tag-like" tools have emerged to specify the roles (e.g. noun or adjective) of items in a text such as part-of-speech (POS) [102]. However, POS still cannot address the semantics of a tagged term (e.g. the meaning of the term still cannot be known). Ruch et al. [91, 92] adopted MEDTAG - a framework to assign categories (e.g. disease name, disease symptoms, name, and address) for the terms of a medical domain text. Based on semantic-tagged terms, the authors also proposed rules to remove information that may violate privacy.

However, protecting text data by scrubbing is not sufficient because an adversary can still identify an individual by unique combinations of the information left in the text. Like in the case of relational and transaction data, both identity and sensitive information may be disclosed as a result. In the next section, we will review another approach which addresses this problem.

[1]UMLS is a database that contains more than 2 million medical terms.

2.4.2 Transformation and Anonymisation

With relational and transaction data, generalisation has been shown to be a better approach than removing information, in terms of preserving utility while protecting privacy. On the other hand, scrubbing often uses dictionaries or statistical learning techniques that may miss the detection of some identifiers. Thus, it is worth considering if text data can be transformed so that generalisation may be applied. In [40, 41], Gardner et al. proposed HIDE, a framework to anonymise text in three steps: (1) attributes are extracted from text using a *named entity recogniser* [68]; (2) a person-centric identifier is used to classify extracted attributes into QID and SA, and as a result, the text data is transformed into relational data; and finally, (3) k-anonymity is adopted to anonymise the relational data.

A limitation of HIDE is that it only seeks to protect extracted data (e.g. extracted relation or transaction) and not the text itself. Therefore, only anonymised structured data can be published, which limits its usefulness. Also, the HIDE framework extracts items and protects them without considering the semantics of the terms. This may not be sufficient as we have shown in Figure 1.5 in the Introduction.

2.4.3 Text Generalisation

Another alternative to protect privacy in text is to make the overall content more general. A typical method of doing this is to identify terms (e.g. nouns and noun phrases) and then generalise them in text by using an ontology such as WordNet[2]. Jiang et al. proposed the t-*plausibility* [58] model for generalising text.

Definition 9 (t-Plausibility)**.** *Let d be a text that can be generalised to $\tilde{d}$ by replacing terms in d from an ontology o. $\tilde{d}$ is t-plausible if at least t base texts (including d) can*

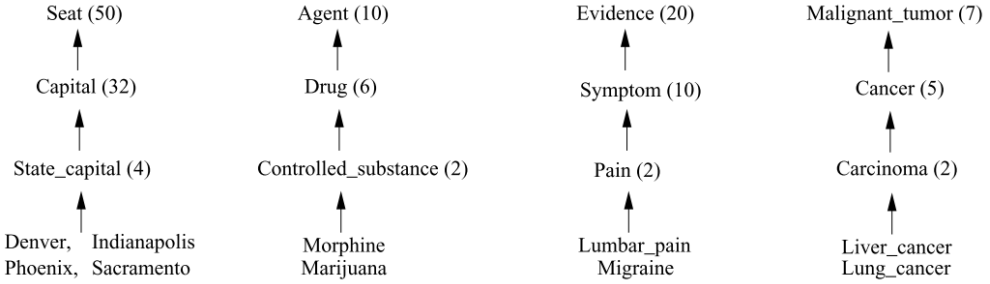
[2]WordNet [80] is the largest lexical database for English. Its words - noun, verb, adjective and adverb - are organised into an ontology where a parent node is a general form of child node. Because of these properties, WordNet is popularly used in generalising text.

A **Sacramento** resident purchased **marijuana** for the **lumbar pain** caused by **liver cancer**.

(a) Original text

A **state capital** resident purchased **drug** for the **pain** caused by **carcinoma**.

(b) Generalised text



(c) A sample ontology from WordNet

Figure 2.15: An example of text generalisation [58]

be generalised to $\tilde{d}$.

A base text is text that contains *the most specific* terms selected from an ontology. A text is called t-plausibility when it *associates* with at least t base texts. The number of associations is defined as the possible replacements of terms in the text. For example, Figure 2.15 (a) contains a sample text that is to be generalised, where the bold terms are the targeted nouns and noun phrases. Figure 2.15 (c) contains an ontology used for generalisation. Terms in a child node can be generalised to its parent node. For instance, *Sacramento* can be generalised to *State capital*. An integer number next to each term indicates how many terms can be generalised to it. For instance, *State capital* can be generalised from 4 other terms, *Drug* can be generalised from 6 other terms (note that other sub-branches of Drug are not shown in Figure 2.15 (c)).

By using the ontology in Figure 2.15 (c), a text in Figure 2.15 (a) is generalised to a text in Figure 2.15 (b) which satisfies t-plausibility with any $t \leq 96$. t-plausibility ensures that the original text cannot be uniquely identified because there are at least 96 possible texts which can be constructed, based on the ontology in Figure 2.15 (c).

Cumby et al. [30] treat the protection problem as a multi-class classification problem by proposing the k-confusability privacy model. The model is very similar to k-anonymity in relational data, in which each document is required to be classified into at least k different topics, assuming that both sensitive and non-sensitive topics are known, and each topic is defined as a set of related terms. A document matches a topic when it contains all terms of the topic. The Cumby model ensures that given a set of items $X = \{x_1, ..., x_{|X|}\}$ in a document, the probability X is classified into a sensitive topic s is:

$$P(s|X) = \frac{P(s)}{P(X)} \prod_{i}^{n} P(x_i|s)$$

Scrubbing, transformation-then-anonymisation and text generalisation are different, in terms of how a data publisher wants an output to be based on the purpose of publishing data. In scrubbing, terms are replaced by a code which may not have any meaning or be completely removed from the text. In transformation-then-anonymisation, the output is not text itself, but the algorithm generates an anonymous relational or transaction data. In text generalisation, it generates a more general text than the original text. However, the same issue remains in these approaches in that they do not consider semantic relationships between the terms in the text. Therefore, an adversary can still infer sensitive information, using the non-sensitive information. In the next section, we will review some related works that take into account semantic relationships, while anonymising text.

2.4.4 Context Awareness

Semantic relationships among the terms in text create a context. Protecting privacy in text without considering its context may not guarantee privacy because an adversary may use the context to narrow down their "guess". We call this type of attack *semantic attack*.

Chakaravarthy et al. [20] proposed a scheme, called ERASE, which detects sensitive

elements using a database of entities (e.g. persons, products, and diseases). Each entity in this database is associated with a certain context, which contains a set of terms related to that entity. For example, the context of a disease could include symptoms, treatments, etc. Using this information, their method detects terms to be sanitised by looking for sensitive entities and their context in the database. Due to the cost of manually compiling such a database, the method is mainly designed for domain-specific application.

In [5], Anandan and Clifton argued that t-plausibility [58] is not practical to protect sensitive information because the method generalises terms independently. In practice, some sets of terms are more likely to appear together in a specific context. For example, a context about a cancer disease is more likely to contain other terms such as common symptoms of this disease. This means that when two terms are often used together in the same context, generalising one but not the other is not useful, as an adversary can use the non-generalised item to infer the generalised one. For example, if *IPhone* and *Apple* are two terms in a context, t-plausibility may generalise *IPhone* to *smartphone*. However, the term is not protected because an adversary still can easily infer this *smartphone* is *IPhone*. This work showed that learning the dependency between terms in a document can improve privacy protection. However, this dependency can be strong or weak, depending on different terms. The main contribution in [5] is to propose a measurement to estimate how a term is dependent on another term, measured by the relative information gain of the two terms [62].

Sánchez et al. [93] consider a similar problem. Their method measures the semantic relationship between two terms by using point-wise mutual information (PMI) [12] and adopts the World Wide Web (WWW) as a repository to detect related terms [96, 93, 94, 26].

2.5 Summary

This chapter has reviewed some important privacy models for relational, transaction, statistical and text data. In relational data, we mainly considered the privacy problem under link attacks and how k-anonymity and its extensions protect privacy from link attacks. In transaction data, we discussed how privacy models differ from k-anonymity for relational data, including k^m-anonymity and constraint-based anonymisation. In statistical data, we briefly discussed several typical perturbation methods and privacy models such as (c, t)-isolation and ϵ-differential privacy. In text, we showed how the privacy issue is different from privacy problems in other types of data, and we then discussed how to protect privacy in text, taking into account semantic relationships among items.

Although various contributions have been made in this research area, our work is unique compared with the published works. Firstly, many de-anonymisation techniques rely on an adversary's background knowledge to attack the data, but we will show that without their background knowledge, it is still possible to practically reconstruct original data from anonymised data. Secondly, our method considers the relationships within information, and we use WWW to establish such relationships. We will describe our method in detail in the following chapters.

Chapter 3

Semantic Attack

Set-based generalisation is one common approach to protecting privacy in transaction and relational data. However, this method protects data *syntactically*, whereby items are assumed to be contextless or even meaningless literals, and how they form a set to replace (or generalise) an item in a particular context is insignificant.

In Chapter 1, we analysed privacy issues of set-based generalisation transaction data. In this chapter, we propose an attacking framework, called *semantic attack*, and we discuss one component of the framework, the *scoring component*. Our work is different from previous attacks in that we use semantic relationships between items in the set-based generalised data to break protection, whereas other works assume that an adversary has some knowledge about individuals [81].

3.1 Semantic Relationship

Measuring semantic relationship is important in our work as we use it to specify how items are related in the context of a transaction. In this section, we first discuss two common types of semantic relationship that may exist among the terms. For example, *string* and *cord* are related in terms of sharing a similar concept, while *iphone* and *apple* are related because they are meaningful in a specific context. We then discuss how such relationships are used in our Semantic Attack.

3.1.1 Ontology-based vs Corpus-based Approaches

There are two common approaches to measuring semantic relationship between items: Ontology-based and Corpus-based.

Ontology-based Measurement

In the context of computer and natural language processing, an ontology defines a set of representational primitives with which to model a domain of knowledge [47]. By organising concepts (terms) of a domain in a hierarchical way and describing relationships between terms using a small number of relational descriptors, an ontology supplies a standardised vocabulary for representing entities in the domain. For example, Figure 3.1 shows an ontology of cancer where higher nodes in the hierarchy are items that have more general meanings and lower ones have more specific meanings, which are subsumed by their parent nodes.

Ontology-based semantic measurement uses an ontology to specify the semantic relation between items based on the relative position of items in an ontology. For example, items that have the same parent node (e.g. *skin cancer* and *oral cancer* in Figure 3.1) are considered to be more related than the items having different parent nodes (e.g. *skin cancer* and *Hodgkin's disease* in Figure 3.1). In other words, the method defines semantic similarity, in terms of how likely two items are deemed to have the same or similar meaning.

This property of ontology is useful for protecting privacy, especially in generalisation. An ontology can be adopted to generalise an item to a more general but semantically equivalent item [118, 76, 35, 58]. The generalisation following this approach considers the semantic relationship between items but does not take into account the context within which the items appear.

Another problem is that although ontologies such as MeSH [53] and WordNet [80]

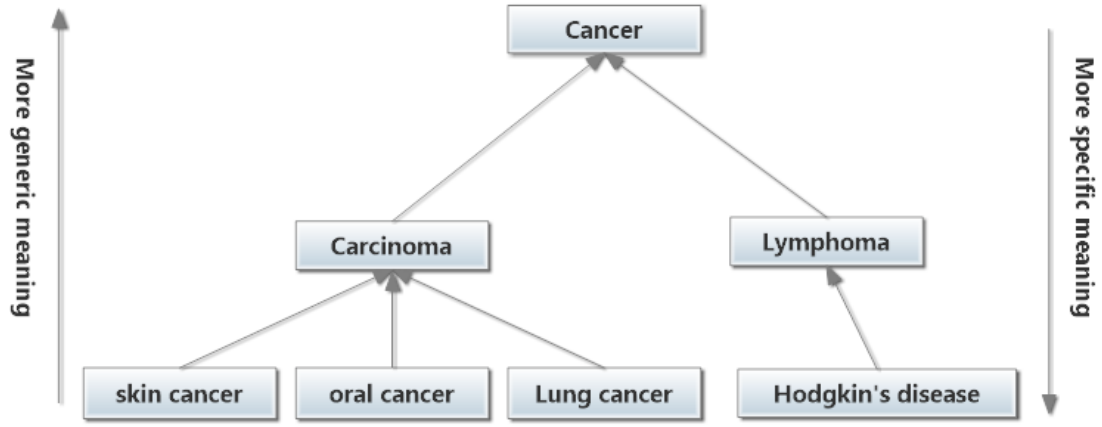


Figure 3.1: A small branch WordNet's ontology of "Cancer"

have a large number of entities[1], they are not sufficient to cover all application domains. This limitation makes it difficult to use an ontology-based method in an application domain which is not fully specified. Manually constructing an ontology can be time-consuming and difficult because it requires expert knowledge.

As items can have multiple senses/interpretations, an ontology may not be used to specify the appropriate sense or interpretation effectively. For example, *string* and *cord* may be considered to be closely related if they are perceived as a line-like object, but if *string* is interpreted as a data type in programming language, then *string* is not related to *cord*.

Corpus-based Measurement

Corpus-based similarity measurement is another approach to specifying the semantic relationship between two items, in terms of how likely they are to be used together in a particular context [118, 76, 35, 58, 57]. For example, *lung cancer* is considered to be more related to *pain coughing* than to *Hodgkin's disease* because *lung cancer* and

[1]MeSH [53] and WordNet [80] they are commonly used in the privacy research area. MeSH ontologies contain items that are present in the medical and health-care application domain, while WordNet ontologies are for generic purposes

pain coughing often appear together in patient discharge reports.

This type of semantic relationship between items is often measured based on the co-occurrence of items in a particular context. Therefore, this approach requires a dataset to determine co-occurrences. The larger the dataset is, the more precise the measurement can be as items can be validated in many contexts.

Our Approach

Imagine a transaction about a patient that contains diseases and symptoms. To protect against sensitive disclosure, symptoms are generalised by using set-based generalisation. However, an adversary can still identify some symptoms as they are common symptoms of a specific disease in a transaction's context. Note that diseases and symptoms do not have similar meanings, hence they may not appear in an ontology with the same parent, but they can be closely related in a particular context. In our attack, we consider this situation, and we argue that it is not necessary to have two items having a similar meaning in order to attack the data, instead, corpus-based measurement is used to identify if two items in a transaction are closely related because the frequency of items used in similar context can help recover original items. In the next section, we will explain in detail how we adopt Normalised Google Distance, a method to measure similarity, based on the corpus-based approach, in our work.

3.1.2 Normalised Google Distance (NGD)

To sufficiently measure the semantic relationship between items, we adopt Normalised Google Distance (NGD) for three reasons:

- NGD is developed according to the corpus-based approach, in which semantic relationship between items is measured, based on how likely items are used in a particular context.

- NGD uses Google repository, which covers various application domains. As a result, our method does not depend on a specific application domain and it does not require much effort to manually construct a dataset for the measurement.
- Google search engine supports operators such as “AND" operator to search the combination of terms which appear in the same document; this feature is useful as it allows the relationship between two sets of items to be assessed.

NGD measures the relationship between items based on the estimated number of Web pages or documents that contain the items from the Google repository, using the *Google Index Function*:

Definition 10 (Google Indexing Function)**.** *A Google index function, $f(x)$, returns the approximate number of documents that contain item x in the Google repository.*

For example, $f(\text{“}HIV\text{”}) = 48,500,000$ as it is shown in Figure 3.2 that Google's search engine estimates that there are $48,500,000$ Web pages or documents containing HIV. This definition can also be extended to $f(x, y)$, which returns the number of documents that contain both x and y, such as $f(\text{“}HIV\text{”}\,AND\,\text{“}Fever\text{”}) = 29,200,000$, which is the number of Web pages that contain both HIV and Fever.

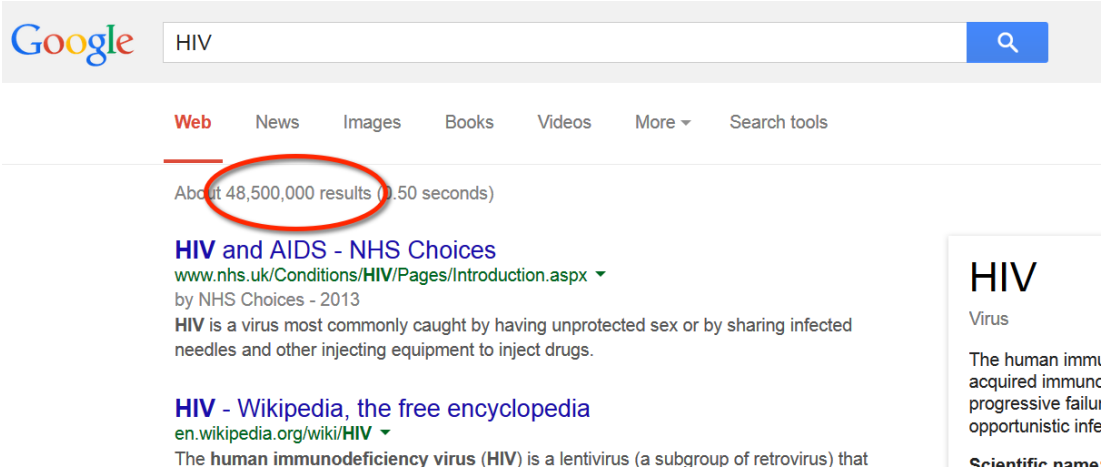


Figure 3.2: An example of $f(\text{“}HIV\text{”})$ by Google

NGD is derived from *information distance theory* [10] and *Kolmogorov Complexity* theory (K-complexity) [70]. Generally speaking, Kolmogorov Complexity theory defines that K-complexity of a variable x (written as $K(x)$) is the shortest logic (measured in bits) that can be used by a programming language to explain variable x; and in [10], authors showed that the semantic similarity between two terms x and y, given their K-complexity, is computed by:

$$E(x, y) = K(x, y) - min\{K(x), K(y)\} \tag{3.1}$$

where $(K(x, y))$ is the shortest length of a logic used to explain the combination of variables x and y. In [27], Cilibrasi and Vitányi proved that the index function of Google search engine $f(x)$ (Definition 10) is approximately similar to $K(x)$. From that, authors proposed a measurement extended from Equation 3.1, called Normalised Google Distance (NGD):

$$NGD(x, y) = \frac{max\{log(f(x)), log(f(y))\} - log(f(x, y))}{log(N) - min\{log(f(x)), log(f(y))\}} \tag{3.2}$$

where $f(x)$ denotes the number of Google pages containing x, $f(y)$ the number of pages containing y, $f(x, y)$ the number of pages containing both x and y, and $log(N) - min(log(f(x)), log(f(y)))$ is a normalising factor with N being the size of Google repository. The range of NGD is between 0 and ∞. The lower the NGD score is, the more closely the two terms are considered to be semantically related. For example, we have

$$NGD(paracetamol, HIV) > NGD(paracetamol, Cold)$$

which suggests that in general *Paracetamol* is more likely to be associated with *Cold* than with *HIV*. However, in some cases, NGD could be negative due to a conjunction of two terms returning the number of indexed pages greater than that for a single term, i.e. f(“a” AND “b”) > f(“a”). In such a scenario, we consider the result from the Google index function to be “unreliable” and therefore its semantic distance is also unreliable in deciding whether items are related or non-related. To avoid any wrong elimination, our attacking strategies ignore any item that has negative distance values.

Unlike methods that use local data to measure semantic relationships, NGD uses a search engine to query the number of occurrences of items from their repository which is expensive because it is often slow and only a limited number of queries is allowed in a period of time (e.g. Google accepts around 1000 queries in an hour, otherwise Google blocks the IP address and releases the information in the following few hours). This limitation affects the performance of our attack in a very large scale dataset, and could be addressed in future work.

3.2 Accuracy of NGD

Semantic relationships among terms are important in our work as we mainly rely on these results to detect non-original items in anonymised transactions. While there are a number of works (e.g. Li's method [74] and Islam's method [55]), which also provide a corpus-based approach, we prefer NGD as it can perform in various application domains. The aim of this section is to show that NGD is a reliable approach for specifying semantic relationships between items. More specifically, we experimentally test NGD's accuracy with other related semantic measurement, using a well-established benchmark.

3.2.1 The Method

Our method is to compare semantic relationships that are produced by various measurement methods and human judgement. We use the correlation $COV(X,Y) = \frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{n-1}$, where X and Y are the result of the measurement and estimation made by humans, $\bar{X}$ and $\bar{Y}$ are means, and n is the number of pairs of terms in the testing data. To compare this with human judgement, we use the benchmark in [84], which is popularly used in the information retrieval community for comparing semantic measurement methods. The benchmark contains 65 pairs of terms, whose semantic re-

latedness is judged by 32 independent participants, as shown in the first column of Table 3.1.

3.2.2 Results

Figure 3.3 compares semantic similarity between human judgement and NGD, Li's method [74] and Islam's method [55]. The correlation between the methods and human judgement is shown in Table 3.1. Note that, to compare our approach with other methods using a similar scale, we converted NGD to NGD', which has a scale from 0 to 1 by $1 - \frac{NGD}{MAX(NGD)}$, where $MAX(NGD)$ is the highest value in the dataset. The higher the NGD' score is, the more relationship there is between the terms it measures.

There is an important "point" at term ID 10. All terms below ID 10 are considered to be unrelated and the ones above ID 10 are considered to be related. We can see from the Table 3.1 that the correlation of NGD with human judgement is close to that of Li's method. However, using NGD would have an additional advantage as it does not require the preparation of a corpus for deriving measurement.

As a conclusion, we see that NGD is a reliable approach to establishing if two terms are related. Although NGD results do not precisely match human judgement, it can still clearly suggest whether two items are related or not. More specifically, in the comparison between NGD and human judgement in Figure 3.3, all items higher than ID 10 have higher similarity than items below ID 10. We see that there are still errors in the measurement, however, as these are small, the method is considered to be reliable enough for our work.

3.3 Conceptual Framework

In this section, we present a general framework, as shown in Figure 3.4, for our semantic attack. Our framework consists of a number of components, and we focus on

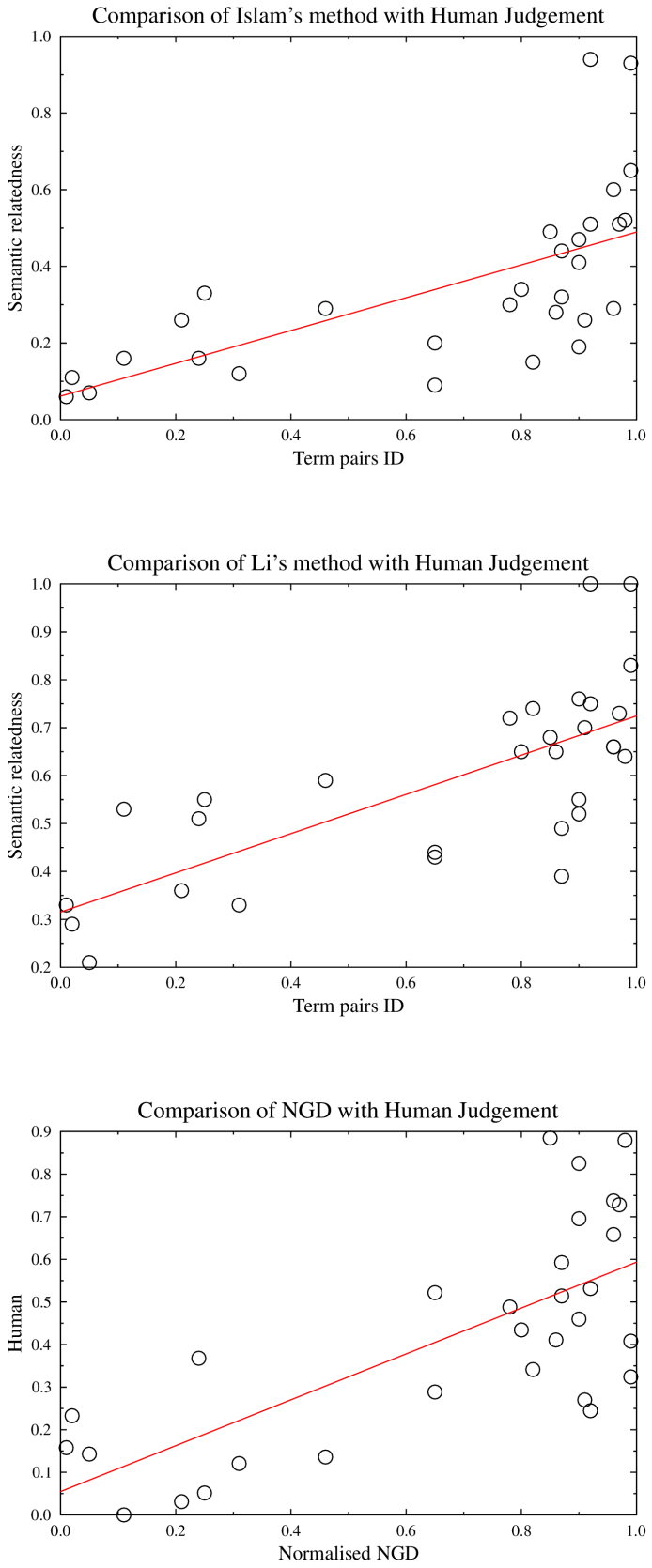


Figure 3.3: Comparison between different semantic measurements

Terms	Human[84]	NGD'	Li [74]	Islam [55]
1. cord-smile	0.01	0.16	0.33	0.06
2. autograph-shore	0.02	0.23	0.29	0.11
3. asylum-fruit	0.05	0.14	0.21	0.07
4. boy-rooster	0.11	0.0	0.53	0.16
5. coast-forest	0.21	0.03	0.36	0.26
6. boy-sage	0.24	0.37	0.51	0.16
7. forest-graveyard	0.25	0.05	0.55	0.33
8. bird-woodland	0.31	0.12	0.33	0.12
9. hill-woodland	0.46	0.14	0.59	0.29
10. magician-oracle	0.65	0.29	0.44	0.2
11. oracle-sage	0.65	0.52	0.43	0.09
12. furnace-stove	0.78	0.49	0.72	0.3
13. magician-wizard	0.8	0.43	0.65	0.34
14. hill-mound	0.82	0.34	0.74	0.15
15. cord-string	0.85	0.88	0.68	0.49
16. glass-tumbler	0.86	0.41	0.65	0.28
17. serf-slave	0.87	0.51	0.49	0.32
18. grin-smile	0.87	0.59	0.39	0.44
19. journey-voyage	0.9	0.46	0.52	0.41
20. autograph-signature	0.9	0.7	0.55	0.19
21. coast-shore	0.9	0.83	0.76	0.47
22. forest-woodland	0.91	0.27	0.7	0.26
23. implement-tool	0.92	0.53	0.75	0.51
24. cock-rooster	0.92	0.24	1	0.94
25. boy-lad	0.96	0.74	0.66	0.6
26. cushion-pillow	0.96	0.5	0.66	0.29
27. cemetery-graveyard	0.97	0.73	0.73	0.51
28. automobile-car	0.98	0.88	0.64	0.52
29. gem-jewel	0.99	0.32	1	0.93
30. midday-noon	0.99	0.41	0.83	0.65
	Correlation	0.72	0.72	0.64

Table 3.1: Comparing NGD with other methods

the development of scoring and elimination in this thesis. We first present an overview below and then describe each component in detail.

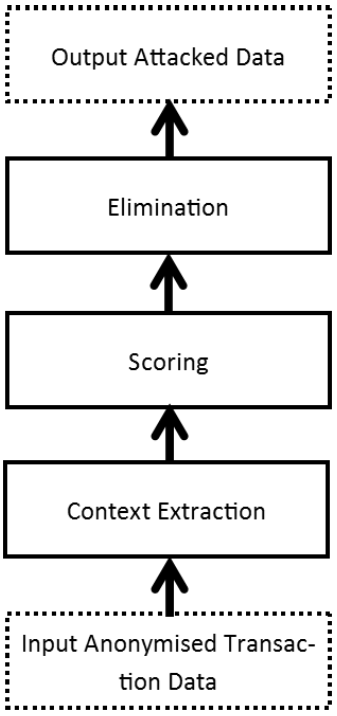


Figure 3.4: Conceptual Framework

We consider an input dataset to be a set of set-based generalisation transactions (e.g. the data produced by COAT [77]). The transactions contain generalised items that consist of both original items in a transaction, and non-original items that are added by the anonymisation process.

3.3.1 Context Extraction

We formally define the context of a transaction as follows:

Definition 11 (Transaction Context)**.** *Given a generalised transaction $\tilde{T}$, $C \subset \tilde{T}$ is a context of $\tilde{T}$, where any item $i_k \in C, 0 \leq k \leq |C|$ is trivially generalised.*

Intuitively, the context of a transaction is a set that contains items that are not generalised in an anonymisation process. In the case where $\tilde{T}$ does not contain any trivially

generalised item, $C = \emptyset$, and we consider the transaction to be incapable of being attacked using context. Note that a context is different from the concept of an application domain. A set of transactions may be obtained from a single application domain, but can contain many different contexts.

In the case where $|C|$ is large, it requires a very expensive computation as our approach analyses semantic relationships between generalised items with each context item of a transaction. Furthermore, using many context items may give a wrong estimation of semantic relationships if a transaction contains multiple contexts. For example, a patient discharge report may contain multiple diseases in the patient's history.

In our work, we consider a specific case where we try to extract context items of a transaction that are suitable for a given generalised item. More specifically, we extract the items in the context that "stayed close" to a given generalised item by the items' positions in the dataset (we will explain in detail how we deal with this for each dataset that we use in our experiment in Chapter 5). In that way, the extracted context will be related to the generalised item and is useful in establishing their semantic relationship. For example, given a transaction $\langle$*heart disease, (blood pressure, icd, limbs, injury), weakness, dizziness*$\rangle$, we use *heart disease* and *weakness* as context items for generalised items $(bloodpressure, icd, limbs, injury)$.

3.3.2 Scoring

Scoring is the component that establishes relationships between items in a generalised item and transaction context. For example, in Figure 1.5, to attack the generalised item (*blood pressure, icd, limbs, injury*) (i.e. identifying non-original and original items), the scoring component measures similarity between items in the generalised item (*blood pressure, icd, limbs, injury*) and the items in the context *heart disease, weakness*. We use a *distance table* to record the result of scoring. For example, Figure 3.6 is a distance table for the generalised item in Figure 1.5. We will explain the

distance table in the next section.

3.3.3 Elimination

Elimination is the component that contains a set of criteria to eliminate non-original items from a distance table, based on the scores. This corresponds to eliminating items which are less related to the context of a transaction. Discussions on such strategies will be given in Chapter 4.

3.4 Basic Denotations and Terminologies

Let $\mathcal{I} = \{i_1, ..., i_m\}$ be a finite set of literals called *items*. A *transaction* T over $\mathcal{I}$ is a set of items $T = \langle a_1, a_2, \ldots, a_k \rangle$, where each $a_j, 1 \leq j \leq k$ is a distinct item in $\mathcal{I}$. A transaction dataset $\mathcal{D} = \{T_1, ..., T_n\}$ is a set of transactions over $\mathcal{I}$.

Definition 12 (Itemset and Support)**.** *Any subset $I \subseteq \mathcal{I}$ is called an itemset. An itemset I is supported by transaction T if $I \subseteq T$. We use $\sigma(I, \mathcal{D})$ to represent the number of transactions in $\mathcal{D}$ that support I, and we call these transactions supporting transactions of I in $\mathcal{D}$.*

For example, $\langle$*gangrene, limbs, injury*$\rangle$ is a transaction in Figure 1.3. $\langle$*limbs, injury*$\rangle$ is an itemset and is supported by T3, and has the support of $\sigma(\langle$*limbs, injury*$\rangle, \mathcal{D}) = 1$. T3 is its supporting transaction.

When the support for an itemset is low, i.e. the itemset appeared infrequently within a transaction dataset, an attacker may use it to identify an individual with a high probability of success. A popular approach to ensuring that such itemsets would not compromise privacy is set-based generalisation [77], where some individual items are replaced by a set of items.

Given the definition of set-based generalisation in Definition 1, we denote a generalised item by listing its items in brackets, e.g. *(blood pressure, icd, limbs, injury)* in Figure 1.5, and we interpret a generalised item as representing any non-empty subset of its member items, e.g. *(blood pressure, injury)* may represent *blood pressure*, *injury* or both. Generalisation can help prevent identity disclosure as it increases the number of transactions in the dataset that may be linked to an individual through a combination of items [77]. For example, consider the mapping of item *injury* in Figure 1.3 to a generalised item *(blood pressure, icd, limbs, injury)* in Figure 1.5. *(Blood pressure, icd, limbs, injury)* is supported by 4 transactions in Figure 1.5, whereas $injury$ is supported by 1 transaction in Figure 1.3.

To protect transactions, we use COAT [77], which uses *privacy constraints* and *utility constraints*, defined as follows:

Definition 13 (Privacy Constraint)**.** *Let $\mathcal{I}$ be an item domain of a transaction dataset. A privacy constraint p is a non-empty set in $\mathcal{I}$ that is specified as potentially linkable to an individual.*

Definition 14 (Utility Constraint Set)**.** *Let $\mathcal{I}$ be an item domain of a transaction dataset. A utility constraint u is a non-empty set in $\mathcal{I}$ where its sub-sets are possible generalised items.*

Intuitively, a privacy constraint defines a set of items that are need to be protected, and a utility constraint defines how items may be generalised. To map an item from $\mathcal{I}$ to $\tilde{\mathcal{I}}$, the item is replaced by a set of items, which is called a generalised item. In a generalised item, we call the item that is in the original transaction *original item*, and an added item a *non-original item*. For example, T4 in Figure 1.5 shows a generalised item *(blood pressure, icd, limbs, injury)*, where *blood pressure, icd* and *limbs* are non-original items and *injury* is an original item.

Various privacy models have been proposed and they require different privacy constraints to be satisfied by the released data [100, 77, 112, 43]. For the purpose of this

paper, we use a simple, but commonly adopted privacy protection model, based on support count.

Definition 15 (Protected Transactions)**.** *Let $\tilde{\mathcal{D}} = \{\tilde{T}_1, \tilde{T}_2, \ldots, \tilde{T}_n, \}$ be a set of set-generalised transactions, and $p = (I, \sigma_{min})$ be a privacy constraint that requires an itemset I to have a minimum support of σ_{min} in $\tilde{\mathcal{D}}$. $\tilde{\mathcal{D}}$ is protected w.r.t. c if either $\sigma(I, \tilde{\mathcal{D}}) \geq \sigma_{min}$ or $\sigma(I, \tilde{\mathcal{D}}) = 0$.*

Given a set of protected transactions w.r.t. a set of privacy constraints, we are interested to see if any constraint may be "violated" by performing some semantic analysis on the transactions. That is, we are interested to know if some items could be removed from a generalised item, based on their semantic relationships with other items in a transaction, thereby reducing the extent of generalisation and recovering some low frequency itemsets from the published transactions.

For each generalised item, we select appropriate items as a context for attacking. We denote $C_T^{\tilde{i}} = \{i_1, ..., i_w\}$ as a set of items that represent the *context* of transaction T for a generalised item $\tilde{i}$ (when $\tilde{i}$ is obvious, we simply use C_T by omitting $\tilde{i}$ in a distance table).

3.5 Scoring

3.5.1 Constructing Distance Table

In this section, we explain how a distance table may be constructed. As has been discussed, given a generalised item, our approach is to measure the semantic relationship between each item $\hat{i}$ in a generalised item and a selected context item set C, by NGD. As C contains a number of context items, it is not straightforward to apply the measurement described in Section 3.1.2 (i.e. NGD is described as a method to measure the semantic relationship between two items). There are two possible approaches to capture semantic relationships between $\hat{i}$ and C:

- Consider C as one item by using multiple AND operators to connect items in C. However, this approach may not be accurate and may even be unnecessary because $\hat{i}$ is not necessarily related to all items in C.

- A more relevant solution is first to measure the semantic relationship between $\hat{i}$ and each item in C and then combine the results. As we are not sure which context item is the most reliable, we consider all context items to be the same and use an average of all distances as the relationship of an item with a transaction context:

$$d_{C,\hat{i}} = \frac{\sum_{j \in C} NGD(j, \hat{i})}{|C|} \tag{3.3}$$

  where $|C|$ is the number of context items in C. That is, when multiple context items are used, an average score between $\hat{i}$ and its context set C is used as a measure of how likely $\hat{i}$ belongs to the transaction. For example, given
  $\tilde{T} = \langle$*heart disease, (blood pressure, injury), weakness, dizziness*$\rangle$,
  the semantic relationship between *blood pressure* and its context
  $C = \{$*heart disease,weakness*$\}$ (here, we select two closest items as context items) is measured by
  $d_{C,\text{blood pressure}} = \frac{NGD(\text{heart disease,blood pressure})+NGD(\text{weakness,blood pressure})}{2}$. Here, we assume that an adversary knows which items are protected, so that they can select appropriate context items to score relationships. In a later section, we will discuss how this assumption may be relaxed.

One requirement of the set-based generalisation is that generalised items form k-equivalent groups. That is, each generalised item will appear at least k times within the released transactions. This is to ensure that the probability of using generalised items to link an individual to a transaction is no more than $1/k$. Therefore, when attacking a generalised item $\tilde{i} = (\hat{i}_1, \hat{i}_2, \ldots, \hat{i}_s)$, we consider the whole equivalent group together by performing NGD on each occurrence of $\tilde{i}$ in different transactions and record the result in a distance table, as shown in Figure 3.5, where columns are items in the generalised item, and rows are context items selected from each transaction in the

equivalence group to attack the generalised item. Note that while the generalised item $\tilde{i}$ is identical in every transaction within the equivalence group, the context items that are selected and used to attack it need not be the same. In fact, each transaction is different and contexts are likely to be different, thereby allowing the membership of $\hat{i}$ in $\tilde{i}$ to be discriminated in a given transaction.

	$\hat{i}_1$	...	$\hat{i}_s$
C_1	$d_{C_1,\hat{i}_1}$	...	$d_{C_1,\hat{i}_s}$
...	...	...	...
C_k	$d_{C_k,\hat{i}_1}$	...	$d_{C_k,\hat{i}_s}$

Figure 3.5: Distance Table

For example, applying our scoring function to the generalised item *(blood pressure, icd, limbs, injury)* in Figure 1.5, we obtain the distance table in Figure 3.6 (to easily distinguish the distance of original and non-original items, we use bold for distances of original item). This generalised item contains 4 items and forms a 4-equivalent group, therefore the distance table has 4 columns and 4 rows. The largest distance is 2.93 between icd and $gangrene$, suggesting that they are not as related as others are, hence icd is likely to be an item introduced into T3 by the generalisation process, rather than an original item in T3. Note that in this example, we used a single context item to attack the generalised item. In general, any number of context items may be used if they are available.

Note that sets of context C are selected for a specific generalised item. While each transaction may contain more than one (different) generalised item, the selected context need not be similar. For example, given generalised transaction $\langle i_1, (i_2, i_3), i_4, i_5, (i_6, i_7), i_8, i_9\rangle$, the selected context for (i_2, i_3) could be $C = \{i_1, i_4\}$, while context items for (i_6, i_7) could be $C = \{i_5, i_8\}$.

	blood pressure	icd	limbs	injury
$C_1 = \{\text{heart disease}\}$	**0.56**	**0.78**	1.57	2.19
$C_2 = \{\text{anesthesia}\}$	1.75	**0.58**	1.74	1.53
$C_3 = \{\text{gangrene}\}$	2.60	2.93	**1.78**	**1.49**
$C_4 = \{\text{knee}\}$	1.60	1.51	1.89	**1.03**

Figure 3.6: An Example Distance Table

3.5.2 Unknown Generalised Items

Semantic measurement performed in the previous section is based on an assumption that an adversary knows which item is generalised, so that they can identify context items to attack the data. One possibility is that a data owner may not release the data with generalised items clearly marked. For example, a generalised transactions dataset is more likely to be released as shown in Figure 3.7:

TID	Items
1	heart disease, blood pressure, icd, limbs, injury, weakness, dizziness
2	anesthesia, blood pressure, icd, limbs, injury, pain, diabetes
3	gangrene, blood pressure, icd, limbs, injury
4	knee, blood pressure, icd, limbs, injury

Figure 3.7: Anonymised transactions with generalised items that are not clearly marked.

As such, it is not immediately clear which items are in a generalised item and which items are context ones. To differentiate them, we use the two following rules:

- Any item that appears in more than one transaction is considered as an item in a generalised item.
- Any item that appears in only one transaction is considered as a context item.

A limitation is that if a transaction contains more than one generalised item, then they cannot be separated, based on these rules. Therefore, the constructed distance table may be large, making the algorithms even more expensive. However, the rules can be extended in various ways to solve this problem, based on how much information an adversary may know about the anonymisation process. For example, it is reasonable to assume that an adversary knows the parameter k that is used in COAT, or utility and privacy constraints. By knowing k, it can be inferred that an item that appears in less than k transactions is not in a generalised item. Or, by knowing a utility constraint set, an adversary can separate generalised items if a transaction contains multiple generalised items as each generalised item is a possible mapping of one utility constraint.

To focus on the main purpose of this thesis, which is using semantic relationships to attack set-based generalised data, we consider a situation where a data publisher releases a dataset with generalised items clearly marked.

3.5.3 Relationship Among Items In A Generalised Item

There are two different semantic relationships that can be exploited by attackers. One is the relationship between items with a transaction context that we discussed in the previous section. In this section, we discuss the relationship between items in the same generalised item and show why it is not useful for our attack.

Since we use COAT to demonstrate our attack, we also notice that COAT generalises data in a specific logic, in which it replaces an item with a group of items to ensure that it preserves the highest utility in anonymised data. The way an item is chosen for generalisation is specified by a utility constraint and utility loss function (*UL*). A utility constraint specifies all possible items that can be generalised, and UL specifies one item that incurs the lowest utility losses. There are two possible ways that UL measures utility loss: (1) the items with the most similar meaning will incur the least utility loss; (2) the item that requires the least modification on the dataset incurs the

least utility loss. Neither of these selection methods can guarantee protection against semantic attack because they do not take into account the context of the data. This means that although an added item is similar in meaning to the original item, it can still be identified as a non-original item if it does not fit into the transaction context. For this reason, using the relationship between items in a generalised item is not useful.

3.5.4 Semantic Relationships Among Data

Our technique is based on the semantic relationship among items in transactions subject to attack. However, this type of relationship may not exist in some datasets, in some applications; for example, a supermarket store customer's shopping items in a shopping basket dataset, where each transaction records items that a customer has purchased. Such datasets may not have the type of relationship that we consider in our work; for example, a transaction $\langle$*milk, (bread, bacon), cheese, medicine*$\rangle$. It may be inferred that *bread* is the most likely to be an original item since people often buy *bread*, *milk* and *cheese* together. However, they may neither be related by semantic relationship nor have similar meaning, but based on a pattern that may be mined in a shopping basket dataset.

Attacking this type of dataset has been considered as part of knowledge hiding, which we reviewed in Chapter 2. In our work, we consider a dataset where the semantic relationships exist among items.

3.6 Summary

In this chapter, we discussed semantic measurement approaches and explained why NGD is used as a method for measuring semantic relationships in our work. We also introduced our semantic attacking framework, which consists of two components: scoring and elimination. Scoring measures the semantic relationship between items in

a generalised item and context items, and these relationships are scored using NGD and are represented in a distance table, which will then be used by the elimination component to get rid of items which are considered to be non-original.

A limitation of this approach is that the measurement is expensive as NGD uses a search engine to capture the number of Web pages that contain the keywords. Unlike other methods that use local repositories, NGD could incur a long response time, making it difficult to apply in a large dataset. However, in a real application, an adversary can choose to attack a small number of equivalent groups that may contain information about some people that the adversary is interested in. Therefore, this limitation may not affect the real purpose of semantic attack.

Chapter 4

Attacking Methods

In this chapter, we propose two types of elimination strategy. The first type is based on a practical assumption that non-original items have a weaker semantic relationship with a transaction's context than original items have. This assumption is reasonable in cases where items in a transaction create a distinct context, and added items from a different context are more likely to have weaker semantic relationships to the context than other items of the transaction. Following this type of strategy, we propose a Threshold-Based Attack that eliminates items that have NGD higher than a particular threshold.

Beside semantic distance measured by NGD, represented in a distance table, in the second type, we analyse how the presence or absence of an item affects other items within a distance table. Following this type of strategy, we propose a heuristic attack which includes Weight-Based, Grouping-Based and Redistribution-Based Attacks. A weight-based attack exploits the relationship between items, in terms of how eliminating an item may affect the other items. Grouping- and redistribution-based attacks divide items into two clusters: one contains the items that are more likely to be non-original, and another contains items that are more likely to be original. The approach is based on relative distances between items in a distance table.

4.1 Maximum Distance Attack (MDA)

Given k equivalent group, it is easy to see from the mechanism of set-based generalisation that there exists at least one item that does not belong to the original transaction. So a conservative method is to consider the one most likely to be a non-original item and eliminate it from the generalised item. We call this attacking technique Maximum Distance Attack (MDA).

Given a distance table where each item in this table represents the semantic relationship between an item and context items, the higher the value is, the less related the two items are. Therefore, the item that has the highest NGD distance is the one that is considered to be the most likely to be non-original. That is, if $\mathcal{D}$ is a set of items in a distance table, the output of MDA is:

$$\mathcal{D}_e = \mathcal{D} \setminus MAX(\mathcal{D} = \{d_1, ..., d_{|\mathcal{D}|}\}), \text{ if } d_i > 0 \text{ where } 1 \leq i \leq |\mathcal{D}| \tag{4.1}$$

where $MAX(\mathcal{D})$ returns the item with the greatest value in $\mathcal{D}$. Note that NGD may return a negative value. The condition $d_i > 0$ is to ensure that we only eliminate the ones that have semantic distance that is not negative. Applying this method to Figure 3.6, we eliminate icd from T3, which is demonstrated in Figure 4.1.

	blood pressure	icd	limbs	injury
$C_1 = \{heartdisease\}$	**0.56**	**0.78**	1.57	2.19
$C_2 = \{anesthesia\}$	1.75	**0.58**	1.74	1.53
$C_3 = \{gangrene\}$	2.60	-	**1.78**	**1.49**
$C_4 = \{knee\}$	1.60	1.51	1.89	**1.03**

Figure 4.1: An Example MDA

The consequence of this attack gives a resultant set of transactions Figure 4.2. The data is considered as not holding the privacy constraint defined in COAT. This means that

TID	Items
1	heart disease, (***blood pressure***, ***icd***, *limbs, injury*), weakness, dizziness
2	anesthesia, (*blood pressure,* ***icd***, *limbs, injury*), pain, diabetes
3	gangrene, (*blood pressure,* ~~*icd*~~, ***limbs***, ***injury***)
4	knee, (*blood pressure, icd, limbs,* ***injury***)

Figure 4.2: Transaction data after applying MDA

if an adversary knows an individual, who has had an *icd* operation, they can identify an individual among three records in the attacked dataset. However, without additional other knowledge, an adversary still cannot uniquely identify the record of an individual based on this attacked result. Although MDA is not effective, its results are useful, in terms of providing practical proof that using semantic relationship among items is a good approach to identifying non-original items.

4.2 Threshold-based Attack (TBA)

MDA is very conservative in that it does not attempt to remove all possible non-original items. Additionally, the mechanism of set-based generalisation replaces an original item with a set of items, some of which contain non-original items. Given that mechanism, we observe that:

Observation 1. *There is at least one original item in each row and column of a distance table.*

A more aggressive attack could consider all items with a distance above a certain threshold to be non-original. That is, given a parameter δ and a distance table, we perform the following, as long as d is not the last item left in a column or row in $\mathcal{D}$:

$$\mathcal{D}_e = \mathcal{D} \setminus \bigcup_{d_{ij} \in \mathcal{D}, d_{ij} > \delta} d_{ij} \text{ if } E_i^r \leq N^r - 2, E_j^c \leq N^c - 2, \tag{4.2}$$

where d_{ij} is a value in $\mathcal{D}$ at i^{th} row and j^{th} column, E_i^r and E_j^c are the numbers of eliminated items in i^{th} row and j^{th} column, respectively, and N^r, N^c are the number of items in each row and column, respectively. Therefore, the condition to eliminate an item is when there is at least 2 items left in both a row and its corresponding column. This attack requires the specification of a suitable δ by the user. Alternatively, the average distance in $\mathcal{D}$ may be used as δ:

$$\delta = \frac{\sum_{d \in \mathcal{D}} d}{|D|} \textit{ where } \mathcal{D} = \{d_1, ..., d_{|\mathcal{D}|}\} \text{ if } d_i \geq 0, 1 \leq i \leq, |\mathcal{D}| \tag{4.3}$$

where $|\mathcal{D}|$ is the number of items in $\mathcal{D}$, and we only consider items in $\mathcal{D}$ that are not negative as they are not reliable, in terms of deducing semantic relatedness.

	blood pressure	icd	limbs	injury
$C_1 = \{heart\,disease\}$	**0.56**	**0.78**	1.57	-
$C_2 = \{anesthesia\}$	-	**0.58**	-	1.53
$C_3 = \{gangrene\}$	-	-	-	**1.49**
$C_4 = \{knee\}$	1.60	1.51	-	**1.03**

Figure 4.3: Result of TBA

Applying TBA on the distance table given in Figure 3.6 gives the result shown in Figure 4.3, and the transaction data following the attack is shown in Figure 4.4. The released data now contains some unique combinations of items that can be used to identify or narrow down possible records of an individual. For example, knowing someone who has a blood pressure problem, an adversary can infer that the individual is more likely to be associated with transaction T1 or T4 as this item is now only in these two transactions.

The effectiveness of TBA relies on the density of a distance table, which is defined as follows:

Definition 16 (Density)**.** *Given a distance table $\mathcal{D}$, $\mathcal{D}_o \subset \mathcal{D}$ is a set of original items*

TID	Items
1	heart disease, (***blood pressure***, ***icd***, *limbs*, ~~*injury*~~), weakness, dizziness
2	anesthesia, (~~*blood pressure*~~, ***icd***, ~~*limbs*~~, *injury*), pain, diabetes
3	gangrene, (~~*blood pressure*~~, ~~*icd*~~, ~~***limbs***~~, ***injury***)
4	knee, (*blood pressure, icd,* ~~*limbs*~~, ***injury***)

Figure 4.4: Transaction data after applying TBA

in $\mathcal{D}$. The density of $\mathcal{D}$ is defined as:

$$\theta = \frac{|\mathcal{D}_o|}{|\mathcal{D}|}$$

For example, the density of the distance table in Figure 3.6 is $\frac{6}{16} = 0.375$. Intuitively, the lower the density is, the more non-original items there are. In such a case, the average value of distances tends to be greater than original items, therefore, the elimination is less likely to affect original items but effectively eliminate non-original ones. For example, Figure 4.5 shows the distribution of distance values from the distance table given in Figure 3.6, where crosses are non-original items and circles are original items. As can be seen, several non-original items have much greater distance values than the others, making average distance a suitable threshold for eliminating them.

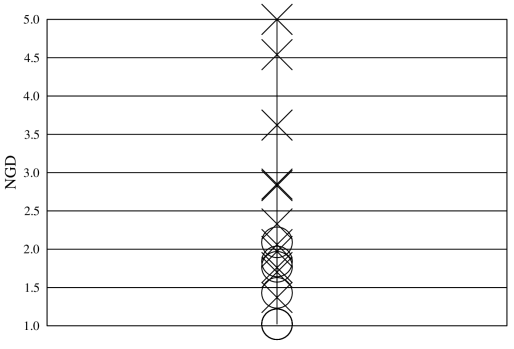


Figure 4.5: Distribution of distance value in Figure 3.6

4.3 Weight-based Attack (WBA)

In a high-density dataset, some of the original items have distances greater than the average distance. These items will be eliminated when the TBA attack is applied. As it is difficult to specify a suitable threshold to avoid wrong eliminations in this case, our approach is to pull the items that are more likely to be original items below the average threshold, by using the information available in the dataset. The aim of this is to make the elimination less dependent on data density and increase the precision in a dataset of high density.

Additionally, as set-based generalisation replaces an item by a set of items, which consists of both non-original and original items, eliminating a non-original item from a set-based generalised item will make the rest of the items in the generalised item more likely to be original. We exploit this property of set-based generalisation and propose an attacking method that takes into account the effect of eliminating items on the rest of the items.

Based on this intuition, we propose a Weighted Distance Attack (WDA), which eliminates items from a distance table in iterations: one item is eliminated in each iteration, then the remaining distances in the table are updated w.r.t. the item eliminated. This continues until no more elimination can be performed.

So initially, we consider that each item has a similar chance of being an original item, without considering semantic relationships. To specify this, we assign weights to items in a distance table. If a row or column has m items left, the weight of each item in this row or column is $1/m$. The higher the weight of an item is, the greater is the chance that the item is original.

Because the effect of an eliminated item is only on items in the same row or column, we consider rows and columns separately and construct row weight and column weight, as shown in Figure 4.6. Figure 4.6 (a) is a row weight table, and the values in the table are weights which are distributed equally among items in the row. That is, the sum of

the weights for the items in a row is equal to 1 (a column weight table in Figure 4.6 (b) is similarly constructed). N^c and N^r are the number of rows and columns, while E_i^r and E_j^c are the number of eliminated items in i^{th} row and j^{th} column, respectively.

	i_1	...	i_{N^c}
C_1	$\frac{1}{N^c-\lvert E_1^r\rvert}$	...	$\frac{1}{N^c-\lvert E_{N^r}^r\rvert}$
...	...	...	...
C_{N^r}	$\frac{1}{N^c-\lvert E_1^r\rvert}$	...	$\frac{1}{N^c-\lvert E_{N^r}^r\rvert}$

(a) Row-weighting Table

	i_1	...	i_{N^c}
C_1	$\frac{1}{N^r-\lvert E_1^c\rvert}$	...	$\frac{1}{N^r-\lvert E_{N^c}^c\rvert}$
...	...	...	...
C_{N^r}	$\frac{1}{N^r-\lvert E_1^c\rvert}$	...	$\frac{1}{N^r-\lvert E_{N^c}^c\rvert}$

(b) Column-weighting Table

Figure 4.6: Weighting Tables

WBA is to use weights to revise the distances recorded in a distance table as follows:

Definition 17 (Weighted Distance)**.** *Let $\mathcal{D}$ be a distance table and $\alpha_{ij} \in \mathcal{D}$ be the distance value at row i and column j in $\mathcal{D}$. The weighted distance α_{ij}^w for α_{ij} is calculated by*

$$\alpha_{ij}^w = \alpha_{ij} \times (1 - \frac{1}{N^r - E_i^r}) \times (1 - \frac{1}{N^c - E_j^c}) \tag{4.4}$$

where N^r and N^c are the number of rows and columns in $\mathcal{D}$, and E_i^r and E_j^c are the number of eliminated items in row i and column j, respectively.

α_{ij} is first revised by the row weights ($\frac{1}{N^r-E_i^r}$) and then by the column weights ($\frac{1}{N^c-E_j^c}$). The more items eliminated from a row (column), the more likely the remaining items in the row (column) will be original, and revision given in Definition 17 reflects that.

The pseudocode of WBA is provided in Algorithm 4.1. The algorithm is a heuristic process, in which the weighted distance table is revised after each elimination, until there is no item satisfying the eliminating criterion.

Algorithm 4.1: *WBA*(D, N^r, N^c)

Input: A distance table D with the number of rows N^r and the number of columns N^c

Output: D with non-original items eliminated

1: $E^c \leftarrow$ initialise

E^c is an array to store the number of eliminations in each column. The initialisation assigns 0 for each element in E^c.

2: $E^r \leftarrow$ initialise

E^r is an array to store the number of eliminations in each row. The initialisation assigns 0 for each element in E^r.

3: $D^w \leftarrow Weighting(D, N^r, N^c, E^r, E^c)$

4: $\delta \leftarrow \frac{\sum_{d \in \mathcal{D}^w} d}{|\mathcal{D}^w|}$

5: $m_{ij} \leftarrow max(D^w)$, if $N^r - E^r_i \geq 2$, $N^c - E^c_j \geq 2$

6: **while** $m_{ij} > \delta$ **do**

7: $D_{ij} \leftarrow \emptyset$

8: $E^r_i \leftarrow E^r_i + 1$

9: $E^c_j \leftarrow E^c_j + 1$

10: $D^w \leftarrow Weighting(D, N^r, N^c, E^r, E^c)$

11: $m_{ij} \leftarrow max(D^w)$, if $N^r - E^r_i \geq 2$, $N^c - E^c_j \geq 2$

12: **return** D

We use E^r and E^c, which are one-dimensional arrays, to store the number of eliminations in rows (E^r) and in columns (E^c). Steps 1 and 2 initialise these arrays by setting 0 for each element. E^r and E^c are updated during the elimination process (from line 6 to 11), so that the algorithm does not need to scan the whole distance table every time when calculating the weighted distance value. Step 3 calculates a weighted distance table D^w by using Algorithm 4.2. Basically, Algorithm 4.2 is a $N^r \times N^c$ loop calculating the weighted distance for each item in D, based on Definition 17. Given a weighted distance table D^w, we want to identify which item in D^w is not related to

a transaction's context by using a constant threshold δ that is calculated on the initial scores in D^w. This is calculated in Step 4, based on the average of all items in D^w. Step 5 gets the item m_{ij} with the greatest distance in D^w as long as m_{ij} is not the last item in row i and column j. The elimination criterion is checked in Step 6 to see if the selected item has a distance greater than δ. If it does, the score at indexes i, j is set to empty at Step 7. The elimination counters E_i^r and E_j^c are then increased, and D^w and m_{ij} are recalculated, based on the new D from Steps 8 to 11. The elimination process is ended when the elimination criterion is not satisfied and the result is returned in Step 12.

Algorithm 4.2 shows how WBA performs weighting on a distance table D, according to Definition 17.

Algorithm 4.2: $Weighting(D, N^r, N^c, E^r, E^c)$

Input: A distance table D, the number of rows N^r and columns N^c, and the number of eliminated items in each row E^r and in each column E^c

Output: A weighted distance table D^w

1: $D^w \leftarrow \emptyset$
2: **for** $i \leftarrow 0$ to N^r **do**
3: **for** $j \leftarrow 0$ to N^c **do**
4: $D^w_{ij} \leftarrow D_{ij} \times (1 - \frac{1}{N^r - E^r_i}) \times (1 - \frac{1}{N^c - E^c_j})$
5: **return** D^w

Consider Figure 3.6 again. To start, we assign weights for each item in the row and column weight tables, as shown in Figure 4.7(a) and Figure 4.7(b). The entries in Figure 3.6 are then revised, using these two weight tables according to Definition 17 to produce Figure 4.7(c).

	blood pressure	icd	limbs	injury
C_1	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
C_2	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
C_3	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
C_4	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$

(a) Row Weights

	blood pressure	icd	limbs	injury
C_1	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
C_2	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
C_3	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
C_4	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$

(b) Column Weights

	blood pressure	icd	limbs	injury
C_1	0.32	0.44	0.88	1.23
C_2	0.98	0.33	0.98	0.86
C_3	1.46	1.65	1.00	0.84
C_4	0.90	0.85	1.06	0.58

(c) Weighted Table

Figure 4.7: Initial Weight Tables and Weighted Distance Table

The elimination of an item from Figure 4.7(c) is then carried out, based on the following conditions: a) the item has the greatest distance in the table; b) the item is not the last one in a row or column; and c) its distance is greater than the average distance in the table. Note that in this case, the average threshold is calculated from the revised table, i.e. $\delta = 0.90$. icd in C3 satisfies these conditions, and hence is eliminated. After icd is removed, the two weight tables are updated and the results are shown in Figure 4.8(a) and Figure 4.8(b). These two tables are then used to revise Figure 3.6 to give Figure 4.8 (c).

	blood pressure	icd	limbs	injury
C_1	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
C_2	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
C_3	$\frac{1}{3}$	-	$\frac{1}{3}$	$\frac{1}{3}$
C_4	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$

(a) Row Weights

	blood pressure	icd	limbs	injury
C_1	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{4}$
C_2	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{4}$
C_3	$\frac{1}{4}$	-	$\frac{1}{4}$	$\frac{1}{4}$
C_4	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{4}$

(b) Column Weights

	blood pressure	icd	limbs	injury
C_1	0.32	0.39	0.88	1.23
C_2	0.98	0.29	0.98	0.86
C_3	1.30	-	0.89	0.75
C_4	0.90	0.76	1.06	0.58

(c) Weighted Table

Figure 4.8: The First Iteration in Weighted attack

The elimination process is carried on until the 4th iteration, at which point there are no more items satisfying the threshold criterion. The result is shown in Figure 4.9. Consequently, the attacked transactions are shown in Figure 4.10.

Although both TBA and WBA use the same criterion to eliminate items (i.e. average

	blood pressure	icd	limbs	injury
C_1	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	-
C_2	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
C_3	-	-	$\frac{1}{2}$	$\frac{1}{2}$
C_4	$\frac{1}{3}$	$\frac{1}{3}$	-	$\frac{1}{3}$

(a) Row Weights

	blood pressure	icd	limbs	injury
C_1	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	-
C_2	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$
C_3	-	-	$\frac{1}{3}$	$\frac{1}{3}$
C_4	$\frac{1}{3}$	$\frac{1}{3}$	-	$\frac{1}{3}$

(b) Column Weights

	blood pressure	icd	limbs	injury
C_1	0.25	0.35	0.70	-
C_2	0.88	0.29	0.87	0.77
C_3	-	-	0.59	0.50
C_4	0.71	0.67	-	0.46

(c) Weighted Table

Figure 4.9: The fourth (final) iteration with the WDA method

TID	Items
1	heart disease, (***blood pressure***, ***icd***, *limbs,* ~~*injury*~~), weakness, dizziness
2	anesthesia, (blood pressure, **icd**, limbs, injury), pain, diabetes
3	gangrene, (~~*blood pressure*~~, ~~*icd*~~, ***limbs***, ***injury***)
4	knee, (*blood pressure, icd,* ~~*limbs*~~, ***injury***)

Figure 4.10: Transaction data after applying the WBA method

threshold), WBA can avoid eliminating original items as the original items can be pulled below the threshold values if the non-original items are correctly eliminated. For example, considering Figure 4.7 (c), *limbs* in T3 has a distance greater than that in T1 and T2, although it is original in T3 due to the context of T3 generating greater distances for all items in the transaction. As the first elimination is *icd*, it makes *limbs*

in T3 more likely to be original and pull down the semantic distance of *limbs* in T3, which is shown in Figure 4.8 (c). Therefore, it is less likely to be eliminated in the later iterations. Because of this mechanism, WBA is a better approach than TBA, in terms of it being less dependent on data density and more effective than TBA when data density is high.

We consider the performance of WBA in the worst case scenario, where WBA eliminates items until there is only one item left in each row and column of a distance table. In this case, the algorithm can perform maximum $N^r \times N^c - MAX(N^r, N^c)$ number of iterations. For instance, the previous example may take a maximum of 12 iterations. In each iteration, the *Weighting* algorithm generates a weighted distance table from a distance table that recalculates each item in the table. This process costs $N^r \times N^c$. Therefore, the overall complexity of WBA is $\mathcal{O}((N^r \times N^c - MAX(N^r, N^c)) \times (N^r \times N^c))$ or $\mathcal{O}((N^r \times N^c)^2)$.

4.4 Grouping-based Attack (GBA)

NGD measures semantic relationship by the number of occurrences of items in WWW, contributed by communities. If an item is less commonly used (a rare term), its NGD with any other items will be higher than expected, even though they are closed related. Because NGD scales from 0 to ∞, it is difficult to specify a range of closely related and non-related items. Those issues concerning NGD make it difficult to compare all items in a distance table, fairly. For example, in considering the distance table in Figure 3.6, distances between $C3$ for all terms are greater than others. One of the reasons for this is that the term (e.g. *gangrene*) is less commonly used in WWW, which makes the distance greater for all terms that it is measured with (e.g. *blood pressure, icd, limbs, injury*). Therefore, considering values in a distance table equally will eliminate most of the items in rows or columns that contain a rare term.

Furthermore, we consider a problem where the NGD of two pairs of terms $NGD(A, B) =$

d_1 and $NGD(C, D) = d_2$, although $d_1 < d_2$, it does not mean that A is related to B more than C is related to D. However, given that $NGD(A, C) = d_3$, if $d_1 < d_3$, it is more "reliable" to consider that A is related to B than to C because we have a "common" base to compare it with (i.e. item A). This implies that the distances in a distance table should not always be compared based on their absolute values.

To address the above problem, we attempt to divide a distance table into *groups*, where each group contains distances that are *comparable*:

Definition 18. *Two distances are comparable when they involve one common item.*

That is, we consider the distance of two pairs of items that are comparable if they have a common item which can be used to compare semantic relationships, relatively.

To apply this approach, we need to specify comparable distances in a distance table. Formally, given a distance table D, as shown in Figure 3.5, and d_{ij} and d_{gh} are two distinct distances in D (i.e. d_{ij} is a distance of the pair of items at i^{th} row and j^{th} column in D), d_{ij} and d_{gh} are comparable distances if $i = g$ or $j = h$. Intuitively, this means that groups of comparable distances are items in the same row or column. For example, given the distance table in Figure 3.6, we have 8 groups

g_1 ={{blood pressure, heart disease}, {blood pressure, anesthesia}, {blood pressure, gangrene}, {blood pressure, knee}}

g_2 ={{icd, heart disease}, {icd, anesthesia}, {icd, gangrene}, {icd, knee}}

g_3 ={{limbs, heart disease}, {limbs, anesthesia}, {limbs, gangrene}, {limbs, knee}}

g_4 ={{injury, heart disease}, {injury, anesthesia}, {injury, gangrene}, {injury, knee}}

g_5 ={{heart disease, blood pressure}, {heart disease, icd}, {heart disease, limbs}, {heart disease, injury}}

g_6 ={{anesthesia, blood pressure}, {anesthesia, icd}, {anesthesia, limbs}, {anesthesia, injury}}

g_7 ={{gangrene, blood pressure}, {gangrene, icd}, {gangrene, limbs}, {gangrene, injury}}

g_8 ={{knee, blood pressure}, {knee, icd}, {knee, limbs}, {knee, injury}}

The way that items are eliminated is similar to that in WBA, where the item with the greatest distance in a group would be eliminated first. However, in GBA, not all distances are comparable, therefore, our strategy is to compare and eliminate items in each group, separately. To effectively specify non-original items in groups, we consider a group having two clusters, one contains lower distance items, which are more likely to be original items, and another contains higher distance items, which are more likely to be non-original items. They are classified by the largest distance between two items, one from each cluster. We call the two clusters *lower cluster* and *higher cluster*, respectively.

Definition 19 (Clusters)**.** *Given a group of distances $D_c = \{d_1, d_2, ..., d_{|D_c|}\}$, where $d_1 \leq d_2 \leq ... \leq d_{|D_c|}$, let d_{max} be the greatest difference between two distances in D_c such that $d_{max} = MAX(d_{i+1} - d_i)$, $0 < i < |D_c| - 1$ and $d_i > 0$. $d_j \in D_c$ is in the lower cluster if $j \leq i$, otherwise it is in the higher cluster.*

Note that this does not imply that the items in the cluster are definitely non-original or original, but they have a greater chance of being so. We also introduce a group's *vulnerability level*, which indicates how likely it is for a group to contain a non-original item.

Definition 20 (Vulnerability Level)**.** *Given a group of comparable distances D_i, the vulnerability level of D_i is specified by*

$$\mathcal{V}_i = \frac{MAX_{D_i=\{d_1,d_2,...,d_{|D_i|}\}}(|d_k - d_{k-1}|)}{MAX_{d \in D_i} d - MIN_{d \in D_i} d} \text{ where } 1 < d_k \leq |D_i| \tag{4.5}$$

where D_i contains distances in an ascending order, and the distance between the clusters or vulnerability level is specified by the largest gap between two neighbouring items. The denominator is the greatest distance in D_i that is to scale the vulnerability level

from 0 to 1, in which the higher the vulnerability level is, the more likely the group is to contain a non-original item. For example, in Figure 4.7 (c), considering a group:

{{heart disease, blood pressure}, {heart disease, icd}, {heart disease, limbs}, {heart disease, injury}}

where {heart disease, blood pressure} and {heart disease, icd} are in the lower cluster, {heart disease, limbs} and {heart disease, injury} are in the higher cluster, and the distance between the two clusters is specified by the distance between {heart disease, icd} and {heart disease, limbs}, which is *0.44*, and the vulnerability level of this group is $\frac{0.44}{1.23-0.32} = 0.48$.

The idea behind using the vulnerability level is that when it is high, the distance between the two clusters is relatively high. This means that the division between related and non-related pairs of items is more obvious. Attacking "obvious" groups first will give us a lower rate of wrong eliminations.

Similar to previous attacks, a criterion is required to determine if an item is likely to be non-original. We call this criterion *Vulnerable Threshold*. A distance table is vulnerable when there is at least one item that can be eliminated based on this criterion, which is defined as follows:

Definition 21 (Vulnerable Threshold)**.** *Given a weighted distance table D^w, a parameter δ, D^w is vulnerable when*

$$\delta < MAX_{D_i \in D^w} \mathcal{V}(D_i)$$

where $MAX_{D_i \in D^w} \mathcal{V}(D_i)$ is the highest vulnerability level of a comparable group D_i in D^w and

$$\delta = \frac{1}{N^r + N^c} \sum_{D_i \in D^w} \mathcal{V}(D_i)$$

where N^r and N^c are the number of rows and columns in D^w, respectively. $N^r + N^c$ is the number of comparable groups in D^w.

Based on that, our attacking strategy is a heuristic process where each iteration eliminates an item in a group that has the highest vulnerability level, then it analyses the effect that the eliminated item may cause to other groups. The process is continued until a distance table is no longer vulnerable. The pseudocode of GBA is provided in Algorithm 4.3. This algorithm is similar to WBA except for the way that we select m_{ij} in Steps 5 and 11. In WBA, m_{ij} is selected as the greatest distance in a distance table, while GBA selects m_{ij}, based on the greatest vulnerability level among groups w.r.t. Formula 4.5. Note that in this algorithm, we use m_{ij} to denote the vulnerability level of a group where i and j are the row and column, respectively, of the greatest distance value of an item in this group, and the item must satisfy the condition $N^r - E^r_i \geq 2$, $N^c - E^c_j \geq 2$, which ensures that there are at least 2 items left in the row and column.

Algorithm 4.3: *GBA(*D, N^r, N^c*)*

Input: A distance table D with the number of rows N^r and the number of columns N^c

Output: D with non-original items eliminated

1: $E^c \leftarrow$ initialise
2: $E^r \leftarrow$ initialise
3: $D^w \leftarrow Weighting(D, N^r, N^c, E^r, E^c)$
4: $\delta \leftarrow \frac{1}{N^r+N^c} \sum_{D_i \in D^w} \mathcal{V}(D_i)$
5: $m_{ij} \leftarrow max_{D_k \in D^w} \mathcal{V}(D_k)$, if $N^r - E^r_i \geq 2$, $N^c - E^c_j \geq 2$
6: **while** $\delta < m_{ij}$ **do**
7: $\quad D_{ij} \leftarrow \emptyset$
8: $\quad E^r_i \leftarrow E^r_i + 1$
9: $\quad E^c_j \leftarrow E^c_j + 1$
10: $\quad D^w \leftarrow Weighting(D, N^r, N^c, E^r, E^c)$
11: $\quad m_{ij} \leftarrow max_{D_k \in D^w} \mathcal{V}(D_k)$, if $N^r - E^r_i \geq 2$, $N^c - E^c_j \geq 2$
12: **return** D

Applying a grouping attack to the weighted distance table in Figure 4.7 gives results

in Figure 4.11, where the group for column *icd* is selected (because it has the highest vulnerability level *0.80* and is higher than the vulnerability threshold, which is *0.45*), so it is eliminated from C3. After eliminating an item, the weight tables are updated in the same way, as in the case of WBA.

	blood pressure	icd	limbs	injury
C_1	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
C_2	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
C_3	$\frac{1}{3}$	-	$\frac{1}{3}$	$\frac{1}{3}$
C_4	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$

(a) Row Weights

	blood pressure	icd	limbs	injury
C_1	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{4}$
C_2	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{4}$
C_3	$\frac{1}{4}$	-	$\frac{1}{4}$	$\frac{1}{4}$
C_4	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{4}$

(b) Column Weights

	blood pressure	icd	limbs	injury
C_1	0.32	0.39	0.88	1.23
C_2	0.98	0.29	0.98	0.86
C_3	1.30	-	0.89	0.75
C_4	0.90	0.76	1.06	0.58

(c) Weighted Table

Figure 4.11: The First Iteration in Grouping attack

Continuing the process, at the fifth iteration (Figure 4.12), the highest vulnerability level is the *blood pressure* column, which is *0.41* - a value lower than the vulnerable threshold. Therefore the attack is terminated and the resultant transaction data is shown in Figure 4.13.

	blood pressure	icd	limbs	injury
C_1	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	-
C_2	-	$\frac{1}{2}$	-	$\frac{1}{2}$
C_3	-	-	$\frac{1}{2}$	$\frac{1}{2}$
C_4	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$

(a) Row Weights

	blood pressure	icd	limbs	injury
C_1	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{3}$	-
C_2	-	$\frac{1}{3}$	-	$\frac{1}{3}$
C_3	-	-	$\frac{1}{3}$	$\frac{1}{3}$
C_4	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$

(b) Column Weights

	blood pressure	icd	limbs	injury
C_1	0.19	0.35	0.70	-
C_2	-	0.19	-	0.51
C_3	-	-	0.59	0.50
C_4	0.60	0.76	0.95	0.52

(c) Weighted Table

Figure 4.12: The Fifth (final) Iteration in Grouping attack

Comparing this result with that produced by WBA, the former had more identified non-original items. Compared with TBA, GBA has made fewer eliminations. As we have analysed in the previous sections, TBA is effective in a low-density dataset and less effective in a high-density dataset, therefore GBA is expected to be a better approach than WBA in general because it is less affected by dataset density and has a better trade-off between the precision of elimination and the number of non-original items that can be identified.

With regard to the methodology of eliminating non-original items, previous methods

TID	Items
1	heart disease, (***blood pressure***, ***icd***, *limbs,* ~~*injury*~~), weakness, dizziness
2	anesthesia, (~~blood pressure~~, **icd**, ~~limbs~~, injury), pain, diabetes
3	gangrene, (~~*blood pressure*~~, ~~*icd*~~, ***limbs***, ***injury***)
4	knee, (*blood pressure, icd, limbs,* ***injury***)

Figure 4.13: Transaction data after applying GBA

consider absolute distances (e.g. a fixed threshold) to specify if an item is non-original or not, whereas GBA uses the relative distance of items. We see that relative distance is a better approach for this measurement to address the problem since items in a distance table cannot always be fairly compared to each other.

Comparing the complexity of GBA with that of TBA, an additional task in GBA is to classify items into clusters in each iteration. This process costs $N^r \times N^c$ because it needs to read each item in a distance table. Therefore, the complexity of GBA is $\mathcal{O}((N^r \times N^c - MAX(N^r, N^c)) \times (N^r \times N^c + N^r \times N^c))$, where the first term is the maximum number of iterations possible and the second term relates to the process in each iteration. Overall, the complexity of GBA is $\mathcal{O}((N^r \times N^c)^2)$, which is similar to that of WBA.

4.5 Redistribution-based Attack (RBA)

With WBA and GBA, once an item is eliminated, we consider the effect of the eliminated item on the rest of the items to be equal. That is, we redistribute the weight of the eliminated item to the rest of the items equally. This makes the rest of the items less likely to be non-original items. In this section, we consider the case where when an item is eliminated, not all items in a distance table are affected equally, but some items become more likely to be original items and some items become more likely to be non-original items.

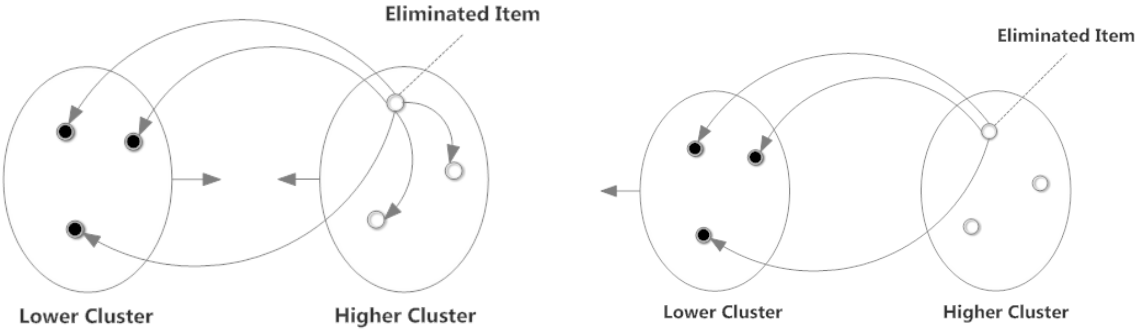
With GBA, after eliminating an item, the weight of any original item is equally divided among the rest of the items in the group. This mechanism literally assumes that the rest of the items in the group are equally likely to be an original item, and more importantly, it scales down the overall distance of all items in the group, which makes it even harder for the algorithm to detect non-original items.

Theorem 1. *Given a comparable group D_i and weighted distances in D_i being distributed equally among items, eliminating items in D_i always decreases the vulnerability level of D_i.*

Proof. The weighted distance calculation in Formula 4.4 shows that when the number of eliminated items in a row or column increases, the weighted distance decreases because the distance value and the number of rows N^r and columns N^c are fixed for each distance table. Because weighted distance is always decreased when there is an elimination, the vulnerability level will also decrease according to Formula 4.5. This means that eliminating items in a comparable group will always decrease the vulnerability level of this group.

The mechanism of always scaling down the vulnerability level of groups is not a problem. However, the weights that are redistributed in those methods are not realistic because some items which are less likely to be original items should receive higher weights than items which are likely to be original items. This difference is demonstrated in Figure 4.14. In Figure 4.14 (a), the weight of an eliminated item is redistributed to all items in the group, causing all distances to be reduced, and therefore the vulnerability level reduces because the distance between the two clusters decreases. While the vulnerability threshold is fixed, this mechanism makes some non-original items even harder to detect as semantic distances of non-original and original items are closer in later iterations. In Figure 4.14 (b), the distribution only affects the lower cluster, and the items in this cluster are harder to be eliminated in later iterations because their distances decrease, while items in the higher cluster are not affected and therefore they retain the same likelihood of being eliminated in later iterations.

We argue that this “redistribution" favours the items in the lower cluster because items in this cluster have a higher chance of being original items. Based on this observation, the weight tables should be calculated in such a way that the redistribution of the weight following the elimination of an item is divided between items in the lower cluster. We call this attack Redistribution-based Attack (RBA).



(a) Redistributing the weight equally to all items in a group

(b) Redistributing the weight equally to items in the lower cluster

Figure 4.14: Different redistributions of weights after eliminating an item

The pseudocode of the method for redistributing the weights, called *RWeighting*, is provided in Algorithm 4.4. The inputs include the distance table, weight tables of columns and rows, and an index of an item at i^{th} row and j^{th} column of the distance table that is eliminated. The aim of this algorithm is to distribute the weight of the eliminated item to other items in the lower cluster.

Algorithm 4.4: $RWeighting(D, W^r, W^c, i, j)$

Input: A distance table D, row and column weight tables W^r, W^c and indexes i and j of the row and column that is affected by the elimination

Output: Updated weight tables W^r and W^c

1: $D^r \leftarrow \pi^r_i(D)$

2: $D^c \leftarrow \pi^c_j(D)$

3: **for** $d_{xy} \in low(D^r)$ **do**

4: $\quad W^r_{xy} \leftarrow W^r_{xy} + \frac{W^r_{ij}}{|low(D^r)|}$

5: **for** $d_{xy} \in low(D^c)$ **do**

6: $\quad W^c_{xy} \leftarrow W^c_{xy} + \frac{W^r_{ij}}{|low(D^c)|}$

7: **return** (W^r,W^c)

We use π^r and π^c to denote projection functions, which project a distance table to a group by row and column index, respectively. For example, $D^r = \pi^r_1(D)$ is the group of items in the first row of the distance table. The low function returns items in the lower cluster of a group following Definition 19. Steps 1 and 2 project the distance table into groups D^r and D^c, which contain the eliminated item at i^{th} row and j^{th} column. So the weights of items in these groups need to be adjusted. Step 3 loops each item d_{xy} (i.e. a distance value at row x and column y) in the lower cluster of D^r. Step 4 adjusts the weight value of d_{xy} in the row weight table at the same row and column index (i.e. row x and column y) by adding $\frac{W^r_{ij}}{|low(D^r)|}$ to the current weight value, where W^r_{ij} is the weight of the eliminated item in the row weight table and $|low(D^r)|$ is the number of items in the lower cluster. As such, the weight is divided equally between each item in the lower cluster. This process is similar for the column weight table from Steps 5 to 6. Step 7 returns both weight tables.

The pseudocode of RBA is provided in Algorithm 4.5. The algorithm takes a distance table as input and generates a distance table as an output, where items that are detected as non-original are eliminated.

Algorithm 4.5: *RBA(*D, N^r, N^c*)*

Input: A distance table D with the number of rows N^r and the number of columns N^c

Output: An attacked result D

1: $D^w \leftarrow Weighting(D, N^r, N^c, \text{empty}, \text{empty})$
2: $\delta \leftarrow \frac{1}{N^r+N^c} \sum_{D_i \in D^w} \mathcal{V}(D_i)$
3: $(W^r, W^c) \leftarrow initW(N^r, N^c)$
4: $m_{ij} \leftarrow MAX_{D_k \in D^w} \mathcal{V}(D_k)$, if $N^r - E^r_i \geq 2$, $N^c - E^c_j \geq 2$
5: **while** $\delta < m_{ij}$ **do**
6: $\quad D_{ij} \leftarrow \emptyset$
7: $\quad (W^r, W^c) \leftarrow RWeighting(D, W^r, W^c, i, j)$
8: $\quad$ **for** $k \leftarrow 0$ to N^r **do**
9: $\quad\quad D^w_{kj} \leftarrow D_{kj} \times (1 - W^r_{kj}) \times (1 - W^c_{kj})$
10: $\quad$ **for** $k \leftarrow 0$ to N^c **do**
11: $\quad\quad D^w_{ik} \leftarrow D_{ik} \times (1 - W^r_{ik}) \times (1 - W^c_{ik})$
12: $\quad m_{ij} \leftarrow MAX_{D_k \in D^w} \mathcal{V}(D_k)$, if $N^r - E^r \geq 2$, $N^c - E^c_j \geq 2$
13: **return** D

The general mechanism of RBA is to find a group in a weighted distance table D^w that has the highest vulnerability level to attack. The attack is designed to eliminate the highest distance item in the identified group. After eliminating an item, the algorithm redistributes weights using Algorithm 4.4. The process is terminated when there is no remaining group that has a vulnerability level higher than threshold δ. The main difference between GBA and RBA is the way we redistribute weights; GBA equally redistributes the weight of the eliminated item to all items in the group, while RBA redistributes the weights of an eliminated item to all items in the lower cluster of a group.

Step 1 generates a weighted distance table using the *Weighting* algorithm. Note that in this algorithm, we do not keep track of the number of eliminated items in rows and

columns, and therefore, we do not initialise E^r and E^c, as we did in GBA. Instead, we assign empty arrays to the *Weighting* algorithm. Step 2 generates a constant δ as a threshold, which is the criterion for attacking a group in D^w. We use $initW$ to denote a function for generating the initial row and column weight tables in Step 3. It divides weight values equally among all items in the table, as shown in Figure 4.6. Step 4 obtains a group that has the highest vulnerable level from D^w w.r.t. Formula 4.5 (see Algorithm 4.3 for a detailed explanation of this step). Step 5 validates if the elimination criterion is satisfied. If so, it proceeds to elimination processing from Steps 6 to 14. Step 6 removes the item from the index and then recalculates new row and column weight tables at Step 7, using Algorithm 4.4. Steps 8 and 9 recalculate weighted distance values for the row that is affected by the eliminated item. The process then makes similar recalculations for the column, in Steps 10 to 11. Step 12 obtains the highest vulnerable value for the next validation, and the elimination process is terminated when m_{ij} is below the threshold and the result is returned in Step 13.

Consider again the example we used to illustrate GBA. Applying RBA to the distance table in Figure 4.7 gives the same result in the first iteration, in which the group of items in the column of *icd* is selected for attacking, and *icd* in *C3* is eliminated as it has the highest distance value in the group. Next, the weight of *icd* in *C3* is redistributed to the items in the *icd* column and *C3* row. Specifically, at row *C3* in Figure 4.15 (a), the weight is distributed to *limbs* and *injury*, while *blood pressure* in *C3*, on the other hand, does not receive any additional weight because it is in the higher cluster. In column *icd* of Figure 4.15 (b), all items receive weight because they are all in the lower cluster, with the only item in the higher cluster being at row *C3*, which is eliminated. A new weight distance table is generated in Figure 4.15 (c).

	blood pressure	icd	limbs	injury
C_1	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
C_2	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
C_3	$\frac{1}{4}$	-	$\frac{3}{8}$	$\frac{3}{8}$
C_4	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$

(a) Row Weights

	blood pressure	icd	limbs	injury
C_1	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{4}$
C_2	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{4}$
C_3	$\frac{1}{4}$	-	$\frac{1}{4}$	$\frac{1}{4}$
C_4	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{4}$

(b) Column Weights

	blood pressure	icd	limbs	injury
C_1	0.32	0.39	0.88	1.23
C_2	0.98	0.29	0.98	0.86
C_3	1.46	-	0.83	0.70
C_4	0.90	0.76	1.06	0.58

(c) Weighted Table

Figure 4.15: The First Iteration in Redistributing attack

Continuing the same process in the next iteration, we have the result in Figure 4.16.

	blood pressure	icd	limbs	injury
C_1	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
C_2	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
C_3	-	-	$\frac{1}{2}$	$\frac{1}{2}$
C_4	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$

(a) Row Weights

	blood pressure	icd	limbs	injury
C_1	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{4}$
C_2	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{4}$
C_3	-	-	$\frac{1}{4}$	$\frac{1}{4}$
C_4	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{4}$

(b) Column Weights

	blood pressure	icd	limbs	injury
C_1	0.21	0.39	0.88	1.23
C_2	0.98	0.29	0.98	0.86
C_3	-	-	0.67	0.56
C_4	0.90	0.76	1.06	0.58

(c) Weighted Table

Figure 4.16: The Second Iteration in Redistributing attack

After ten iterations, there is no group that has a vulnerability level higher than the vulnerable threshold and the attack is therefore terminated.

	blood pressure	icd	limbs	injury
C_1	$\frac{1}{2}$	$\frac{1}{2}$	-	-
C_2	-	1.0	-	-
C_3	-	-	$\frac{1}{2}$	$\frac{1}{2}$
C_4	-	-	-	1.0

(a) Row Weights

	blood pressure	icd	limbs	injury
C_1	1.0	$\frac{1}{2}$	-	-
C_2	-	$\frac{1}{2}$	-	-
C_3	-	-	1.0	$\frac{1}{2}$
C_4	-	-	-	$\frac{1}{2}$

(b) Column Weights

	blood pressure	icd	limbs	injury
C_1	0.0	0.2	-	-
C_2	-	0.0	-	-
C_3	-	-	0.0	0.37
C_4	-	-	-	0.0

(c) Weighted Table

Figure 4.17: The Tenth Iteration in Redistributing attack

The transactions following the attack are shown in Figure 4.18, where many unique combinations of items exist, which can be used to link to individuals, representing real potential to break anonymity.

TID	Items
1	heart disease, (***blood pressure***, ***icd***, ~~*limbs*~~, ~~*injury*~~), weakness, dizziness
2	anesthesia, (~~blood pressure~~, **icd**, ~~limbs~~, ~~injury~~), pain, diabetes
3	gangrene, (~~*blood pressure*~~, ~~*icd*~~, ***limbs***, ***injury***)
4	knee, (~~blood pressure~~, ~~icd~~, ~~limbs~~, **injury**)

Figure 4.18: Transaction data after applying RBA

In WBA, the mechanism is mainly to pull down the distances of some items to avoid eliminating original items wrongly, in the case of high-density data. However, the method applies to all the items in a generalised item as it does not differentiate between items which should or should not be eliminated. In GBA, we classify items into clusters to indicate how likely items are to be original or non-original items although non-original items are difficult to detect because the distances are getting closer in the later iterations, due to weights being redistributed equally among all items. RBA combines the advantages of both methods to allow it to detect non-original items more effectively and avoid eliminating original items in a high-density dataset.

The main difference between RBA and GBA algorithms is in terms of the weighting algorithm (i.e. GBA uses a *Weighting* algorithm 4.2 and RBA uses an *RWeighting* algorithm 4.4). As these two weighting algorithms have a similar complexity $\mathcal{O}(N^r \times N^c)$ because they generate weighted values for each item in a distance table, the overall complexity of RBA is similar to that of GBA, which is $\mathcal{O}((N^r \times N^c)^2)$.

4.6 Summary

In this chapter, we proposed various methods to attack transactions by exploiting the relationship between items, established in Chapter 3, to eliminate non-original items:

- Maximum Distance Attack (MDA) is developed based on the observation that there is always at least one item that is a non-original item in a distance table. Therefore, the strategy is to find the item that is the most likely to be that one, based on semantic distances. The method eliminates the item that has the greatest semantic distance. This approach is not effective in terms of its conservative nature, so that many non-original items can still be left in the transactions. However, as MDA only attempts to eliminate one item, it tends to get that one right, hence it is practically useful in showing that using semantic relationships is a relevant approach to eliminate a non-original item.

- Threshold-based Attack (TBA) is developed to eliminate any item which has semantic distance greater than the average distance of a distance table. Using average distance as a threshold is effective in a low-density dataset, which contains more non-original items than original ones. In such a case, the average distance is above most original items, therefore, eliminating items above it will avoid wrong elimination. However, the approach depends heavily on the data's density to be effective.

- Weight-based Attack (WBA) eliminates non-original items based on the observation that when an item is eliminated, the rest of the items in the row and column are more likely to be original items. To achieve this, we set weights for items in a distance table, and weights are redistributed to items in rows and columns when an item is eliminated. The main role of weights is to pull some items, which are likely to be original, below the threshold through iterations of eliminations. As such, the items in the generalised item, where one is eliminated, become more likely to be original. As their distances are pulled down following the elimination, the remaining items become harder to eliminate in the later iterations.

- Group-based Attack (GBA) and Redistribution-based Attack (RBA) solved two important problems. Firstly, they use the relative distance between items for identifying non-original items as using a threshold can be problematic, given the difficulty in determining a suitable one. Secondly, the RBA method combines the approaches of WBA and GBA. As a result, RBA can be used to improve overall results of the attack, in terms of removing more non-original items, thereby depending less on data density.

In the next chapter, we will report on our experimental results and analyse the performance and efficiency of our proposed algorithms.

Chapter 5

Experiments and Results

In this chapter, we evaluate our attacking methods, described in Chapter 4. We compare the performance and effectiveness of different methods, in terms of the number of eliminated non-original items and the precision of eliminations.

The chapter begins with a discussion on the datasets used in our experiments: AOL[1], I2B2[2] and GoArticle[3]. Each dataset has unique properties to ensure that our algorithms are evaluated in a range of conditions. We then discuss the methodology used to compare the effectiveness of our algorithms. Finally, we analyse the experiment results, using recall and precision measures.

5.1 Dataset Preparation and Experiment Setup

Our experiments are conducted on three real datasets: AOL (queries from a search engine), I2B2 (medical and health-care free text dataset) and GoArticle (manually constructed dataset of general free text articles). Table 5.1 summarises and compares the following properties of the datasets:

[1]The AOL dataset contains 20M search queries from 650k users.

[2]Deidentified clinical records used in this research were provided by the i2b2 National Center for Biomedical Computing funded by U54LM008748 and were originally prepared for the Shared Tasks for Challenges in NLP for Clinical Data organized by Dr. Ozlem Uzuner, i2b2 and SUNY.

[3]GoArticles.com is a dual-curation layered resource with millions of quality articles and a membership base exceeding several hundred thousand authors.

- *Data's density* is the ratio between non-original and original items in a generalised item. These density levels are specified based on Definition 16. As the density level can affect the effectiveness of our algorithms, we use both low- and high-density datasets to evaluate our methods in these experiments.

- *Type* describes the format of the original datasets. Some datasets are not transaction data. In the data preparation step, we will discuss how we transform them into transactions.

- *Length of transaction* is measured in terms of the number of terms extracted from the original dataset to form a transaction.

- *Quality* of data indicates if data contains many typographical errors and abbreviations. The more of these contained in a dataset, the harder it is for terms to be extracted, and they can also affect semantic distance measurement because Google may not understand the terms.

- *Domain:* We evaluate our algorithm in both single and multiple domains, and expect datasets that are from multiple domains to be easier to attack because an item from one domain is more distinguishable from that from another domain.

- Finally, *Semantic* is an important property because we mainly base attacks on this property of a dataset. The selected datasets are ensured to have this property, so that the attack results reflect our hypothesis that we proposed in Chapter 1.

Properties	AOL	I2B2	GoArticle
Origin	From AOL released dataset	From I2B2 released dataset	Manually collected dataset
Average density (defined in Definition 16)	0.22	0.28	0.55
Average transaction's length	23	216	104
Type	Transactions	Free text	Free text
Quality	There are typographical errors and abbreviations	There are typographical errors and abbreviations	There is almost no typographical error and has less abbreviations
Domain	Multiple	Medical & healthcare	Multiple
Semantic	Search queries from a user's session are often about a specific topic. Therefore, items extracted from these queries are often related, which are considered as semantically related.	Each document of these datasets is about a particular topic. Therefore, extracted terms are considered as semantically related as they are used to discuss a topic in the document.	

Table 5.1: Dataset properties

Additionally, we do not distinguish between negative and positive sentences from text in the original data. We only focus on items to anonymise and then attempt to de-anonymise them. In some semantic-measuring algorithms, it matters whether singular or plural terms are measured. However, it does not affect the semantic measurement when using NGD because Google automatically searches for both cases, and we do not consider them.

5.1.1 Datasets

I2B2

The medical and health-care domain is one of the most relevant application domains for privacy protection. We choose the I2B2 dataset which is in this domain. I2B2 contains 630 free text documents which are about clinical data and patient discharge reports. Each document is de-identified (i.e. identifiers are removed) and contains disease and symptom information.

The main goal of our experiments is to anonymise transactions by set-based generalisation and then attack it by means of our semantic attack. As the data is in free text format, we take the following steps to prepare and transform the data into transactions:

- We focus on extracting the main part of documents and ignore other parts such as headers and footers of the report which repeatedly appear in all documents such as the ones in Figure 5.1. Therefore, in the first step, we manually filter out those parts before extracting the content.
- A Part-Of-Speech Tagger (POS Tagger) is a software system that reads text in some language and assigns parts of speech to each word (and other tokens) such as noun, verb, and adjective. To focus on extracting nouns and noun phrases, we tag terms in the document in the second step. To achieve this, we use Stan-

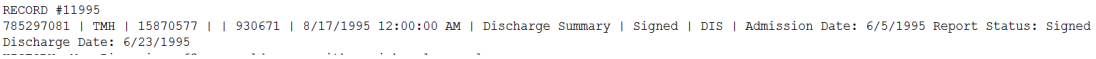
ford POS tagger which is open-source and provided by The Stanford Natural Language Processing Group from Stanford University[4].

- We only select nouns and noun phrases by following patterns that are tagged in the second step: Noun + Noun, Adjective + Noun or Noun. At the end of the third step, for each document, we have extracted items and put them into a transaction.

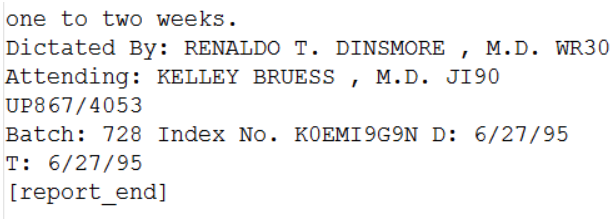
- Finally, we remove any item that satisfies the following conditions: (1) the item is a stop word; (2) the item is duplicated (we check singular and plural forms of a term); (3) a noun already appears in another noun phrase (e.g. in Figure 5.2 (b) "history" is removed because there is another term "long history" that has also been extracted) because a noun phrase is considered to be more meaningful than a noun, in many cases.

The documents in the dataset contain many typographical errors and abbreviations, which can affect our results as Google may not recognise such terms, and therefore it cannot return the correct number of indices for them. To avoid this problem, we do not remove those typographical errors in extracted transactions as it is a time-consuming task on a large dataset, but we do not attack generalised items that contain typographical errors (we have a few hundred generalised items to attack and therefore, it is easier to manually control). However, we still use abbreviations because many common abbreviations can be understood by Google Search Engine. Figure 5.2 (a) shows an example of the dataset after manually filtering out unnecessary information and Figure 5.2 (b) shows the extracted transaction.

[4]http://nlp.stanford.edu/software/tagger.shtml



(a) Headers of a document need to be filtered out



(b) Footers of a document need to be filtered out

Figure 5.1: Example of filtering part in I2B2 dataset

HISTORY: Ms. Pizzo is a 63-year-old woman with peripheral vascular disease who recently underwent revision of her left superior femoral artery anterior tibial bypass graft , who now presents with a cool , ischemic left foot. Mrs. Denman is a 63-year-old , insulin-dependent diabetic with a long history of peripheral vascular disease as well as multiple surgical procedures. She underwent a right transmetatarsal amputation in 1990 and subsequently underwent a right femorla distal saphenous vein bypass graft in 1991 which was later revised in 1992. She seems to be doing well with the left side until July of this year , at which time she underwent a left superficial femoral artery to anterior tibial artery bypass using non-reversed basilic vein harvested from the right arm. She , however , had a large great toe ulcer , possibly attributed to hammertoe , which subsequently underwent a left great toe amputation performed on the 21 of October . After this time , she was discharged to the Nut Hospital in Amore Pu , where she was making progress in physical therapy and rehabilitation. On the day prior to admission , she was exercising with 4 pound weights on her legs with the physical therapist when she described a cool sensation in her foot. She reported that her foot had been blue , and there were no Dopplerable pulses. Color later returned. The absence of pulses persisted over the course of the night , after which point she was referred back to Largrine Medical Center for evaluation. She denies any significant pain or any other complications.

(a) A sample of I2B2 text

ms. pizzo, 63-year-old woman, vascular disease, revision, femoral artery, tibial bypass, graft, left foot, mrs. denman, long history, surgical procedure, transmetatarsal, amputation, right femorla, saphenous vein, by pass graft, left side, july, time, tibial artery, basilic vein, right arm, great toe, ulcer, hammertoe, october, nut hospital, amore pu, progress, physical therapy, rehabilitation, day, admission, pound, weight, leg, physical theparist, cool sensation, dopplerable pulse, color, absence, course, night, point, largrine medical, center, evaluation, significant pain, other complication, medical history, insulin-dependent diabete, coronary artery, hypertension, cataract, mrsa, toe wound, september, surgical history, debridement, toe amputation, largrine medical center.

(b) A sample of I2B2 extracted transaction

Figure 5.2: An example of I2B2 original text and extracted transaction

AOL

The AOL dataset contains 20M search queries from 650k users. Each record has user_id, timestamp of the search, the query which has one or more keywords, the url that is clicked and the clicked rank. Although AOL is already in a transaction form, we reformat the data and remove unnecessary information. Figure 5.3 (a) shows the data in the original dataset and Figure 5.3 (b) shows transactions that we can obtain after following the preparation steps:

- We first need to divide the dataset into transactions by user sessions. This means that any query that is posed by a use_id (denoted as AnonID in Figure 5.3 (a)) will be put into one transaction. We also remove user_id, timestamp and clicked url from the dataset. A keyword may be searched multiple times, but we only count it once.
- We then tag items and only use nouns and noun phrases as we did in the I2B2 dataset. However, it is difficult to do this for the AOL dataset because search queries are often short and they do not follow proper grammar. As we saw from our experiments, the NGD of an item that contains more than two words is often imprecise, so we consider a query as an item if the query contains two or fewer words, otherwise, we use the POS tagger to tag words and extract items based on patterns: Noun+Noun, Adjective+Noun or Noun.
- Similarly to the I2B2 dataset, we also apply several rules to avoid using duplicated items and remove items which are stop words (see the rules used in preparing the I2B2 dataset).

A major reason for using AOL dataset is because the application domain of I2B2 is about the medical domain only. In other applications, a dataset could be drawn from multiple domains. In the AOL dataset, search queries are about various topics therefore when generalising items, keywords from different domains may be grouped together[5].

[5]This would help enhance privacy protection

1	AnonID	Query	QueryTime	ItemRank	ClickURL
8303	13508	good wine	3/1/2006 21:29	7	http://www.wines.com
8304	13508	accent marks	3/1/2006 22:04	1	http://faculty.weber.edu
8305	13508	accent marks	3/1/2006 22:04	2	http://fog.ccsf.cc.ca.us
8306	13508	accent marks	3/1/2006 22:04	4	http://users.ipfw.edu
8307	13508	accent marks	3/1/2006 22:04	3	http://www.starr.net
8308	13508	accent marks	3/1/2006 22:04	10	http://en.wikipedia.org
8309	13508	body mass index	3/2/2006 20:34	2	http://www.halls.md
8310	13508	por que te vas	3/2/2006 21:04	1	http://members.fortunecity.es
8311	13508	por que te vas	3/2/2006 21:04	8	http://lyricsplayground.com
8312	13508	sudden weight loss	3/3/2006 19:21	1	http://menshealth.about.com
8313	13508	sudden weight loss	3/3/2006 19:21	3	http://www.ivillage.co.uk
8314	13508	not enough sleep	3/3/2006 19:26	4	http://search400.techtarget.com
8315	13508	not enough sleep	3/3/2006 19:26	3	http://www.webmd.com
8316	13508	cancer.org	3/3/2006 19:39	1	http://www.cancer.org
8317	13508	chemotherapy and fertility	3/3/2006 19:44	3	http://www.chemocare.com
8318	13508	chemotherapy and fertility	3/3/2006 19:44	6	http://www.ncbi.nlm.nih.gov
8319	13508	chemotherapy and fertility	3/3/2006 19:44	1	http://www.cancernews.com
8320	13508	chemotherapy and fertility	3/3/2006 19:44	4	http://www.cancerhelp.org.uk
8321	13508	chemotherapy and fertility	3/3/2006 19:44	2	http://www.cancernews.com
8322	13508	chemotherapy and fertility	3/3/2006 19:44	7	http://www.netdoctor.co.uk
8323	13508	chemotherapy and fertility	3/3/2006 19:44	2	http://www.cancernews.com
8324	13508	methotrexate	3/3/2006 19:45	1	http://www.medicinenet.com
8325	13508	when you are in love	3/3/2006 20:02		
8326	13508	when you are in love	3/3/2006 20:03		
8327	13508	eid al fitr	3/4/2006 11:20		
8328	13508	eid al fitr 2003	3/4/2006 11:21	3	http://www.student.virginia.edu
8329	13508	eid al fitr 2003	3/4/2006 11:21	9	http://www.hilal-sighting.com

(a) A sample of AOL's raw data

good wine, accent mark, body mass, index, por que, te va, sudden weight, loss, cancer.org, chemotherapy, fertility, methotrexate, love, eid al, fitr, al fitr, crush

(b) A sample of AOL extracted transaction

Figure 5.3: An example of AOL's raw data and extracted transaction

This may make added terms easier to be detected. Therefore, we expect this result to be better for AOL than for I2B2.

GoArticle

AOL and I2B2 datasets have a similar property: the generalised items in both tend to have a low density. In the previous chapters, we have explained that the density of data can affect an attack result. Therefore, we construct another dataset which has a higher density to evaluate our algorithms. GoArticle is a dataset that we manually

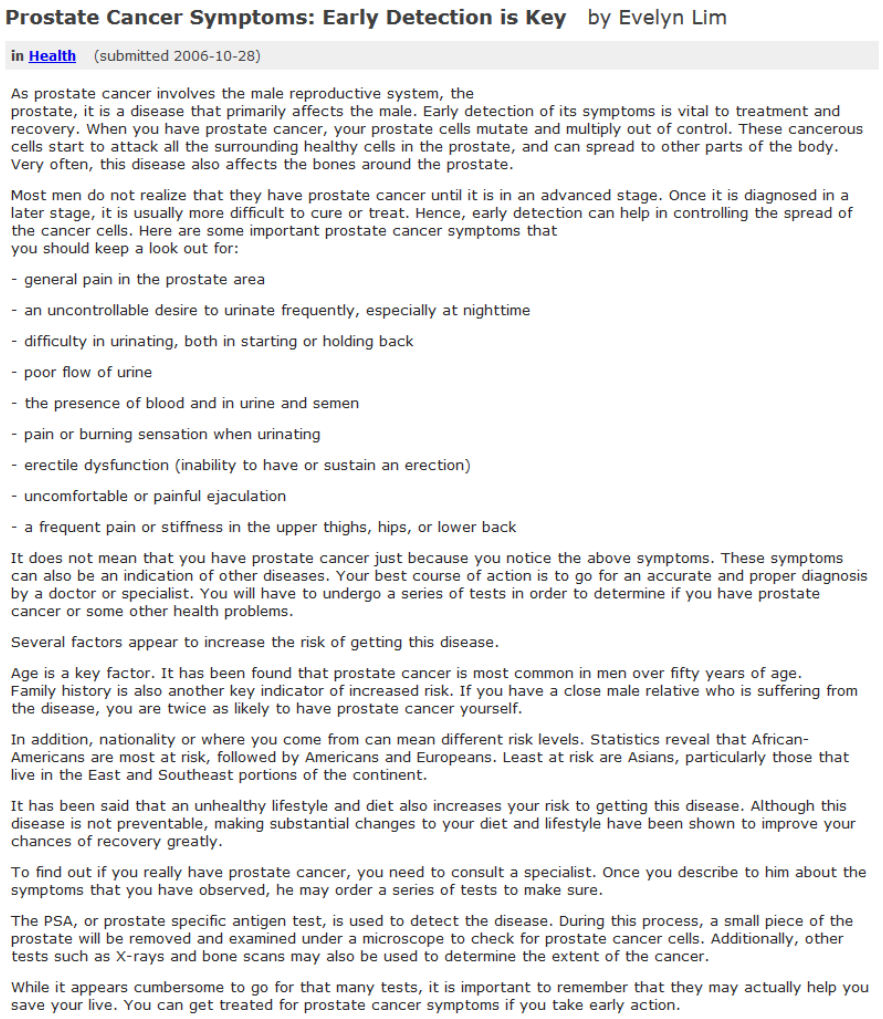
constructed to evaluate our algorithm with dense transactions. The dataset is collected from GoArticles.com on some specific topics, and we manually chose articles which share many common keywords, and hence, in generalisation, fewer non-original items are added into transactions.

GoArticle contains free text which is similar to the I2B2 dataset. Therefore, the process to prepare and extract free text into transactions is similar to what we did for I2B2 (see I2B2's preparation steps for more details). Figure 5.4 (a) shows a sample raw document and Figure 5.4 (b) shows an extracted transaction.

5.1.2 Experiment Setup

In this section, we give details of how we anonymise transactions by using set-based generalisation. More specifically, we use COAT to anonymise the data with the following inputs and constraints:

- Privacy constraints are required input. Each constraint is a set of items that need to be protected. This ensures that if an adversary has knowledge about an individual which are items in a privacy constraint, they still cannot link the individual to a record in the released transactions. In real applications, privacy constraints could be manually constructed by a data owner. In our experiments, we randomly pick 3-6 items from the transactions to form a privacy constraint. For efficiency, these constraints are constructed as follows. We first randomly select a set of 3-item constraints. We then add one more random item to each 3-item constraint to make a set of 4-item constraints. We then do the same for 5-item and 6-item constraints. The reason for constructing our different sized constraints this way is to minimise the effort required to measure item relationships: measuring NGD involving 4-items constraints can re-use the result of measuring NGD of 3-item constraints. This resulted in the number of generalised items for the experimental data as shown in Table 5.2.



Prostate Cancer Symptoms: Early Detection is Key by Evelyn Lim

in Health (submitted 2006-10-28)

As prostate cancer involves the male reproductive system, the prostate, it is a disease that primarily affects the male. Early detection of its symptoms is vital to treatment and recovery. When you have prostate cancer, your prostate cells mutate and multiply out of control. These cancerous cells start to attack all the surrounding healthy cells in the prostate, and can spread to other parts of the body. Very often, this disease also affects the bones around the prostate.

Most men do not realize that they have prostate cancer until it is in an advanced stage. Once it is diagnosed in a later stage, it is usually more difficult to cure or treat. Hence, early detection can help in controlling the spread of the cancer cells. Here are some important prostate cancer symptoms that you should keep a look out for:

- general pain in the prostate area
- an uncontrollable desire to urinate frequently, especially at nighttime
- difficulty in urinating, both in starting or holding back
- poor flow of urine
- the presence of blood and in urine and semen
- pain or burning sensation when urinating
- erectile dysfunction (inability to have or sustain an erection)
- uncomfortable or painful ejaculation
- a frequent pain or stiffness in the upper thighs, hips, or lower back

It does not mean that you have prostate cancer just because you notice the above symptoms. These symptoms can also be an indication of other diseases. Your best course of action is to go for an accurate and proper diagnosis by a doctor or specialist. You will have to undergo a series of tests in order to determine if you have prostate cancer or some other health problems.

Several factors appear to increase the risk of getting this disease.

Age is a key factor. It has been found that prostate cancer is most common in men over fifty years of age. Family history is also another key indicator of increased risk. If you have a close male relative who is suffering from the disease, you are twice as likely to have prostate cancer yourself.

In addition, nationality or where you come from can mean different risk levels. Statistics reveal that African-Americans are most at risk, followed by Americans and Europeans. Least at risk are Asians, particularly those that live in the East and Southeast portions of the continent.

It has been said that an unhealthy lifestyle and diet also increases your risk to getting this disease. Although this disease is not preventable, making substantial changes to your diet and lifestyle have been shown to improve your chances of recovery greatly.

To find out if you really have prostate cancer, you need to consult a specialist. Once you describe to him about the symptoms that you have observed, he may order a series of tests to make sure.

The PSA, or prostate specific antigen test, is used to detect the disease. During this process, a small piece of the prostate will be removed and examined under a microscope to check for prostate cancer cells. Additionally, other tests such as X-rays and bone scans may also be used to determine the extent of the cancer.

While it appears cumbersome to go for that many tests, it is important to remember that they may actually help you save your live. You can get treated for prostate cancer symptoms if you take early action.

(a) A sample of GoArticle's raw text

prostate cancer, reproductive system, disease, male, detection, symptom, treatment, recovery, prostate cell, healthy cell, part, body, bone, man, advanced stage, later stage, early detection, spread, cancer cell, important prostate, cancer symptom, general pain, prostate area, uncontrollable desire, difficulty, poor flow, urine, presence, blood, seman, burning sensation, erectile dysfunction, inability, erection, painful ejaculation, frequent pain, stiffness, upper thigh, hip, indication, course, action, proper diagnosis, doctor, specialist, series, test, order, health, problem, several factor, risk, key factor, year, family history, key indicator, close male, relative, addition, nationality, different risk, level, statistic, african-american, european, least, asian, southeast portion, continent, unhealthy lifestyle, diet, substantial change, chance, psa, specific antigen, process, small piece, microscope, test, x-ray, bone scan, extent

(b) A sample of GoArticle extracted transaction

Figure 5.4: An example of GoArticle's raw text and extracted transaction

- COAT uses an input k to specify how much protection is required to satisfy privacy constraints. That is, given a parameter k, COAT ensures that any subset of items in each privacy constraint appears at least k times. Therefore, the higher the k, the more non-original items may need to be added into transactions. In our experiments, we evaluate with k from 3 to 6 to see how our algorithm performs in low- and high-protection levels.

- Utility constraints specify how to replace an item in transactions. The basic idea is that items in the same utility constraint should have similar meaning, so that when used together, the generalised item is still meaningful. In our experiment, for simplicity, we use all items in datasets as one utility constraint. More specific utility constraints may be specified, but substantial domain knowledge is required. Also, from the privacy protection point of view, it is useful to have a more "diverse" utility constraints, e.g. in the case of AOL dataset.

For example, given a set of privacy and utility constraints constructed as we have described above and $k = 4$, COAT transforms transaction in Figure 5.4 (b) to a generalised transaction in Figure 5.5. Note that we have only shown one transaction in this example, and a generalised item (e.g. (*prostate cancer, unstable angina, depression, coronary artery*) should also occur in at least other 3 transactions given $k = 4$. For a generalised item, e.g. (*important prostate, cough, osteomyelitis, chest pain*), we use the two closest non-generalised items to it as its context items, e.g. {*cancer cell, cancer symptom*}. This will then allow a distance table to be constructed and de-anonymisation to be performed.

Table 5.2 summarises the statistical information of the datasets we used in our experiments. Extracted transaction is the number of transactions that are constructed from the original data (e.g. Free text). Generalised Items is the number of generalised items to be attacked. This number is relatively small compared with the real number of generalised items in the dataset due to the expensive process of our experiments. For example, in AOL with 127 generalised items and $k = 4$, there are about 2k items (i.e.

(prostate cancer,unstable angina,depression, coronary artery), reproductive system, disease, detection, symptom, treatment, (recovery,vomiting,fever,dry gangrene), prostate cell, healthy cell, part, body, bone, advanced stage, later stage, early detection, spread, cancer cell, *(important prostate,cough,osteomyelitis,chest pain)*, cancer symptom, general pain, prostate area, uncontrollable desire, difficulty, poor flow, urine, presence, *(blood, liver biopsy,chest pain,ophthalmology)*, seman, (burning sensation,syncope,gastric,sleep), erectile dysfunction, (inability,motion,diet,unstable angina), erection, *(icd placement,fusion,myocardial infarction,painful ejaculation)*, frequent pain, stiffness, upper thigh, hip, indication, course, action, proper diagnosis, doctor, specialist, series, test, order, health, problem, several factor, risk, key factor, year, family history, key indicator, close male, relative, addition, nationality, different risk, level, statistic, african-american, european, least, (asian,man,male,women), southeast portion, continent, unhealthy lifestyle, substantial change, chance, psa, specific antigen, process, small piece, microscope, test, x-ray, bone scan, extent

Figure 5.5: An example of anonymised Transaction in our experiment

127×16 because the data is sparse, each distance table may contain up to 16 items) whose semantic relationships need to be measured. However, it may not be necessary to attack all possible generalised items in a dataset, as revealing some original items may be enough to re-identify some individuals.

Dataset	Extracted transaction	Generalised items
AOL	758	127
I2B2	643	112
HC	263	45

Table 5.2: Number of attacked items in our experiments

5.2 Evaluation Methods

We compare our proposed methods to a baseline method which performs a *Random Attack* on generalised items. The baseline method essentially assumes that an adversary has no information other than the released dataset, and they can only randomly guess whether an item in a generalised item is an original one or not. The process is that in each distance table, an adversary is assumed to have x attempts to eliminate items, where x is drawn from a uniform distribution. On each elimination attempt, we also perform a random decision regarding whether or not to eliminate an item.

We use *precision* (p) and *recall* (r) to measure how well our methods can detect non-original items correctly:

$$r = \frac{\text{correct eliminations}}{\text{all non-original items}} \qquad p = \frac{\text{correct eliminations}}{\text{all eliminations}}$$

We use F-score to measure the overall effectiveness of our methods:

$$F_1 = 2 \times \frac{p \times r}{p + r}$$

which provides a better aggregation of recall and precision than a simple arithmetic average.

5.3 Results and Discussions

This section evaluates various properties of our algorithms on three datasets. For each dataset, we evaluate how precisely the algorithm can perform, in terms of correctly eliminating non-original items and the number of non-original items that can be eliminated. We then consider the overall effectiveness of each algorithm and how one compares with another. While some of our algorithms use the average distance as a criterion (i.e. Threshold) to eliminate items, we also evaluate how these algorithms perform under manually specified thresholds. We then evaluate how our measurement

is affected in different ways by choosing affected context items. Finally, we compare the time efficiency of the algorithms.

5.3.1 AOL

Figure 5.6 shows the results obtained from testing our methods on the AOL dataset. In this figure, we study the effectiveness of our algorithms, in terms of how accurately they eliminate non-original items from anonymised data. k is a privacy constraint that anonymised data must satisfy. Increasing k typically requires adding more items into a generalised item. In terms of precision, increasing k has two possible effects on our algorithms: (1) if non-original items (added items) are not related to the transaction's context, our algorithms are more effective because the ratio of the number of wrongly eliminated items to the total number of eliminated items will be higher; (2) if non-original items are related to a transaction's context, it is harder to attack. However, because COAT does not consider the transaction context when adding items, case (1) is more likely to happen in our experiments. This effect turned out to be significant w.r.t. the AOL dataset because its context is more diverse compared with other datasets.

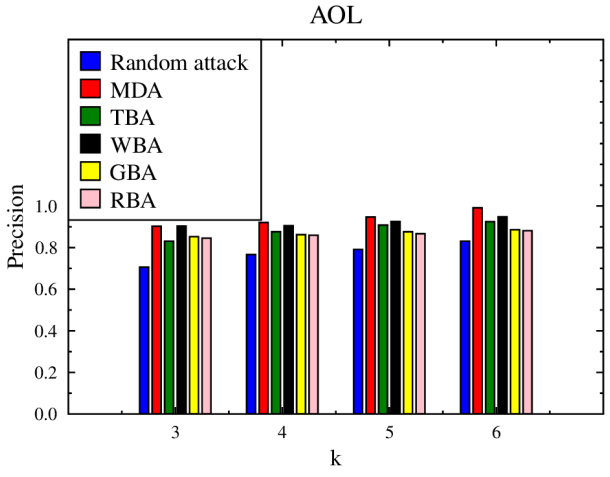


Figure 5.6: Comparing precisions on the AOL dataset

It is easy to see that the overall precisions of the algorithms are higher than 80%, which means that only a small number of items are wrongly eliminated. Compared with other datasets, which we will analyse in the later sections, the precision of the AOL dataset is better because of two main reasons: (1) AOL has low density, therefore, expectedly,

the chance of eliminating the wrong item is low, even in a random attack (i.e. Random attack has 70% precision); (2) AOL has distinct contexts, therefore adding items from other contexts is relatively easier to identify.

Figure 5.7 shows the effectiveness of the algorithms, in terms of the number of non-original items that can be eliminated. The more items that are correctly eliminated, the higher the chance an adversary has of uniquely identifying an individual. Therefore, the recall is important to justify whether data is secure or not.

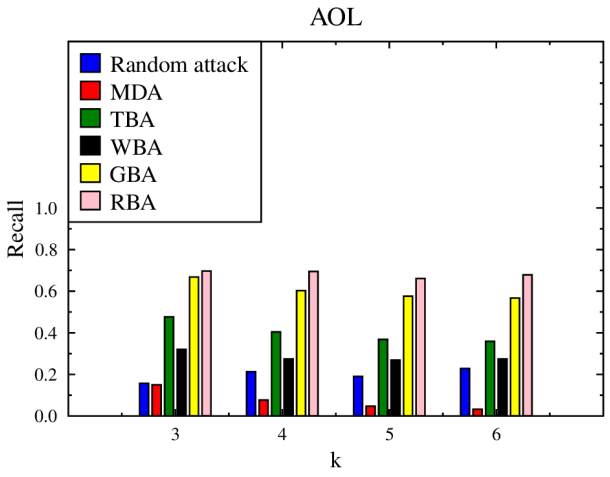


Figure 5.7: Comparing recalls in AOL dataset

Although the levels of precision are very close to each other, their recalls are distinguishable. Specifically, GBA and RBA outperform other algorithms in which about 70% of non-original items can be eliminated. MDA gives a low recall because the algorithm only eliminates the most likely non-original item. This also explains why MDA has high precision in Figure 5.6. Therefore, MDA is only adept at demonstrating our idea of using the semantic relationship to attack the data, but it is less effective, in terms of de-anonymising the data. Because Random Attack eliminates items by a probability drawn from a uniform distribution, both precision and recall are based on the dataset's density. As the dataset has low density and Random Attack does not attempt to eliminate all non-original items, its recall is also low.

WBA and TBA have high precision in Figure 5.6, however, it is shown in Figure 5.7 that the methods are less effective because they can only eliminate a small number of non-original items, compared with GBA and RBA. The main difference in GBA and RBA is that these methods divide items into comparable groups, therefore, comparing the semantic distance among items in a group is more precise. Since some first eliminations affect the choice of later items for elimination, the grouping method helps the algorithm detect correct items for elimination in early eliminations, therefore the algorithm can detect more non-original items for eliminating in the overall result.

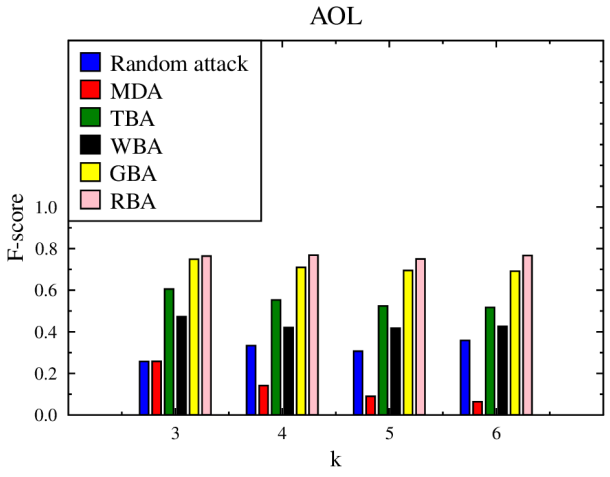


Figure 5.8: Comparing overall effectiveness in AOL dataset

Figure 5.8 shows the overall result of our algorithms in the AOL dataset. In comparing our algorithms with those of Random Attack, the implication is that the semantic relationship among items is a good clue to use in identifying non-original items as Random Attack does not consider semantic relationships. The low score results from MDA and Random Attack are not useful or reliable, in terms of de-anonymising or re-identifying individuals because the attacked data may not contain unique combinations of items and therefore, it is difficult to uniquely identify individuals, or the data has many original items eliminated, and therefore, some combinations may link to non-existent or wrong individuals.

5.3.2 I2B2

There are two important properties that we want to evaluate with our datasets: data density and domains (i.e. multiple or single domain). I2B2 and AOL have a similar density, however, I2B2 focuses on the medical domain only, while AOL is about multiple domains. Our experiments in this section are to evaluate how the domain of a dataset affects the effectiveness of our algorithms.

Figure 5.9 shows the results of the same algorithms and parameters as those used for AOL on the I2B2 dataset. It is easy to see that the overall precision is lower than for the AOL dataset. That is because when the data (e.g. AOL) has multiple domains, items from different domains may be grouped in one generalised item. Therefore, an item that is not relevant to a particular domain will have a greater semantic distance and is easier to eliminate.

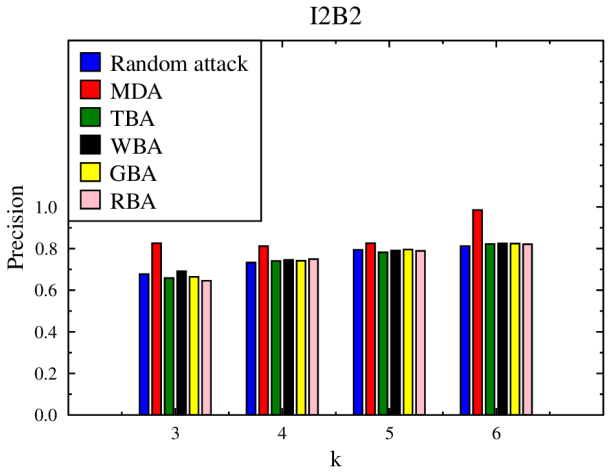


Figure 5.9: Comparing precisions with the I2B2 dataset

However, this property does not affect MDA because this method eliminates one most likely non-original item, therefore the chance of eliminating a wrong one (i.e. an original item) is still very low in both datasets. As a result, the precision of MDA is still the highest in the I2B2 dataset.

Figure 5.10 compares the number of eliminated items made in our algorithms on the I2B2 dataset. It is noticed that our algorithms have a lower recall for I2B2 than for AOL, while Random Attack is consistent in both datasets. That is because Random Attack is not based on semantic measurement, unlike other methods. Because semantic distances in I2B2 are less distinguishable than those in AOL due to the single domain in the I2B2 dataset, it is more difficult to identify non-original items, hence the recalls are lower.

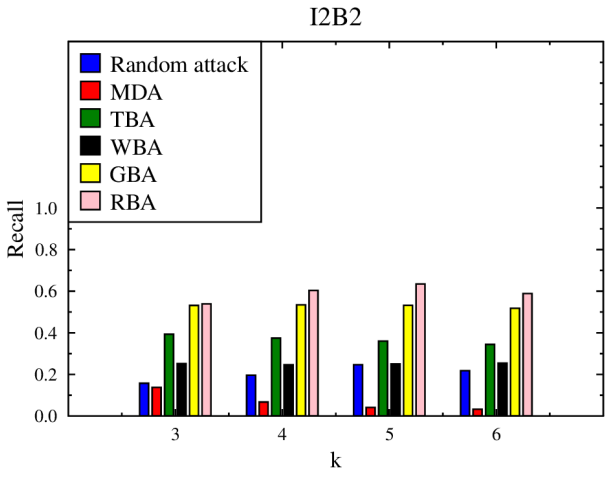


Figure 5.10: Comparing recalls in I2B2 dataset

The overall effectiveness of the algorithms is shown in Figure 5.11. Compared with the AOL dataset, this result shows that the algorithms are more effective if the dataset contains various topics. With a lower k, where fewer non-original items are added, TBA is close to GBA and RBA, and outperforms WBA. With a higher k, more non-original items are added, and GBA and RBA perform better because they can detect more non-original items (demonstrated in Figure 5.10), although TBA has the similar result. WBA has the lowest score among our algorithms (except for MDA) because the method eliminates very few items due to the mechanism of distributing weights that scales down all distances, following the elimination of items.

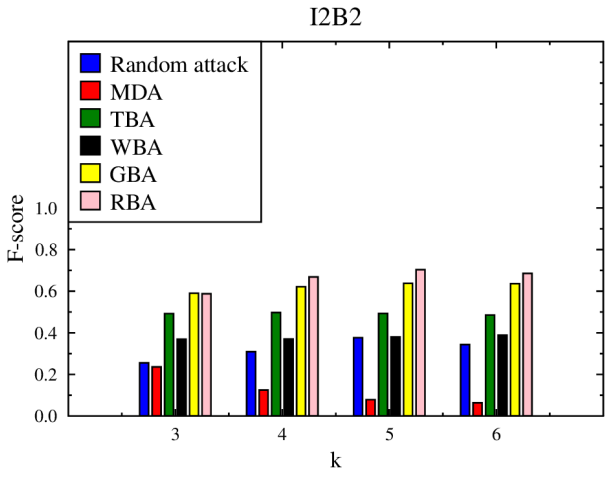


Figure 5.11: Comparing overall effectiveness in I2B2 dataset

5.3.3 GoArticle

In this section, we focus on evaluating how data density affects our algorithms. We use the GoArticle dataset which contains items from multiple domains and has density higher than that of AOL and I2B2. Density in I2B2 and AOL is around 0.2-0.4 while most of the generalised items in GoArticle have a density of 0.4-0.7. Based on our analysis in the previous chapters, when density is high, the dataset may contain more original items than non-original items, making the attack harder. Therefore, it is expected that our algorithm can detect fewer non-original items in the GoArticle datase than AOL and I2B2.

Figure 5.12 shows the precision of the algorithms performance on the GoArticle dataset. Because the weighting mechanism in WBA pulls distances below the average distance, this method has fewer eliminations than others and the precision is the highest.

Observing precision in Figure 5.12 and recall in Figure 5.13, the precision of TBA is close to that of RBA and GBA. However, it eliminates fewer non-original items than RBA and GBA. This indicates that TBA may eliminate more original items. This result is expected because TBA uses the average threshold to eliminate items, and when most

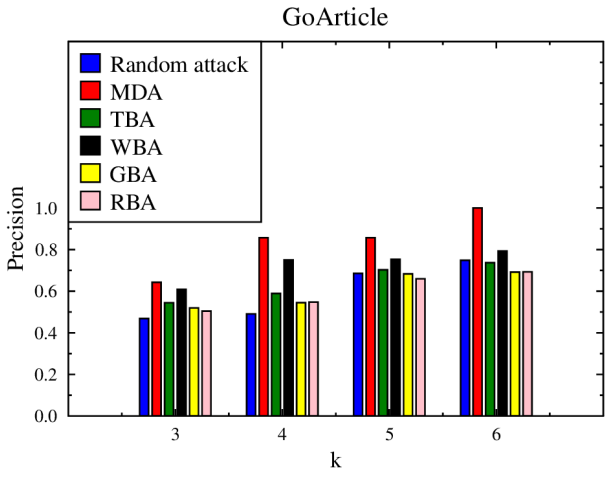


Figure 5.12: Comparing precisions in GoArticle dataset with sparsity 0.4-0.5

of the items are original, this average distance will be below many original items.

GBA and RBA also have low precision, like TBA, however, the main reason is because these methods attempt to eliminate all possible non-original items (they have high recall in Figure 5.13) and therefore, some wrong elimination may occur as the data has high density.

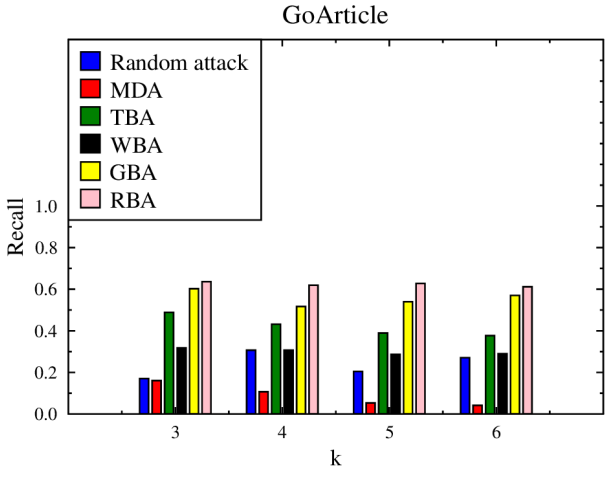


Figure 5.13: Comparing recalls in GoArticle dataset

In AOL and I2B2, the precision and recall of Random Attack are not significantly changed because the data has similar density as this method eliminates items in a random manner, and its overall results are only based on the density of a dataset. Indeed, in a GoArticle dataset, precision and recall of Random Attack are significantly changed as its precision is lower and its recall is slightly higher, due to the data containing few non-original items.

The overall result on GoArticle is shown in Figure 5.14. It turns out that the high-density dataset has a slightly higher impact on the result of our algorithms than using a single domain dataset has.

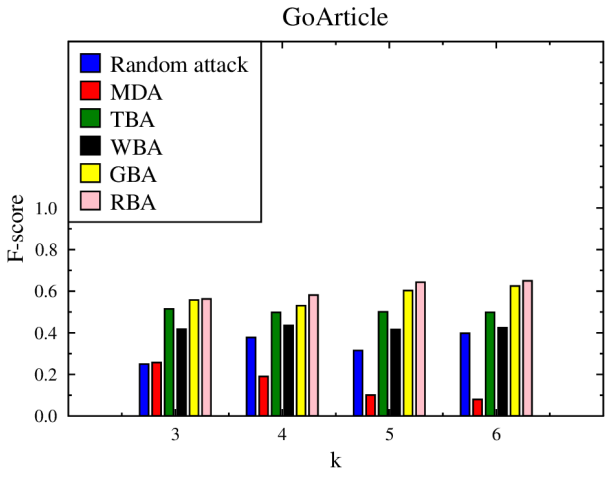


Figure 5.14: Comparing overall effectiveness in GoArticle dataset

5.3.4 Effectiveness of Thresholds

The WBA did not perform as well as the TBA, in terms of the overall F-score in our experiments. This is a surprise, but we believe that this is mainly due to the characteristics of the datasets used in the experiments. For AOL and I2B2 datasets, we observed that a relatively large number of items were added into generalised items, because the data was high dimensional and sparse. This resulted in the NGD scores for the original

items being mostly below the average threshold. With the TBA, this gave a very good recall (and F-score), as all items above the threshold were removed. The WBA, on the other hand, is more conservative. Anytime an item is removed, it makes the rest more likely to be original. Consequently, it eliminates fewer, and has a lower recall and a higher precision. To verify this, we undertook further experiments to vary the thresholds used in elimination. The result is shown in Figure 5.15, where the *threshold ratio* varies in WBA and TBA.

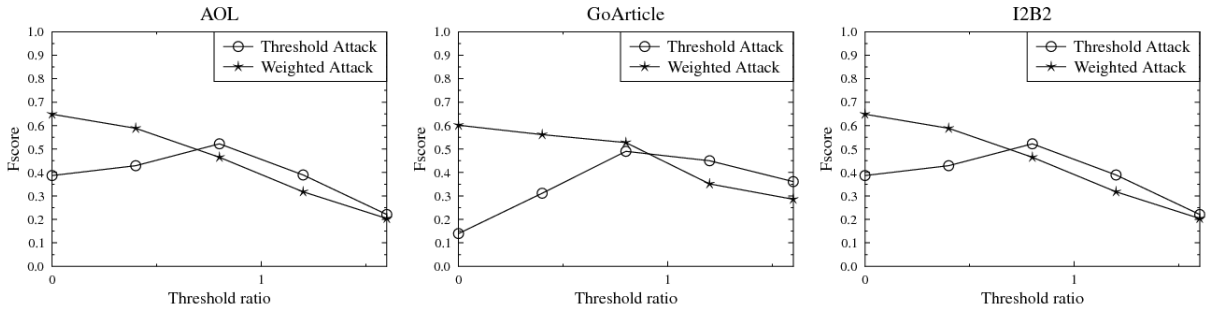


Figure 5.15: Comparison of Effect of Threshold

As can be seen, when thresholds are set very low (i.e. anticipating that most of the items are non-original), the WBA performed better. This is because, as the thresholds lowered, more original items will have NGD scores that are above the threshold. They will therefore be wrongly removed by the TBA, significantly reducing precision and the F-score. The WBA, on the other hand, is able to use the "enlarged" range to remove more non-original items, while maintaining relatively good precision due to its iterative process of elimination. This results in a better overall F-score. When the thresholds have increased to a point where they place most of the original scores below it, the threshold method works better. Again, how to find an appropriate threshold needs to be investigated further.

5.3.5 Effect of Data Density

As the density of a dataset is one of the important properties that can affect the results of our algorithms, this section studies the variation of the algorithms' performance with different density levels. To set up this experiment, we select subsets of documents in previous datasets to have average density levels from 0.1 to 0.7.

In Figure 5.16, the algorithms' precision is low when the density level is high. That is because, when increasing the density level, less non-original items are added into generalised transactions, and as our algorithms try to eliminate all possible non-original items, their precision is decreased as a result of wrong eliminations.

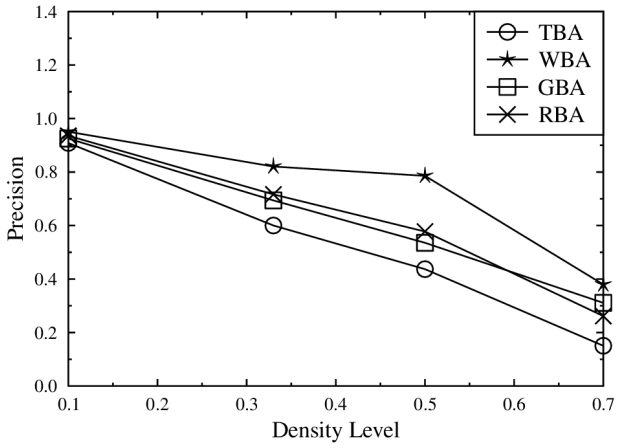


Figure 5.16: Comparing Precision in Different Sparsity Levels

The precisions of the algorithms are easier to distinguish when increasing the density. WBA's precision was often higher than others because that method eliminates fewer non-original items. TBA's precision decreases faster than other methods as the elimination's criterion of TBA is only based on an average distance, and therefore, when most of the items are original items, the average distance is below that of many original items, making the precision low due to many wrong eliminations. The precision of GBA and RBA is slightly better than in the TBA method. However, overall, GBA

and RBA are better, in terms of the method of detecting more non-original items than others, as shown in Figure 5.17.

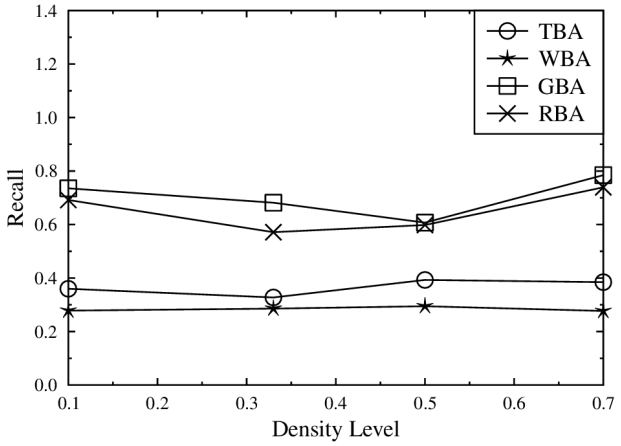


Figure 5.17: Comparing Recall in Different Sparsity Levels

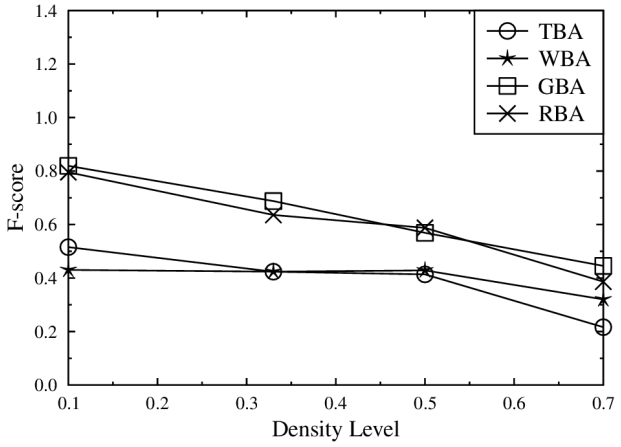


Figure 5.18: Comparing F-score in Different Sparsity Levels

Figure 5.17 shows the recall of our algorithms. As we use average distance and average vulnerability level as criteria for eliminating items in our algorithm, the values also scale according to distance values in a distance table. Therefore, recalls of the algorithms are not significantly changed with increasing density.

5.3.6 Time Efficiency

Figure 5.19 shows the time efficiency of our algorithms. As the complexity of our algorithm is dependent on the size of the distance tables, the aim is to evaluate the performance when varying distance table size. We use $N^r \times N^c$ to denote the size of a distance table, where N^r is the number of rows and N^c is the number of columns. Timing is the time needed to attack a dataset that is shown in Table 5.2.

Our methods were implemented in Java which has a garbage collection mechanism. To ensure fair comparisons, we run each experiment 10 times and the timings reported in Figure 5.19 is the average of 10 runs.

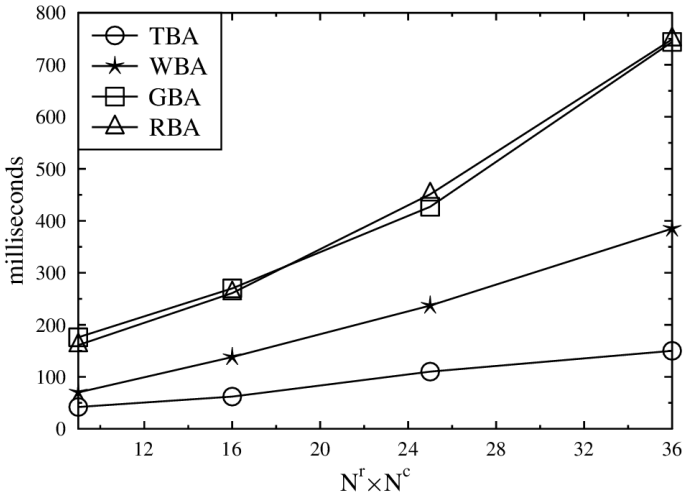


Figure 5.19: Performance of attacking algorithms

In Chapter 4, we showed that complexity is $\mathcal{O}(N^r \times N^c)$ for TBA and $\mathcal{O}((N^r \times N^c)^2)$ for WBA, GBA and RBA. Indeed, in our experiments, TBA timing grows linearly, while WBA, GBA and RBA are grow exponentially. Times for GBA and RBA are often higher than those of WBA although they have similar complexity. This is because RBA and GBA cluster items need more time for this processing.

We do not include time for scoring (i.e. NGD) in this experiment as the process is very

expensive and run remotely on Google servers. Therefore, it is mostly dependent on Internet speed and external searching algorithms. In our experiment, it took more than 24 hours to run all these experiments (not counting the time when Google blocks the IP address because of mass requests to the server, in a short period).

5.4 Summary

In this chapter, we evaluated our algorithms, using three datasets that have distinct properties. The AOL dataset contains items that are from multiple domains: the I2B2 contains items that are from a single domain; and GoArticle is a dataset that has a higher density level than others. Based on these properties, we studied how many non-original items can be eliminated (i.e. measured by recall) and the preciseness of the algorithms (i.e. measured by precision).

This is significant from the privacy protection point of view. Any eliminated item is a risk for privacy, as it may reduce the "cover" for the original data. That is, it may reduce a dataset to containing combinations of items that occur less than k times. Although we did not actually carry out a link attack on the de-anonymised transactions in this study, our observation during the experiments showed a very high proportion of individuals could be at risk of identification, especially with the AOL dataset.

As a conclusion, MDA gives a very high precision in all experiments. These results show that using semantic relationships is reliable for eliminating non-original items. TBA is a method that is effective at eliminating non-original items in a low density dataset. Although the method may have some wrong eliminations, the attack result can still practically be used to re-identify individuals in a real application as many non-original items are eliminated. WBA often has high precision even in a high density dataset, however, this method does not eliminate as many non-original items as TBA does, making the overall effectiveness of WBA less than that of TBA. GBA and RBA have a similar precision with other methods in most of the experiments, however, be-

cause these two methods often have more non-original items eliminated from datasets, they are considered to be more effective.

Chapter 6

Conclusion

This chapter summarises our contributions and discusses the possible future research directions, based on our current work.

6.1 Research summary

Publishing data that contains personal and sensitive information is in great demand by many organisations. Therefore, privacy is an important issue and needs to be addressed. To protect privacy, one common method that is broadly used is set-based generalisation, because of its flexibility to anonymise data and that it can produce low information-loss in anonymised data [77]. In this thesis, we have studied an important property of the method, which is to determine how secure released data is, when it is protected by set-based generalisation.

In Chapter 1, we analysed the privacy issue of set-based generalisation and saw that this method can be used to protect privacy of data, however, it is based on the impractical assumption that protecting items are considered as contextless or even meaningless literals. Our hypothesis for attacking is that: set-based generalisation may not provide adequate protection for transaction data and an adversary can infer original items based on the semantic relationships that exist among the items in transactions.

In Chapter 2, we studied some common privacy models to see how privacy is protected and how it can be broken in different ways. Chapter 3 gives an overview of our

attacking framework, which contains two main components:

- **Scoring** is used to establish the semantic relationship between items and a transaction's context. As there are different ways to address this type of relationship, in our work, we use Normalised Google Distance to measure how likely it is that two items belong to the same context. Based on this measurement, we construct the distance table for a generalised item to represent the semantic relationship between each item and each transaction's context.

- **Elimination** takes a distance table as an input and attempts to eliminate non-original items from it. The result of this component produces a generalised item that has some non-original items eliminated.

In Chapter 4, we have developed five elimination methods. Maximum Distance Attack (MDA) attempts to eliminate only one item which has the greatest item distance, from a distance table. As a result, this method achieves very high precision, which indicates that most of the items eliminated by this approach are non-original items. However, in a real application, this method is not appropriate for use to attack privacy as the attacked data still contains many non-original items. With Threshold-based Attack (TBA), we attempt to eliminate all possible non-original items, in which any item that has semantic distance greater than a threshold is eliminated. TBA is developed based on the assumption that when a dataset contains more non-original items than original items, the average distance of all items in a distance table will be greater than most of the original items. Therefore, using an average threshold can eliminate non-original items whilst avoiding the elimination of original items. However, TBA depends on the density of data, and therefore its attack result in a high density dataset may not be very good, in that many original items may be eliminated.

We also exploit other properties of transactions and generalised items to improve the effectiveness of our attacks. Specifically, with the Weight-based Attack (WBA), when an item is eliminated, intuitively it suggests that the rest of the items in a generalised

item are more likely to be original items. To model this relationship, WBA assigns weights for items in a distance table and eliminates items in iterations. After eliminating an item in an iteration, this method then distributes its weight to other items. Furthermore, we see that comparing distance with an absolute value (threshold value) may be difficult because there are cases where the threshold is wrongly estimated. To solve this problem, Grouping-based Attack (GBA) and Redistribution-based Attack (RBA) attempt to classify items into two clusters: a cluster that is more likely to contain non-original items, and a cluster that is more likely to contain original items, based on the relative distance of the items.

Semantic Attack addresses a significant limitation of previous privacy models as it does not rely on an adversary's background knowledge about individuals. In our experiments, this method can eliminate more than 85% of non-original items, with a precision higher than 70%. With a high ratio of removing non-original items, it shows that the current privacy protection methods are not safe, in terms of protecting data privacy. In practice, most of the dataset may contain semantic relationships between items, the assumption that items are contextless or even meaningless literals being impractical.

6.2 Future work

Although this thesis has shown that our attacking methods can achieve good results, the work can be extended in various directions:

- **Improve Semantic Measurement.** Since this research focuses on developing elimination methods, investigating a more powerful and accurate scoring approach for Semantic Attack is still an avenue open to pursuit. The performance of NGD is low and therefore, this method is difficult to apply, in the case of a large-scale dataset.

  The context of a transaction is essential to measure the semantic relationship between an item and a transaction. In our work, we used close items as context

items. When there are multiple context items available, we measure semantic relationships with each context item and use the average result as a semantic distance between an item and a transaction. The limitation of NGD prevents us from specifying the semantic relationship of an item with multiple contexts simultaneously. Addressing this problem may improve both performance and the measuring of the semantic relationship between an item and a transaction, more precisely.

- **Data Types and Privacy Models.** This thesis concentrates on set-based generalisation which is commonly used to protect privacy in data publishing. Because of this, we mainly deal with transaction data, which often contains items that are semantically related to each other. In the case of this particular use, it is easy to illustrate the usefulness of semantic attack. However, the attacking approach has the potential to be applied to other types of data and privacy model.

- **Attacking High Density Data.** Our methods worked relatively better on datasets that have low density. However, there are also a number of real datasets that have high density, especially, when more specific utility constraints are used to group data items together. Attacking a high density dataset needs to be addressed in the future as our analysis and experiments in Figure 5.16 showed that with a density level of 0.5 or above, more than half of TBA eliminations are incorrect.

  Attacking high density data is difficult as when transactions have high density, they contain many similar items, and therefore, they may have similar context as well. Since our approach is based on the difference between attacking contexts, attacking high density data may require a different approach.

- **Measuring Confidence.** Semantic attack uses approximately semantic relationships among items. This means that when some items satisfy the elimination criterion and are eliminated from transactions, it would be useful to establish the confidence of how likely the eliminated decision is a correct one. Therefore, it is useful to study the "confidence" of the attack [115].

References

[1] N. Adam and J. Wortmann. Security-Control Methods for Statistical Databases: A Comparative Study. *ACM Computing Surveys (CSUR)*, 21(4):515–556, 1989.

[2] N.R. Adam and J.C. Wortmann. Security-control methods for statistical databases: A comparative study. *ACM Computing Surveys (CSUR)*, 21(4):515–556, 1989.

[3] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD international conference on Management of data*, pages 207–216, 1993.

[4] R. Agrawal and R. Srikant. Privacy-preserving data mining. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pages 439–450, 2000.

[5] B. Anandan and C. Clifton. Significance of term relationships on anonymization. In *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, pages 253–256, 2011.

[6] M. Askari, R. Safavi-Naini, and K. Barker. An information theoretic privacy and utility measure for data sanitization mechanisms. In *Proceedings of the second ACM conference on Data and Application Security and Privacy*, page 283, 2012.

[7] M.J. Atallah, E. Bertino, A.K. Elmagarmid, M. Ibrahim, and V.S. Verykios. Disclosure Limitation of Sensitive Rules. In *Workshop on Knowledge and Data Exchange*, 1999.

[8] M. Atzori, F. Bonchi, F. Giannotti, and D. Pedreschi. Blocking anonymity threats raised by frequent itemset mining. In *Fifth IEEE International Conference on Data Mining*, pages 561–564, 2005.

[9] B. Barak, K. Chaudhuri, C. Dwork, S. Kale, F. McSherry, and K. Talwar. Privacy, accuracy, and consistency too: a holistic solution to contingency table release. In *Proceedings of the twenty-sixth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 273–282, 2007.

[10] C.H. Bennett, P. Gacs, M. Li, P.M.B. Vitanyi, and W.H. Zurek. Information distance. *Transactions on Information Theory*, 44(4):1407–1423, 1998.

[11] J.J. Berman. Concept-match medical data scrubbing. How pathology text can be used in research. In *Archives of pathology and laboratory medicine*, pages 680–686, 2003.

[12] G. Bouma. Normalized (Pointwise) Mutual Information in Collocation Extraction. In *Proceedings of the Biennial GSCL Conference*, 2007.

[13] R. Brand. Microdata Protection through Noise Addition. In *Inference Control in Statistical Databases*, pages 97–116, 2002.

[14] J. Brickell and V. Shmatikov. The Cost of Privacy: Destruction of Data-Mining Utility in Anonymized Data Publishing Categories and Subject Descriptors. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 70–78, 2008.

[15] Y. Bu, A.W. Fu, R.C. Wong, L. Chen, and J. Li. Privacy preserving serial data publishing by role composition. *Proceedings of the VLDB Endowment*, 1(1):845–856, 2008.

[16] J. Byun, A. Kamra, E. Bertino, and N. Li. Efficient k -Anonymization Using Clustering Techniques. In *Proceedings of the 12th international conference on Database systems for advanced applications*, pages 188–200, 2007.

[17] J. Byun, Y. Sohn, E. Bertino, and N. Li. Secure Anonymization for Incremental Datasets. In *Third VLDB Workshop, SDM*, pages 1–15, 2006.

[18] J. Cao, P. Karras, C. Raissi, and K. Tan. p-uncertainty: Inference-Proof Transaction Anonymization. *Proceedings of the VLDB Endowment*, 3(1):1033–1044, 2010.

[19] R.M. Caplan. HIPAA. Health Insurance Portability and Accountability Act of 1996. In *Dental assistant (Chicago, Ill. : 1994)*, pages 6–8, 1996.

[20] V.T. Chakaravarthy, H. Gupta, P. Roy, and M.K. Mohania. Efficient techniques for document sanitization. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 843–852, 2008.

[21] S. Chawla, C. Dwork, F. McSherry, A. Smith, and H. Wee. Toward privacy in public databases. In *Second Theory of Cryptography Conference, TCC*, pages 363–385, 2005.

[22] B. Chen, D. Kifer, K. LeFevre, and A. Machanavajjhala. Privacy-Preserving Data Publishing. *Foundations and Trends in Databases*, 2(1-2):1–167, 2009.

[23] K. Chen and L. Liu. Privacy Preserving Data Classification with Rotation Perturbation. In *Fifth IEEE International Conference on Data Mining*, pages 589–592, 2005.

[24] K. Chen and L. Liu. Privacy preserving data classification with rotation perturbation. In *Fifth IEEE International Conference on Data Mining*, pages 589–592, 2005.

[25] C. Chow and M.F. Mokbel. Trajectory Privacy in Location-based Services and Data. *ACM SIGKDD*, 13(1):19–29, 2011.

[26] R. Chow, I. Oberst, and J. Staddon. Sanitization's Slippery Slope : The Design and Study of a Text Revision Assistant. In *Proceedings of the 5th Symposium on Usable Privacy and Security*, 2009.

[27] R.L. Cilibrasi and P.M.B. Vitányi. The google similarity distance. *IEEE Transactions on Knowledge and Data Engineering*, 19(3):370–383, 2007.

[28] C. Clifton and D. Marks. Security and Privacy Implications of Data Mining. In *ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery*, pages 15–19, 1996.

[29] G. Cormode, D. Srivastava, T. Yu, and Q. Zhang. Anonymizing bipartite graph data using safe groupings. *The VLDB Journal*, 19(1):115–139, 2009.

[30] C. Cumby and R. Ghani. Inference control to protect sensitive information in text documents. *ACM SIGKDD Workshop on Intelligence and Security Informatics*, 2010.

[31] T. Dalenius and S.P. Reiss. Data-swapping: A technique for disclosure control. *Journal of Statistical Planning and Inference*, 6(1):73–85, 1982.

[32] A. Datta, D. Sharma, and A. Sinha. Provable de-anonymization of large datasets with sparse dimensions. In *Principles of Security and Trust*, pages 229–248, 2012.

[33] M. M. Douglass, G.D. Cliffford, A. Reisner, W. J. Long, G. B. Moody, and R. G. Mark. De-Identification Algorithm for Free-Text Nursing Notes. In *Computers in Cardiology*, pages 331–334, 2005.

[34] C. Dwork. Differential privacy. In *33rd International Colloquium, ICALP*, pages 1–12, 2006.

[35] A. Erola, J. Castellá-Roca, G. Navarro-Arribas, and V. Torra. Semantic microaggregation for the anonymization of query logs using the open directory project. *SORT - Statistics and Operations Research Transactions*, 35(1):41–58, 2011.

[36] A. Evfimievski, J. Gehrke, and R. Srikant. Limiting privacy breaches in privacy preserving data mining. In *Proceedings of the twenty-second ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 211–222, 2003.

[37] B.C.M. Fung, K. Wang, R. Chen, and P.S. Yu. Privacy-preserving data publishing: A survey of recent developments. *Computing Surveys*, 42(4):1–53, 2010.

[38] B.C.M. Fung, K. Wang, A.W. Fu, and J. Pei. Anonymity for continuous data publishing. In *Proceedings of the 11th international conference on Extending database technology: Advances in database technology*, pages 264–275, 2008.

[39] B.C.M. Fung, K. Wang, and P.S. Yu. Top-Down Specialization for Information and Privacy Preservation. In *Proceedings 21st International Conference on Data Engineering*, pages 205–216, 2005.

[40] J. Gardner and L. Xiong. In *IEEE International Symposium on Computer-Based Medical Systems (CBMS)*, pages 254–259.

[41] J. Gardner, L. Xiong, F. Wang, A. Post, J. Saltz, and T. Grandison. An evaluation of feature sets and sampling techniques for de-identification of medical records. In *Proceedings of the 1st ACM International Health Informatics Symposium*, pages 183–190, 2010.

[42] L. Geng, Y. You, Y. Wang, and H. Liu. Privacy measures for free text documents: bridging the gap between theory and practice. In *Trust, Privacy and Security in Digital Business*, pages 161–173, 2011.

[43] G. Ghinita, Y. Tao, and P. Kalnis. On the anonymization of sparse high-dimensional data. In *IEEE 24th International Conference on Data Engineering*, pages 715–724, 2008.

[44] C. R. Giannella, K. Liu, and H. Kargupta. Breaching Euclidean distance-preserving data perturbation using few known inputs. *Data & Knowledge Engineering*, 84:93–110, 2013.

[45] A. Gkoulalas-divanis and V.S. Verykios. Exact knowledge hiding through database extension. *IEEE Transactions on Knowledge and Data Engineering*, 21(5):699–713, 2009.

[46] P. Golle. Revisiting the uniqueness of simple demographics in the US population. pages 77–80, 2006.

[47] T. Gruber. Ontology. In *Encyclopedia of Database Systems*. 2008.

[48] J. A. Hartigan and M. A. Wong. A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108, 1979.

[49] X. He, Y. Xiao, Y. Li, Q. Wang, W. Wang, and B. Shi. Permutation anonymization: Improving anatomy for privacy preservation in data publication. In *New Frontiers in Applied Data Mining*, pages 111–123, 2011.

[50] Y. He and J. F. Naughton. Anonymization of set-valued data via top-down, local generalization. *Proceedings of the VLDB Endowment*, 2(1):934–945, 2009.

[51] S. Hedegaard, S. Houen, and J.G. Simonsen. LAIR: A Language for Automated Semantics-Aware Text Sanitization Based on Frame Semantics. In *Semantic Computing, 2009. ICSC '09. IEEE International Conference*, pages 47–52, 2009.

[52] L. Hirschman and J. Aberdeen. Measuring risk and information preservation: toward new metrics for de-identification of clinical texts. In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 72–75, 2010.

[53] A. Hliaoutakis. Semantic Similarity Measures in MeSH Ontology and their application to Information Retrieval on Medline. In *Master's Thesis Technical University of Crete, Greek*, 2005.

[54] T. Hong, C. Lin, C. Chang, and S. Wang. Hiding sensitive itemsets by inserting dummy transactions. In *IEEE International Conference on Granular Computing (GrC)*, pages 246–249, 2011.

[55] A. Islam and D. Inkpen. Semantic text similarity using corpus-based word similarity and string similarity. *Transactions on Knowledge Discovery from Data*, 2(2):1–25, 2008.

[56] V. S. Iyengar. Transforming Data to Satisfy Privacy Constraints. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 279–288, 2002.

[57] J.J. Jiang and D.W. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. *In the Proceeding of International Conference on Research on Computational Linguistics*, pages 19–33, 1997.

[58] W. Jiang, M. Murugesan, C. Clifton, and L. Si. t-Plausibility : Semantic Preserving Text Sanitization. In *Proceedings IEEE CSE'09, 12th IEEE International Conference on Computational Science and Engineering*, pages 68–75, 2009.

[59] T. Johnsten and V.V. Raghavan. A methodology for hiding knowledge in databases. In *Workshop on Privacy, Security and Data Mining*, pages 9–17, 2014.

[60] H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar. On the privacy preserving properties of random data perturbation techniques. In *Third IEEE International Conference on Data Mining*, pages 99–106, 2003.

[61] H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar. Random-data perturbation techniques and privacy-preserving data mining. 7(4):387–414, 2005.

[62] J.T. Kent. Information gain and a general measure of correlation. *Biometrika Trust*, 70(1), 1982.

[63] D. Kifer. Attacks on privacy and deFinetti's theorem. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*, pages 127–138, 2009.

[64] D. Kifer and A. Machanavajjhala. No free lunch in data privacy. In *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*, pages 193–204, 2011.

[65] J.J. Kim and W. E. Winkler. Multiplicative noise for masking continuous data. In *Government publications, Census., Census report*, 2003.

[66] P. Kiran. A Survey on Methods , Attacks and Metric for Privacy Preserving Data Publishing. *International Journal of Computer Applications*, 53(18):20–28, 2012.

[67] E.M. Knox and R. T. Ng. Algorithms for Mining Distance-Based Outliers in Large Datasets. In *Proceedings of the 24rd International Conference on VLDB*, pages 392–403, 1998.

[68] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289, 2001.

[69] K. LeFevre, D.J. DeWitt, and R. Ramakrishnan. Incognito: Efficient full-domain k-anonymity. In *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, pages 49–60, 2005.

[70] M. Li and P.M.B. Vitányi. An introduction to kolmogorov complexity and its applications. In *Develops Kolmogorov theory in detail and outlines the wide range of illustrative applications*. 2008.

[71] N. Li, T. Li, and S. Venkatasubramanian. t-Closeness : Privacy Beyond k - Anonymity and l-diversity. In *IEEE 23rd International Conference on Data Engineering*, pages 106–115, 2007.

[72] T. Li and N. Li. Injector: Mining background knowledge for data anonymization. In *ICDE 2008. IEEE 24th International Conference on Data Engineering*, pages 446–455, 2008.

[73] T. Li and N. Li. On the Tradeoff Between Privacy and Utility. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2009.

[74] Y. Li, D. McLean, Z.A. Bandar, J.D. O'Shea, and K. Crockett. Sentence similarity based on semantic nets and corpus statistics. *IEEE Transactions on Knowledge and Data Engineering*, 18(8):1138–1150, 2006.

[75] J. Liu and K. Wang. Anonymizing transaction data by integrating suppression and generalization. In *Proceedings of the 14th Pacific-Asia conference on Advances in Knowledge Discovery and Data Mining*, pages 171–180, 2010.

[76] J. Liu and K. Wang. Enforcing Vocabulary k-Anonymity by Semantic Similarity Based Clustering. In *2010 IEEE 10th International Conference on Data Mining (ICDM)*, pages 899–904, 2010.

[77] G. Loukides, A. Gkoulalas-Divanis, and B. Malin. COAT: COnstraint-based anonymization of transactions. *Knowledge and Information Systems*, 28(2):251–282, 2010.

[78] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam. l-Diversity: Privacy Beyond k-Anonymity. 1(1), 2007.

[79] D.J. Martin, D. Kifer, A. Machanavajjhala, J. Gehrke, and J. Y. Halpern. Worst-Case Background Knowledge for Privacy-Preserving Data Publishing. In *Proceedings of the 23rd IEEE International Conference on Data Engineering (ICDE)*, pages 126–135, 2007.

[80] G. A. Miller. WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41, 1995.

[81] A. Narayanan and V. Shmatikov. Robust De-anonymization of Large Sparse Datasets. In *IEEE Symposium on Security and Privacy*, pages 111–125, 2008.

[82] G. Navarro-Arribas, V. Torra, A. Erola, and J. Castellá-Roca. User k-anonymity for privacy preserving data mining of query logs. *Information Processing and Management: an International Journal*, 48(3):476–487, 2011.

[83] M. E. Nergiz, C. Clifton, and A. E. Nergiz. Multirelational k-anonymity. *IEEE Transactions on Knowledge and Data Engineering*, 21(8):1104–1117, 2009.

[84] J. O'Shea, Z. Bandar, K. Crockett, and D. McLean. Pilot Short Text Semantic Similarity Benchmark Data Set: Full Listing and Description. In *Speech and Language Processing*, 2008.

[85] H. Park and K. Shim. Approximate algorithms for k-anonymity. In *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, pages 67–78, 2007.

[86] S.P. Patil. A novel approach for efficient mining and hiding of sensitive association rule. In *Nirma University International Conference on Engineering (NUiCONE)*, pages 1–6, 2012.

[87] J. Pei, J. Xu, Z. Wang, W. Wang, and K. Wang. Maintaining K -Anonymity against Incremental Updates. In *SSBDM '07. 19th International Conference on Scientific and Statistical Database Management*, page 5, 2007.

[88] S. R. Rajagopalan, L. Sankar, S. Mohajer, and H. V. Poor. Smart Meter Privacy: A Utility-Privacy Framework. In *2011 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pages 190–195, 2011.

[89] V. Rastogi, D. Suciu, and S. Hong. The boundary between privacy and utility in data publishing, 2007.

[90] D. Rebollo-monedero, J. Parra-arnau, C. Diaz, and J. Forné. On the Measurement of Privacy as an Attacker's Estimation Error. *International Journal of Information Security*, 12(2):129–149, 2011.

[91] P. Ruch, R.H. Baud, A. Rassinoux, P. Bouillon, and G. Robert. Medical document anonymization with a semantic lexicon. In *Proc AMIA Symp.*, pages 729–733, 2000.

[92] P. Ruch, J. Wagner, P. Bouillon, R.H. Baud, A. Rassinoux, and J. Scherrer. MEDTAG: tag-like semantics for medical document indexing. In *Proc AMIA Symp.*, pages 137–141, 1999.

[93] D. Sánchez. Detecting Term Relationships to Improve Textual Document Sanitization. pages 105–119, 2013.

[94] D. Sánchez, M. Batet, and A. Viejo. Detecting sensitive information from textual documents: An information-theoretic approach. pages 173–184, 2012.

[95] M. Sramka, R. Safavi-naini, J. Denzinger, and M. Askari. A Practice-oriented Framework for Measuring Privacy and Utility in Data Sanitization Systems. In *Proceedings of the 2010 EDBT/ICDT Workshops*, 2010.

[96] J. Staddon, P. Golle, and B. Zimny. Web-based inference detection. In *Proceedings of 16th USENIX Security Symposium on USENIX Security Symposium*, 2007.

[97] L. Sweeney. Replacing personally-identifying information in medical records, the Scrub system. In *AMIA Conference*, pages 333–337, 1996.

[98] L. Sweeney. Achieving k-Anonymity Privacy Protection Using Generalisation and Suppression. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(5):571–588, 2002.

[99] L. Sweeney. k-Anonymity: A Model For Protecting Privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(5):557–570, 2002.

[100] M. Terrovitis, N. Mamoulis, and P. Kalnis. Anonymity in unstructured data. In *VLDB*, pages 1–21, 2008.

[101] M. Terrovitis, N. Mamoulis, and P. Kalnis. Privacy-preserving Anonymization of Set-valued Data. *PVLDB*, 1(1):115–125, 2008.

[102] K. Toutanova, D. Klein, C. Manning, and Y. Singer. Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 252–259, 2003.

[103] Ö. Uzuner, Y. Luo, and P. Szolovits. Evaluating the state-of-the-art in automatic de-identification. *Journal of the American Medical Informatics Association*, 14(5):550–563, 2007.

[104] V.S. Verykios, A.K. Elmagarmid, E. Bertino, Y. Saygin, and E. Dasseni. Association rule hiding. *IEEE Transactions on Knowledge and Data Engineering*, 16:434–447, 2004.

[105] K. Wang, B.C.M. Fung, and P.S. Yu. Handicapping attacker's confidence: an alternative to k-anonymization. *Knowledge and Information Systems*, 11(3):345–368, 2007.

[106] K. Wang, P.S. Yu, and S. Chakraborty. Bottom-Up Generalization: A Data Mining Solution to Privacy Protection. In *Fourth IEEE International Conference on Data Mining*, pages 249–256, 2004.

[107] Wikipedia. Association rule learning.

[108] R.C. Wong, A.W. Fu, J. Liu, K. Wang, and Y. Xu. Global privacy guarantee in serial data publishing. In *IEEE 26th International Conference on Data Engineering (ICDE)*, pages 956–959, 2010.

[109] R.C. Wong, A.W. Fu, K. Wang, and J. Pei. Minimality Attack in Privacy Preserving Data Publishing. In *Proceedings of the 33rd international conference on VLDB*, pages 543–554, 2007.

[110] R.C. Wong, J. Li, A.W. Fu, and K. Wang. (α,k)-Anonymity : An Enhanced k-Anonymity Model for Privacy-Preserving Data Publishing. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 754–759, 2006.

[111] X. Xiao and Y. Tao. Anatomy: Simple and effective privacy preservation. In *Proceedings of the 32nd international conference on VLDB*, pages 139–150, 2006.

[112] Y. Xu, B.C.M. Fung, K. Wang, A.W. Fu, and P. Jian. Publishing Sensitive Transactions for Itemset Utility. In *Eighth IEEE International Conference on Data Mining*, pages 1109–1114, 2008.

[113] Y. Xu, K. Wang, A.W. Fu, and P.S. Yu. Anonymizing transaction databases for publication. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 767–775, 2008.

[114] C. Yao, X.S. Wang, and S. Jajodia. Checking for k-Anonymity Violation by Views. In *Proceedings of the 31st international conference on VLDB*, pages 910–921, 2005.

[115] L.A. Zadeh. A Simple View of the Dempster-Shafer Theory of Evidence and its Implication for the Rule of Combination. In *AI Magazine*, pages 85–90, 1986.

[116] L. Zhang, S. Jajodia, and A. Brodsky. Information disclosure under realistic assumptions: Privacy versus optimality. In *Proceedings of the 14th ACM conference on Computer and communications security*, pages 573–583, 2007.

[117] Q. Zhang, N. Koudas, D. Srivastava, and T. Yu. Aggregate Query Answering on Anonymized Tables. In *IEEE 23rd International Conference on Data Engineering (ICDE)*, pages 116–125, 2007.

[118] Y. Zhu, L. Xiong, and C. Verdery. Anonymizing user profiles for personalized web search. In *Proceedings of the 19th international conference on World wide web*, page 1225, 2010.