Hiding Sensitive Information When Sharing Distributed Transactional Data

Abstract

Retailers have been sharing transactional data with supply chain partners for a long time, to the benefit of all involved. However, many are still reluctant to share, and there is evidence that the extent of sharing would be greater if information sensitive to retailers is concealed before data is shared. While there has been considerable research into methods to hide sensitive information from transactional data, extant research has focused only on sensitive information at the organizational level. This is rarely the case in reality – the retail industry has recognized and adapted their offerings to region-wide differences in customer tastes for decades, and when stores offer a mix of standardized and customized products, the differences in customer characteristics across regions lead to sensitive information that is region-specific, in addition to sensitive information at the organizational level. To date, this version of the problem has been overlooked, and no effective methods exist to solve it; this paper fills that gap. While some existing approaches can be adapted to this more realistic context, the existence of region-level requirements substantially increases the size of an already difficult (NP-hard) problem to be solved, making such adaptations impractical. Traditional decomposition-based approaches like Lagrangian relaxation are not viable either, as they require the repeated solution of NP-hard problems involving millions of variables multiple times. In this paper, we present an ensemble approach that draws intuition from Lagrangian relaxation to maximize the accuracy of a shared transactional dataset. Extensive computational experiments show that this approach not only identifies near-optimal solutions, it can do so even when other approaches fail. We also show that the precision of recommendations made using datasets that have been modified using the ensemble approach is not statistically different from that of recommendations made using the original datasets; this demonstrates that using the ensemble approach to hide sensitive information before sharing transactional data has negligible negative impact.

Keywords: Data quality, Lagrangian relaxation, ensemble, itemset hiding, privacy.

Hiding Sensitive Information When Sharing Distributed Transactional Data

1 INTRODUCTION

Information sharing between partners in supply chains has been standard practice for decades (Chen and Deng 2015). *RetailLink* (WalMart), *HomeDepotLink* (Home Depot) *VendorDart* (Lowe's), and *MerchIQ* (Target) are but some prominent examples of data-sharing IT systems currently in use (RetailVelocity 2017). Historically, the focus of sharing has primarily been cost and inventory, and its contribution to reducing the bullwhip effect is well documented (Croson and Donohue 2003). In the retail context, one particularly desirable type of information being shared by the retailer takes the form of transactional, or market basket, data – essentially, a collection of records of products purchased by customers in individual transactions (GMA 2009, Chandra 2012). The value of such data is underscored by Lobel et al. (2017), who note that retailers consider it more valuable than all other types of data combined.

While transactional data is shared at various levels of granularity, recent trends suggest that more retailers are beginning to share complete transactional data (i.e., data at the finest level of granularity) with suppliers (GXS 2013, Retail TouchPoints 2017). For example, Seymour (2014, paragraph 4) points out that Lowes provides "suppliers access to in-store transaction data allowing them to analyze cross-sells, geographic penetration and orders to co-develop insights and strategies." Along similar lines, Petersen (2013, paragraph 7) notes that "Walmart essentially gives suppliers all of their sell through data … by store". One important objective of sharing transactional data is to work alongside suppliers to mine and identify associations among items that are sold together, such as complementary products (Retail TouchPoints 2017) – for example, Kraft found that a retailer's sales increased when Kraft's salad dressings were displayed next to fresh produce (Manthan 2017). Such cooperative arrangements help both suppliers and retailers, increasing sales and providing insights on where to focus marketing efforts (Munves 2013).

Despite such initiatives, many retailers remain unwilling to do so. Konzak (2012, Page 4) notes that several retailers in a Modern Distribution Management (MDM) survey were not sharing because *"distributors are concerned that manufacturers will use that information to bypass the distributor and go*

direct. Or that manufacturers will hand off the data or leads from the data to competing distributors – something that has happened to some of the distributors who responded to the MDM survey." The 2013 Retailer/Supplier Shared Data Study echoed these concerns (Alaimo 2013). Market research companies recognize this hesitation on the part of retailers, and have suggested that sharing, even if incomplete, can be of mutual benefit. For example, Computer Market Research (2017, paragraph 23) explicitly lists compromise as a strategy, noting that "*there is no reason your channel partners cannot confidently provide you with some useful information without turning over their most sensitive data. By discovering a compromise that is mutually beneficial, everyone has the opportunity to walk away satisfied (and without feeling "used")*."

This realization – that many of the benefits of sharing can be achieved while hiding sensitive information – is not new to researchers. Indeed, the tradeoff between the benefits of sharing transactional data and the potential risks inherent in it has been long recognized, and many approaches have been suggested to hide sensitive information prior to sharing (e.g., Oliveira and Zaïane 2002, Verykios 2013, Stavropoulos et al. 2016). Sensitive information in the context of transactional data usually involves information derived from frequently occurring sets of items (*frequent itemsets*, or simply *itemsets*). Such information could be a result of promotions, cross-selling efforts, and shelf layouts that were surprisingly successful, or serendipitous associations observed when mining data. Therefore, most research that attempts to hide sensitive information in the context of transactional data hides patterns in some form – either directly as itemsets (as we do in this paper), or in a more nuanced form (like association rules and sequential patterns).

Typically, the hiding of sensitive itemsets is accomplished in two steps. In the first, a subset of transactions from which sensitive itemsets need to be hidden is identified. The sensitive itemsets are then hidden from each of the identified transactions by selectively removing items from the transaction such that the altered transactions no longer contain anything sensitive – a process referred to as *sanitization*.

While many researchers have looked into the hiding of sensitive itemsets, all extant research has assumed that sensitive information exists only at the organizational level. This however is seldom the case

– knowledge has long been recognized as specific in time and place (Hayek, 1945), and Anand and Mendelson (1997) highlight the importance of local knowledge in retailers' decision making. Region-specific differences in customer tastes are well known to affect what they buy (Gilman 1987), and retailers like Walmart and Macy's have been adapting their assortments based on local needs for a long time (Zimmerman 2006, O'Connell 2008). Over a decade ago, Vishwanath and Rigby (2006) pronounced that localization is a "*revolution in consumer markets*," noting that "*consumer communities are growing more diverse – in ethnicity, wealth, lifestyle, and values*." They observed that companies are mining data in order to obtain "*deep insight into local preferences and buying behaviors*", which "*make it possible to "localize" stores, products, and services with unprecedented precision*." Pearson (2016) emphasizes that customization allows shoppers in different geographical regions to have different experiences tailored to their specific needs. This is echoed in Lobel et al. (2017), who note that "*brands may perform differently across stores*" and further that "*a store system that can optimize each store for their target customer and demographic market vs. an average for an overall multilocation system is well positioned to increase profitability of each store.*"

Recognizing that existing methods for conducting market basket analysis may "*fail to discover important purchasing patterns in a multi-store environment*", Chen et al. (2005) propose a method for mining itemsets in a distributed, multi-store environment. In this setting – where retail stores offer a mix of standardized and customized products – the differences in customer characteristics across regions lead to sensitive information that is store-specific, in addition to sensitive information at the organizational level. Consequently, data owners need to hide sensitive information both at the store and organizational levels before sharing datasets comprising transactions across different stores.

An example database involving 20 transactions and 2 stores is presented in Table 1. We use this example to illustrate various aspects of the problem studied here. As indicated by the presence of antifreeze, Partition D_1 represents a store in a cold region. Partition D_2 on the other hand, represents a store in a warm coastal region (where swimwear is common, and antifreeze is not). Together, the two partitions constitute the

consolidated organizational database D.

Table 1: Example Database

		Transactions			
		ID	**Items**	**ID**	**Items**
Consolidated Dataset D	**Partition D_1**	t_1	Chips, Beer, Swimwear, Bag, Cake Mix, Bottled Water, Antifreeze	t_6	Bag, Bottled Water, Vitamins, Antifreeze
		t_2	Mixed Nuts, Antifreeze	t_7	Beer, Mixed Nuts
		t_3	Chips, Beer, Swimwear, Bag, Cake Mix, Energy Bar, Vitamins, Antifreeze	t_8	Cake Mix, Bottled Water, Energy Bar
		t_4	Chips, Bag, Cake Mix, Bottled Water, Energy Bar, Vitamins, Antifreeze	t_9	Chips, Beer, Cake Mix, Antifreeze
		t_5	Bag, Cake Mix, Mixed Nuts, Energy Bar, Vitamins, Antifreeze	t_{10}	Chips, Energy Bar, Antifreeze
	Partition D_2	t_{11}	Chips, Beer, Swimwear, Bag, Cake Mix, Energy Bar, Vitamins	t_{16}	Chips, Beer, Vitamins
		t_{12}	Swimwear, Bottled Water, Mixed Nuts, Energy Bar	t_{17}	Swimwear, Bottled Water, Vitamins
		t_{13}	Chips, Beer, Swimwear, Bottled Water, Mixed Nuts, Energy Bar, Vitamins	t_{18}	Swimwear, Cake Mix, Energy Bar
		t_{14}	Chips, Beer, Swimwear	t_{19}	Beer, Swimwear, Mixed Nuts
		t_{15}	Chips, Beer, Swimwear, Bag, Energy Bar, Vitamins	t_{20}	Swimwear, Cake Mix, Bottled Water

Suppose the organization finds that a nationwide promotion that bundled energy bars and vitamins together was very successful, and they do not want to divulge this knowledge to others. They could choose not to share their data, or to suppress this sensitive information before sharing. If the decision is to share, the sensitive information can be hidden by making sure that not many transactions in the database contain it. As six transactions (t_3, t_4, t_5, t_{11}, t_{13}, t_{15}) contain both energy bars and vitamins, hiding would require the removal of at least one of the two items from some of these transactions. For example, suppose the organization is comfortable if fewer than 5 transactions contain this pair. In that case, removing either item from any two of the six transactions – say, t_3 and t_4 – will achieve that goal; t_3^s = {Chips, Beer, Swimwear, Bag, Cake Mix, Energy Bar, Antifreeze} and t_4^s = {Chips, Bag, Cake Mix, Bottled Water, Vitamins,

Antifreeze} are possible sanitized versions of t_3 (from which vitamins have been removed) and t_4 (from which the energy bar has been dropped).

As mentioned earlier, all itemset-hiding approaches proposed to date have assumed that sensitive information exists only at the organizational level. While some of these approaches can be adapted to the new context to some degree, the multi-store environment comes with a new set of challenges that make effective adaptation difficult. In order to understand exactly what these challenges are, we need to recall that the original problem – where sensitive information exists only at the organizational level – is NP-Hard (Atallah et al. 1999). It can be formulated as a generalized set-covering problem (Menon et al. 2005), with as many binary variables as transactions in the database, and as many constraints as sensitive itemsets. As each store can have its own set of distinct sensitive itemsets to conceal, the number of constraints in the problem increases dramatically relative to the original version of the problem. Loosely speaking, the number of constraints increases by a factor of the number of stores with sensitive information; in practical terms, this corresponds to the number of constraints increasing from the tens into the thousands (as most large retailers have hundreds of stores). Combined with the fact that the number of variables in the problem can be in the hundreds of millions, this makes the new version of the problem significantly more difficult to solve from a practical perspective. This implies that, for problems of realistic size, solution via optimal approaches are effectively ruled out.

While the structure of the problem naturally lends itself to decomposition-based solution procedures, traditional approaches like Lagrangian relaxation are not viable as they would require the repeated solution of the integer programs associated with each store (involving millions of variables and hundreds of constraints), which is impractical. However, the intuition behind Lagrangian relaxation can still be helpful. We present an ensemble approach that draws inspiration from Lagrangian relaxation to identify the transactions for sanitization. The first step of this approach solves two different relaxations of the problem, to obtain two different sets of solutions. Both incorporate a simple but effective forward-looking approach to exploit the fact that multiple optimal solutions often exist for the sub-problems to be solved. The second

step takes the results from the two relaxations and attempts to identify a better solution by leveraging the commonality between them. Computational experiments conducted to examine the effectiveness of the proposed approach show that the ensemble approach consistently finds near-optimal solutions. In addition, it is able to identify solutions in situations where other approaches fail. We also find that the quality of recommendations made using the sanitized datasets are essentially just as good as that of recommendations made using the original (unaltered) datasets.

2 LITERATURE REVIEW

The problem of hiding sensitive information before sharing transactional data is not new. While the problem can be viewed from different perspectives, Atallah et al. (1999) show that most versions of the problem are NP-hard. Verykios et al. (2004) view the problem from the perspective of hiding association rules and some of their approaches are also applicable to the hiding of sensitive itemsets. Menon et al. (2005) present an integer programming formulation to minimize the number of transactions modified while hiding sensitive itemsets, while Menon and Sarkar (2016) show that ideas from linear programming column-generation can be used to address scalability concerns by solving problems involving databases with as many as 100 million transactions. However, the structure exploited there – a hierarchical structure among the columns of the problem – exists only within partitions, and does not extend across different partitions in the multi-partition version of the problem addressed in this paper. Consequently, while that approach can be applied to solve the problems associated with each partition when they are very large, it is not appropriate for the multi-partition version addressed in this paper.

Gkoulalas-Divanis and Verykios (2006) hide sensitive association rules while minimizing the distance between the original dataset and its sanitized version. Hong et al. (2013) hide sensitive itemsets by changing transactions that are similar (based on a modified version of Term Frequency/Inverse Document Frequency) to the sensitive itemsets. Lin et al. (2015, 2016) propose genetic algorithms for deleting items to hide sensitive itemsets, while limiting the side effects on the dataset. Moustakides and Verykios (2006) propose two approaches they term Max-Min algorithms to remove sensitive patterns. Stavropoulos et al. (2016)

develop a method based on minimal traversals of a hypergraph to identify the transactions to sanitize. The integer program presented there reduces to that of Menon et al. (2005) when maximizing accuracy is the objective of interest. Cheng at al. (2016) apply distortion-based method to hide sensitive rules by removing some items from transactions to reduce the support or confidence of the sensitive rules. Distortion-based approaches to hide sensitive association rules are proposed by Verykios et al. (2007) and Cheng et al. (2015), while Wu et al. (2007) try to hide sensitive rules by limiting undesirable side effects.

Oliveira and Zaïane (2002) introduce a framework to hide sensitive itemsets while keeping the number of non-sensitive itemsets hidden to a minimum. Using the same objective, Sun and Yu (2007) propose a border-based approach that efficiently evaluates the impact of any modification to a dataset during the process of hiding sensitive itemsets. Menon and Sarkar (2007) provide an integer programming formulation to minimize the number of non-sensitive itemsets lost while concealing the sensitive ones. Leloglu et al. (2014) and Kagklis et al. (2014) modify the coefficients in the objective function of the formulation of Menon et al. (2005) to achieve a similar objective. Kagklis et al. (2018) provide a toolbox for hiding frequent itemsets that incorporates seven state-of-the-art approaches to solve the single-partition version problem; specifically, it includes both approaches of Moustakides and Verykios (2006), along with those of Gkoulalas-Divanis and Verykios (2006), Sun and Yu (2007), Menon et al. (2005), Leloglu et al. (2014) and Kagklis et al. (2014).

As mentioned earlier, despite the rich literature on hiding sensitive information from transactional datasets, all work to date on this topic has overlooked the fact that most retail organizations have multiple locations, and that data from different locations can be quite different from each other in terms of customer purchasing patterns. There has, however, been some research on the mining of distributed data – for example, Vaidya and Clifton (2002) address the problem of mining globally valid frequent itemsets and association rules from vertically distributed data, while Kantarcioglu and Clifton (2004) do the same with datasets that are partitioned horizontally. Procedures to mine a distributed dataset using partition-specific mining parameters have also been proposed (Chen et al. 2005). However, there has been no attempt to date

to hide sensitive information from transactional data involving many partitions, with sensitive itemsets specific to the region or store associated with the partition, in addition to sensitive itemsets at the organizational level. As noted before, while some existing approaches can be adapted to the new context to some extent, most existing methods do not scale well to this very relevant, yet overlooked version of the problem.

3 DEFINITIONS AND PROBLEM FORMULATION

The formulation for the problem being studied here is introduced in this section, along with all relevant notation.

3.1 Notation and Definitions

I is a set of items, and $j \subseteq I$ an *itemset*. A transaction i defined over I is the set of items purchased by a customer in one visit. P is the set of locations (stores/regions), and the dataset from location p is denoted D_p. $D = \bigcup_{p \in P} D_p$ is the dataset obtained by consolidating all partitions D_p. If an itemset j is contained in transaction i (i.e., $j \subseteq i$), we say that transaction i supports itemset j. The *support* σ^j (σ_p^j) of itemset j is the number of transactions in D (D_p) that contains j. An itemset j is considered *frequent* if there are at least σ_{min} transactions in D (σ_{min}^p transactions in D_p) supporting it, where the owner-specified value σ_{min} (σ_{min}^p) is called the *mining threshold* for support. The set of frequent itemsets F (F_p) for D (D_p) is the set of all itemsets with a minimum support of σ_{min} (σ_{min}^p). Based on their business strategies, the owner of the data identifies some itemsets $F^S \subseteq F$ as sensitive in the consolidated dataset D and some itemsets $F_p^S \subseteq F_p$ as sensitive in partition D_p. The itemsets in F^S and F_p^S are the ones the owner wishes to hide, and as in prior literature, an itemset is considered hidden if its support in the sanitized dataset falls below an owner-specified *hiding threshold* σ_h^j for D (σ_{hp}^j for D_p). The mining and hiding thresholds can also be expressed as relative values, as a percentage of the number of transactions in the corresponding dataset. The hiding thresholds are selected by the owner such that they are comfortable revealing the sensitive itemsets if the receiver were to mine the consolidated dataset D with a mining threshold lower than σ_h^j, or mine partition

D_p with a mining threshold lower than σ_{hp}^{j}.

We clarify these concepts using the dataset introduced in Table 1. In the store from the colder region (D_1), suppose the organization finds that antifreeze and mixed nuts have sold together unexpectedly well (perhaps to have something to snack on while waiting for a frozen vehicle to become road-worthy again), as have chips and bottled water (possibly for similar reasons). In the store from the coast (D_2), they may find that swimwear is purchased with bags surprisingly often (perhaps because both are useful at a beach), and that energy bars are often purchased with bottled water and mixed nuts (perhaps because people in the region are more health conscious). In addition to these patterns at the store-level, suppose they find three interesting patterns at the organizational level as well – chips and beer often being sold with swimwear (all are relevant in the context of a pool party), bags being sold with cake mixes (potentially driven by birthday parties), and energy bars being purchased with vitamins (perhaps because both have health implications). Note that some of these might be a result of promotions, cross-selling efforts, or shelf layouts that were surprisingly successful (e.g., energy bars and vitamins), while others might be fortuitous associations observed when mining the data (e.g., antifreeze and mixed nuts). Irrespective of how the relationships were discovered, the organization may consider these unexpected purchasing patterns – i.e., the itemsets that comprise the sets F^S, F_1^S and F_2^S listed in Table 2 – can be of strategic importance, as their competitors are unlikely to be aware of them. Consequently, the organization would like to hide these patterns before sharing the data with anyone else. If they have set the relative hiding thresholds to 0.2 (20%), it would translate to absolute hiding thresholds of $\sigma_h^j = 0.2 \times 20 = 4$ for the sensitive itemsets at the organizational level, and $\sigma_{hp}^j = 0.2 \times 10 = 2$ for the sensitive itemsets in each partition. That is, they will consider the sensitive itemsets hidden if their supports are brought below these values (i.e., to no more than 3 and 1 for the organization-level and store-level sensitive itemsets, respectively).

Table 2: Sensitive Itemsets

	Sensitive Itemsets		Supports (σ^j, σ_p^j)	Supported By	Hiding Thresholds $(\sigma_h^j, \sigma_{hp}^j)$
Consolidated Dataset D	$F^S = \{S_1, S_2, S_3\}$	S_1: {Chips, Beer, Swimwear}	6	$t_1, t_3, t_{11}, t_{13}, t_{14}, t_{15}$	4
		S_2: {Bags, Cake Mixes}	5	$t_1, t_3, t_4, t_5, t_{11}$	4
		S_3: {Energy Bars, Vitamins}	6	$t_3, t_4, t_5, t_{11}, t_{13}, t_{15}$	4
Partition D_1	$F_1^S = \{s_1^1, s_2^1\}$	s_1^1: {Chips, Bottled Water}	2	t_1, t_4	2
		s_2^1: {Mixed Nuts, Antifreeze}	2	t_2, t_5	2
Partition D_2	$F_2^S = \{s_1^2, s_2^2\}$	s_1^2: {Bottled Water, Mixed Nuts, Energy Bars}	2	t_{12}, t_{13}	2
		s_2^2: {Swimwear, Bags}	2	t_{11}, t_{15}	2

Concealing a sensitive itemset j in D (D_p) implies that at least $(\sigma_j - \sigma_h^j + 1)$ transactions in D (($\sigma_p^j - \sigma_{hp}^j + 1$) transactions in D_p) need to be sanitized – this reduces the support of itemset j in the sanitized dataset to a value below the owner-specified hiding threshold σ_h^j (σ_{hp}^j). For example, hiding sensitive itemset s_2^1 = {Mixed Nuts, Antifreeze} in Partition D_1 involves sanitizing t_2 and/or t_5 since $(\sigma_1^1 - \sigma_{h1}^1 + 1) = (2 - 2 + 1) = 1$ for this itemset. If t_2 is chosen for sanitization, it could be altered to either {Mixed Nuts} or {Antifreeze}. Notice that transactions $t_6 - t_{10}$ of D_1 and $t_{16} - t_{20}$ of D_2 do not support any sensitive itemset. As sanitizing these transactions will not help reduce the support for any sensitive itemset in the database, they can be ignored when identifying transactions for sanitization.

We have chosen accuracy – the proportion of transactions that do not need to be sanitized to hide sensitive information (Menon et al. 2005) – as the measure of quality in this paper as in many others in the literature. The accuracy of a relation has a nice interpretation from the perspective of data quality; it was defined by Reddy and Wang (1995) as the proportion of accurate tuples in the relation, where a tuple is said to be accurate if and only if every attribute value in the tuple is accurate. Consequently, accuracy in our context is the proportion of transactions left unsanitized, and minimizing the number of transactions to sanitize maximizes accuracy. This is the only measure available in the literature that is appropriate for use when specific details of the mining parameters (such as minimum support) to be used by the receiver of the

data are not available.

Once transactions are identified for sanitization, they need to be sanitized. There are many ways to sanitize an individual transaction (e.g., Atallah et al. 1999, Menon et al. 2005). However, the specific sanitization technique does not impact accuracy, as accuracy – the number of transactions modified – is not affected by how each transaction is sanitized.

3.2 Integer Programming Formulation

The problem of maximizing the accuracy of a shared transactional dataset while hiding all sensitive itemsets (the *Frequent Itemset Hiding* problem for the distributed dataset *D*) can be formulated as FIH_D, after defining the following parameters and variables. Parameter a_{ij} (a_{ij}^p) is 1 if transaction i supports the organizational-level sensitive itemset $j \in F^S$ (partition-level sensitive itemset $j \in F_p^S$), and 0 otherwise. Decision variable x_i is 1 if transaction i is marked for sanitization, and 0 otherwise. The objective is to minimize the total number of transactions to be sanitized in the distributed dataset. The current support for the organizational-level sensitive itemset j in F^S (partition-level sensitive itemset $j \in F_p^S$) is σ^j (σ_p^j) and σ_h^j (σ_{hp}^j) is the hiding threshold specified by the owner for sensitive itemset j in F^S (F_p^S).

$$
\begin{aligned}
\text{FIH}_D \quad \min \quad & \sum_{i \in D} x_i & & & & \\
\text{s.t.} \quad & \sum_{i \in D} a_{ij} x_i & \geq & \left(\sigma^j - \sigma_h^j + 1\right) & \forall j \in F^S & \quad (OC) \\
& \sum_{i \in D_p} a_{ij}^p x_i & \geq & \left(\sigma_p^j - \sigma_{hp}^j + 1\right) & \forall j \in F_p^S, \forall p \in P & \quad (PC) \\
& x_i & \in & \{0,1\} & \forall i \in D &
\end{aligned}
$$

Constraint (*OC*) states that at least $(\sigma^j - \sigma_p^j + 1)$ out of σ^j transactions supporting organizational-level sensitive itemset $j \in F^S$ need to be sanitized. Constraint (*PC*) states similar requirements for the sensitive itemsets in each partition D_p. To date, all existing research has considered problems without constraint set (*PC*). As there can be many regions or stores contributing data to the consolidated dataset, the number of constraints in (*PC*) can be quite large even if the number of sensitive itemsets in each partition is small. For example, in the case of a distributed dataset with 100 partitions and 100 sensitive itemsets in each partition,

there would be 10,000 more constraints in FIH_D than in the formulation involving just the consolidated dataset. This increase in the number of constraints makes the integer program substantially harder to solve from a practical perspective. As was the case for the problem addressed in Menon et al. (2005), FIH_D is a generalized set covering problem, and is NP-Hard. Formulation FIH_D for the example in Table 1 (with sensitive itemsets F^S, F_1^S and F_2^S identified in Table 2, and variables corresponding to the relevant transactions $t_1 - t_5$ and $t_{11} - t_{15}$) is shown below; the outlined blocks highlight the decomposition structure underlying the problem. If sensitive itemsets are considered only at the organizational level, the formulation would involve only the first block, comprising the constraints S_1, S_2 and S_3.

$$
\begin{array}{lll}
\text{FIH}_D & \min & x_1+x_2+x_3+x_4+x_5+x_{11}+x_{12}+x_{13}+x_{14}+x_{15} \\
& \text{s.t.} & \boxed{\begin{array}{llllllllll} x_1 & & +x_3 & & & +x_{11} & & +x_{13}+x_{14}+x_{15} & \geq & 3 \quad (S_1) \\ x_1 & & +x_3+x_4+x_5+x_{11} & & & & & & \geq & 2 \quad (S_2) \\ & & x_3+x_4+x_5+x_{11} & & & & +x_{13} & +x_{15} & \geq & 3 \quad (S_3) \end{array}} \\
& & \begin{array}{|llll|llll|} \hline x_1 & & +x_4 & & & & & & \geq & 1 \quad (s_1^1) \\ & x_2 & & +x_5 & & & & & \geq & 1 \quad (s_2^1) \\ \hline & & & & & x_{12}+x_{13} & & & \geq & 1 \quad (s_1^2) \\ & & & & x_{11} & & & +x_{15} & \geq & 1 \quad (s_2^2) \\ \hline \end{array} \\
& & x_1 \, , \, x_2 \, , \, x_3 \, , \, x_4 \, , \, x_5 \, , \, x_{11} \, , \, x_{12} \, , \, x_{13} \, , \, x_{14} \, , \, x_{15} \in \{0,1\}
\end{array}
$$

Solving (FIH_D) to optimality sets x_1, x_3, x_{11}, and x_{13} to 1 and all other variables to 0, resulting in an optimal objective function value of 4.

4 THE *ENSEMBLE* APPROACH

Solving formulation FIH_D directly is not practical in most realistic scenarios – as the number of constraints in formulation FIH_D increases with the number of partitions, and as large retailers have hundreds of stores, most realistic versions of FIH_D will likely involve thousands of constraints and many millions of variables. Indeed, even established decomposition-based approaches that could potentially be employed on problems with the general structure of FIH_D – like Lagrangian relaxation and Dantzig-Wolfe decomposition – are not viable, as they require the (NP-hard) generalized set covering problems associated with each store (each involving millions of variables) to be solved multiple times. This is impractical – solving the extremely large NP-hard problems associated with each partition many times over, even when feasible, will take a substantial amount of time. Therefore, a solution approach that exploits the underlying decomposition

structure while avoiding the need to solve the sub-problems multiple times will make the solution of otherwise intractable problem instances feasible.

Although Lagrangian relaxation cannot be used directly, the intuition behind it can be leveraged to arrive at approaches grounded in theory to find good solutions to FIH_D. The ensemble approach developed in this paper first solves FIH_D using two different relaxations, and exploits the commonality between the two solutions in the next step to arrive at a better one.

4.1 *Ensemble* - Step 1

We now discuss the heuristics involved in the first step of the ensemble approach, and the relaxations that lead to their development.

4.1.1 Procedure *PartitionsFirst*

If we associate multipliers $\lambda_j \geq 0$ with each of the organizational-level constraints j in constraint set (OC) of FIH_D and dualize them, we get the Lagrangian dual problem $\mathcal{L}^{OC}(\lambda)$ below.

$$\begin{aligned}
\mathcal{L}^{OC}(\lambda) \quad \min \quad & \sum_{i \in D} x_i \quad + \quad \sum_{j \in F^S} \lambda_j \left(\left(\sigma^j - \sigma_h^j + 1\right) - \sum_{i \epsilon D} a_{ij} x_i \right) & & \\
\text{s.t.} \quad & \sum_{i \in D_p} a_{ij}^p x_i \quad \geq \quad \left(\sigma_p^j - \sigma_{hp}^j + 1\right) & & \forall j \in F_p^S, \forall p \in P \quad (PC) \\
& x_i \quad \in \quad \{0,1\} & & \forall i \in D
\end{aligned}$$

The objective function of $\mathcal{L}^{OC}(\lambda)$ can be re-written as

$$\min \sum_{i \in D} \left(1 - \sum_{j \in F^S} \lambda_j a_{ij} \right) x_i + \sum_{j \in F^S} \lambda_j \left(\sigma^j - \sigma_h^j + 1\right).$$

For any $\lambda_j \geq 0$, $\mathcal{L}^{OC}(\lambda)$ is a lower bound on the optimal objective function value of FIH_D. In our first heuristic (*PartitionsFirst*), we fix λ_j to a very small value $\epsilon > 0$ to get $\mathcal{L}^{OC}(\epsilon)$ with objective function

$$\min \sum_{i \in D} \left(1 - \epsilon \left(\sum_{j \in F^S} a_{ij} \right) \right) x_i + \epsilon \left(\sum_{j \in F^S} \left(\sigma^j - \sigma_h^j + 1\right) \right).$$

Note that $\left(\epsilon\left(\sum_{j \in F^S}\left(\sigma^j - \sigma_h^j + 1\right)\right)\right)$ is a constant, and can be ignored when optimizing $\mathcal{L}^{OC}(\epsilon)$.

$\mathcal{L}^{OC}(\epsilon)$ decomposes into disconnected subproblems $\mathcal{L}^{OC}_{D_p}(\epsilon)$ for each partition, each of which can be solved independently of the others. If the solution to $\mathcal{L}^{OC}(\epsilon)$ is not feasible to FIH_D, one approach to obtaining a feasible solution to FIH_D – and consequently, an upper bound to it – is to identify the fewest number of transactions that will make $\left(\left(\sigma^j - \sigma^j_h + 1\right) - \sum_{i \epsilon D} a_{ij} x_i\right)$ non-positive for every $j \in F^S$. This ensures that all the constraints in FIH_D corresponding to the organizational-level constraints are satisfied, thereby making the resulting solution feasible to FIH_D.

Here, ϵ is selected such that its effect is recognizable by the computer, and the total impact of the terms $\epsilon\left(\sum_{j \in F^S} a_{ij}\right)$ on the objective function is less than 1. This ensures that the optimal solution of each partition is chosen from the set of optimal solutions of the partition where the organizational-level constraints are ignored (i.e., $\mathcal{L}^{OC}_{D_p}(0)$). $\left(\sum_{j \in F^S} a_{ij}\right)$ simply adds up the number of sensitive itemsets in the organizational-level constraints supported by transaction i, thereby giving priority to transactions that support more of them. Therefore, the term $\left(1 - \epsilon\left(\sum_{j \in F^S} a_{ij}\right)\right)$ makes the formulation look forward, and drives the solutions of the partitions towards transactions that also support sensitive itemsets at the organizational level.

The difference in the objective function values of $\mathcal{L}^{OC}_{D_p}(0)$ and $\mathcal{L}^{OC}_{D_p}(\epsilon)$ for partition p is

$$\sum_{\{i \in D_p | x_i = 1\}} x_i - \sum_{\{i \in D_p | x_i = 1\}} \left(1 - \epsilon\left(\sum_{j \in F^S} a_{ij}\right)\right) x_i = \sum_{\{i \in D_p | x_i = 1\}} \epsilon\left(\sum_{j \in F^S} a_{ij}\right) x_i = \epsilon \times \sum_{\{i \in D_p | x_i = 1\}} \left(\sum_{j \in F^S} a_{ij}\right) x_i$$

In the extreme case where all the x_i are set to 1, this is equal to $\epsilon \times \sum_{i \in D_p} \sum_{j \in F^S} a_{ij}$. We know that the number of transactions marked for sanitization will be integral – i.e., the objective function value of the problem without ϵ is integral. Therefore, if the total impact of ϵ is less than 1, the ceiling of the objective function value of the problem with ϵ will be the same as the objective function value of the problem without ϵ – i.e., it will be one of the optimal solutions of the original problem. As we want $\epsilon \times \sum_{i \in D_p} \sum_{j \in F^S} a_{ij} < 1$, setting ϵ to any value less than $\frac{1}{\sum_{i \in D_p} \sum_{j \in F^S} a_{ij}}$ will ensure that the total impact of ϵ is less than 1. One such value is

$\frac{1}{|D_p| \times |F^S|}$, since $|F^S|$ is an upper bound on $\sum_{j \in F^S} a_{ij}$ and $|D_p|$ is an upper bound on $\sum_{\{i \in D_p | x_i = 1\}} x_i$. We note that in general, the best way to avoid rounding issues is to use the largest valid value of epsilon.

Proposition 1 identifies one situation when the solution from *partitionsFirst* will be optimal.

Proposition 1: If the optimal solutions to the partition problems in *partitionsFirst* result in a feasible solution to the original integer program FIH_D, this solution is optimal to FIH_D.

Proof: Let x_p^* be an optimal solution to partition *p*, for a valid value of ϵ. We know that ϵ is chosen such that x_p^* is one of the multiple optima of $\mathcal{L}^{OC}(0)$ for partition *p*. So x_p^* is optimal to $\mathcal{L}^{OC}(0)$ for partition *p*. As $x^* = \bigcup_p x_p^*$ is feasible to FIH_D, we know that x^* satisfies the organizational level constraints (OC) (for notational convenience, we will refer to these as $Ax \geq b$). This implies that $\mathcal{L}^{OC}(\epsilon) = cx^* + \epsilon(b - Ax^*) \leq cx^*$. But as x^* is optimal to $\mathcal{L}^{OC}(0)$ for partition *p*, we know that $\mathcal{L}^{OC}(0) = cx^* + 0(b - Ax^*) = cx^*$, and therefore x^* is optimal to FIH_D. □

As there is no overlap among the partitions, the sequence in which the partitions are solved does not affect the optimal solutions of the partitions (so they can be solved in parallel). Once the subproblems for the partitions have been solved, we check whether the solution is feasible to FIH_D. If so, we conclude with the optimal solution to FIH_D, based on Proposition 1. If not, we identify the fewest number of transactions that will make $\left(\left(\sigma^j - \sigma_h^j + 1\right) - \sum_{i \epsilon D} a_{ij} x_i\right)$ non-positive for every $j \in F^S$, thereby identifying an upper bound to FIH_D. To do this, we substitute the solutions obtained for the partition-level problems $\mathcal{L}_{D_p}^{OC}(\epsilon)$ into FIH_D, and obtain a reduced version of the problem involving only organizational-level constraints (as this solution is feasible to the subproblems of each partition, all the partition-level constraints are satisfied by it). The reduced formulation is $\mathcal{L}_{D'}^{OC}(\epsilon)$ below, where $\sigma^{j'}$ is the new support corresponding to the j^{th} organizational-level itemset, and D' is the reduced version of dataset *D* after eliminating transactions that are already marked for sanitization as a result of the solutions to $\mathcal{L}_{D_p}^{OC}(\epsilon)$.

$$\mathcal{L}_{D'}^{OC}(\epsilon) \quad \min \sum_{i \in D'} x_i$$
$$\text{s.t.} \sum_{i \in D'} a_{ij} x_i \geq \left(\sigma^{j'} - \sigma_h^j + 1\right) \forall j \in F^S$$
$$x_i \in \{0,1\} \quad \forall i \in D'$$

As the partition-level problems are forward-looking, the selected solution reduces the right-hand sides of the organizational-level constraints as much as possible while maintaining optimality of the subproblems $\mathcal{L}_{D_p}^{OC}(\epsilon)$. It is likely therefore, that only a few additional transactions will need to be sanitized when we solve $\mathcal{L}_{D'}^{OC}(\epsilon)$. As noted earlier, $\frac{1}{|D_p| \times |F^S|}$ is a valid value for ϵ. As 5 transactions in D_p support sensitive itemsets in our example, $\frac{1}{|D_p| \times |F^S|} = \frac{1}{5 \times 3} = \frac{1}{15}$. The modified partitions-level formulations are $\mathcal{L}_{D_1}^{OC}(\epsilon)$ and $\mathcal{L}_{D_2}^{OC}(\epsilon)$ below.

$\mathcal{L}_{D_1}^{OC}(\epsilon)$:

$$\min \quad \frac{13}{15}x_1 + x_2 + \frac{12}{15}x_3 + \frac{13}{15}x_4 + \frac{13}{15}x_5$$
$$\text{s.t.} \quad x_1 + x_4 \geq 1 \quad (s_1^1)$$
$$x_2 + x_5 \geq 1 \quad (s_2^1)$$
$$x_1, x_2, x_3, x_4, x_5 \in \{0,1\}$$

$\mathcal{L}_{D_2}^{OC}(\epsilon)$:

$$\min \quad \frac{12}{15}x_{11} + x_{12} + \frac{13}{15}x_{13} + \frac{14}{15}x_{14} + \frac{13}{15}x_{15}$$
$$\text{s.t.} \quad x_{12} + x_{13} \geq 1 \quad (s_1^2)$$
$$x_{11} + x_{15} \geq 1 \quad (s_2^2)$$
$$x_{11}, x_{12}, x_{13}, x_{14}, x_{15} \in \{0,1\}$$

One optimal solution to $\mathcal{L}_{D_1}^{OC}(\epsilon)$ is $x_4 = x_5 = 1$, while $x_{11} = x_{13} = 1$ is an optimal solution to $\mathcal{L}_{D_2}^{OC}(\epsilon)$. Substituting these solutions into FIH_D results in the reduced formulation $\mathcal{L}_{D'}^{OC}(\epsilon)$ below.

$$\mathcal{L}_{D'}^{OC}(\epsilon) \quad \min \; x_1 + x_2 + x_3 + x_{12} + x_{14} + x_{15}$$
$$\text{s.t.} \; x_1 + x_3 + x_{14} + x_{15} \geq 1 \quad (S_1)$$
$$x_1, x_2, x_3, x_{12}, x_{14}, x_{15} \in \{0,1\}$$

$\mathcal{L}_{D'}^{OC}(\epsilon)$ can be solved by setting exactly one variable – say x_1 – to 1. Therefore, solution via *PartitionsFirst* results in *five* transactions – t_1, t_4, t_5, t_{11}, and t_{13} – being marked for sanitization.

One advantage of *PartitionsFirst* is that the optimal solution of each partition when tackled in isolation (i.e., $\mathcal{L}_{D_p}^{OC}(0)$) is included in its solution. Therefore, if the reduced problem $\mathcal{L}_{D'}^{OC}(\epsilon)$ is significantly smaller than FIH_D with just the organizational-level constraints, the solution to *PartitionsFirst* is likely to be close to the optimal solution of FIH_D. Another advantage is that it solves the problems associated with each partition and the consolidated dataset exactly once.

Note that the complexity of the problem remains NP-Hard. By recognizing and exploiting the underlying problem structure, we decompose the problem into a collection of much smaller problems, which while still NP-Hard, are much easier to solve from a practical perspective.

4.1.2 Procedure *OrgFirst*

Another Lagrangian dual can be obtained by dualizing the partition-level constraints (PC) instead of the organizational level ones. If we associate multipliers $\mu_j^p \geq 0$ with each constraint j of partition p in constraint set (PC) and dualize these constraints, we get the Lagrangian dual problem $\mathcal{L}^{PC}(\mu)$ below. As with $\mathcal{L}^{OC}(\lambda)$, $\mathcal{L}^{PC}(\mu)$ is a lower bound on the optimal objective function value of FIH_D for any $\mu_j^p \geq 0$.

$$\begin{aligned}
\mathcal{L}^{PC}(\mu) \quad \min \quad & \sum_{i \in D} x_i \quad + \quad \sum_{p \in P} \sum_{j \in F_p^S} \left(\mu_j^p \left(\left(\sigma_p^j - \sigma_{hp}^j + 1 \right) - \sum_{i \epsilon D_p} a_{ij}^p x_i \right) \right) & \\
\text{s.t.} \quad & \sum_{i \in D} a_{ij} x_i \quad \geq \quad \left(\sigma^j - \sigma_h^j + 1 \right) & \forall j \in F^S \quad (OC) \\
& x_i \quad \in \quad \{0,1\} & \forall i \in D
\end{aligned}$$

Once again, we can fix the value of the dual multipliers μ_j^p to a very small value $\epsilon > 0$ to get $\mathcal{L}^{PC}(\epsilon)$; the corresponding objective function is

$$\min \sum_{i \in D} \left(1 - \epsilon \left(\sum_{p \in P} \sum_{j \in F^S} a_{ij}^p \right) \right) x_i + \epsilon \left(\sum_{p \in P} \sum_{j \in F_p^S} \left(\sigma_p^j - \sigma_{hp}^j + 1 \right) \right).$$

The second term is a constant, and can be ignored when solving $\mathcal{L}^{PC}(\epsilon)$. As before, we select the value of ϵ such that the net impact on the optimal solution is less than 1; this ensures that the solution to $\mathcal{L}^{PC}(\epsilon)$ is chosen from one of the multiple optimal solutions of the organizational problem (i.e., of $\mathcal{L}^{PC}(0)$, the problem where the partition constraints are ignored). Using similar arguments as in *partitionsFirst*, we can identify one possible value for ϵ as $\frac{1}{|D| \times \max_p |F_p^S|}$, and we have used $\epsilon = \frac{1}{|D| \times \max_p |F_p^S|} = \frac{1}{10 \times 2} = \frac{1}{20}$ in our example (as 10 transactions in D support sensitive itemsets). Just as with $\mathcal{L}^{OC}(\epsilon)$, if the optimal solution to $\mathcal{L}^{PC}(\epsilon)$ is feasible to FIH_D, it is optimal to FIH_D; this is established in Proposition 2.

Proposition 2: If the optimal solution to the organizational problem in *orgFirst* results in a feasible solution to the original integer program FIH_D, this solution is optimal to FIH_D.

Proof: Let x_o^* be an optimal solution to the organizational level problem for a valid value of ϵ. ϵ is chosen such that x_o^* is one of the multiple optima of $\mathcal{L}^{PC}(0)$. So x_o^* is optimal to $\mathcal{L}^{OC}(0)$.

As x_o^* is feasible to FIH_D, we know that x_o^* satisfies all the partition level constraints (PC) (for notational convenience, we will refer to these as $Bx \geq d$). This implies that $\mathcal{L}^{PC}(\epsilon) = cx_o^* + \epsilon(d - Bx_o^*) \leq cx_o^*$. But as x_o^* is optimal to $\mathcal{L}^{PC}(0)$, we know that $\mathcal{L}^{PC}(0) = cx_o^* + 0(d - Bx_o^*) = cx_o^*$, and therefore x_o^* is optimal to FIH_D. □

Together, propositions 1 and 2 imply that that if the union of the optimal solutions to the partition problems in *partitionsFirst* is identical to the optimal solution to the organizational problem in *orgFirst*, the solution is optimal to FIH_D.

If the optimal solution to $\mathcal{L}^{PC}(\epsilon)$ is not feasible to FIH_D, we incorporate the solution of $\mathcal{L}^{PC}(\epsilon)$ into FIH_D (by fixing at 1 all variables set to 1 by the solution of $\mathcal{L}^{PC}(\epsilon)$) to obtain a reduced formulation $\mathcal{L}_{D_p'}^{PC}(\epsilon)$ for each partition. The reduced formulations for the partitions $\mathcal{L}_{D_p'}^{PC}(\epsilon)$ are then solved separately, to arrive at a feasible solution to FIH_D. Formulation $\mathcal{L}^{PC}(\epsilon)$ for our example is below.

$$
\begin{aligned}
\mathcal{L}^{PC}(\epsilon)\ \min\ & \tfrac{19}{20}x_1+\tfrac{19}{20}x_2+x_3+\tfrac{19}{20}x_4+\tfrac{19}{20}x_5+\tfrac{19}{20}x_{11}+\tfrac{19}{20}x_{12}+\tfrac{19}{20}x_{13}+x_{14}+\tfrac{19}{20}x_{15} \\
\text{s.t.}\ & x_1 + x_3 + x_{11} + x_{13} + x_{14} + x_{15} \geq 3 \quad (S_1) \\
& x_1 + x_3 + x_4 + x_5 + x_{11} \geq 2 \quad (S_2) \\
& x_3 + x_4 + x_5 + x_{11} + x_{13} + x_{15} \geq 3 \quad (S_3) \\
& x_1, x_2, x_3, x_4, x_5, x_{11}, x_{12}, x_{13}, x_{14}, x_{15} \in \{0,1\}
\end{aligned}
$$

One optimal solution is $x_3 = x_{11} = x_{15} = 1$. Incorporating this solution into FIH_D results in the reduced formulations $\mathcal{L}_{D_1'}^{PC}(\epsilon)$ and $\mathcal{L}_{D_2'}^{PC}(\epsilon)$ below.

$$
\begin{aligned}
\mathcal{L}_{D_1'}^{PC}(\epsilon)\ \min\ & x_1+x_2+x_4+x_5 \\
\text{s.t.}\ & x_1 + x_4 \geq 1 \quad (s_1^1) \\
& x_2 + x_5 \geq 1 \quad (s_2^1) \\
& x_1, x_2, x_4, x_5 \in \{0,1\}
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{L}_{D_2'}^{PC}(\epsilon)\ \min\ & x_{12}+x_{13}+x_{14} \\
\text{s.t.}\ & x_{12}+x_{13} \geq 1 \quad (s_1^2) \\
& x_{12}, x_{13}, x_{14} \in \{0,1\}
\end{aligned}
$$

One set of optimal solutions for $\mathcal{L}_{D_1'}^{PC}(\epsilon)$ and $\mathcal{L}_{D_2'}^{PC}(\epsilon)$ is $x_1 = x_2 = 1$, and $x_{13} = 1$ respectively. Therefore, *OrgFirst* results in the sanitization of six transactions (and is suboptimal, as was *PartitionsFirst*).

This procedure has the advantage that the optimal solution of the organizational-level problem where the partition constraints are ignored (i.e., of $\mathcal{L}^{PC}(0)$) are guaranteed to be included in the solution of *OrgFirst*. If the reduced problems $\mathcal{L}_{D_p'}^{PC}(\epsilon)$ of the partitions are significantly smaller than the original ones, the solution of *OrgFirst* will be close to the optimal solution of FIH_D. As with *PartitionsFirst*, *OrgFirst* also solves the problems associated with each partition and the consolidated dataset exactly once.

4.2 *Ensemble* - Step 2

PartitionsFirst can be expected to perform well when the bulk of the transactions identified in the solution of FIH_D appear in the solutions for the partitions, i.e., when most of the transactions identified for sanitization in this procedure are obtained before solving $\mathcal{L}_{D'}^{OC}(\epsilon)$. This is likely when the right-hand sides of the partition-related constraints are large in a relative sense. On the other hand, *OrgFirst* can be expected to perform well when the majority of the transactions identified in the solution of FIH_D appear in the solution for $\mathcal{L}^{PC}(\epsilon)$; this is likely to happen when the number of transactions to sanitize based on the solution for $\mathcal{L}^{PC}(\epsilon)$ is large and relatively evenly spread out throughout the partitions. Ideally, it would be useful to have a procedure that exploits the features of both approaches, and in this section, we present a procedure that does exactly that.

It is reasonable to expect that the transactions that have not been identified for sanitization by either *PartitionsFirst* or *OrgFirst* are less likely to feature in the optimal solution to FIH_D. Therefore, in this step of the *Ensemble*, we modify FIH_D by eliminating all variables that are set to zero by both *PartitionsFirst* and *OrgFirst*. The resulting, and often significantly smaller, reduced problem FIH_D^r (shown below) is then solved to identify a feasible solution to FIH_D.

$$
\begin{aligned}
\text{FIH}_D^r \quad \min \quad & \sum_{i \in D^r} x_i & & & \\
\text{s.t.} \quad & \sum_{i \in D^r} a_{ij} x_i & \geq & \left(\sigma^j - \sigma_h^j + 1\right) & \forall j \in F^S \qquad (OC^r) \\
& \sum_{i \in D_p^r} a_{ij}^p x_i & \geq & \left(\sigma_p^j - \sigma_{hp}^j + 1\right) & \forall j \in F_p^S, \forall p \in P \quad (PC^r) \\
& x_i & \in & \{0,1\} & \forall i \in D^r
\end{aligned}
$$

Here, D^r corresponds to the reduced version of D, and D_p^r to the reduced version of D_p, obtained by eliminating all transactions that were not identified for sanitization by either *PartitionsFirst* or *OrgFirst*. The right-hand sides of the constraints are unaffected as only the x_i's set to zero by both *PartitionsFirst* and *OrgFirst* have been eliminated. The reduced problem can be viewed as the original problem with many of the variables set to 0 – i.e., it has the same structure as the original problem. However, if *PartitionsFirst* and *OrgFirst* are run again, the solutions will not change, as the eliminated variables were set to 0 by both in the original run.

Proposition 3 identifies another condition where the solution from *Ensemble* will be optimal.

Proposition 3: If no transaction supports both partition-level and organizational-level itemsets, then the solutions from *partitionsFirst*, *orgFirst*, and *Ensemble* are optimal to FIH_D.

Proof: If no transaction supports both partition-level and organizational-level itemsets, there are no columns connecting the partition-level constraints to the organization-level ones. Consequently, the partition-level problems have to be solved independently from the organizational level one, and the union of the solutions from the partition-level problems and the organizational-level one is optimal to FIH_D. □

If a problem satisfies any of the conditions in Propositions 1 – 3 or Corollary 1, both FIH_D and FIH_D^r will mark the same number of transactions for sanitization. Using a valid non-zero ϵ increases the chances of identifying such situations – for example, the partition problems in *partitionsFirst* have a higher chance of resulting in a feasible solution to FIH_D as using a positive ϵ will selectively choose transactions that also reduce the supports of the organizational-level sensitive itemsets.

Returning to our example, the variables set to zero by both *PartitionsFirst* and *OrgFirst* are x_7 and x_9. Therefore, we remove these variables from FIH_D to obtain the reduced formulation FIH_D^r below.

$$
\begin{array}{lll}
\text{FIH}_D^r & \min & x_1+x_2+x_3+x_4+x_5+x_{11}+x_{13}+x_{15} \\
& \text{s.t.} & x_1 \quad +x_3 \quad +x_{11}+x_{13}+x_{15} \geq 3 \quad (S_1^r) \\
& & x_1 \quad +x_3+x_4+x_5+x_{11} \geq 2 \quad (S_2^r) \\
& & x_3+x_4+x_5+x_{11}+x_{13}+x_{15} \geq 3 \quad (S_3^r) \\
& & x_1 \quad +x_4 \geq 1 \quad (s_1^{1r}) \\
& & x_2 \quad +x_5 \geq 1 \quad (s_2^{1r}) \\
& & x_{13} \geq 1 \quad (s_1^{2r}) \\
& & x_{11} \quad +x_{15} \geq 1 \quad (s_2^{2r}) \\
& & x_1 \, , \, x_2 \, , \, x_3 \, , \, x_4 \, , \, x_5 \, , \, x_{11} \, , \, x_{13} \, , \, x_{15} \in \{0,1\}
\end{array}
$$

The optimal solution to FIH_D^r is to mark four transactions (t_1, t_5 t_{11}, and t_{13}) for sanitization. This is better than the solutions from both *PartitionsFirst* and *OrgFirst*. In fact, the ensemble approach terminates with the optimal solution of FIH_D in this example.

As this procedure eliminates variables set to zero by both *PartitionsFirst* and *OrgFirst*, the search for solutions is restricted to a space that likely contains the optimal one(s). While the size of the formulation in Step 2 need not be smaller than the size of the corresponding optimal formulation, it is likely to be much smaller in most instances (this was the case in all our experiments). Solving the resulting formulation will be quick when the number of variables eliminated is large. As Step 2 is solved after *PartitionsFirst* and *OrgFirst*, the solutions of those heuristics are available when Step 2 starts. The better of these two solutions serves as an upper bound that can speed up solution by reducing the search space further. As the solutions from *PartitionsFirst* and *OrgFirst* are feasible to the formulation being solved in Step 2, the solution of the *Ensemble* has to be at least as good as those solutions; in all our experiments, they have been strictly better.

As is the case in many situations where Lagrangian relaxation is used to solve integer programs, the complexity of the problem remains NP-Hard (for example, in the basic version of the cutting-stock problem, the sub-problem being solved is a knapsack problem, which is NP-hard). The contributions in most of these situations is a practical, rather than a theoretical one. Recognizing and exploiting underlying problem structure allows for the solution of an extremely difficult problem (from a practical perspective) by solving a series of much smaller problems, which while theoretically just as difficult, are easier from a practical

standpoint. Our problem falls into this category – the subproblems, while NP-Hard, are significantly smaller than the original problem, making them easier to solve (practically). For instance, a problem involving 500 million transactions and 200 sensitive itemsets in each partition and at the organizational level results in a problem that has 100,200 constraints, and (up to) 500 million variables. Each problem in *partitionsFirst* on the other hand, involves 200 constraints and (up to) 1 million variables. While each of these remains NP-Hard, the partition problems are much easier to solve from a practical perspective.

4.3 *A Discussion on Average-Time Complexity*

In this section, we investigate the average-time complexity of *Ensemble*, which provides some insight into what could be expected if the conditions of the analysis hold. We leverage the result of Lifschitz and Pittel (1983), who have shown that the average-time complexity of set-covering problems is $n^{O(log(n))}$, when a uniform probability distribution is assumed on the set of inputs of size *n*. For the set-covering problem, the inputs are the elements of the set that needs to be covered with subsets containing one or more of these elements. In the context of our problem, the sensitive itemsets are the elements that need to be covered by transactions that represent subsets containing sensitive itemsets. Consequently, *n* in our context is the number of sensitive itemsets. Based on this result, the average time complexity of a partition would be $|F_p^S|^{O(log(|F_p^S|))}$ for partition *p*; if the number of sensitive itemsets in each of *m* partitions and at the organizational level is the same (and equal to $|F_p^S|$), this implies that the average-time complexities of *partitionsFirst* and *orgFirst* are $m\left(|F_p^S|^{O(log(|F_p^S|))}\right)$. This compares to $\left((m|F_p^S|)^{O(log(m|F_p^S|))}\right)$ for FIH_D. Since $\left((m|F_p^S|)^{O(log(m|F_p^S|))}\right) = \left((m)^{O(log(m|F_p^S|))} \times \left((|F_p^S|)^{O(log(m|F_p^S|))}\right)\right) = \left((m)^{O(log(m|F_p^S|))} \times \left((|F_p^S|)^{O(log(m))}\right) \times \left((|F_p^S|)^{O(log(|F_p^S|))}\right)\right)$, and since $m \ll (m)^{O(log(m|F_p^S|))} \times \left((|F_p^S|)^{O(log(m))}\right)$, *partitionsFirst* and *orgFirst* are more efficient than FIH_D on average. Part 2 of the ensemble is harder to compare on this dimension, as we are explicitly forcing many variables to 0, and the average-time

complexity argument of Lifsehitz and Pittel (1983) relies on imposing a uniform probability distribution on all possible subsets of the set of sensitive itemsets.

5 COMPUTATIONAL EXPERIMENTS: DATA QUALITY

We evaluate the quality of solutions obtained through the ensemble approach. Experiments are conducted on real and simulated datasets of various sizes, the largest of which has 500 million transactions. As mentioned earlier, all existing approaches consider sensitive itemsets only at the organizational level. In order to identify potential benchmarks, we tried all the approaches available in the Frequent Itemset Hiding Toolbox (Kagklis et al. 2018). Menon et al.'s (2005) formulation, adjusted for the existence of constraints at the partition-level, is formulation FIH_D, and we have solved FIH_D directly in CPLEX to obtain optimal solutions where possible (i.e., we have not resorted to the toolkit to solve it). In addition, we tried the approaches of Verykios et al. (2004), Cheng at al. (2016) and Lin at al. (2016). As the approach of Stavropoulos et al. (2016) reduces to that of Menon et al. (2005) when the objective is to maximize accuracy, we know that our optimal formulation is smaller and easier to solve than the one presented there. None of the approaches from the Toolkit were able to solve single-partition problems involving databases with 5 million transactions and 30 constraints. The approaches of Cheng et al. (2016) and Lin et al. (2016) did not hide a significant fraction of the sensitive itemsets[1], and leaves the data owner vulnerable as a result. One of the approaches in Verykios et al. (2004) (Algorithm 2.b) can be applied to distributed datasets even though it was developed to hide sensitive itemsets at the organizational level. Therefore, we have used that as a basis for comparison. This algorithm first sorts the sensitive itemsets in descending order of size and support. For each sensitive itemset in the sorted list, it sorts the transactions supporting it in increasing order of size. It then removes the item with the highest support from the smallest transactions until the support of the itemset drops below the hiding threshold.

[1] Specifically, Lin et al. (2016) left 72.74% and 69.83% of sensitive itemsets exposed across all the experiments in *Retail* and *BMS-POS* respectively. The corresponding numbers for Cheng et al. (2016) were 83.99% and 94.28%.

5.1 Data

We conduct experiments on real and synthetic datasets. The real datasets – *Retail* and *BMS-POS* – are obtained from the Frequent Itemset Mining Implementations Dataset repository (fimi.ua.ac.be/data/). *Retail* is collected from a Belgian retail store (Brijs et al. 1999), and has 88,162 transactions across 16,470 items, with an average transaction length of 10.3 items. *BMS-POS* is from a large electronics retailer (Zheng et al. 2001); it has 515,597 transactions, 1,657 items and an average transaction length of 6.5 items. Each of these datasets were randomly divided into ten partitions of equal sizes, with the 2 and 5-partition versions of these datasets being created by combining the appropriate number of partitions from that dataset.

As large real world distributed transactional datasets are not publicly available, the larger datasets used in our experiments are generated synthetically using the IBM Quest synthetic data generator (Agrawal and Srikant 1994). Zheng et al. (2001) observe that data generated by the IBM Quest data generator is more right skewed in its transaction size distribution than real-world datasets. This means that the formulations based on synthetic datasets have denser constraint matrices, making our results conservative. Each partition in the synthetic datasets (*10m*, *50m*, *200m* and *500m*) has 1 million transactions, 10,000 items and an average transaction length of 20. *10m*, *50m*, *200m* and *500m* have 10, 50, 200 and 500 partitions respectively, resulting in datasets with 10, 50, 200 and 500 million transactions. We also conduct experiments on two additional 500 million transaction datasets, to see the impact of changes in average transaction length and the number of items: *500m2* with 10,000 items and an average transaction length of 10, and *500m3* with 100,000 items and an average transaction length of 10. The frequent itemsets are mined using Apriori (Agrawal and Srikant 1994), as implemented by Bodon (2003). All integer programs are solved in CPLEX 12.6.1 (IBM 2014) on a personal computer with quad-core processors (i7, 3.60 GHz) and 64 GB of RAM.

5.2 Computational Experiments and Results

In our first set of experiments, we compare the accuracies of the ensemble approach to the corresponding values from the optimal approach and the approach of Verykios et al. (2004). A relative mining threshold

of 0.2% is used, with the hiding threshold being set equal to the mining threshold. The experiments on the real datasets (*Retail*, *BMS-POS*) involve two, five, and ten partitions, with 30 sensitive itemsets in each partition and at the organizational level (for a total of 90, 180, and 330 sensitive itemsets, respectively). The experiments on the synthetic datasets (*10m*, *50m*, *200m* and *500m*) involve 200 sensitive itemsets in each partition and also at the organizational level. Therefore, the largest experiment involves 500 million transactions and 100,200 sensitive itemsets. All sensitive itemsets are selected randomly from the frequent itemsets of the corresponding datasets. The results of these experiments are in Table 3.

Table 3: Comparing the *Ensemble* with the Alternative and the Optimal

Procedures		Real Datasets						Synthetic Datasets			
		Retail			*BMS-POS*			*10m*	*50m*	*200m*	*500m*
		# of Partitions						# of Partitions			
		2	*5*	*10*	*2*	*5*	*10*	*10*	*50*	*200*	*500*
Optimal # of Transactions Sanitized		1,609	5,537	8,009	4,759	20,052	35,158	4,406,115	20,754,518	88,801,837	120,478,438
Verykios et al. (2004)	***Transactions Sanitized***	2,137	6,821	10,430	8,156	28,647	49,906	–	–	–	–
	Gap	528 (32.81%)	1284 (23.20%)	2422 (30.24%)	3397 (71.38%)	8595 (42.86%)	14748 (41.95%)				
Ensemble	***Transactions Sanitized***	1,610	5,537	8,013	4,768	20,061	35,160	4,414,022	20,788,028	88,942,239	120,609,576
	Gap	1 (0.06%)	0 (0.00%)	4 (0.05%)	9 (0.19%)	9 (0.04%)	2 (0.01%)	7,907 (0.18%)	33,510 (0.16%)	140,402 (0.16%)	131,138 (0.11%)
	Solution Time (sec)	0.02	0.05	0.09	0.09	0.24	0.44	69.22	591.92	1,723.09	4,872.14

These results show that the solutions identified by the ensemble approach are extremely close to the optimal ones. The solutions from the ensemble approach are off from the optimal by an average of only 1.66 transactions (i.e., the average gap relative to the optimal solution is 0.04%) across the three experiments on *Retail*; the corresponding numbers are 6.67 and 0.08% for the experiments on *BMS-POS*. Not only are the average gaps resulting from the approach of Verykios et al. (2004) much larger – 28.75% across the three experiments on *Retail* and 52.06% over the experiments on *BMS-POS* – it was not able to

solve any of the problems on the synthetic databases. In contrast, *Ensemble* was able to solve all the problems in reasonable time, with all experiments on the real datasets taking less than half a second to complete. These results indicate that the approach is quite scalable.

Table 4 provides additional information on the impact of the ensemble approach on the resulting problem size. As this is an issue of relevance primarily to large problems, we have focused on the four largest ones here, where there were 200 sensitive itemsets in each partition and at the organization level.

Table 4: Number of Variables in Step 2 of the *Ensemble* Relative to Optimal

		10m	**50m**	**200m**	**500m**
# of Partitions		10	50	200	500
# of Transactions Sanitized	(A)	4,414,022	20,788,028	88,942,239	120,609,576
# of Transactions Supporting Sensitive Itemsets (# of Variables in Complete Formulation)	(B)	7,250,185	35,308,803	148,760,766	250,317,080
# of "Excess" Variables in Optimal Formulation	$(C = B - A)$	2,836,163	14,520,775	59,818,527	129,707,504
# of Variables in Step 2 of Ensemble	(D)	5,409,906	25,418,921	107,833,821	146,473,483
# of "Excess" Variables in Step 2 of Ensemble	$(E = D - A)$	995,884	4,630,893	18,891,582	25,863,907
% Reduction in "Excess" Variables	$\left(1 - \frac{E}{C}\right)\%$	64.89%	68.11%	68.42%	80.06%

The number of transactions that do not support any sensitive itemsets are 2,749,815 (*10m*), 14,691,197 (*50m*), 51,239,234 (*200m*), and 249,682,920 (*500m*), respectively. The number of "excess" variables essentially represent the number of variables that are set to 0 in the solution, representing the transactions that are not marked for sanitization. This represents "unnecessary baggage" – the set of variables that potentially could have been eliminated before-hand (in hind-sight). The number of such "excess" variables reduce substantially in Step 2 of the *Ensemble* relative to the complete formulation FIH_D, as seen in the last two rows of Table 4. Viewed another way, the fraction of excess variables in the complete formulation FIH_D relative to the number of transactions sanitized are 64.25% (*10m*), 69.85% (*50m*), 67.26% (*200m*) and 107.54% (*500m*), while the corresponding numbers are 22.56%, 22.28%, 21.24% and 21.44% in Step 2 of the *Ensemble*.

In our next set of experiments, we focus on datasets with 500 million transactions to investigate the impact of various problem parameters (the number of items, the average transaction length, the hiding threshold, and the number of sensitive itemsets at the organizational level) on solvability and the quality of solutions. Two additional 500 million transaction datasets are used in these experiments, along with *500m*. *500m2* has the same characteristics as *500m* (10,000 items and 500 partitions of 1 million transactions each), except that the average transaction length is 10. *500m3* is similar to *500m2* in all respects (average transaction length of 10 and 500 partitions of 1 million transactions each), except that it has 100,000 items. The results of these experiments are presented in Table 5.

Table 5: Sensitivity Analysis

		500m			**500m2**	**500m3**
# of Items		**10,000**			**10,000**	**100,000**
Average Transaction Length		**20**			**10**	
Mining/Hiding Thresholds		**0.20%**	**0.20%**	**0.10%**	**0.10%**	
Sensitive Itemsets at Org Level		**200**	**1,000**	**200**	**200**	
Transactions Sanitized	***Ensemble***	120,609,576	137,497,118	150,888,910	144,466,434	148,088,246
	Optimal	120,478,438	–	150,844,862	144,455,751	148,080,615
Solution Times		4,872.14	38,712.63	8,884.25	1,859.94	2,207.30

The gaps between the *Ensemble* and optimal solutions remain low, at an average of 0.04% across the four problems where optimal solutions were obtained. The optimal solution could not be obtained for the problem with 1,000 sensitive itemsets at the organizational level. While not guaranteed, Step 2 of the *Ensemble* improved on the solutions from *OrgFirst* and *PartitionsFirst* in every problem we solved. The extent of improvement varied; as an example, the *OrgFirst* and *PartitionsFirst* solutions for the problem with 1,000 sensitive itemsets at the organizational level were 149,701,253 and 144,503,010 respectively. The *Ensemble* solution of 137,497,118 therefore, represents improvements of 8.88% and 5.10% respectively over *OrgFirst* and *PartitionsFirst*. Given the extremely small gaps between the *Ensemble* solution and the optimal one, this indicates that the improvements resulting from Step 2 of the *Ensemble* can be significant.

As expected, the number of transactions that need to be sanitized increases with the number of sensitive itemsets at the organizational level. Specifically, a five-fold increase in the number of sensitive itemsets at

the organization level (from 200 to 1,000) results in 16,887,542 (14%) more transactions being marked for sanitization (as shown by the first two columns under *500m*). This was reflected in the time needed for solution – the problem involving 1,000 sensitive itemsets at the organization level took 8 times as long to solve as the original 200-sensitive itemset version of *500m*. There are half as many items per transaction on average in *500m2* (column 6 of Table 5) and *500m3* (column 7 of Table 5) than in *500m* (column 5 of Table 5). This implies that transactions are likely to support fewer sensitive itemsets on average. In general, this leads to sparser constraint matrices, which in turn leads to faster solution. Given that these problems do not need to be solved in real time, these times show that *Ensemble* is practical even on very large datasets.

Hiding at a lower threshold level also results in more transactions being marked for sanitization. The experiments on all three datasets (*500m, 500m2* and *500m3*) result in over 140 million transactions being marked for sanitization. In particular, halving the hiding threshold from 0.2% to 0.1% results in 30,279,334 (25.11%) more transactions being marked by the *Ensemble* on *500m* (as shown by the first and last columns under *500m*). This too is expected, as a hiding threshold of 0.1% requires the supports of the organizational level sensitive itemsets to be brought below 500,000 for a 500 million transaction dataset, rather than 1,000,000 when the threshold is 0.2% (and to 1,000 rather than 2,000 in each 1-million transaction partition).

The impact of a lower average transaction length is to reduce the number of transactions marked for sanitization (comparing the results from *500m2* to the last column of *500m*). This is expected as well, as a lower average transaction length translates to a sparser dataset when the number of items is unchanged. The supports of itemsets in sparser datasets tend to be lower in general, and fewer transactions will need to be sanitized to bring the supports of sensitive itemsets below the hiding threshold. The impact of increasing the number of items from 10,000 to 100,000 is similar (comparing columns *500m2* and *500m3*), for the same reason – more items imply a sparser dataset when the average transaction length is the same. The *Ensemble* performs better on sparser datasets, based on the four problems for which optimal solutions are available – the average gap was 0.006% on the two problems where the average transaction length was 10,

while the corresponding value for the two problems was 0.07% where the average transaction length was 20. As real datasets tend to be sparse, this suggests that the *Ensemble* will perform well on large real datasets.

6 COMPUTATIONAL EXPERIMENTS: RECOMMENDATION QUALITY

One common use of transactional data is to make recommendations to customers. Hiding sensitive itemsets reduces the quality of the dataset, and it is important to evaluate the impact of this process in an actual context where transactional data is used. Therefore, we compare the predictions from a recommender system that uses a dataset sanitized using the *Ensemble*, with that of a similar system that uses the original (unsanitized) dataset.

6.1 Sanitization and Recommender System

The *Ensemble* only identifies transactions for sanitization. Once these transactions are identified, the sanitization process needs to ensure that a transaction that has been marked for sanitization will not support any sensitive itemset after it has been sanitized. This is usually done by removing items from the transaction – for example, removing all but one of the items in a transaction will ensure that the transaction will not support any sensitive itemset. As this could result in the removal of many more items than necessary, we follow the approach from Menon and Sarkar (2007). Here, the item to be removed is the one with the highest ratio of the number of sensitive itemsets involving that item supported by the particular transaction, to the number of non-sensitive itemsets involving that item supported by the transaction. If the transaction continues to support other sensitive itemsets after this item is removed, another item is removed using the same logic. This process is continued until the transaction no longer supports sensitive itemsets.

Zaine (2002) proposed an approach that recommends items based on association rules with the highest confidence, from the set of rules whose antecedents are contained in the customer's current basket. We use this approach to compare the quality of recommendations made with rules mined from the sanitized and non-sanitized datasets.

6.2 Experiments and Results

We use *Retail*, *BMS-POS*, *10m* and *500m* in these experiments, with *Retail*, *BMS-POS* and *10m* having 10 partitions apiece, and *500m* having 500. The datasets are sanitized based on the results reported in Table 3. For the synthetic datasets (*10m* and *500m*), the complete original and sanitized versions were used as the training sets, and new datasets with 250,000 transactions per partition were generated (using the same parameters as the original ones) for testing. As recommendations should be made to customers based on the dataset that is local to them, we create separate training and testing datasets for each partition. As new datasets cannot be generated for the real datasets (*Retail* and *BMS-POS*), the original datasets are divided into training and testing sets, with 80% of the transactions going into the training set and rest going into the testing set. The sanitization method discussed earlier is applied to every transaction (in the training sets) identified for sanitization by the *Ensemble*. Association rules are mined from both versions of the datasets using a support threshold of 0.2%; we have tried five different confidence thresholds – 20%, 30%, 40%, 50%, and 60% – in our experiments.

The transactions in the testing set are used to create baskets. Half the items in each transaction are placed randomly into the basket, and recommendations made based on them. A recommendation is considered successful if the recommended item is present in the rest of the transaction, indicating that the recommended item was selected by the customer. Recommendations are made for all test transactions, and the percentages of successful recommendations made – i.e., the precisions of the recommender systems – are reported in Table 6.

Table 6: Comparing Recommendation System Precision

Real Datasets				Synthetic Datasets			
Dataset	**Confidence**	**Prediction Precision (%)**		**Dataset**	**Confidence**	**Prediction Precision (%)**	
		Sanitized	**Original**			**Sanitized**	**Original**
Retail	20%	48.05	48.45	*10m*	20%	60.15	60.27
	30%	48.89	49.27		30%	63.81	63.94
	40%	49.29	49.66		40%	68.78	68.91
	50%	49.85	50.20		50%	73.81	73.95
	60%	52.93	53.29		60%	76.78	76.92
BMS – POS	20%	49.41	49.51	*500m*	20%	66.55	66.70
	30%	51.00	51.08		30%	69.82	69.98
	40%	53.71	53.78		40%	73.85	74.02
	50%	56.09	56.17		50%	78.08	78.25
	60%	57.94	58.01		60%	80.46	80.63

The results show that while recommendations made using the sanitized datasets have lower precision than recommendations made using the original datasets, the differences are negligible in most cases. In fact, none of the differences are statistically significant at the 90% level. These results demonstrate that hiding all sensitive itemsets using the *Ensemble* has minimal impact on the quality of recommendations made.

7 CONCLUSIONS

Retailers recognize the potential benefits of sharing transactional data with partners in the supply chain for mutual benefit. However, many are still hesitant for fear of revealing sensitive information, and there is evidence that the extent of sharing would be greater if information considered sensitive by the retailer could be hidden prior to sharing. While prior research has looked into how to hide sensitive itemsets before transactional data is shared with business partners, there has been no work done to address this problem when data comes from different regions. That however, is the reality – most large retailers recognize that region-specific differences in customer tastes affect purchases, and customize offerings based on local need. Such customization results in sensitive information that needs to be hidden at the regional level, in addition to those at the organizational level. In this paper, we present an ensemble approach to hide sensitive itemsets in distributed datasets, thereby filling a key gap in the literature.

The distributed nature of the data makes the problem considerably more complex than the corresponding problem on a centralized dataset because each partition has sensitive information specific to

it, in addition to the sensitive information to be hidden at the organizational level. While this problem is conceptually similar to the non-distributed version of the problem, the fact that each partition can have distinct sensitive itemsets increases the problem size dramatically, making it significantly harder to solve using existing approaches. If the data owner cannot solve the problem, her decision essentially reduces to choosing between sharing and not sharing. This is a problem firms are currently dealing with, with many recognizing the risks and choosing not to share. Hiding sensitive information allows the data owner to benefit from the advantages of sharing risk free, which is why alleviating the need to worry about this trade-off provides the basis for all versions of the itemset hiding problem.

In the context of distributed data, we leverage the intuition behind Lagrangian relaxation, and propose an ensemble approach to solve this problem. Essentially, the procedure first solves two different relaxations of the problem; these relaxations are forward looking, as they try to choose variables that will help in the next step. The procedure then eliminates all variables set to zero by both the relaxations, and solves a substantially reduced version of the original problem.

We conduct computational experiments to investigate the impact of the proposed approach from two perspectives. The first set of experiments examines the effectiveness and scalability of the ensemble approach, while the second examines whether sensitive information can be hidden without significantly reducing the benefits of sharing. We find that the ensemble approach performs well, identifying optimal or near-optimal solutions in all our experiments where optimal solutions are available. It is also able to solve problems involving as many as 500 million transactions and over 100,000 sensitive itemsets. We also find that the quality of recommendations made based on datasets sanitized using the ensemble approach is comparable to the quality of recommendations made using the original (non-sanitized) version of the dataset. This establishes the practical viability of the approach – not only is it effective on datasets with hundreds of millions of transactions, the hiding of sensitive information has minimal impact on the quality of recommendations made. Thus, data owners using the ensemble approach can share transactional data knowing that sensitive information will not be revealed to the parties receiving the data. At the same time,

the ability to accomplish this with limited distortion will help recipients derive higher value from the data, which can benefit the retailers as well.

REFERENCES

Agrawal, R., and R. Srikant. 1994. "Fast Algorithms for Mining Association Rules in Large Databases," in *Proceedings of the 20th International Conference on Very Large Data Bases*, J. Bocca, M. Jarke, and C. Zaniolo (eds.), Santiago de Chile, September 12-15, pp. 487-499.

Alaimo, D. 2013. "CGT/RIS Retailer/Supplier Shared Data Study: A Supplement to Consumer Goods Technology and RIS News," Sponsored by *RSi Retail Solutions*, 2013.

Anand, K. and H. Mendelson. 1997. "Information and Organization for Horizontal Multimarket Coordination," *Management Science* 43(12) pp. 1609-1627.

Atallah, M., E. Bertino, A. Elmagarmid, M. Ibrahim and V. Verykios. 1999. "Disclosure Limitation of Sensitive Rules," *Proceedings of 1999 Workshop Knowledge Data Engineering Exchange (KDEX'99)*, IEEE Computer Society, Washington, D.C., pp. 45-52.

Bodon, F. 2003. "A Fast APRIORI Implementation," *IEEE ICDM Workshop on Frequent Itemset Mining Implementations (FIMI'03)*, Melbourne, Florida, USA, 2003.

Brijs T., G. Swinnen, K. Vanhoof and G. Wets. 1999. "The Use of Association Rules for Product Assortment Decisions: A Case Study," in: Proceedings of the Fifth International Conference on Knowledge Discovery and Data Mining, San Diego (USA), August 15-18, 1999, pp. 254-260.

Chandra, N. 2012. "Unraveling the Customer Mind," *Cognizant 20-20 Insights*, December, 2012.

Chen, Y., and M. Deng. 2015. "Information Sharing in a Manufacturer–Supplier Relationship: Suppliers' Incentive and Production Efficiency," *Production and Operations Management* 24(4) pp. 619-633.

Chen, Y., K. Tang, R. Shen and Y. Hu. 2005. "Market Basket Analysis in a Multiple Store Environment," *Decision Support Systems* (40:2), pp. 339-354.

Cheng, P., J. Roddick, S. Chu and C. Lin. 2016. "Privacy Preservation Through a Greedy, Distortion-based Rule-hiding Method," *Applied Intelligence* (44), pp. 295–306.

Computer Market Research. 2017. "9 Strategies That Will Encourage Distributors to Submit Channel POS Data," *Channel POS Articles, Computer Market Research*, (https://computermarketresearch.com/inspire-channel-pos-data-submission/, paragraph 23) July 13, 2017.

Croson, R. and K. Donohue. 2003. "Impact of POS Data Sharing on Supply Chain Management: An Experimental Study," *Production and Operations Management* 12(1) pp. 1-11.

Garfinkel, R., R. Gopal, and P. Goes. 2002. "Privacy Protection of Binary Confidential Data Against Deterministic, Stochastic, and Insider Threat," *Management Science* (48:6), pp. 749-764.

Gilman, H. 1987. "J. C. Penney Decentralizes Its Purchasing," *The Wall Street Journal*, May 08, 1987.

Gkoulalas-Divanis, A., and V. Verykios. 2006. "An Integer Programming Approach for Frequent Itemset Hiding," *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*, Arlington, VA, pp. 748-757.

GMA. 2009. "Retailer Direct Data Report," *Grocery Manufacturers Association*, July, 2009.

GXS. 2013. "Point of Sale Data Sharing – A Case Study in Standards Deviation," *GXS Market Perspectives*, June 2013.

Hawkins, G. 2012. "Will Big Data Kill All But the Biggest Retailers?," *The Promise and Challenge of Big Data*, A Harvard Business Review Insight Center Report, 2012.

Hayek, F. 1945. "The Use of Knowledge in Society," *American Economic Review* 35(4) 519-530.

Hong, T., C. Lin, K. Yang and S. Wang. 2013. "Using TF-IDF to Hide Sensitive Itemsets," *Applied Intelligence* (38) pp. 502-510.

IBM. 2014. *IBM ILOG CPLEX Optimization Studio CPLEX User's Manual* (Version 12, Release 6), IBM Corp., Armonk, NY.

Kagklis, V., V. Verykios, G. Tzimas and A. Tsakalidis. 2014. "An Integer Linear Programming Scheme to Sanitize Sensitive Frequent Itemsets." *International Conference on Tools with Artificial Intelligence (ICTAI 14)*, p Limassol, Cyprus, p. 771-775, November 2014.

Kagklis V., E. Stavropoulos and V. Verykios. 2018. "A Frequent Itemset Hiding Toolbox," Computing Research Repository (CoRR), February 2018. http://arxiv.org/abs/1802.10543.

Kantarcioglu, M. and C. Clifton. 2004. "Privacy-Preserving Distributed Mining of Association Rules on Horizontally Partitioned Data," *IEEE Transactions on Knowledge Data Engineering* (16:9), pp. 1026-1037.

Konzak, L. 2012. "Sharing Point-of-Sale Data: Challenges & Opportunities," *MDM Special Report*, Page 4, 2012.

Leloglu, E., B. Ayav and B. Ergenc. 2014. "Coefficient-Based Exact Approach for Frequent Itemset Hiding," *The Sixth International Conference on Information, Process, and Knowledge Management (eKNOW 2014)*, Barcelona, Spain, March 2014.

Lifschitz, V. and Pittel, B. 1983. "The Worst and the Most Probable Performance of a Class of Set-Covering Algorithms," *SIAM Journal on Computing* (12:2), pp. 329-346.

Lin, J., Q. Liu, P. Fournier-Viger, T. Hong, M. Voznak and J. Zhan, J. 2016. "A Sanitization Approach for Hiding Sensitive Itemsets Based on Particle Swarm Optimization," *Engineering Applications of Artificial Intelligence* (53), pp. 1-18.

Lin, C., T. Hong, K. Yang and L. Wang. 2015. "The GA-Based Algorithms for Optimizing Hiding Sensitive Itemsets through Transaction Deletion," *Applied Intelligence* (42), pp. 210-230.

Lobel, J., B. Bishop and V. Youshaei. 2017. "Realizing the Value of Supermarket POS Data," *SwiftIQ eBook* 2017 (http://www.swiftiq.com/ebooks/realizing-the-value-of-supermarket-pos-data).

Manthan. 2017. "Top 3 Insights That Suppliers Gain From Downstream Data," *Manthan Insights* (https://www.manthan.com/cpg-solutions/insights/504-top-3-insights-that-suppliers-gain-from-downstream-data).

Menon, S., S. Sarkar and S. Mukherjee. 2005. "Maximizing Accuracy of Shared Databases when Concealing Sensitive Patterns," *Information Systems Research* (16:3), pp. 256-270.

Menon, S., and S. Sarkar. 2007. "Minimizing Information Loss and Preserving Privacy," *Management Science* (53:1), pp. 101-116.

Menon, S. and S. Sarkar. 2016. "Privacy and Big Data: Scalable Approaches to Sanitize Large Transactional Databases for Sharing," *Management Information Systems Quarterly* (40:4), pp.963-981.

Monga, V. 2014. "The Big Mystery: What's Big Data Really Worth?," *The Wall Street Journal*, October 6, 2014.

Moustakides, G., and V. Verykios. 2006. "A Max-Min Approach for Hiding Frequent Itemsets," *Sixth IEEE International Conference on Data Mining - Workshops (ICDMW'06)*, pp. 502-506.

Munves, G. 2013. "Wake Up, Retailers! Make Money From Your Big Data," *Chain Store Age*, April 03, 2013 (https://www.chainstoreage.com/article/wake-retailers-make-money-your-big-data/)

Nielsen. 2016. "For The First-Time, Nielsen Opens CPG Data To Connected Partners On A Broad Scale," *Nielsen Press Release, The Nielsen Company*, October 20, 2016 (http://www.nielsen.com/us/en/press-room/2016/for-the-first-time-nielsen-opens-cpg-data-to-connected-partners.html).

O'Connell, V. 2008. "Reversing Field, Macy's Goes Local," *The Wall Street Journal*, April 21, 2008.

Oliveira, S. and O. Zaïane. 2002. "Privacy Preserving Frequent Itemset Mining," *Proceedings of IEEE ICDM Workshop on Privacy, Security, and Data Mining*, Australian Computer Society, Sydney, Australia, pp. 43-54.

Pearson, B. 2016. "Kroger Deal Highlights 3 Reasons Blended Data Is A Must In Retail," *Forbes* August 17, 2016 (https://www.forbes.com/sites/bryanpearson/2016/08/17/kroger-deal-highlights-3-reasons-blended-data-is-a-must-in-retail/#433120d52752).

Petersen, C. 2013. "Walmart's Secret Sauce … How the largest survives & thrives," (https://www.imsresultscount.com/resultscount/2013/03/walmarts-secret-sauce-how-the-largest-survives-thrives.html, paragraph 7)

Reddy, M., R. Wang. 1995. "Estimating Data Accuracy in a Federated Database Environment," *Proceedings 6th International Conference of Information Systems Management Data (CISMOD)*. Springer-Verlag, Secaucus,NJ, pp. 115–134.

Retail TouchPoints. 2017. "Winning With Data Sharing: Driving New Analytical and Revenue Opportunities," *Retail TouchPoints White Paper*, Sponsored by *1010data*.

RetailVelocity. 2017. "Retailer POS Data Sources," (https://www.retailvelocity.com/downstream-pos-data-retail-sales-analysis-sources)

Seymour, S. 2014. "Why Retailers and CPG Suppliers Should Form Data Sharing Partnerships," June 17, 2014 (http://www.swiftiq.com/blog/why-retailers-and-cpg-suppliers-should-form-data-sharing-partnerships, paragraph 4).

Stavropoulos, E., V. Verykios and V. Kagklis. 2016. "A Transversal Hypergraph Approach for the Frequent Itemset Hiding Problem," *Knowledge and Information Systems* 47(3), June 2016.

Sun, X., and P. Yu. 2007. "Hiding Sensitive Frequent Itemsets by a Border-Based Approach," *Journal of Computing Science and Engineering* (1:1), pp. 74-94.

Terry, L. 2015. "CGT/RIS Retailer/Supplier Shared Data Study: A Supplement to Consumer Goods Technology and RIS News," Sponsored by *RSi Retail Solutions*, 2015.

Vaidya, J. and C. Clifton. 2002. "Privacy Preserving Association Rule Mining in Vertically Partitioned Data," *Proceedings of 8th International Conference on Knowledge Discovery and Data mining (ACM SIGKDD)*, Edmonton, Alberta, Canada, pp. 639-644.

Verykios, V., A. Elmagarmid, E. Bertino, Y. Saygin, and E. Dasseni. 2004. "Association Rule Hiding," *IEEE Transactions on Knowledge Data Engineering* (16:4), pp. 434-447.

Verykios, V., E. Pontikakis, T. Yannis and L. Chang. 2007. "Efficient Algorithms for Distortion and Blocking Techniques in Association Rule Hiding," *Distributed and Parallel Databases* (22:1), pp. 85-104.

Verykios, V. 2013. “Association Rule Hiding Methods,” *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* (3:1) pp. 28-36.

Vishwanath V. and D. Rigby. 2006. “Localization: The Revolution in Consumer Markets,” *Harvard Business Review* April 1, 2006.

Wu, Y., C. Chiang and A. Chen. 2007. “Hiding Sensitive Association Rules with Limited Side Effects,” *IEEE Transactions on Knowledge and Data Engineering* (19:1), pp. 29-42.

Zaïane, O. 2002. “Building a Recommender Agent for e-Learning Systems,” *Proceedings of the International Conference on Computers in Education*, Auckland, New Zealand, December 2002.

Zheng, Z., R. Kohavi and L. Mason. 2001. “Real World Performance of Association Rule Algorithms,” *Proceedings of 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 401-406, New York, NY.

Zimmerman, A. 2006. “Wal-Mart Drops One-Size-Fits-All Approach,” *The Wall Street Journal*, September 07, 2006.

APPENDIX

Table A1: Notation

	Notation	Definition
Section 3.1	I	Set of items
	$j \subseteq I$	Itemset
	$i \subseteq I$	Transaction (set of items purchased by a customer in one visit)
	P	Set of locations (stores/regions/partitions)
	D_p	Dataset from location p
	$D = \cup_{p \in P} D_p$	Organizational-level dataset (obtained by consolidating all partitions D_p).
	σ^j (σ_p^j)	Support of itemset j (the number of transactions in D (D_p) that contains j)
	σ_{min} (σ_{min}^p)	Mining threshold for support for D and D_p
	F (F_p)	Set of frequent itemsets for D (D_p)
	$F^S \subseteq F$ $\left(F_p^S \subseteq F_p\right)$	Sensitive itemsets in D (D_p)
	σ_h^j $\left(\sigma_{hp}^j\right)$	Hiding threshold for D and D_p
Section 3.2	FIH_D	Complete formulation for the frequent itemset hiding problem for D
	a_{ij} $\left(a_{ij}^p\right)$	1 if transaction i supports the sensitive itemset j in $F^S\left(F_p^S\right)$; 0 otherwise
	x_i	1 if transaction i is marked for sanitization; 0 otherwise
	(OC)	Organizational-level constraints in FIH_D
	(PC)	Partition-level constraints in FIH_D
Section 4	$\mathcal{L}^{OC}(\lambda)$	Lagrangian dual with organizational-level constraints (OC) dualized using multipliers λ
	$\mathcal{L}_{D_p}^{OC}(\lambda)$	Subproblem of $\mathcal{L}^{OC}(\lambda)$ corresponding to partition p
	$\epsilon > 0$	Small value such that $\epsilon \times \sum_{i \in D_p} \sum_{j \in F^S} a_{ij} < 1$
	$\mathcal{L}_{D'}^{OC}(\epsilon)$	Reduced version of FIH_D after fixing (at 1) all variables set to 1 by all $\mathcal{L}_{D_p}^{OC}(\epsilon)$
	$\mathcal{L}^{PC}(\mu)$	Lagrangian dual with partition-level constraints (PC) dualized using multipliers μ
	$\mathcal{L}_{D_p'}^{PC}(\epsilon)$	Reduced version of FIH_D associated with partition D_p after fixing (at 1) all variables set to 1 by $\mathcal{L}^{PC}(\epsilon)$
	FIH_D^r	Reduced version of FIH_D obtained by eliminating all variables set to zero by *PartitionsFirst* and *OrgFirst*
	D^r $\left(D_p^r\right)$	Reduced version of D $\left(D_p\right)$ obtained by eliminating all transactions not identified for sanitization by either *PartitionsFirst* or *OrgFirst*