k Nearest Neighbor Classification across Multiple Private Databases [*]

Li Xiong
Department of Mathematics and
Computer Science
Emory University
lxiong@mathcs.emory.edu

Subramanyam Chitti, Ling Liu
College of Computing
Georgia Institute of Technology
chittis, lingliu@cc.gatech.edu

ABSTRACT

Distributed privacy preserving data mining tools are critical for mining multiple databases with a minimum information disclosure. We present a framework including a general model as well as multi-round algorithms for mining horizontally partitioned databases using a privacy preserving k Nearest Neighbor (kNN) classifier.

Categories and Subject Descriptors: H.2.8 [Database Management]: Database Applications

General Terms: Algorithms, Experimentation, Security

Keywords: Privacy, k Nearest Neighbor, Classification, Distributed Databases

1. INTRODUCTION

The information age has enabled many organizations to collect large amounts of data. Privacy-preserving data mining [5] becomes an important enabling technology for mining data from multiple private databases provided by different and possibly competing organizations. For example, many insurance companies collect data on disease incidents, seriousness of the disease and patient background. One way for the Center for Disease Control to identify disease outbreaks is to train a classifier across the data held by the various insurance companies for patterns that are indicative of disease outbreaks and use it to classify a query pattern as an outbreak or the opposite. However, commercial and legal reasons prevent the insurance companies from revealing their data. It is important and beneficial to have a distributed data mining algorithm that is capable of identifying potential outbreaks while respecting the privacy requirements of its participants.

Design Goals. There are three dimensions that we should consider when designing a privacy preserving classification algorithm, namely, *accuracy, efficiency, and privacy.* Ideally, we would like the algorithm to have a comparable accuracy to its non-privacy preserving counterpart, and an absolute privacy wherein no information other than the trained classifier and the classification of the query instance should be revealed to any node. At one end of the spectrum, we have the non-privacy preserving classifier algorithms, which are highly efficient but are not secure. At the other end, we have the secure multi-party computation protocols [4], using which we can construct classifiers which are provably secure in the sense that they reveal the least amount of information and have the highest accuracy; but are very inefficient. Our design goal is to look for algorithms that can provide a desired level of tradeoff between the accuracy of the classifier constructed and the stringency of the privacy requirements while maintaining efficiency.

Contributions. With these design objectives in mind, we present a privacy-preserving framework for constructing a kNN classifier across multiple private databases. This framework consists of a general model for privacy preserving kNN classification and a set of concrete algorithms for realizing this model. We discuss how well our algorithm achieves the specified requirements as demonstrated by our analytical study and experimental evaluation. To the best of our knowledge, this is the first paper to show how kNN classification can be achieved in a privacy preserving manner for horizontally partitioned data without a centralized trusted third party. Our approach has an important trait − it offers a trade-off between accuracy, efficiency and privacy, allowing our privacy-preserving kNN classifier to be applied in a variety of problem settings and meeting different optimization criteria.

2. MODEL AND ALGORITHMS

Consider n private databases distributed at n different nodes where all databases have the same schema, i.e. data is horizontally partitioned. We consider the problem where the nodes want to train a kNN classifier on the union of their databases while revealing as little information as possible to the other nodes during the construction of the classifier (*training* phase) and the classification of a new query (*test* phase).

Privacy Preserving kNN Classification Model. To solve the above problem, we need to adapt the basic distance weighted kNN classification algorithm to work in a distributed setting in a privacy preserving manner. The central thesis of our proposed model is to divide the problem into two sub-problems, and to ensure that each step is accomplished in a privacy preserving manner.

1. **Privacy preserving nearest neighbor selection:** Given a query instance x to be classified, the databases need to identify all points among the k nearest neighbors of x in a privacy preserving manner.

[*]The research was partially supported by NSF ITR, NSF CyberTrust and NSF CSR grants. A full version of this paper is available as [2].

CIKM'06, November 5–11, 2006, Arlington, Virginia, USA.
ACM 1-59593-433-2/06/0011.

2. **Privacy Preserving Classification:** With the knowledge of the points in its local database that are among the k nearest neighbors of x, each node can then calculate its local classification of x and determine the global classification of x in a privacy preserving manner.

Algorithm. It is important to note that if executed naively, the above steps can violate the privacy requirements of the individual databases. Given a query point x, we should ensure that the instances in a database are not revealed to other databases in the nearest neighbor selection, and that the local classification of each database is not revealed to other databases during global classification. We next present the concrete algorithms for each of these two steps.

In order to determine the points in their database that are among the k nearest neighbors of x, each node calculates k smallest distances between x and the points in their database (locally) and then we can use a privacy preserving algorithm to determine k smallest distances between x and the points in the union of the databases or kth nearest distance (globally). We can assume that the distance is a one-way function so that nodes do not know the exact position of each other node by distance. There has been privacy preserving algorithms recently proposed [1] for finding kth element that we can use for implementing this step. Although information-theoretically secure, it is still computationally expensive. In this paper, we adapt the multi-round top$k$ algorithm proposed in [6] to determine k smallest distances before determining kth smallest distance and achieve a tradeoff between accuracy, efficiency and privacy.

After each node determines the points in its database which are within the kth nearest distance from x, each node computes a local classification vector of the query instance where the ith element is the amount of vote the ith class received from the points in this node's database which are among the k nearest neighbors of x. The nodes then participate in a privacy preserving term-wise addition of these local classification vectors to determine the global classification vector. For the term-wise addition, we use the multi-round privacy-preserving addition protocol suggested in [3]. Once each node knows the global classification vector, it can find the class with the global majority of the vote by determining the index of the maximum value in the global classification vector. We present a sketch of the complete algorithm below.

1. Given an instance x to be classified, each node computes the distance between x and each point y in its database, $d(x, y)$, selects k smallest distances (locally), and stores them in a local distance vector ldv.

2. Using ldv as inputs, the nodes use the adapted privacy preserving top$k$ selection protocol [6] to select k nearest distances (globally), and stores them in gdv.

3. Each node selects the kth nearest distance Δ: $\Delta = gdv(k)$.

4. Assuming there are v classes, each node calculates a local classification vector lcv for all points y in its database: $\forall 1 \leq i \leq v$, $lcv(i) = \sum_y w(d(x, y)) * [f(y) == i] * [d(x, y) \leq \Delta]$, where $d(x, y)$ is the distance between x and y, $f(y)$ is the classification of point y, and $[p]$ is a function that evaluates to 1 if the predicate p is true, and 0 otherwise.

5. The nodes use the privacy preserving addition protocol [3] to do an element-wise addition of their local classification vectors lcv to calculate the global classification vector gcv: $gcv(i) = \sum_{j=1}^{n} lcv_j(i)$.

6. Each node assigns the classification of x as $classification(x) \leftarrow \arg\max_{i \in V} gcv(i)$.

3. ANALYSIS AND EXPERIMENTS

We conducted an analytical study as well as experimental evaluations on real datasets for the proposed algorithms in terms of its correctness, efficiency, and privacy characteristics. We present the key findings below.

The accuracy of our algorithm is very close to the accuracy obtained by a distributed kNN classifier. In particular, we are able to make the privacy preserving classifier as accurate as an ordinary classifier by running the algorithm for a larger number of rounds. A smaller randomization parameter also helps maximizing the relative accuracy of the algorithm.

With a communication complexity of $\Theta(n)$, where n is the number of nodes involved in the classification, the algorithm can accommodate a large number of nodes as well as large databases at every node.

The algorithm achieves a strong data privacy by having a very low probability of revealing the values (local distances) to other nodes. The algorithm parameter k plays an important role in affecting the accuracy and privacy of the algorithm. Using a larger k increases the privacy preserving nature of the protocol, however, with a price in the accuracy and efficiency of the kNN classifier constructed when each database has a large number of points. We note that it is possible to pick a k to achieve both good accuracy and good privacy.

4. CONCLUSION

We presented a general model for kNN classification across multiple private databases and multi-round algorithms for realizing the model. Our analysis and experiments showed the feasibility of the approach and its ability to achieve a balance between three important performance metrics: relative accuracy, efficiency, and privacy.

5. REFERENCES

[1] G. Aggarwal, N. Mishra, and B. Pinkas. Secure computation of the kth ranked element. In *IACR Conference on Eurocrypt*, 2004.

[2] S. Chitti, L. Xiong, and L. Liu. Mining multiple private databases using a privacy preserving knn classifier. Technical report, Emory University, Department of Mathematics and Computer Science, 2006. TR-2006-008-A.

[3] C. Clifton, M. Kantarcioglu, X. Lin, J. Vaidya, and M. Zhu. Tools for privacy preserving distributed data mining. In *SIGKDD Explorations*, 2003.

[4] O. Goldreich. Secure multi-party computation, 2001. Working Draft, Version 1.3.

[5] V. S. Verykios, E. Bertino, I. N. Fovino, L. P. Provenza, Y. Saygin, and Y. Theodoridis. State-of-the-art in privacy preserving data mining. *ACM SIGMOD Record*, 33(1), 2004.

[6] L. Xiong, S. Chitti, and L. Liu. Topk queries across multiple private databases. In *25th International Conference on Distributed Computing Systems (ICDCS)*, 2005.