Poster: Privacy Preserving Distributed Data Mining for Enterprises - Towards Application in Practical Setups

Daniel Tasche, M.Sc., Jens Heider, M.Sc., and Prof. Dr.-Ing. Jörg Lässig
University of Applied Sciences Zittau/Görlitz, Department of Computer Science
Brückenstr. 1, 02826 Görlitz, Germany
E-Mail: dtasche, jheider, jlaessig@hszg.de
http://ead.hszg.de

1) Introduction: In today's globalized world, organizations and companies are within a business environment, which is characterized by a competition with emerging economies and steadily rising cost pressure. To implement cross-company cooperation that also protect the interests of the individual companies, this work presents an approach for the implementation of cooperative data mining algorithms that comes without the drawbacks shown above. This work is based on a scenario of collaborative data exchange with three example implementations of privacy preserving data mining (PPDM) algorithms that are based on methods that have been introduced in the literature have yet not been implemented in productive applications: K-Means clustering on horizontally and vertically partitioned data [8], neural network learning on horizontally partitioned data [7] and the ID3 decicion tree algorithm on horizontally partitioned data [6]. As a platform for implementation and exchange for businesses and enterprises in the cross-company cooperation context, the open source platform RapidMiner is used.

In enterprise infrastructures, there are various obstacles in applying state of the art privacy preserving data mining algorithms. Besides the algorithmic setups, addressing security issues and following privacy preserving protocols, there are various data sources and infrastructure components, that the analyisis components have to be connetced to. In our studies we applied standard approaches as ESB systems or other well known integration tools and strategies. The privacy preserving analysis methods are injected as additional layer between local data sources on the one hand, and interconnecting infrastructure components for data exhange with other companies on the other hand.

The systematic algorithmic integration of information disclosure techniques and the development of new data mining algorithms which guarantee the security of sensitive data is called *Privacy Preserving Data Mining*. In general, two different categories of PPDM methods can be distinguished. On the one hand, anonymization tries to suppress the access to critical data. This approach is very general but poses the risk of significant loss of quality in the analysis, since a certain amount of information, which is in principle available, is not used for the analysis. On the other hand, there are the secure distributed data mining methods which try to avoid information leaks from the very beginning for the whole data set [2]. PPDM in general seeks for win-win situations, extracting and using knowledge over several sites [3]. But there are also concerns. in this process the individual's privacy has to be preserved and the data holders have to be protected against misuse or disclosure of information [5]. So every PPDM technique has to ensure that all information that is disclosed cannot be traced to an individual or does not contribute an intrusion [9]. The problem of data loss as described in the second approach is solved by Secure Multi-Party Computation [1]; methods which permit a precise analysis of cross-company data, without the need to reveal this data to the other participants.

Homomorphic encryption is the key part to preserve privacy in various algorithms in the literature for our framework. It is a form of encryption which allows to perform computations on ciphertexts. The result of such computations is an encrypted representation of the result as it would have been computed on the plaintexts. After all operations are finished, the plaintext result can be generated by simply decrypting the ciphertext result. Almost all PPDM methods, that do not rely on privacy information sanitization, are based on cryptosystems with such properties. In 1999, Pascal Paillier published a new public-key encryption method [4]. Later it became, because of its additive homomorphic property, a fundamental part of several privacy preserving computations. Though, it is not mandatory to use exactly this scheme, because every additive homomorphic encryption will work.

2) Architecture: PPDM algorithms like privacy preserving ID3, neural networks and K-Means have in common that they require an additional security layer. This means e. g., that an infrastructure with pair-wise public-key encrypted channels for secure communication of different entities is necessary. Furthermore, most complex privacy preserving data mining algorithms make use of a limited set of basic operations as secure multi-party addition, secure multi-party multiplication or more complex operations [5]. The library integrates those operations as a basic layer on which the PPDM algorithms build up.

First, there is a security layer, which consists of libraries for encryption and decryption of data. Almost all algorithms require a homomorphic encryption system to calculate with encrypted data. Hence "JPaillier" has been implemented as instance of the Paillier cryptosystem. Second, there is a layer for required atomic operations. E. g. the privacy preserving ID3 algorithm on horizontally partitioned data requires a Secure-Add

operation while the privacy preserving K-Means algorithm on vertically partitioned data requires the add_and_permute-operation. Within the framework they are offered as atomic operation to use in other algorithms as well. Finally, there is the algorithm layer with different methods, including privacy preserving ID3 and neural networks on horizontally partitioned data, as well as K-Means on vertically partitioned data, as mentioned here.

Distributed computation of the algorithms requires also a network layer. But since the later application of the framework is open, the network layer has not been integrated in the core framework. Instead, the possibility for distributed communication has been prepared, but it is open whether the parties communicate directly on the same system or via the network. A reference implementation of network communication has been implemented as an example.

The secure multi-party computation operations which have been implemented are designed in an independent way. Hence, it is possible to use them in further algorithm implementations as well. There are *Secure Multi-party Addition*, *Secure Multi-party Multiplication*, *Secure Multi-party Square Division* and *Add-And-Permute*.

3) Evaluation for K-Means: Based on the Iris data set, the runtime of the privacy preserving K-Means algorithm has been measured for vertically partitioned data and based on different input sizes. Vertically partitioned means here that each party in the distributed setup has its own attributes. Besides the runtime of the overall method, also the cumulative time for all add-and-permute-operations of the privacy preserving version of the algorithm has been measured. The Add-And-Permute-operation, a necessary privacy feature of the algorithm, is an important additional security step compared to the local algorithm. Thus, the test data set which consists of four attributes has been divided so that each party has one attribute per entity. The final test setup was configured with four parties in the distributed privacy preserving implementation and a threshold for the algorithm termination of 0.1 in both algorithms.The comparison shows that there is a huge gap between the runtime of both algorithms. Most time of the distributed version is spent in the Add-And-Permute-operation, hence in security related features, even if their implementation is heavily optimized.

4) Evaluation for Neural Network: As for K-Means, the runtime of the privacy preserving neural network algorithm on horizontally partitioned data was compared to a local version of a neural network with offline/batch-learning. Based on the Tic-Tac-Toe Endgame data set, time was measured for the total execution of the algorithm, both local and distributed on different input sizes. Also, the runtime for all distributed k-secure sum-operations of the privacy preserving has been measured separately, since this is the extra step needed to secure to communication. In the horizontally partitioned case, each party has its own entities. Thus, the test data set has been divided so that each party has the same number of entities. The tests show that the distributed privacy preserving version performs better than the local one, which is a remarkable fact. This behavior results from the parallel processing due to the distributed setup. The communication between the parties is only used for the exchange of the matrices weights of the neural network.

5) Conclusion and Future Work: We implemented different privacy preserving data mining algorithms from published theoretical research and analyzed three of them in detail. These are the K-Means algorithm on vertically partitioned data as well as neural network and ID3 decision tree algorithm on horizontally partitioned data. As we have seen, these privacy preserving data analysis methods are in general applicable but have significant performance deficits concerning complexity and runtime. The implemented algorithms and provided basic operations have been bundled to a expendable framework, which turned out to be a complex task. We conclude by remarking that we focused on the theoretical algorithms without considering any parallelization approaches of modern programming languages. The focus was on feasibility and practical applicability of the methods. So there is potential for further optimization.

Future work includes the implementation of other algorithms, in particular K-Means on horizontally partitioned data and neural networks and ID3 on vertically partitioned data as well as further, not yet considered, data mining methods. A more detailed analysis of the privacy preserving implementation of the single algorithms and an in-deep discussion of individual obstacles is given elsewhere, in upcoming publications in the CoPPDA project, since this needs further information about the methods, which cannot be covered by this work. Furthermore, the described optimization approaches concerning parallelization are currently implemented. The overall goal is to create a rapid miner plug-in which integrates the described methods.

REFERENCES

[1] Wenliang Du and Mikhail J. Atallah. Secure multi-party computation problems and their applications: A review and open problems. In *Proceedings of the 2001 Workshop on New Security Paradigms*, NSPW '01, pages 13–22, New York, NY, USA, 2001. ACM.

[2] Henrik Grosskreutz, Benedikt Lemmen, and Stefan Rping. Privacy-preserving data-mining. *Informatik-Spektrum*, 33(4):380–383, 2010.

[3] Joerg Laessig and Michael Hahsler. Cooperative data analysis in supply chains using selective information disclosure. In *Operations Research and Computing: Algorithms and Software for Analytics*, pages 245–256, Jan 2015.

[4] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In Jacques Stern, editor, *Advances in Cryptology EUROCRYPT 99*, volume 1592 of *Lecture Notes in Computer Science*, pages 223–238. Springer Berlin Heidelberg, 1999.

[5] S. Samet and A. Miri. *Privacy-Preserving Data Mining: Secure Protocols for Privacy-Preserving Data Mining and Machine Learning Techniques*. VDM Publishing, 2011.

[6] Saeed Samet and Ali Miri. Privacy preserving id3 using gini index over horizontally partitioned data. In *AICCSA*, pages 645–651, 2008.

[7] Nico Schlitter. A protocol for privacy preserving neural network learning on horizontally partitioned data. *PSD*, 2008.

[8] Jaideep Vaidya and Chris Clifton. Privacy-preserving k-means clustering over vertically partitioned data. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 206–215. ACM, 2003.

[9] Jaideep Vaidya, Christopher W Clifton, and Yu Michael Zhu. *Privacy preserving data mining*, volume 19. Springer Science & Business Media, 2006.