Security-preserving Support Vector Machine with Fully Homomorphic Encryption

Saerom Park*, Jaeyun Kim, Joohee Lee, Junyoung Byun, Jung Hee Cheon, Jaewook Lee**
Seoul National University
1 Gwanak-ro, Gwanak-gu, Seoul, South Korea 08826
drsaerompark@gmail.com *, jaewook@snu.ac.kr **

Abstract

Recently, security issues have become more and more important to apply machine learning models to a real-world problem. It is necessary to preserve the data privacy for using sensitive data and to protect the information of a trained model for defending the intentional attacks. In this paper, we want to propose a security-preserving learning framework using fully homomorphic encryption for support vector machine model. Our approach aims to train the model on encrypted domain to preserve data and model privacy with the reduced communication between the servers. The proposed procedure includes our protocol, data structure and homomorphic evaluation.

As machine learning models have been effectively applied to various real-world problems, collecting data from various sources became crucial for many service providers(Park, Hah, and Lee 2017). In this situation, privacy issues have become a major problem in the learning society. Therefore, it is necessary to preserve the privacy of the data and the security of the learning process without compromising the performance of the learning algorithms.

Homomorphic encryption (HE) enables computations on encrypted data (ciphertext) which are equivalent to operations on decrypted data (plaintext) (Gentry 2009). In recent years, privacy-preserving machine learning applications have developed, but they have high computational cost and huge memory burden. Cheon et al. developed a homomorphic encryption scheme for approximate arithmetic (HEAAN) (Cheon et al. 2017).

Support vector machine (SVM) is one of the most effective machine learning algorithms for classification (Cortes and Vapnik 1995). The training data and the model parameters should be protected to apply the SVM model to the security-preserving scenario. Therefore, in this paper, we propose the implementation for the training phase of the SVM classifier on the encrypted domain with the HEAAN scheme.

Design Components

Fully Homomorphic Encryption

Fully homomorphic encryption (FHE) is a cryptographic scheme which aims to enable homomorphic operations such as additions and multiplications on encrypted data. Recently, HEAAN scheme (Cheon et al. 2017) was developed to carry out approximate computations efficiently.

For the (leveled) HEAAN of depth L, we set the parameters such as a power of two M', integers p, q_0, $q_L = p^L \cdot q_0$, h and P, and a real σ for λ-bit security. The specifications of the algorithm are as follows.
- $(pk, sk, evk) \leftarrow \mathsf{KeyGen}(1^\lambda)$
- $\vec{c} \leftarrow \mathsf{Encrypt}(pk,\ m)$: Sample $\vec{r} \in \mathcal{R}^2$ and $e_0, e_1 \leftarrow \mathcal{DG}_{q_L}(\sigma^2)$.
Output $\vec{c} \leftarrow \vec{r} \cdot pk + (m + e_0,\ e_1) \in \mathcal{R}_{q_L}^2$.
- $m \leftarrow \mathsf{Decrypt}(sk, \vec{c} = (c_0, c_1))$: Output $m \leftarrow c_0 + c_1 \cdot s \mod q_\ell$.
- $c_{\mathsf{add}} \leftarrow \mathsf{Add}(\vec{c}_1, \vec{c}_2)$: Output $\vec{c}_{\mathsf{add}} \leftarrow \vec{c}_1 + \vec{c}_2 \mod q_\ell$.
- $c_{\mathsf{mult}} \leftarrow \mathsf{Mult}(\vec{c}_1, \vec{c}_2)$: Set $c_1 = (b_1, a_1)$ and $c_2 = (b_2, a_2)$. Let $(d_0, d_1, d_2) \leftarrow (b_1 \cdot b_2, a_1 \cdot b_2 + a_2 \cdot b_1, a_1 \cdot a_2) \in \mathcal{R}_{q_\ell}^3$. Output $\vec{c}_{\mathsf{mult}} \leftarrow (d_0, d_1) + \lfloor \frac{1}{P} \cdot (d_2 \cdot evk \pmod{P \cdot q_\ell}) \rceil \in \mathcal{R}_{q_\ell}^2$.
- $\vec{c}' \leftarrow \mathsf{Rescaling}_{\ell \to \ell'}(\vec{c})$: For a level ℓ ciphertext $\vec{c}$, output $\vec{c}' \leftarrow \lfloor \frac{q_{\ell'}}{q_\ell} \cdot \vec{c} \rceil \in \mathcal{R}_{q_{\ell'}}^2$ at level ℓ'.
For more details, we recommend to see (Cheon et al. 2017).

Least Square Support Vector Machine

The SVM aims to find the maximum margin hyperplane, given the training examples $\{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)\} \subset \mathbb{R}^d \times \{-1, 1\}$. To train the SVM model over encrypted data, we used a nonlinear least-square SVM with a basis function $\phi(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}^l$ that minimizes the following optimization problem (Suykens and Vandewalle 1999):

$$\min_{\mathbf{w},b,e_i} \quad \frac{1}{n}\sum_{i=1}^n e_i + \lambda \|\mathbf{w}\|^2 \qquad (1)$$
$$\text{s.t.} \quad y_i \left[\mathbf{w} \cdot \phi(\mathbf{x}_i) + b\right] = 1 - e_i, \forall i = 1, \ldots, n,$$

where $\mathbf{w} \in \mathbb{R}^l$. To solve the problem (1), we construct the Lagrangian function: $L(\mathbf{w}, b, \mathbf{e}, \boldsymbol{\alpha}) = \lambda \|w\|^2 + \frac{1}{n}\sum_{i=1}^n e_i^2 - \sum_{i=1}^n \alpha_i \{[\mathbf{w} \cdot \phi(\mathbf{x}_i) + b] + e_i - 1\}$. With the optimality conditions of the Lagrangian function, the following linear system is obtained by removing $\mathbf{w}$ and $\mathbf{e}$:

$$\mathbf{A}\mathbf{b} = \begin{bmatrix} 0 & \mathbf{y}^T \\ \mathbf{y} & \boldsymbol{\Omega} + \lambda \mathbf{I}_n \end{bmatrix} \begin{bmatrix} b \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{1}_n \end{bmatrix} = \tilde{\mathbf{1}} \qquad (2)$$

where $\boldsymbol{\Omega} \in \mathbb{R}^{(n+1)\times(n+1)}$ s.t. $\boldsymbol{\Omega}_{i,j} = k(x_i, x_j)$. We introduce an additional least square problem with gradient descent method for the system (2), which convergence can be

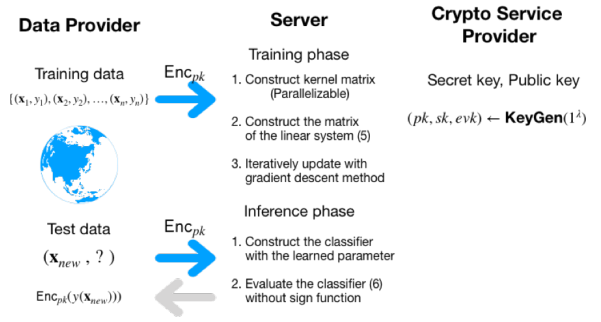

Figure 1: Our procedure with protocol

guaranteed by the well-posedness of. The matrix $\mathbf{A}$ can be decomposed as follows:

$$\mathbf{A} = \left(\begin{bmatrix}1\\ \mathbf{y}\end{bmatrix}\begin{bmatrix}1 & \mathbf{y}^T\end{bmatrix}\right) \odot \begin{bmatrix}0 & \mathbf{1}^T\\ \mathbf{1} & \mathbf{\Omega}\end{bmatrix} + \eta \begin{bmatrix}0 & \mathbf{0}^T\\ \mathbf{0} & \mathbf{I}_n\end{bmatrix} \quad (3)$$

where $\odot$ represents a Hadmard product. By Schur product theorem, the first term becomes PSD with Mercer kernel. Therefore, the linear system (2) is well-posed with some $\eta > 0$, and the convergence of iterative method can be guaranteed. With these design components, the intermediate dycryptions for training the SVM model can be reduced.

Implementation

In this paper, we propose a protocol which can reduce the communication to construct kernel matrix and to learn the model parameters with FHE operations. Figure 1 illustrates the whole procedure of our protocol which secures server-side and data provider-side information because all operations can be performed on encrypted domain.

Data Structure

As mentioned previously, we first compute the kernel matrix from the encrypted data. For simplicity, we assume that the kernel matrix is encrypted as a ciphertext. We want to build a parallelizable procedure because of computational cost and large memory of ciphertext. The HEAAN scheme supports slot-wise operation over ciphertext, so the operations of matrices should be replaced with addition and Hadmard product. We propose data structure for calculating the inner product matrix. Assume that $\mathbf{Z} \in \mathbb{R}^{n\times d}$ and $\mathbf{Z}\mathbf{Z}^T, \hat{\mathbf{Z}}_j \in \mathbb{R}^{n\times n}$.

$$\mathbf{Z}\mathbf{Z}^T = \hat{\mathbf{Z}}_1 \odot \hat{\mathbf{Z}}_1^T + \cdots + \hat{\mathbf{Z}}_d \odot \hat{\mathbf{Z}}_d^T,$$

$$\mathbf{Z} = \begin{bmatrix} z_{11} & \cdots & z_{1d} \\ \vdots & \ddots & \vdots \\ z_{n1} & \cdots & z_{nd} \end{bmatrix}, \quad \hat{\mathbf{Z}}_j = \begin{bmatrix} z_{1j} & z_{2j} & \cdots & z_{nj} \\ \vdots & \vdots & \ddots & \vdots \\ z_{1j} & z_{2j} & \cdots & z_{nj} \end{bmatrix}.$$

The multiplication of cihpertexts in this operation can be performed on d different machines in parallel.

Secure-Preserving Iterative Training

In this study, to efficiently implement the gradient descent method, we utilize the symmetricity of the matrix (3) in matrix multiplication. To learn $\mathbf{b}$, the updated equation is $\mathbf{b}_{k+1} = \mathbf{b}_k - \alpha\mathbf{A}^T(\mathbf{A}\mathbf{b}_k - \tilde{\mathbf{1}})$. Matrix-vector multiplication on the encrypted domain is efficiently implemented by rotating the slots. We used pre-computed $\mathbf{A}^T\mathbf{A}$ by using the symmetric property instead of two matrix-vector multiplications per iteration. Therefore, the resulting update equation is $\mathbf{b}_{k+2} = \mathbf{M}\mathbf{b}_k - \mathbf{c}$, where $\mathbf{M} = (\mathbf{I} - \alpha\mathbf{A}^T\mathbf{A})^2$ and $\mathbf{c} = ((1+\alpha)\mathbf{I} - \alpha\mathbf{A}^T\mathbf{A})\mathbf{A}^T\mathbf{b}$ are pre-computed.

data	Dual(RBF)	GD(RBF)	Dual(poly)	GD(poly)
hea	0.90	0.89	0.79	0.79
pid	0.83	0.82	0.76	0.75
wbc	0.98	0.97	0.97	0.95

Table 1: Classification accuracy for real datasets

Comparison

Our procedure replaces the dual convex optimization with numerical gradient descent to implement the security-preserving LSSVM. To illustrate the result of this replacement, we compare the classification performances with RBF and polynomial kernels for some real datasets used in (Suykens and Vandewalle 1999). Table 1 shows that the replacement does not severely affect the performances. From this results, we can expect to develop a secure-preserving SVM without compromising the performance.

Conclusion

In this paper, we present a new framework to train the LSSVM model using HEAAN. This framework includes protocol, data structure and secure-preserving iterative training procedure. Our method considers the reduction of computational cost and memory burden that are the common problem for the application of HE scheme. The proposed solution can be helpful to show the potential for the practical utilization of machine learning models without concerns on security and privacy issues.

Acknowledgement

This work was supported in part by NRF of Korea grants No. 2018R1D1A1A02085851, 2016R1A2B3014030 and 2017R1A5A1015626.

References

[Cheon et al. 2017] Cheon, J. H.; Kim, A.; Kim, M.; and Song, Y. 2017. Homomorphic encryption for arithmetic of approximate numbers. In *ASIACRYPT 2017*, 409–437. Springer.

[Cortes and Vapnik 1995] Cortes, C., and Vapnik, V. 1995. Support-vector networks. *Machine learning* 20(3):273–297.

[Gentry 2009] Gentry, C. 2009. *A fully homomorphic encryption scheme*. Ph.D. Dissertation, Stanford University. http://crypto.stanford.edu/craig.

[Park, Hah, and Lee 2017] Park, S.; Hah, J.; and Lee, J. 2017. Inductive ensemble clustering using kernel support matching. *Electronics Letters* 53(25):1625–1626.

[Suykens and Vandewalle 1999] Suykens, J. A., and Vandewalle, J. 1999. Least squares support vector machine classifiers. *Neural processing letters* 9(3):293–300.