

Privacy-Preserving Machine Learning for Mental Health Prediction Using Homomorphic Encryption

Shahroz Abbas

*Faculty of Computer Science and Tech.
Algoma University
Brampton, ON, Canada
sabbas@algonau.ca*

Ajmery Sultana

*Faculty of Computer Science and Tech.
Algoma University
Brampton, ON, Canada
ajmery.sultana@algonau.ca*

Mahreen Nasir

*Faculty of Computer Science and Tech.
Algoma University
Saulte Ste. Marie, ON, Canada
mahreen.nasir@algonau.ca*

Miguel Garcia-Ruiz

*Faculty of Computer Science and Tech.
Algoma University
Sault Ste. Marie, ON, Canada
miguel.garcia@algonau.ca*

Wenjun Lin

*Faculty of Computer Science and Tech.
Algoma University
Brampton, ON, Canada
randy.lin@algonau.ca*

Abstract—Student mental health issues, such as stress, anxiety, and depression, are increasingly prevalent in academic institutions, significantly affecting well-being and academic performance. Recent machine learning (ML)-based systems have demonstrated promise in predicting mental health conditions using survey data, but these approaches often process sensitive information in plaintext, risking privacy breaches or relying on centralized data storage vulnerable to leaks. Homomorphic encryption (HE) has been proposed for secure ML, but existing implementations either focus on simpler datasets (e.g., numerical/IoT data) or incur impractical computational overhead (e.g., high RAM usage or prolonged training times) for real-world mental health applications. To address these gaps, we introduce a privacy-preserving predictive model for student mental health using logistic regression trained directly on encrypted data via the TenSEAL library. Our work uniquely combines a leveled fully homomorphic encryption (FHE) scheme to ensure end-to-end confidentiality, replacing the standard sigmoid with a quadratic approximation for homomorphic compatibility. We also perform a comprehensive efficiency analysis that evaluates RAM usage and training time across polynomial-modulus degrees to balance security and practicality, a trade-off underexplored in prior HE-based mental health studies. Experimental results show that our encrypted model achieves 84% accuracy (vs. 96% unencrypted) with minimal performance loss, while benchmarks demonstrate scalable resource consumption. This work advances the feasibility of implementing FHE in sensitive domains such as mental health, offering a rigorous template for privacy-preserving ML without compromising predictive utility.

Index Terms—Student Mental Health, Predictive Modeling, Homomorphic Encryption, Privacy-Preserving Machine Learning

I. INTRODUCTION

Mental health challenges among students have become a critical concern for academic institutions around the world [1]. The pressure of educational performance, social expectations, and life transitions make university students especially unsafe for stress, anxiety, depression, and other psychological problems. According to recent studies, a significant percentage of college students experience mental health problems that

can affect their educational performance, social engagement, and overall well-being [2], [3]. In some severe cases, untreated mental health issues can also cause suicidal ideation or dropouts, highlighting the urgency of early detection and intervention mechanisms within academic institutions [4].

Recently, there has been a lot of interest in using Machine Learning (ML) approaches to predict student mental health diagnoses [5]. These ML models could help identify students with high stress levels in the early stages. This could lead to enabling initial intervention and customized support services, students can improve results and reduce dropout rates. However, it also raises concerns about sensitivity and data privacy [6]. The data used for the training and prediction of these models contain highly personal and confidential information. Therefore, the privacy of these models is a very important concern that should not be neglected.

To address this issue, the privacy-preserving machine learning model (PPML) could be utilized [7]. Techniques like PPML are designed to ensure that sensitive data used during the training and inference phases of ML models remain secure and confidential, as ML models are highly dependent on personal data.

The primary goal of PPML is to mitigate risks such as unauthorized access, data leakage, and inference attacks, and one of the types of PPML is Homomorphic Encryption (HE) [8]. The HE allows the computation directly on encrypted data, which means that the data are first encrypted and then trained on the ML model, and the results can be decrypted later to reveal the output. HE supports various schemes, including the Brakerski Vaikuntanathan (BFV) scheme and the Cheon Kim Song (CKKS) scheme. The BFV scheme supports accurate arithmetic on the integer, which is suitable for disconnected data, while the CKKS scheme consistently involves the actual number, which is ideal for machine learning.

In 2009, Gentry [9] introduced the first fully homomorphic encryption (FHE) scheme that had the ability to perform

arbitrary additions and multiplications. Since then, there have been different types of HE based on the types and complexity of operations they support. To turn these theoretical breakthroughs into practical tools, several high-performance open-source libraries have emerged HELib [10], Microsoft-SEAL [11], HEAAN [12], and TFHE [13]. FHE supports both addition and multiplication operations on ciphertexts, enabling arbitrary computations on encrypted data. This means that, in theory, any function that could be computed on plaintext data can also be computed securely on encrypted data using FHE [14], making it a strong candidate for end-to-end privacy in ML.

In this work, we used HE techniques in the ML model using TenSEAL [15]. we employ the CKKS scheme, which is part of HE. By integrating it, we ensure that student mental health data can be used to train while maintaining strict privacy standards. This balance between utility and privacy is important to deploy an ML system in sensitive domains such as health and education. To this end, the main contributions of this study are as follows.

- **Development of privacy preserving student mental health predictive model:** We build a secure and privacy-preserving ML model to analyze student mental health data.
- **Integration of HE for secure analysis:** We implemented HE to allow computations on student mental health data without requiring decryption. We encrypt the input feature vectors using the CKKS scheme through TenSEAL before model training and then train the model based on encrypted data. This ensures that student mental health records remain protected, reducing the risk of data breaches and unauthorized access.
- **Evaluation of encrypted vs. unencrypted model performance:** We compare the effectiveness of our privacy-preserving model with traditional unencrypted approaches in terms of accuracy, computational efficiency, and feasibility for large-scale deployment in academic institutions.
- **Analysis of RAM consumption based on different polynomial degrees:** We investigate the computational overhead of HE by analyzing RAM consumption for different polynomial degrees in the encryption scheme. This helps balance security and efficiency, ensuring optimal performance while maintaining strong privacy guarantees in the encrypted mental health prediction model.
- **Time consumption analysis based on different polynomial degrees:** We evaluate the impact of polynomial degree variations in HE on computation time. Our analysis covers encryption, model training, and inference, highlighting how increasing polynomial degrees affects processing time and overall system performance. The time consumption analysis provides critical insights into the trade-offs between computational efficiency and security. This enables a practical deployment of the privacy-preserving model while ensuring real-world feasibility.

The rest of this paper is organized as follows. Section II describes related works in the field of HE. Section III provides an overview of the proposed model architecture, Section IV provides the experimental results, and Section V concludes the paper.

II. RELATED WORK

Researchers have used HE to protect data in ML models. This section focuses on related work in the field of HE for keeping sensitive information private. Fawaz et al. [16] conducted a comparative analysis of BFV and CKKS HE schemes using the Microsoft SEAL library. Their experiments on encrypted vector operations showed that BFV was faster in most cases, while CKKS handled larger vector sizes more effectively, though at the cost of increased noise. However, they did not train the ML model. Sagarika et al. [17] applied the Paillier HE scheme to perform linear regression on a sales dataset. The encrypted model's outputs matched those of the plaintext model after decryption, demonstrating the correctness of encrypted computation. Aono et al. [18] developed a secure logistic regression framework using additively HE, including Paillier and LWE-based schemes. Their system achieved up to 80.7% accuracy in the Pima dataset and 75.4% in SPECTF, showing minimal performance loss. Liu et al. [19] proposed a privacy-preserving logistic regression model using Microsoft SEAL, achieving 98.9% accuracy on a subset of the scikit-learn Digits dataset. Similarly, Wiryen et al. [20] trained encrypted logistic regression models on IoT data using the TenSEAL library. Their results showed that the CKKS-based model achieved 69.76% accuracy, identical to its plaintext counterpart, supporting the feasibility of the HE scheme for the privacy-preserving ML model.

The works of [17]–[20] have demonstrated the feasibility of a privacy-preserving ML model using HE scheme, they primarily focused on structured or numerical datasets (e.g., digits, IoT data, or healthcare records), which are less complex and sensitive compared to survey-based mental health data. Mental health surveys contain highly personal and nuanced responses that require specialized privacy-preserving techniques. Addressing this gap ensures the framework's applicability to real-world, sensitive domains where data confidentiality is critical. Moreover, the existing work [17]–[20] did not thoroughly analyze the trade-offs between security parameters (e.g., polynomial degrees) and computational resources (RAM usage, training time). Understanding these trade-offs is essential for practical deployment, as it helps institutions balance privacy guarantees with feasible resource allocation, especially in resource-constrained environments like academic settings. Although some studies [17]–[20] reported accuracy, comprehensive evaluations of encrypted models (e.g., Mean Squared Error (MSE), Mean Absolute Error (MAE), and runtime under varying HE parameters) were missing. Rigorous benchmarking ensures transparency and trust in the model's reliability. It also guides stakeholders in selecting optimal configurations for specific use cases, enhancing scalability and adoption.

TABLE I
COMPARATIVE ANALYSIS OF HE SCHEMES IN PRIVACY-PRESERVING ML MODEL

Study	Scheme	Library	Task	Acc. (enc/plain)
Fawaz et al. [16]	BFV vs CKKS	Microsoft SEAL	Encrypted vector operations	–
Sagarika et al. [17]	Paillier	Custom	Linear regression (sales dataset)	– (plaintext match)
Aono et al. [18]	Paillier, LWE	Custom	Logistic regression (Pima, SPECTF)	80.7%, 75.4% / same
Liu et al. [19]	BFV(LR)	SEAL+PyTorch	Digit classification	98.9% / 99.1%
Wiryen et al. [20]	CKKS(LR)	TenSEAL	IoT anomaly detection	69.8% / 69.8%
Our work	CKKS(LR)	TenSEAL	Student MH survey prediction	84% / 96%

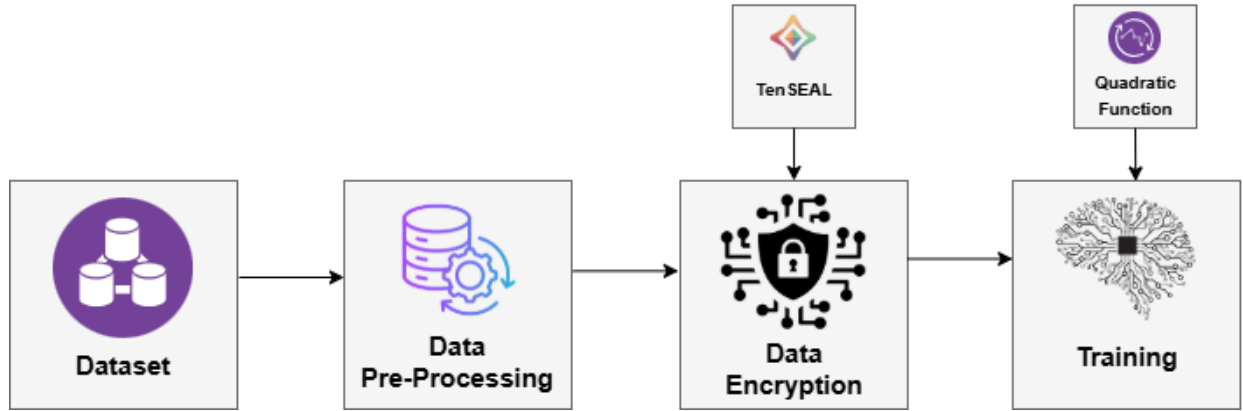


Fig. 1. Privacy preserving system architecture using ML model

III. METHODOLOGY

This section describes the methodology adopted for developing the PPML model for student mental health using the HE scheme. As illustrated in Fig.1, our privacy-preserving framework follows a structured workflow: (1) raw student mental health surveys are collected and preprocessed (label encoding, scaling, and train-test splitting), (2) processed data are encrypted using the CKKS homomorphic scheme via TenSEAL, and (3) logistic regression training occurs directly on the encrypted data while preserving confidentiality throughout the pipeline.

A. Dataset

The readily available dataset from the Kaggle [21] is utilized, which represents several attributes relevant to the mental health of the student. It contains several features related to demographics, lifestyle, and psychological indicators, which serve as input variables for the model.

Fig. 2 presents a correlation heatmap that visually represents the relationships among the features in the dataset. This analysis is crucial to understanding which variables have strong associations with mental health outcomes, which helps to select the best features and optimize the model. Among all features, suicidal thoughts, academic pressure, and financial stress show the strongest positive correlations with depression. The dataset captures complex non-linear relationships that contribute to strong model performance as shown in Table. II.

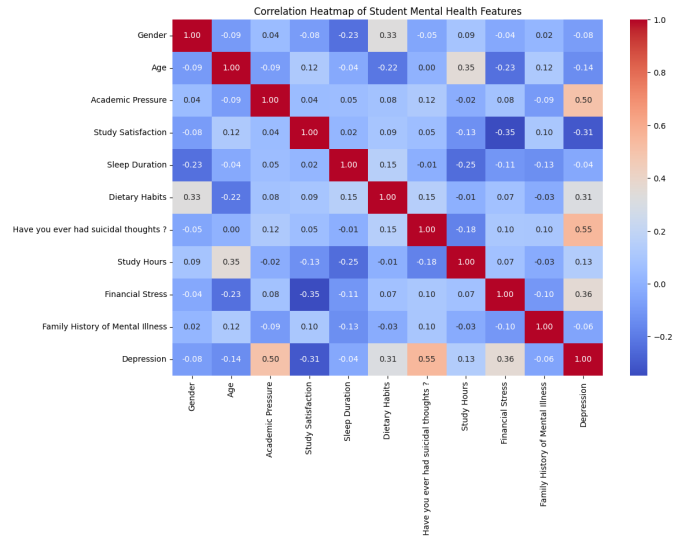


Fig. 2. Correlation heatmap of the features in the student mental health dataset

This dataset serves as the foundation for building a privacy-preserving predictive model using the HE scheme, ensuring that sensitive student data remain confidential throughout the machine learning pipeline.

B. Preprocessing

We performed the following pre-processing steps:

- **Label encoding:** Converted each categorical feature (e.g., gender, duration of sleep) to integer codes using Label Encoder. CKKS operates on numeric vectors, so converting categories to integers allows encrypted arithmetic while keeping the dimensionality low compared to one-hot encoding.
- **Target mapping:** Mapped the Depression column from Yes/No to floating 1.0, 0.0. Logistic regression loss and CKKS vectors require the mapping of numerical labels to 0,1, which enables the direct computation of gradients and predictions under encryption.
- **Feature scaling:** Used Min-Max Scaler to rescale all features to the 0,1 range. CKKS encodes real values at a fixed precision, bounding inputs to 0,1 maximizes use of the noise budget, reduces noise growth per operation, and stabilizes gradient descent by keeping all features on the same scale.
- **Train/Test split:** The dataset was split into 75% training and 25% testing sets.

C. Privacy-Preserving Encryption

The proposed system utilizes a PPML framework built using the CKKS scheme [22] provided by the TenSEAL library [15]. This allows computations to be performed directly on encrypted data, ensuring that the privacy of sensitive information is preserved throughout training and inference processes.

Algorithm 1 Privacy-Preserving CKKS Algorithm

Require: raw input x , transformation ϕ , CKKS settings σ

Ensure: decrypted result $\phi(x)$

- 1: $(PK, SK) \leftarrow \text{GenerateKeys}(\sigma)$
- 2: $C \leftarrow \text{EncryptCKKS}(PK, x)$
- 3: $C' \leftarrow \text{EvaluateCKKS}(C, \phi)$
- 4: $y \leftarrow \text{DecryptCKKS}(SK, C')$
- 5: **return** y

The pseudocode in Algorithm 1 is the consensus of the CKKS HE scheme and is based on the following four parts. The Generate Keys are used to generate the public key (PK) or private key (SK) based on the parameters. The EncryptCKKS takes the public key PK and the raw input x or the plaintext to produce a ciphertext C . EvaluateCKKS applies the transformation ϕ directly to C , performing all necessary additions and multiplications under encryption to obtain a new ciphertext C that contains $\phi(x)$. DecryptCKKS uses the secret key SK to decrypt C .

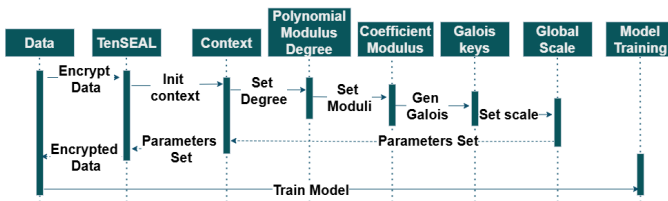


Fig. 3. CKKS Encryption Flowchart

Fig.3 illustrates the process flow from data encryption to model training based on the TenSEAL Library.

The TenSEAL library uses context as shown in Fig.3, which is an object that holds all of the homomorphic encryption parameters and keys, and serves as the environment in which you create, manipulate and decrypt ciphertexts.

The TenSEAL context is configured by four key parameters: the polynomial modulus degree n , the coefficient modulus chain q , the Galois keys and the global scale. The polynomial modulus degree n sets the ring dimension for encryption. The larger n increases the rigidity and security of the lattice and allows more plaintext slots per ciphertext. The coefficient modulus chain q is specified as a sequence of bit-lengths, each consumed by one homomorphic multiplication. Its length therefore determines the maximum multiplicative depth before a ciphertext refresh is required. Galois keys are special key-switching matrices that enable cyclic rotations of encrypted slot vectors, which are essential for computing inner products and other aggregated operations directly on ciphertexts, and the global scale factor controls the fixed-point encoding precision of real numbers. A higher scale reduces rounding error, but also accelerates noise growth during multiplication.

A context is created with polynomial modulus degree $n = 8192$ and a coefficient modulus chain $q = [60, 40, 40, 40, 40, 40, 40, 60]$, which offers a balance between security and computation depth. Galois keys are generated to support encrypted vector operations, such as rotation. The global scale is set to 2^{15} to manage precision in encrypted computations.

Due to the limitations of HE scheme in supporting non-polynomial functions, the traditional sigmoid function is approximated using a second-degree polynomial quadratic sigmoid. This quadratic sigmoid approximation balances computational efficiency and functional performance, enabling encrypted prediction without violating the algebraic constraints of the CKKS scheme.

The encrypted feature vectors (e.g., gender, age, academic pressure, etc.) and the corresponding labels are used for training, and during evaluation, the encrypted feature vectors from the test set are passed through the model to generate predictions.

D. Security Analysis

The proposed system ensures end-to-end confidentiality of student mental health data by using the CKKS HE scheme at every stage except within the data owner's environment. The TenSEAL-based implementation guarantees that without the secret key, only ciphertexts could be seen, and no plaintext information about individual responses, intermediate gradients, or final weights can be recovered. The choice of polynomial modulus degree ($n=8192$) and coefficient modulus chain ($q = [60, 40, 40, 40, 40, 40, 40, 60]$) provides a 192-bit security level, resistant to lattice-based attacks. This modulus chain supports 8 levels of multiplicative depth, which was specifically chosen to accommodate the homomorphic computation of logistic regression, including dot product operations and a

quadratic approximation of the sigmoid function. Since the sigmoid cannot be directly computed in encrypted form, we use a second-degree polynomial, which keeps the multiplicative depth low. As a result, this configuration enables us to complete training and inference on encrypted data without the need for bootstrapping, which is computationally expensive. Since all computations (training and inference) are performed on encrypted data, sensitive information is never exposed in plaintext to unauthorized parties, including cloud servers or third-party processors. This satisfies the core requirement of data privacy in ML pipelines. The data holder (e.g., academic institution) is trusted to manage encryption/decryption keys securely that ensure the client-side exclusivity of sensitive output.

IV. EXPERIMENTAL RESULTS

This section illustrates the experimental results obtained from our PPML model. All experiments were carried out on Google Colaboratory using the default CPU runtime, 16GB of RAM, Python 3.11.12, TenSEAL 0.3.16, and scikit-learn 1.6.1. For all experiments, we used a fixed coefficient modulus chain $q = [60, 40, 40, 40, 40, 40, 40, 60]$, which supports 8 levels of multiplicative depth in the CKKS encryption scheme and the global scale is set to 2^{15} .

TABLE II
COMPARISON OF UNENCRYPTED AND ENCRYPTED MODELS

Type	Model	Accuracy	MSE	MAE
Unencrypted	Logistic Regression	0.96	0.0396	0.0396
Encrypted	Logistic Regression	0.8462	0.1538	0.1538

Table. II illustrates the accuracy, MSE, and MAE values of the unencrypted logistic regression model and the encrypted logistic regression model, respectively, using the TenSEAL library. As shown in Table.II, the plaintext logistic regression model achieved 96.0% accuracy, whereas its CKKS-encrypted model, as shown in Table.II, reached 84.62%, a slight drop due to the approximation and noise inherent in the HE scheme.

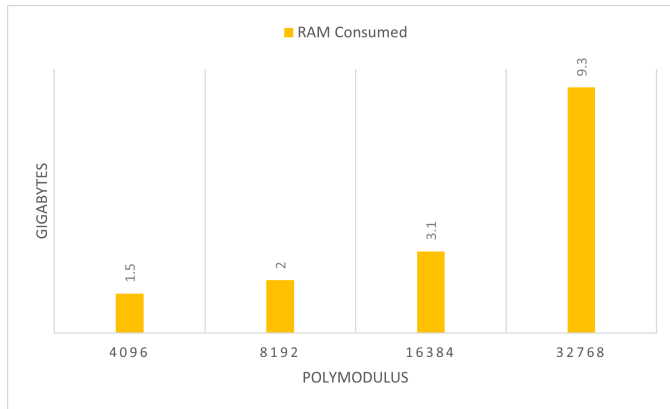


Fig. 4. System Usage

Fig.4 shows the consumption of RAM seen in Google Colab during the training phase of the encrypted model using

TenSEAL library for various polynomial degrees. As the modulus degree increases, both the ciphertext size and the number of coefficient moduli in the CKKS context grow, leading to larger in-memory representations and more intermediate data during homomorphic operations. Consequently, higher polynomial modulus settings incur significantly greater memory usage.

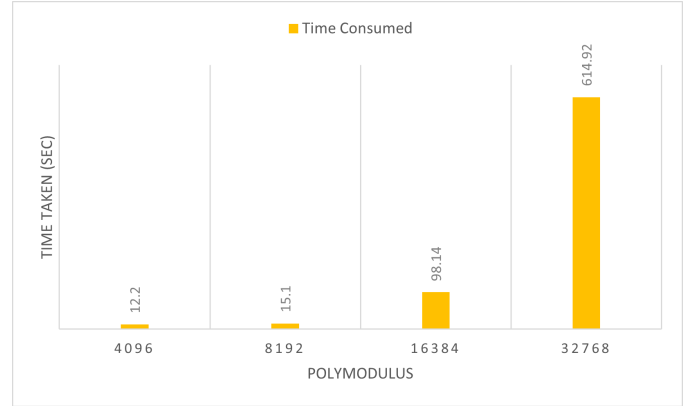


Fig. 5. Time Taken

Fig. 5 illustrates the total time taken for training of the TenSEAL encrypted logistic regression model as a function of the CKKS polynomial modulus degree. As the modulus degree increases, the ciphertexts grow in size and each homomorphic operation requires more underlying coefficient-modulus arithmetic. This added complexity extends the encryption, evaluation, and decryption phases, resulting in significantly longer training times at higher precision settings.

V. CONCLUSIONS AND FUTURE WORK

In this work, we presented a privacy-preserving solution for student mental health prediction through an encrypted logistic regression model. Our approach trains directly on encrypted data using the CKKS homomorphic encryption scheme implemented via the TenSEAL library. To enable efficient computation on ciphertexts, we replaced the standard sigmoid function with a quadratic approximation, maintaining compatibility with HE constraints while working with sensitive survey-style datasets. The experimental results demonstrate that our model achieves 84% accuracy compared to 96% for the unencrypted version, proving that strong privacy guarantees can be maintained with only a modest performance trade-off. Furthermore, we conducted a comprehensive analysis of computational overhead, evaluating memory usage and training time across different polynomial modulus degrees. These findings offer practical insights for balancing security parameters with resource requirements in real-world deployments.

To further strengthen the security and scalability of our framework, in the future, we will integrate Multi-Party Computation (MPC) to decentralize trust in key management and mitigate single points of failure. Additionally, adopting post-quantum-resistant HE schemes (e.g., Kyber-CKKS hybrids)

will future-proof the system against quantum attacks. Finally, dynamic parameter adjustment mechanisms can optimize polynomial degrees in real-time based on data sensitivity and computational constraints, improving efficiency without compromising security. These advances will improve the robustness of privacy-preserving ML in sensitive domains such as mental health.

REFERENCES

- [1] M. Mofatteh, "Risk factors associated with stress, anxiety, and depression among university undergraduate students," *AIMS Public Health*, vol. 8, no. 1, pp. 36–65, 2020.
- [2] Mortier, P., Cuijpers, P., Kiekens, G., et al. (2018), The prevalence of suicidal thoughts and behaviours among college students: a meta-analysis. *Psychological Medicine*, 48(4), 554–565.
- [3] Bruffaerts, R., Mortier, P., Kiekens, G., Auerbach, R. P., Cuijpers, P., Demyttenaere, K., ... & Kessler, R. C. (2018), Mental health problems in college freshmen: Prevalence and academic functioning. *Journal of Affective Disorders*, 225, 97–103.
- [4] Z. Zhang, "Evaluation model of college students' mental health based on neural network," *Journal of Physics: Conference Series*, Vol. 1744, No. 4. IOP Publishing, 2021.
- [5] P. Bhar, A. Bhar, S. Bhattacharyya, P. Shee, et al., "Psychometric Educational Stress and Anxiety Analysis of Undergraduate Students Using Machine Learning Approaches," in *Smart Innovation, Systems and Technologies*, vol. 433, 2025, pp. 413–423, doi:10.1007/978-981-96-1348-9_31.
- [6] Z. Wang, Z. Yang, I. Azimi, and A. M. Rahmani, "Differential Private Federated Transfer Learning for Mental Health Monitoring in Everyday Settings: A Case Study on Stress Detection," *arXiv Preprint*, arXiv:2402.10862, Feb. 2024. [Online]. Available: <https://arxiv.org/abs/2402.10862>.
- [7] Y. Aono, T. Hayashi, L. Wang, and S. Matsuo, "Privacy-preserving deep learning via additively homomorphic encryption," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 5, pp. 1333–1345, May 2018, doi:10.1109/TIFS.2018.2829204.
- [8] J.Fan and F.Vercauteren, "Somewhat Practical Fully Homomorphic Encryption," *IACR Cryptology ePrint Archive*, Report2012/144, 2012.
- [9] Gentry, Craig. "Fully homomorphic encryption using ideal lattices," *Stoc*. Vol. 9. No. 2009. 2009.
- [10] S. Halevi and V. Shoup, "Algorithms in HELib," in *Advances in Cryptology – CRYPTO 2014*, LNCS, vol.8616, Springer, 2014, pp.155–182.
- [11] H.Chen, K.Laine, and R.Player, "Simple Encrypted Arithmetic Library— SEAL v2.2," Microsoft Research, Tech. Rep.MSR-TR-2017-03, 2017. [Online]. Available: <https://github.com/Microsoft/SEAL>
- [12] J.H.Cheon, M.Kim, A.Kim, and Y.Song, "Homomorphic Encryption for Arithmetic of Approximate Numbers," in *Proc.ASIACRYPT2017*, LNCS, vol.10624, Springer, 2017, pp.335–362.
- [13] I.Chillotti, N.Gama, M.Georgieva, and M.Izabachene, "TFHE: Fast Fully Homomorphic Encryption over the Torus," *IACR Cryptol. ePrint Arch.*, Report2016/421, 2016. [Online]. Available: <https://eprint.iacr.org/2016/421>
- [14] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC)*, 2009, pp.169–178.
- [15] A.Benaissa, B.Retiat, B.Cebere, and A.E.Belfedhal, "TenSEAL: A library for encrypted tensor operations using homomorphic encryption," *arXiv preprint arXiv:2104.03152*, 2021. [Online]. Available: <https://arxiv.org/abs/2104.03152>
- [16] S. Fawaz, N. Belal, A. Elrefaey, and M. W. Fakhr, "A Comparative Study of Homomorphic Encryption Schemes Using Microsoft SEAL," in *Proceedings of the 6th International Conference on Advanced Technology and Applied Sciences (ICaTAS 2021)*, Cairo, Egypt, Oct. 2021.
- [17] S. Behera and J. R. Prathuri, "Application of Homomorphic Encryption in Machine Learning," 2020 2nd PhD Colloquium on Ethically Driven Innovation and Technology for Society (PhD EDITS), Bangalore, India, 2020, pp. 1-2, doi: 10.1109/PhDEDITS51180.2020.9315305.
- [18] Y. Aono, T. Hayashi, L. T. Phong, and L. Wang, "Scalable and Secure Logistic Regression via Homomorphic Encryption," in **Proceedings of the 6th ACM Conference on Data and Application Security and Privacy (CODASPY)**, 2016, pp. 142–144, doi: 10.1145/2857705.2857731.
- [19] C. Liu et al., "Efficient and Privacy-Preserving Logistic Regression Scheme based on Leveled Fully Homomorphic Encryption," *IEEE INFOCOM 2022 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, New York, NY, USA, 2022, pp. 1-6, doi: 10.1109/INFOCOMWKSHPS54753.2022.9797933.
- [20] Y. B. Wiryen, N. A. Vigny, J. M. Ngono, and L. Fono, "Leveraging TenSEAL: A Comparative Study of BFV and CKKS Schemes for Training ML Models on Encrypted IoT Data," **International Journal of Information Security and Privacy**, vol. 18, pp. 1–17, 2024, doi: 10.4018/IJISP.356402.
- [21] I.Kynahidwin, "Depression Student Dataset," Kaggle, 2024. [Online]. Available: <https://www.kaggle.com/datasets/ikynahidwin/depression-student-dataset>.
- [22] J.H.Cheon, A.Kim, M.Kim, and Y.Song, "Homomorphic Encryption for Arithmetic of Approximate Numbers," in *Proc. ASIACRYPT 2017*, LNCS, vol.10624, Springer, 2017, pp.3–30.