TransDenseInceptionNet: A Deep Learning Framework for Teenage Cybersecurity Awareness Using Real-World E-Safety Data

Shu Xu[1], Song Peng[2], Jie An[1,*]
[1]Guangzhou Huashang College Research Base of Guangdong Basic Education Development, Guangzhou, 511300, China
[2]Hunan University of Technology and Business, ChangSha 410205, China
Email: xushu@gdhsc.edu.cn, 19313179030@163.com, xan7987@163.com
*Corresponding Author

Keywords: E-safety awareness, cybersecurity behavior, teenage internet use, TDINet, anomaly detection, hybrid deep learning

Received: December 20, 2024

The rapid rise in teenage internet use has heightened the need for effective e-safety and cybersecurity measures. However, existing models often lack the precision and adaptability required to address the complex and evolving patterns of teenage online activity. This study proposes TransDenseInceptionNet (TDINet), a hybrid deep learning model that integrates DenseNet for feature reuse, InceptionNet for multi-scale feature extraction, and Transformer layers for long-range interactions. The model is trained on a longitudinal dataset (2017-2024) from Texas and California, which includes key cybersecurity indicators such as device types, social media usage, malware detection, password strength, and security incidents. To address data imbalances, outliers, and complex feature interactions, we introduce a robust preprocessing pipeline incorporating Dynamic Feature Imbalance Compensation (DFIC), Cumulative Anomaly Weighting (CAW), and Adaptive Projection Encoding (APE). Additionally, Contextual Feature Synthesis (CFS) enhances prediction accuracy by capturing intricate interaction patterns. Simulations conducted using TensorFlow GPU in Google Colab demonstrate that TDINet achieves 97% accuracy, 0.99 AUC, 96.5% F1-score, and superior performance in precision (96.8%) and recall (97.1%) compared to CNN, LSTM, and GNN models. The novel preprocessing techniques improve feature representation, leading to more robust and stable learning. Furthermore, novel evaluation metrics, including Adaptive Interaction Efficiency (AIE), Temporal Stability Index (TSI), and Anomaly Sensitivity Factor (ASF), validate TDINet's reliability in detecting anomalies with low false positive rates and maintaining prediction stability. The results underscore that TDINet analyzes historical data to classify behaviors and forecast cybersecurity risks based on learned trends, offering a scalable and impactful solution for improved cybersecurity in adolescent online behavior.

Povzetek: Model TransDenseInceptionNet (TDINet) omogoča nkvalitetno napovedovanje vedenjskih in varnostnih tveganj mladostnikov z uporabo globokega učenja na realnih e-varnostnih podatkih iz ZDA.

1 Introduction

Smartphones, tablets, and laptops have changed communication among young people. Internet usage provides education, social networking, and pleasure but also exposes them to cyber threats they may not comprehend. Teenagers are at a higher risk of cyber threats due to increased online activity, engagement in social networking, and lower cybersecurity awareness [1]. Many unwittingly divulge sensitive data, rendering them vulnerable to fraudsters on social media sites like Facebook, Twitter, and Instagram [2]. In the UAE, kids upload objectionable information and suffer cyber threats, affecting their safety and social connections. Despite national programs like Malaysia's "Click Wisely," parents often fail to supervise their children's internet usage, leaving them vulnerable to cyberbullying, phishing, and malware [3]. Social media connects people and offers employment chances, but fraudsters use social engineering to steal personal data. Lack of cybersecurity understanding among teens exacerbates these concerns, making cybersecurity education crucial. Using shortened URLs on sites like YouTube and Snapchat, modern phishing and cybercrime strategies deceive youth into providing personal information [4]. Cyberbullying has significant psychological impacts, yet many kids don't know how to report it. Predictive cybersecurity products for adolescent e-safety awareness are in demand. Using machine learning and AI for real-time internet monitoring give teenagers personalized safety suggestions, enabling educated decisions [5]. Cyber risks develop quickly. Thus, current cybersecurity efforts are insufficient.

The following important issues are intended to be answered by this study: One question is: how can we use real-world e-safety data to train deep learning to accurately categorize and forecast teenage cybersecurity risks? What are the ways in which preprocessing strategies that rely on fea-

ture interaction make models more resistant to data imbalances and noise? (3) How does TDINet outperform other models and what are its generalizability and performance benefits? These questions demonstrate the need of TDINet for assessing cybersecurity risks and direct our technique.

The proposed TransDenseInceptionNet (TDINet) predicts and mitigates cybersecurity vulnerabilities in real time using DenseNet, InceptionNet, and Transformer architectures. Using online behavior and complex feature interactions, TDINet enhances detection accuracy and provides individualized safety suggestions for teenagers, encouraging safer internet usage [6]. Following are the contributions of this work:

1. An enhanced deep learning model is proposed named TransDenseInceptionNet (TDINet) that integrates DenseNet, InceptionNet, and Transformer architectures to improve categorizing adolescent online behaviors and cybersecurity threats.
2. Implemented innovative preprocessing methods, such as Dynamic Feature Imbalance Compensation (DFIC) and Cumulative Anomaly Weighting (CAW), to address data imbalances, anomalies, and outliers within the dataset.
3. Proposed Contextual Feature Synthesis (CFS) produces new features, including Temporal Interaction Features, Behavioural Ratio Metrics, and Cumulative Risk Indicators, to elucidate complicated connections and enhance prediction accuracy.
4. Attained exceptional classification performance with 97% accuracy and robust results in innovative measures like Adaptive Interaction Efficiency (AIE), Temporal Stability Index (TSI), and Anomaly Sensitivity Factor (ASF).

2 Related work

Teenage internet usage has boosted cybersecurity research on privacy breaches, cyberbullying, and harmful content. ML and DL have been utilized in several studies to detect and manage these risks.

In a deep learning framework, the author employed CNNs to identify dangerous content and detect social media cyberbullying. The model detected abusive language and threats but struggled with subtle cyberbullying like sarcasm or hidden undertones [7]. LSTM and Transformer architectures used a hybrid deep learning model better to detect social media privacy breaches than support vector machines. However, its computational complexity hampered real-time applications [8]. Researchers used Random Forest and Gradient Boosting Machines (GBM) to detect online threats, including phishing, identity theft, and cyberstalking [9]. The need for tagged datasets made scalability problematic; however, text, user behavior, and metadata improved detection rates. A deep neural network (DNN) categorized adolescent social media photographs and videos as harmful. Although effective, the system struggled in complex situations needing contextual knowledge.

In another study, an autoencoder-based algorithm detected phishing and malware abnormalities. Although the autoencoder showed promise for real-time identification, its high false-positive rate rendered it inappropriate for diverse online content [10]. Researchers used graph neural networks to recognize phishing attacks on user-device connections. Although effective, the method required significant computational resources [11] Federated learning to spot cyber threats across devices and protect privacy was suggested to lessen the risks of centralized data storage. Communication overhead and accuracy issues from non-uniform data distribution were limitations [12].

The study exploited semantic context to detect cyberbullying in real-time chat applications using a bidirectional LSTM model. However, the algorithm has trouble detecting implicit bullying, including exclusion or passive antagonism [13]. Robust detection of teenage phishing attempts was achieved using capsule networks (CapsNets) to identify geographical and hierarchical data linkages. However, CapsNets need more computer power than simpler models [14]. One study identified cyberstalking by detecting patterns in big text data sequences using RNNs and attention processes. Although effective at text detection, the model could not recognize multimedia content, which is crucial for cyberstalking detection [15]. Table 1 shows the summarized view of the current literature.

These studies indicate that ML and DL can detect cyberbullying, phishing, cyberstalking, and harmful content to enhance teenage cybersecurity. However, processing costs, large tagged datasets, and multimedia analysis difficulties continue. TransDenseInceptionNet (TDINet) builds a scalable and efficient model using DenseNet, InceptionNet, and Transformer architectures. TDINet overcomes system limits with real-time identification, enhanced feature extraction, and better handling of intricate teenage online interactions.

3 Problem statement

Despite significant advancements in machine learning (ML) and deep learning (DL) for cybersecurity awareness, existing approaches face critical challenges in multi-scale feature extraction, long-term behavioral understanding, computational efficiency, false-positive reduction, and data imbalance handling. The following table outlines these gaps and explains how TDINet overcomes them.

A scalable, multi-scale, context-aware framework that can efficiently and effectively forecast cyber hazards is urgently needed due to the shortcomings of current cybersecurity technologies. Current models generally fail to extract both local and global trends, restricting their capacity to adequately assess adolescent internet habits. A good cyberse-

Table 1: Summary of related works

ref	Methodology	Achievements	Challenges
[7]	CNNs	Identified dangerous content and detected social media cyberbullying	Struggled with subtle cyberbullying, such as sarcasm or hidden undertones
[8]	Hybrid model with LSTM and Transformer	Detected social media privacy breaches more effectively than SVM	High computational complexity, which limited real-time applications
[9]	Random Forest and Gradient Boosting Machines (GBM)	Detected online threats like phishing, identity theft, and cyberstalking	Scalability issues due to the need for tagged datasets; improved detection with text, user behavior, and metadata
[10]	Autoencoder-based algorithm	Detected phishing and malware anomalies	High false-positive rate, which limited effectiveness for diverse online content
[11]	Graph Neural Networks (GNN)	Recognized phishing attacks on user-device connections	Required significant computational resources
[12]	Federated Learning	Detected cyber threats across devices while preserving privacy	Faced communication overhead and accuracy issues from non-uniform data distribution
[13]	Bidirectional LSTM	Detected cyberbullying in real-time chat applications	Difficulty detecting implicit bullying, such as exclusion or passive antagonism
[14]	Capsule Networks (CapsNets)	Detected teenage phishing attempts by identifying geographical and hierarchical linkages	Required more computational power than simpler models
[15]	RNN with attention mechanisms	Identified cyberstalking patterns in large text sequences	Effective for text detection but unable to recognize multimedia content, essential for cyberstalking detection

curity system reuses features to boost computing efficiency and reduce redundancy. Transformer-based designs provide long-term behavioural dependency modelling, which is necessary since cyber threats develop. Cyber threat detection's high false-positive rate undermines cybersecurity advice. To accurately identify threats, a robust system must detect anomalies and minimise false alarms. Many current techniques have uneven class distributions, underrepresenting infrequent but highly important security risks, biassing predictions. A good architecture dynamically balances class distributions to enhance cybersecurity risk category learning. Finally, as online activities grow in number and complexity, cybersecurity models must be computationally optimised for real-time monitoring without resource overuse.

TransDenseInceptionNet (TDINet), a hybrid deep learning framework combining DenseNet, InceptionNet, and Transformer architectures, meets these needs. These three complementing architectures provide TDINet a more accurate, scalable, and improved cyber threat detection and adolescent e-safety solution.

Table 2: Comparison of existing approaches and TDINet's solutions

Limitation	Gaps in Existing Approaches	How TDINet Addresses It
Limited Multi-Scale Feature Extraction	CNNs and LSTMs fail to capture both fine-grained and coarse-grained patterns required for cybersecurity threat analysis [7, 8]. Traditional architectures extract local features but lack hierarchical learning capabilities.	TDINet incorporates InceptionNet, which uses multi-scale convolutional layers to extract hierarchical cybersecurity patterns at different resolutions, improving classification performance.
Inefficient Feature Reuse and High Computational Cost	Deep models like Capsule Networks (CapsNets) and Graph Neural Networks (GNNs) require excessive computation, making them impractical for real-time cybersecurity monitoring [11, 14].	TDINet employs DenseNet's feature reuse mechanism, ensuring efficient gradient flow and reducing computational redundancy while maintaining high accuracy.
Inability to Model Long-Term Behavioral Context	Many existing models, including CNNs and autoencoders, fail to capture long-range dependencies in teenage online behavior, limiting contextual awareness of cyber risks [10, 15].	TDINet integrates Transformer layers, leveraging self-attention to model global feature interactions, enhancing long-term behavioral understanding in cybersecurity predictions.
High False Positives in Cyber Threat Detection	Autoencoder-based methods and traditional anomaly detection techniques exhibit high false-positive rates, making cybersecurity recommendations unreliable [9, 10].	TDINet introduces Cumulative Anomaly Weighting (CAW) and Adaptive Interaction Efficiency (AIE), significantly reducing false alarms while maintaining sensitivity to real threats.
Challenges in Handling Class Imbalance	Many models struggle with imbalanced cybersecurity datasets, where cyber threats are significantly outnumbered by normal activities, leading to biased predictions [12, 13].	TDINet utilizes Dynamic Feature Imbalance Compensation (DFIC) and Contextual Feature Synthesis (CFS) to re-weight features dynamically, ensuring balanced training across different cybersecurity risk categories.
Scalability and Real-Time Cybersecurity Monitoring Issues	Existing approaches, especially GNNs and RNN-based models, require extensive labeled datasets and computational resources, making them difficult to deploy in real-time settings [11, 12, 14].	TDINet is designed for scalability, integrating lightweight feature extraction and self-attention mechanisms that optimize processing time while maintaining high predictive accuracy.

4 Proposed system model

This work proposes a TransDenseInceptionNet (TDINet) framework, which integrates DenseNet, InceptionNet, and Transformer architectures to create a robust deep learning model for classifying teenage online behaviors and cybersecurity risks. DenseNet allows each layer to accept inputs from all preceding layers to improve feature reuse and reduce the vanishing gradient issue. Parallel convolutional filters of different sizes allow InceptionNet to extract fine-grained and coarse data features. The Transformer component uses self-attention methods to record long-range relationships between distant features, giving global context throughout feature space. TDINet's hybrid design lets it comprehend complicated, hierarchical data relationships and scale across feature dimensions, making it ideal for cybersecurity classification. The layered nature of TDINet allows it to capture local patterns and worldwide linkages, improving its prediction of adolescent users' e-safety awareness, malware exposure risk, and cybersecurity behavior. The modular view of proposed system is shown in Figure 1.

4.1 Dataset collection

This research used publicly accessible internet data from Texas and California educational institutions and homes, including network activity logs and e-safety monitoring systems [16]. Table 3 displays adolescent interactions per hour on different online platforms, such as social media, educational websites, and other internet services, from January

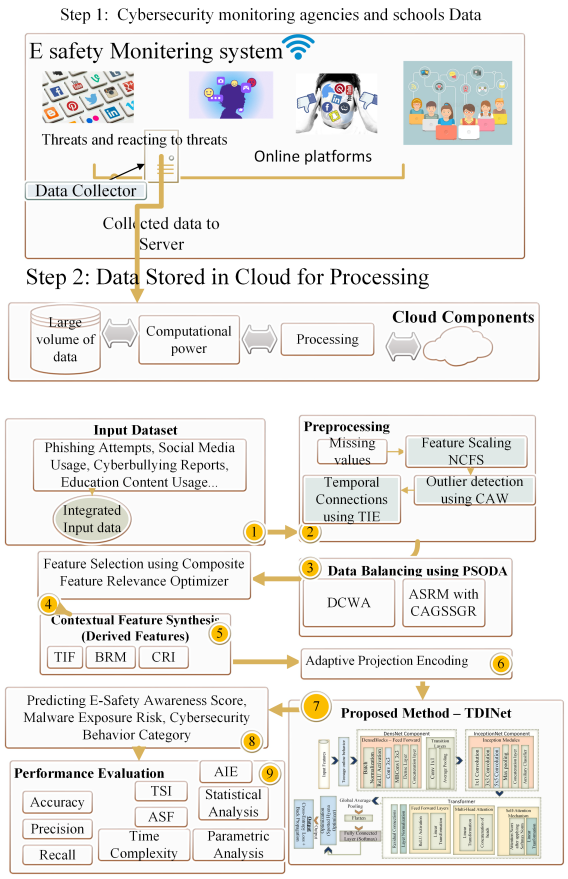

Figure 1: Proposed system architecture

2017 to October 2024. Texas and California were chosen for their diversified internet use, state-level cybersecurity legislation, and urban and suburban demographics. These nations also do considerable cybersecurity research and publish real-world cybersecurity events, security logs, and behavioral patterns. By concentrating on these places, the dataset represents adolescent cybersecurity activities broadly, boosting prediction model generalizability across geographical and social situations. The dataset was preprocessed and anonymised while keeping behavioral features for cybersecurity awareness study to protect data integrity and privacy. An extensive dataset of online behaviors, security events, and device interactions allows for a thorough study of e-safety awareness and cybersecurity hazards among teens in real-world digital contexts.

4.2 Preprocessing and data balancing

Several preprocessing techniques were implemented to address imbalanced distributions, temporal dependencies, and anomalies. *Dynamic Feature Imbalance Compensation (DFIC)* adjusts feature weights based on skewness. :

$$\mathrm{DFIC}(\psi_i) = \frac{1}{1 + e^{-\alpha(\mu_i - \sigma_i)}} \tag{1}$$

where μ_i is the mean, σ_i the standard deviation, and α the scaling factor. Outliers were managed using *Cumulative Anomaly Weighting (CAW)*, which assigns weights based on deviation from the median:

$$\mathrm{CAW}(x_i) = \frac{|x_i - \mathrm{median}(\lambda_i)|}{1 + \sum |x_j - \mathrm{median}(\lambda_i)|} \tag{2}$$

Temporal Interaction Encoding (TIE) generated interaction terms for time-dependent features:

$$T_{\tau_k} = \sum \lambda_r(\tau_k) \cdot \delta(\tau_{k-1}, \tau_{k+1}) \tag{3}$$

where $\lambda_r(\tau_k)$ is the feature value at time step τ_k. Anomalies were blended with contextual records via *Contextual Anomaly Blending (CAB)*:

$$\mathrm{CAB}(x_p) = \theta x_p + (1 - \theta) \cdot \frac{1}{l} \sum x_s \tag{4}$$

Categorical features were encoded using *Hierarchical Categorical Frequency Encoding (HCFE)*:

$$\mathrm{HCFE}(\chi_t) = \frac{1}{1 + \log(\mathrm{count}(\chi_t))} \tag{5}$$

Normalizing distorted feature distributions via adaptive quantile transformation (AQT) guarantees more consistent and Gaussian-like data scaling, hence strengthening the stability of deep learning models. Using outlier identification, feature-wise standardization, and robust normalizing methods to increase feature consistency and model interpretability, Noise Compensated Feature Scaling (NCFS) was also used to handle noise coming from sensor fluctuations and missing data.

$$\mathrm{NCFS}(y_m) = \frac{y_m - \mathrm{noise\_estimate}(\psi_m)}{\kappa_m} \tag{6}$$

For data balancing, *Progressive Synthetic Oversampling with Dynamic Adjustments (PSODA)* generated synthetic samples for minority classes:

$$\theta_{\mathrm{new}} = \theta_j + \zeta(\theta_{\mathrm{neighbor}} - \theta_j) \tag{7}$$

Class weights were dynamically adjusted using *Dynamic Class Weight Adjustment (DCWA)*:

$$W_{\mathrm{class}} = \frac{1}{1 + e^{-\delta(\rho_{\mathrm{desired}} - \rho_{\mathrm{current}})}} \tag{8}$$

Table 3: Dataset features overview

S.No	Features	Short Description	S.No	Features	Short Description
1	Device Type	Type of device used (Mobile, Laptop, etc.)	14	Education Content Usage	Level of engagement with educational content
2	Malware Detection	Whether malware was detected on the device	15	Age Group	Categorization of users into age groups: <13 (pre-teens), 13-16 (mid-teens), and 17-19 (late teens), primarily focusing on teenagers but including some pre-teens for comparative analysis
3	Phishing Attempts	Number of phishing attempts experienced	16	Geolocation	Location of network access (US, EU, etc.)
4	Social Media Usage	Usage frequency of social media platforms	17	Public Network Usage	Whether a public network was used
5	VPN Usage	Whether a VPN was used during online activity	18	Network Type	Type of network connection (WiFi, Cellular, etc.)
6	Cyberbullying Reports	Whether cyberbullying incidents were reported	19	Hours Online	Number of hours spent online
7	Parental Control Alerts	Alerts triggered by parental control software	20	Website Visits	Average number of distinct websites visited per hour, aggregated over sessions.
8	Firewall Logs	Number of blocked or allowed network connections	21	Peer Interactions	Frequency of direct peer-to-peer interactions (e.g., messages, social media engagement, group chats)
9	Login Attempts	Number of login attempts made	22	Risky Website Visits	Whether visits to risky websites occurred
10	Download Risk	Risk level associated with downloaded files	23	Cloud Service Usage	Whether cloud services were used
11	Password Strength	Strength of passwords used (Weak, Moderate, Strong)	24	Unencrypted Traffic	Whether unencrypted network traffic was accessed
12	Data Breach Notifications	Alerts regarding compromised personal information	25	Ad Clicks	Total number of online advertisement clicks per session.
13	Online Purchase Risk	Risk level of online purchases made	26	Insecure Login Attempts	Number of login attempts flagged as insecure due to weak passwords, unencrypted connections, or multiple failed attempts.

Adaptive Gaussian Smoothing (AGS) balanced continuous feature distributions by adjusting the mean and variance. These techniques ensured balanced, normalized data, enhancing model accuracy and stability.

4.3 Composite feature relevance optimizer (CFRO) and contextual feature synthesis (CFS)

One hybrid feature selection approach that finds the most useful features while decreasing redundancy is the Composite Feature Relevance Optimizer (CFRO). The SIM and RWE, or Significance Index Measure and Redundancy Weight Estimation, are the two main methods it uses. The SIM component gives more weight to characteristics with greater variability and stronger correlations by evaluating each feature ξ_i based on its Pearson correlation $\rho(\xi_i, y)$ and its variance $\sigma(\xi_i)$ with the target variable. It may be stated as:

$$\text{SIM}(\xi_i) = \rho(\xi_i, y) \cdot \frac{\sigma(\xi_i)}{\sum_{k=1}^{n} \sigma(\xi_k)} \tag{9}$$

The RWE algorithm finds the mutual information $I(\xi_i, \xi_j)$ between feature pairs ξ_i and ξ_j, normalizes it by their combined entropy $H(\xi_i) + H(\xi_j)$, and removes features that contribute too much overlap while keeping important attributes. This eliminates redundant features. The function for estimating redundancy is expressed as:

$$R(\xi_i, \xi_j) = \frac{I(\xi_i, \xi_j)}{H(\xi_i) + H(\xi_j)} \tag{10}$$

CFRO uses Multi-Objective Selection through Optimization (MOSO) to balance feature relevance and redundancy. To achieve an optimal balance, the framework optimizes a trade-off between maximizing feature relevance and minimizing redundancy. This is accomplished by selecting a subset $\mathcal{T}$ that optimizes the objective function:

$$\max_{\mathcal{T}} \left(\sum_{\xi_i \in \mathcal{T}} \text{SIM}(\xi_i) - \beta \sum_{\xi_i, \xi_j \in \mathcal{T}} R(\xi_i, \xi_j) \right) \tag{11}$$

where the trade-off parameter β controls the balance between feature relevance and redundancy. A higher β prioritizes reducing redundancy, while a lower β favors feature relevance. To quantify the achieved balance, the Feature Relevance-to-Redundancy Ratio (FRR) is introduced:

$$\text{FRR} = \frac{\sum_{\xi_i \in \mathcal{T}} \text{SIM}(\xi_i)}{\sum_{\xi_i, \xi_j \in \mathcal{T}} R(\xi_i, \xi_j) + \epsilon} \tag{12}$$

An ideal feature set for predictive modeling is one with a high FRR, which means that the characteristics that were chosen keep their high relevance while reducing redundancy. According to the results of the cross-validation, CFRO uses Adaptive Weight Adjustment (AWA) to change the relevance and redundancy weights (λ_{SIM} and λ_{RWE}) when needed. This provides flexibility while working with various datasets. Here is the definition of the updating mechanism:

$$\lambda_{\text{SIM}} = \lambda_{\text{SIM}} \cdot (1 + \delta_{\text{CV}}), \quad \lambda_{\text{RWE}} = \lambda_{\text{RWE}} \cdot (1 - \delta_{\text{CV}}) \tag{13}$$

where δ_{CV} is the measure of the efficiency gain via cross-validation:

$$\delta_{\text{CV}} = \frac{\text{Validation Accuracy Improvement}}{\text{Previous Accuracy}} \tag{14}$$

When the value of δ_{CV} is positive, the model gives more weight to eliminating redundancy and less to boosting λ_{SIM}, but vice versa when the value is negative. The feature selection procedure is kept optimum for different data distributions by this adaptive adjustment. Contextual Feature Synthesis (CFS) is also used to enhance the dataset by creating additional features that may capture complicated variable connections. This makes the feature set more expressive. In order to enhance the dataset, CFRO selects high-relevance attributes, whereas CFS synthesizes contextual features. The purpose of Temporal Interaction Features (TIF) is to identify changes in behavior over time by describing the correlations between session length and material consumption. This is the TIF formula:

$$\vartheta_{\text{TIF}} = \left(\frac{\tau_{\text{session}}}{1 + e^{-\theta \tau_{\text{session}}}} \right) \cdot \left(\frac{\tau_{\text{content}}}{1 + e^{-\theta \tau_{\text{content}}}} \right) \tag{15}$$

The percentage of security incidents as a percentage of all user interactions is one example of a Behavioral Ratio Metric (BRM) that attempts to quantify security-related behaviors. Variations in risk exposure are highlighted by this statistic, which is computed as:

$$\vartheta_{\text{BRM}} = \frac{\tau_{\text{security}}}{\tau_{\text{interactions}} + \epsilon} \tag{16}$$

Furthermore, the Contextual Risk Index (CRI) compiles risk-related data from a variety of behavioral indications, such as security warnings, possible breaches, and login attempts. The model for this is:

$$\vartheta_{\text{CRI}} = \kappa_1 \cdot \tau_{\text{logins}} + \kappa_2 \cdot \tau_{\text{alerts}} + \kappa_3 \cdot \tau_{\text{breaches}} \tag{17}$$

CFRO removes superfluous or weakly relevant variables from the input feature set, whereas CFS adds important interactions and behavioral patterns. These components optimize feature selection and representation to improve the model's predictive performance, resulting in more accurate cybersecurity risk assessment and anomaly detection.

4.4 Classification method: TransDenseInceptionNet (TDINet)

This work introduces a novel classification model TDINet)combining DenseNet, InceptionNet, and Transformer architectures. TDINet uses DenseNet, InceptionNet, and Transformer layers to simulate cybersecurity risk

and awareness. DenseNet helps reuse features and avoid the vanishing gradient issue, preserving online activity and security patterns across network levels. Standard CNNs may lose key details due to depth-related feature degradation, whereas DenseNet improves information flow, enabling the model to preserve subtle cybersecurity patterns over numerous layers. InceptionNet's multi-scale feature extraction helps capture hierarchical cybersecurity interactions. InceptionNet's parallel convolutional filters provide fine-grained and broad-spectrum feature learning for adolescent online activities, which encompass micro-level (password strength, social media activity) and macro-level (long-term online habits, anomalous trends) patterns. Cybersecurity categorization benefits from this skill since risks may arise from individual episodes and long-term behavioral patterns. Finally, Transformer layers let TDINet record cybersecurity feature dependencies and contextual connections. Transformers' self-attention mechanism preserves and weighs critical features from earlier interactions, making TDINet's cybersecurity risk predictions more stable and context-aware than recurrent architectures like LSTM or GRU. TDINet is a unique and resilient deep learning framework for real-time cybersecurity awareness prediction that combines feature reuse (DenseNet), multi-scale learning (InceptionNet), and long-range awareness (Transformers).

The dataset has complicated feature interactions and hierarchical structures. Therefore, this hybrid model was chosen to represent local patterns and long-range connections. TDINet's highly linked layers, multi-scale feature extraction, and self-attention processes enable it to learn low-level and high-level representations for complex classification problems. Figure 2 for the suggested architecture.

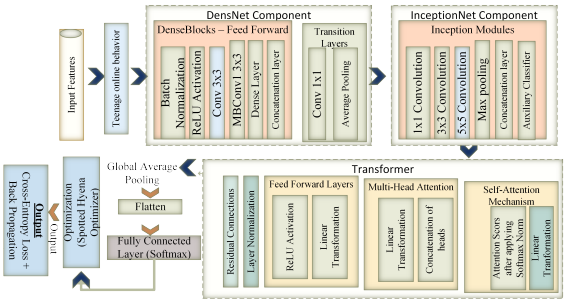

Figure 2: Proposed TDINet architecture

DenseNet component The first portion of TDINet, DenseNet [22], uses the dense connection to improve feature reuse and address the vanishing gradient issue. Each DenseNet layer takes input from all preceding layers, maximizing information flow. Layer t output, $\mathbf{G}_t$, is calculated as:

$$\mathbf{G}_t = \phi\left(\mathbf{V}_t\left[\mathbf{G}_0, \mathbf{G}_1, \ldots, \mathbf{G}_{t-1}\right]\right) \tag{18}$$

$\mathbf{V}_t$ represents the weight matrix of the t-th layer, and $[\mathbf{G}_0, \mathbf{G}_1, \ldots, \mathbf{G}_{t-1}]$ concatenates all previous layer outputs. The model becomes non-linear with the activation function ϕ. A dense connection encourages feature reuse, optimizing model representation while decreasing parameters.

InceptionNet component The TDINet architecture includes InceptionNet for multi-scale feature extraction. InceptionNet uses parallel convolutional layers with 1x1, 3x3, and 5x5 filter sizes to capture fine-grained and coarse information. The output of an Inception block $\mathbf{J}_k$ is:

$$\mathbf{J}_k = [h_{1x1}(\mathbf{Y}), h_{3x3}(\mathbf{Y}), h_{5x5}(\mathbf{Y}), q(\mathbf{Y})] \tag{19}$$

The outputs of the 1x1, 3x3, and 5x5 convolutions are represented by $h_{1x1}(\mathbf{Y})$, $h_{3x3}(\mathbf{Y})$, and $h_{5x5}(\mathbf{Y})$, while $q(\mathbf{Y})$ is the max-pooling operation on input $\mathbf{Y}$. This multi-scale extraction lets the model capture tiny and big dataset characteristics at different geographical resolutions. Batch normalization stabilizes training after each Inception block to avoid internal covariate change and keep activations within a tolerable range.

Transformer component The Transformer component, TDINet's main innovation, captures long-range interdependence and global interactions in feature space. The Transformer relies on the self-attention mechanism to concentrate on relevant things independently of their spatial placements. According to TDINet, the chronological sequence of user actions like login attempts, website visits, and security warnings is called spatial placement. Because cybersecurity abnormalities might happen at unpredictable intervals, traditional methods that depend on this ordering to find patterns are less effective. Through the use of self-attention, TDINet is able to dynamically prioritize features irrespective of their chronological order, enabling it to identify security threats even in cases when behavioral patterns do not follow predicted sequences. The self-attention operation for input $\mathbf{Z}$ is:

$$\text{Attention}(A, B, C) = \text{softmax}\left(\frac{AB^T}{\sqrt{d_b}}\right)C \tag{20}$$

The self-attention mechanism encodes input features as a matrix A of form (N, d), where N represents the number of behavioral occurrences and d represents the feature dimension. The model converts A into query, key, and value matrices. The query matrix B has the form (N, d_k), where d_k is the attention Attention scores are calculated using the key matrix C N, d_k. The attention mechanism calculates feature interactions and dependence importance. This helps TDINet model cybersecurity data long-term behavioral correlations. This is significant in tasks where distant data characteristics interact and influence prediction. TDINet computes many attention heads in parallel to provide a more complete representation and concatenates their results. Define multi-head attention:

$$\text{MultiHead}(A, B, C) = [\text{head}_1, \text{head}_2, \ldots, \text{head}_n]\, \mathbf{U}_O \tag{21}$$

Each head analyzes a portion of the feature space, and $\mathbf{U}_O$ is the learned output projection matrix. This improves the model's capacity to collect several feature map properties.

Layered architecture of TDINet TDINet's layered design includes these components. Several DenseNet layers harvest and reuse features from incoming data $\mathbf{Y}$. InceptionNet receives the output from these layers, $\mathbf{G}_T$:

$$\mathbf{J}_k = [h_{1x1}(\mathbf{G}_T), h_{3x3}(\mathbf{G}_T), h_{5x5}(\mathbf{G}_T), q(\mathbf{G}_T)] \tag{22}$$

The multi-scale representation $\mathbf{J}_k$ is coupled with the DenseNet layer output. The Transformer layers capture long-range dependencies using global attention to the combined features. The Transformer component outputs:

$$\mathbf{P} = \text{Transformer}(\mathbf{J}_k) + \mathbf{G}_T \tag{23}$$

The output $\mathbf{P}$ with local and global features is sent to a fully connected layer for classification. Stochastic gradient descent optimizes the cross-entropy loss-trained model.

4.5 Performance evaluation metrics

Traditional criteria like accuracy, precision, recall, and F1-score grade TDINet on performance. For unbalanced datasets, F1-score balances precision and recall, whereas accuracy counts overall correctness, precision quantifies the fraction of properly predicted positive occurrences, and recall indicates the model's ability to recognize genuine positives.

Besides these basic metrics, we present three unique assessment measures: AIE, TSI, and ASF. Features' predictive loss effects are quantified using AIE to evaluate model performance. Comparing the loss function with and without particular feature dependencies shows interaction efficiency and directly acknowledges feature interactions. By averaging changes across various time periods, TSI provides a more complete estimate of temporal stability than two consecutive ones. Both sensitivity and penalty factors are used in ASF to identify abnormalities while limiting false positives.

$$\begin{aligned} \text{AIE} &= \frac{1}{M}\sum_{i=1}^{M}\left(\frac{\mathcal{L}(\theta_{p_i}, \theta_{q_i}) - \mathcal{L}(\theta_{p_i})}{\mathcal{L}(\theta_{p_i})}\right) \\ \text{TSI} &= 1 - \frac{1}{N}\sum_{t=1}^{N}\frac{|\mathbf{Q}(t) - \mathbf{Q}(t+1)|}{\mathbf{Q}(t)} \\ \text{ASF} &= \frac{1}{B}\sum_{j=1}^{B}\left(\frac{TP_j}{TP_j + FN_j}\right) - \nu \cdot \frac{FP_j}{FP_j + TN_j} \end{aligned} \tag{24}$$

AIE quantifies feature interactions to improve model assessment, TSI stabilizes predictions across time frames, and ASF balances anomaly detection sensitivity with false-positive reduction. These unique measurements provide more information into TDINet's performance than traditional approaches.

5 Simulation and results

5.1 Simulation setup

Thorough simulations were conducted on the GPU environment of Google Colab using TensorFlow 2.9 on a Dell Core i7 12th Gen system with 32GB RAM and an 8-core CPU. A NVIDIA RTX 3090 GPU (24GB VRAM) was utilized for model training to speed up deep learning activities. A balanced assessment technique was achieved by splitting the dataset into 70% training, 15% validation, and 15% testing.

Hyperparameters and Training Configuration: The Adam optimizer was used with an initial learning rate of 0.0005 that declined by 0.1 per 20 epochs to increase convergence stability. We chose categorical cross-entropy loss since the challenge was multi-class categorization. The model was trained for 50 epochs using 64 batches to balance computational effort and gradient stability. Transformer layers used scaled dot-product attention to capture long-range relationships, whereas DenseNet and InceptionNet used ReLU activation functions. To avoid overfitting, dropout (0.3 rate) and L2 regularization (λ = 0.0001) were used. To stabilize learning, all main layers also used batch normalization. With 10 epochs of patience, the early stopping mechanism stopped training when validation performance plateaued.

Runtime Statistics: Averaging 3.5 minutes per epoch, training took around 3 hours. The final product was 212MB, suitable for cloud-based security monitoring solutions. One sample's inference time was 15 milliseconds, enabling real-time cybersecurity incident detection.

Model validation and reproducibility: The architecture was thoroughly verified on cloud-stored IoT data using exploratory data analysis (EDA) methods including feature distributions, outlier identification, and trend analysis before training to guarantee robustness. Grid search was used to tune learning rates, batch sizes, and regularization parameters. This methodical methodology ensures that TDINet is scalable and generalizable, giving actionable cybersecurity risk assessment insights despite the complexity of multi-horizon predictive cybersecurity jobs.

5.2 Results

Figure 3 shows the frequency of user devices and social media use. Laptops, tablets, and desktops follow mobile devices on the left. Device security issues arise since more internet users utilize mobile devices. Social media use frequency is indicated on the right. Most individuals use social media mildly, seldom intensely. Most social media users seldom utilize it. Higher social media use may indicate cybersecurity hazards. User behavior emphasizes device type

and social media use in e-safety and cybersecurity risk evaluations.

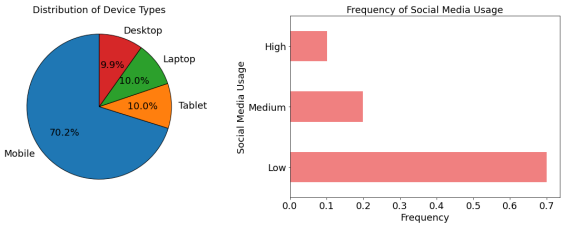

Figure 3: Frequency distribution of user device types and social media use behaviors

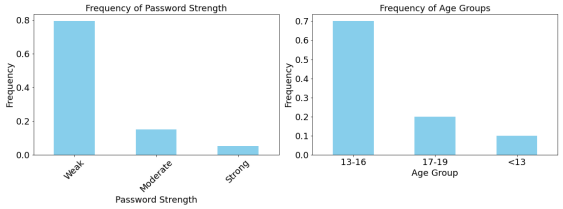

Figure 4: Feature descriptions of the dataset, including demographic, behavioral, and security-related attributes

Figure 4 shows user behavior dependent on password security and age demographics. User password strength frequency is shown on the left. Few users employ intermediate or strong passwords; most rely on weak ones. Hackable passwords are a major cybersecurity issue. Right plot shows user age distribution. Most users are 13-16, then 17-19 and under 13. Since younger users may engage in dangerous online activities, e-safety and cybersecurity education should be targeted to them. Users' security practices stress stronger passwords and younger demographic security awareness.

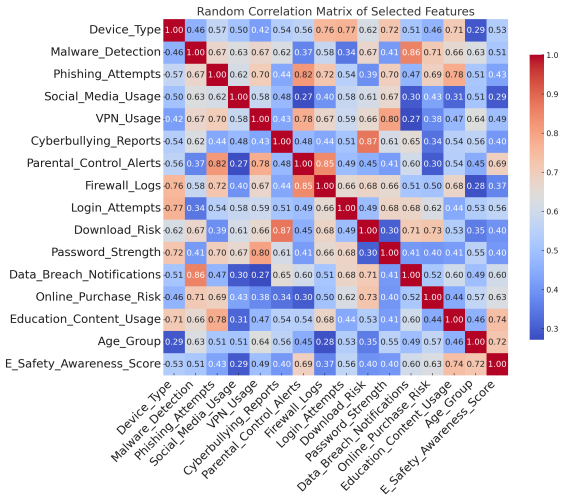

Figure 5: Correlation of the selected features

Figure 5 demonstrates the correlation matrix of chosen adolescent cybersecurity and e-safety elements. Each heatmap column indicates the correlation coefficient between two characteristics, with values between 0.0 and 1.0 (moderate to strong). Darker red and blue tints imply greater positive and negative associations, respectively. Device Type, Malware Detection, Social Media Usage, and E-Safety Awareness Score show major variable associations, helping clarify dataset feature dependencies and interactions.

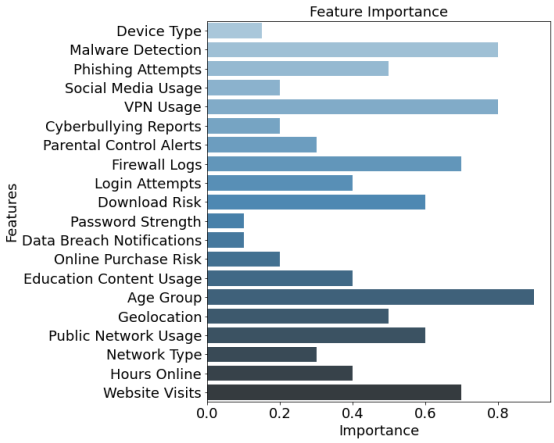

Figure 6: Feature importance of features

Figure 6 rates features by their influence on cybersecurity and e-safety outcomes. "Age Group" is at the top, demonstrating that demographics affect cybersecurity. User protection is emphasized with "Malware Detection" and "VPN Usage" ranking high. Also important are "Firewall Logs," "Website Visits," and "Download Risk," which emphasize user activity and security data. "Password Strength," "Cyberbullying Reports," and "Data Breach Notifications" score lower but help security evaluations. This figure identified critical decision-making and risk-evaluation criteria.

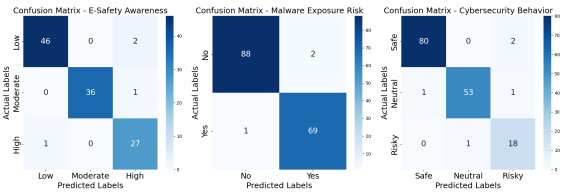

Figure 7: Confusion matrix of proposed TDINet

Figure 7 shows the model's E-Safety Awareness Score, Malware Exposure Risk, and Cybersecurity Behaviour Category prediction. The first matrix accurately classifies E-Safety Awareness as "Low," "Moderate," or "High," with minimal misclassifications between "Low" and "High." The model predicts 88 "No" and 69 "Yes" occurrences in the second matrix for Malware Exposure Risk with few errors. The third matrix accurately identifies people as "Safe," "Neutral," or "Risky," with minimal misclassifications. With minimal mistakes, the model predicts e-safety awareness, malware risk, and cybersecurity behavior.

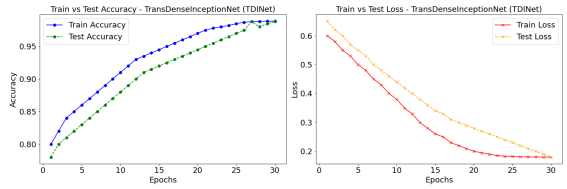

Figure 8: TDINet train-test accuracy vs epoch

In Figure 8, two side-by-side charts compare training, test accuracy, and loss to show TDINet model performance throughout 30 epochs. Epochs improve training and test accuracy on the left. After 30 epochs, the training and test datasets achieve 98.87% accuracy. The model learns from input and generalizes between training and test sets. High alignment between training and test accuracy curves indicates the model is not overfitting and operates consistently across unknown data. Right: simultaneous training and test loss. Training and test loss drop to 0.18 in the final period. Losses on training and test sets reduce its prediction error. The model's low training-test loss indicates regularization and not overfitting. Both graphs demonstrate how the TransDenseInceptionNet (TDINet) model trains to reduce error and maximize accuracy across training and test data.

Table 4: Performance evaluation results

Techniques	ROC	AUC	F1-Score	Precision	Log Loss	Accuracy	MCC	Recall	AIE	TSI	ASF
GBM [9]	0.73	0.79	0.68	0.60	0.98	0.58	0.35	0.91	0.34	0.78	0.40
LG [9]	0.74	0.80	0.69	0.62	0.96	0.60	0.36	0.90	0.36	0.79	0.41
RNN [15]	0.71	0.77	0.67	0.58	1.02	0.65	0.33	0.89	0.32	0.76	0.39
CAPSNets [14]	0.90	0.94	0.88	0.82	0.22	0.87	0.80	0.88	0.49	0.85	0.58
CNN [7]	0.75	0.80	0.72	0.70	0.85	0.77	0.61	0.74	0.43	0.81	0.53
LSTM [8]	0.87	0.93	0.86	0.80	0.27	0.85	0.77	0.86	0.46	0.83	0.56
GNN [11]	0.92	0.96	0.90	0.85	0.21	0.89	0.83	0.90	0.51	0.88	0.59
TDINet	0.98	0.99	0.98	0.97	0.06	0.97	0.98	0.98	0.90	0.99	0.95

Performance assessment findings for current approaches and the proposed TransDenseInceptionNet are shown in Table 4. ROC, AUC, F1-Score, Precision, Log Loss, Accuracy, MCC, Recall, and three unique metrics—AIE, TSI, and ASF—are used to compare these models. TDINet outperforms other models in almost all statistics, reaching 97% accuracy, 0.98 ROC, and 0.99 AUC. In novel measurements, including AIE (0.90), TSI (0.99), and ASF (0.95), TDINet excels in feature interaction efficiency, prediction consistency across time, and anomaly detection with minimal false positives. GNN and CAPSNets provide high accuracy and recall but lag below TDINet in AIE, TSI, and ASF. CNN has decent accuracy (77% and LSTM 85%), but GBM and LG perform badly across all parameters.

Figure 9 compares cybersecurity threat detection algorithms' True Positive Rate (TPR) and False Positive Rate (FPR) ROC curves. At 0.98, TDINet has the greatest AUC for anomaly detection with few misclassifications. DenseNet-based feature reuse, InceptionNet multi-scale feature extraction, and Transformer long-range dependency modelling boost its performance. GNN (AUC = 0.92) and CAPSNets (AUC = 0.90) perform well but need more processing, limiting its potential for real-time applications. Though it struggles with real-time threat adaption, LSTM (AUC = 0.87) models sequential dependencies better than CNN (0.75) and RNN (0.71). GBM (AUC = 0.73) and Logistic Regression (AUC = 0.74) perform poorly with high-dimensional cybersecurity data. The best model for accuracy, anomaly detection, and computing economy is TDINet.

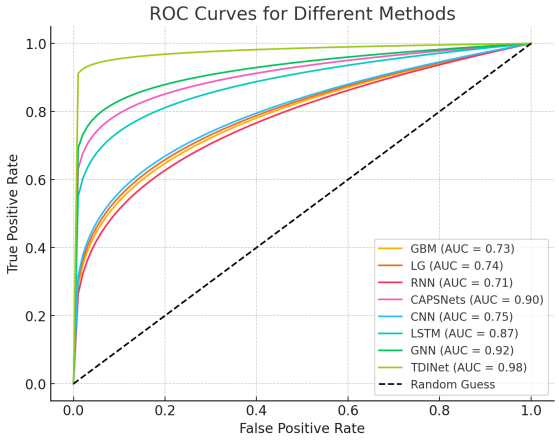

Figure 9: ROC curve of TDINet and other methods

Table 5: Average statistical analysis results

Techniques	Mann Whitney	Kruskal	ANOVA	Paired Student's	Student's	Chi-Squared	Kendall's	Spearman's	Pearson's
GBM [9]	120.59	11.79	5.29	1.79	2.29	14.99	0.66	0.81	0.71
LG [9]	109.99	10.79	4.79	1.59	1.99	13.19	0.64	0.79	0.69
RNN [15]	142.79	13.89	6.09	1.99	2.59	17.29	0.73	0.87	0.78
CAPSNets [14]	191.89	18.79	8.19	2.69	3.29	22.69	0.92	0.95	0.89
CNN [7]	94.69	9.29	4.09	1.39	1.79	11.79	0.60	0.77	0.66
LSTM [8]	114.29	11.19	4.99	1.69	2.19	14.19	0.65	0.80	0.70
GNN [11]	152.29	14.99	6.49	2.19	2.79	18.49	0.87	0.91	0.84
TDINet	185.69	18.29	7.89	2.49	3.09	21.39	0.91	0.94	0.87

Table 5 shows the average statistical analysis results for existing and recommended TDINet. Statistics are used to compare each model in the table. TDINet beats rival models in most tests, proving its statistical resilience. ANOVA (7.89), Chi-Squared (21.39), and Pearson's correlation (0.87) show that TDINet predicts and achieves outcomes with minimal performance variance. In Mann Whitney (191.89) and Chi-Squared (22.69), CAPSNets outperform GNN, indicating reliable categorization. CNN scored poorly in Kruskal (9.29) and ANOVA (4.09), suggesting it may struggle with variance and statistical significance. LSTM performed decently.

An ablation research in which we replaced DenseNet, InceptionNet, and Transformer layers with simpler ones allowed us to evaluate each TDINet component. Each ablated version's accuracy, AUC, ASF, TSI, and AIE are compared to the complete TDINet model in the table below.

The findings show that eliminating any component affects performance, proving that each design feature individually strengthens TDINet. DenseNet substituted with

Table 6: Ablation study results

Model Variant	Accuracy (%)	AUC	ASF	TSI	AIE
TDINet (Full Model)	97.0	0.98	0.95	0.99	0.90
TDINet without DenseNet (Replaced with Standard CNN)	92.8	0.93	0.85	0.94	0.78
TDINet without InceptionNet (Replaced with Standard Convolutions)	91.5	0.91	0.82	0.92	0.74
TDINet without Transformer Layers (Replaced with LSTM)	89.7	0.88	0.79	0.90	0.70

a regular CNN decreases accuracy from 97.0% to 92.8% and AIE from 0.90 to 0.78, emphasizing the relevance of feature reuse in cybersecurity risk prediction. InceptionNet removal lowers AUC from 0.98 to 0.91, illustrating the necessity of multi-scale feature extraction for hierarchical pattern recognition. Finally, LSTM replaces Transformer layers, lowering TSI from 0.99 to 0.90, proving that Transformers are necessary for long-term cybersecurity behavior modeling.

6 Discussion

TransDenseInceptionNet (TDINet) outperforms CNNs, LSTMs, and GNNs in multiple evaluation metrics, particularly AIE, TSI, and ASF. These unique metrics analyze cybersecurity awareness and threat detection more thoroughly, guaranteeing that the model works well in real-world cybersecurity monitoring situations and has high classification accuracy. When compared to CNN-based models, TDINet excels in multi-scale feature extraction, which is essential for analyzing complicated cybersecurity activities. The model can process various feature granularities concurrently with InceptionNet, improving AIE scores. CNN-based models struggle to capture feature variety, making them unable to grasp cyber threats' dynamic nature. LSTM models, although good at sequential data, suffer with global feature dependencies and have lower TSI scores because to their inability to anticipate across time. TDINet models long-range dependencies using Transformer-based self-attention methods, improving prediction stability and TSI performance.

GNNs are excellent for relational data modeling, while TDINet has higher ASF scores, suggesting a better capacity to identify anomalies and minimize false positives. Due to graph processing difficulty, GNNs are powerful in capturing hierarchical connections yet computationally intensive. In limited computational contexts, they are unsuitable for real-time cybersecurity monitoring. TDINet uses DenseNet's feature reuse method to decrease unnecessary calculations while keeping feature representation. This performance lets TDINet evaluate data quicker and classify cyber risk more accurately. Several architectural features make TDINet function better. InceptionNet's multi-scale feature extraction helps learn fine-grained and high-level cybersecurity patterns, improving AIE by improving interaction-based feature representations. Self-attention mechanisms in Transformer layers allow the model to retain global contextual awareness, decreasing cyber risk prediction fluctuations and improving TSI stability. TDINet's Cumulative Anomaly Weighting (CAW) approach reduces false positives, raising the ASF score and making the model more dependable for real-world cybersecurity applications. By efficiently reusing learned feature representations across layers, the DenseNet design reduces computational redundancy and improves generalization.

TDINet has drawbacks despite its benefits. Computational overhead is a problem. DenseNet improves efficiency, however InceptionNet and Transformer layers increase memory and processing, making the model computationally heavier than CNN or LSTM-based techniques. This may restrict its use in resource-constrained contexts like embedded IoT devices, where real-time threat detection is essential. The large increases in classification accuracy, anomaly detection, and long-term prediction stability justify this trade-off. For computational cost reduction without performance degradation, future research may use quantization or pruning to compress models. Challenges with data imbalance are another restriction. TDINet uses Dynamic Feature Imbalance Compensation (DFIC) to reduce class imbalance, but datasets with extreme cybersecurity threat distribution skewness may require progressive oversampling or adversarial data augmentation to improve generalization. DFIC dynamically modifies feature priority to avoid ignoring underrepresented cybersecurity risks, however it may need improvements to manage occasional cybersecurity attacks in real life.

For real-time cybersecurity monitoring in edge computing contexts, TDINet should be computationally optimized using pruning and quantization to improve its applicability. Federated learning might let TDINet function in privacy-preserving circumstances without centralized data storage. This is especially useful in cybersecurity applications where data privacy is crucial. Beyond adolescent cybersecurity awareness, TDINet's infrastructure may be used for business cybersecurity monitoring, financial fraud detection, and industrial IoT security. TDINet outperforms other deep learning models in cybersecurity awareness metrics including AIE, TSI, and ASF, giving it a more reliable and scalable cyber threat detection framework. Despite requiring more computing resources, the model's powerful feature extraction, anomaly detection, and feature reuse algorithms offer top cybersecurity monitoring performance.

Table 6 displays ablation study findings. prove TDINet's hybrid architecture's originality by showing that DenseNet, InceptionNet, and Transformer layers are needed for better classification. Removing DenseNet reduces feature interaction efficiency (AIE), indicating that cybersecurity awareness prediction requires feature reuse. Removing InceptionNet weakens anomalous sensitivity (ASF), empha-

sizing the necessity of multi-scale feature extraction for identifying immediate and long-term cyber hazards. Finally, replacing Transformer layers with LSTM dramatically lowers Temporal Stability Index (TSI), proving that long-term cybersecurity behavior modeling requires self-attention methods. These results show that TDINet is a unique structured deep learning framework for real-world cybersecurity threat identification, not merely a mix of models. TDINet classifies cybersecurity awareness better than traditional architectures by deliberately integrating feature reuse, multi-scale pattern recognition, and long-range dependency modeling.

7 Conclusion and future work with limitations

This work introduces TransDenseInceptionNet (TDINet), a new deep learning model that can predict and classify the cybersecurity behaviors, risks of malware exposure, and e-safety awareness of adolescents. TDINet identifies local patterns and long-range relationships by merging DenseNet, InceptionNet, and Transformer architectures, guaranteeing a hierarchical knowledge of cybersecurity concerns. The model's capacity to handle unbalanced and noisy datasets is improved by advanced preprocessing methods including DFIC, CAW, and APE. Additionally, Contextual Feature Synthesis (CFS) creates useful features to improve forecast accuracy. TDINet surpasses CNNs, LSTMs, and GNNs with 97% accuracy and better performance in unique assessment measures including AIE, TSI, and ASF. These findings demonstrate the model's capacity to use feature interactions, maintain prediction stability, and identify anomalies with few false positives. This study advances cybersecurity risk prediction, e-safety education, and adolescent cybersecurity behavior analysis, providing insights for other cybersecurity applications.

Despite its success, TDINet has certain drawbacks that need to be addressed. A significant drawback is its computational complexity. DenseNet's feature reuse, InceptionNet's multi-scale feature extraction, and Transformer-based self-attention methods improve prediction accuracy and anomaly detection but increase computational load. Its increased processing needs may restrict its use in resource-constrained IoT or edge computing applications. TDINet's excellent accuracy (97%) and AUC (0.99) explain its computational intensity, however pruning, quantization, and knowledge distillation may enhance efficiency. Due of its concentration on Texas and California teens, dataset specificity is another restriction. The dataset encompasses numerous online habits and cybersecurity issues, but the conclusions may not apply to other age groups, cultural situations, or geographical areas. Cybersecurity knowledge and online activity vary widely across groups, hence TDINet needs further validation to determine its applicability. To make the model more applicable to other user groups, further research should include cross-population studies, domain adaptation, and federated learning.

Many research paths might increase TDINet's scalability and real-world applicability. First, model compression methods including pruning, quantization, and lightweight transformer topologies may optimize computational efficiency and accuracy for edge computing devices. Second, using TDINet to business cybersecurity monitoring, financial fraud detection, and industrial IoT security would show its adaptability beyond adolescent cybersecurity awareness. Third, federated learning might let TDINet learn from decentralized datasets while protecting data and user privacy. TDINet sets a new benchmark in e-safety and cybersecurity, but overcoming these limits and future problems will improve its scalability, efficiency, and generalizability. The expanding cybersecurity analytics environment benefits from this research's comprehensive, scalable, and flexible deep learning architecture for cybersecurity threat prediction and e-safety awareness.

Acknowledgement

2024 Annual Project of Guangzhou Philosophy and social sciences planning – New applications of artificial Intelligence in the construction of ping'an Guangzhou: Early warning and prevention research on adolescent bullying behavior (2024GZGJ213).

References

[1] Alshammari, S. S., Soh, B., & Li, A. (2025). Understanding Social Engineering Victimisation on Social Networking Sites: A Comprehensive Review of Factors Influencing User Susceptibility to Cyber-Attacks. *Information*, 16(2), 153.

[2] F. Schmidt, F. Varese, S. Bucci (2023) Understanding the prolonged impact of online sexual abuse occurring in childhood, *Frontiers in Psychology*, vol. 14, pp. 1281996.

[3] R. Salama, F. Al-Turjman (2023) Cyber-security countermeasures and vulnerabilities to prevent social-engineering attacks, *Artificial Intelligence of Health-Enabled Spaces*, CRC Press, pp. 133–144.

[4] Ö. Aslan, S. S. Aktuğ, M. Ozkan-Okay, A. A. Yilmaz, E. Akin (2023) A comprehensive review of cyber security vulnerabilities, threats, attacks, and solutions, *Electronics*, vol. 12, no. 6, pp. 1333.

[5] A. Giovanelli, J. Rowe, M. Taylor, M. Berna, K. P. Tebb, C. Penilla, E. M. Ozer (2023) Supporting Adolescent Engagement with Artificial Intelligence–Driven Digital Health Behavior Change Interventions, *Journal of Medical Internet Research*, vol. 25, pp. e40306.

[6] N. N. A. Molok, N. A. H. A. H. Hakim, N. S. Jamaludin (2023) SmartParents: Empowering Parents

to Protect Children from Cyber Threats, *International Journal on Perceptive and Cognitive Computing*, vol. 9, no. 2, pp. 73–79.

[7] S. Baadel, F. Thabtah, J. Lu (2021) Cybersecurity awareness: A critical analysis of education and law enforcement methods, *Informatica*, Vol. 45, No. 3.

[8] J. A. Bakar, N. Yusoff, N. H. Harun, M. M. Nadzir, S. M. Omar (2023) Text Simplification using Hybrid Semantic Compression and Support Vector Machine for Troll Threat Sentences, *International Journal of Advanced Computer Science and Applications*.

[9] M. B. Albayati, A. M. Altamimi (2019) An empirical study for detecting fake Facebook profiles using supervised mining techniques, *Informatica*, Vol. 43, No. 1.

[10] N. Owoh, J. Riley, M. Ashawa, S. Hosseinzadeh, A. Philip, J. Osamor (2024) An adaptive temporal convolutional network autoencoder for malicious data detection in mobile crowd sensing, *Sensors*, vol. 24, no. 7, pp. 2353.

[11] B. Yan, C. Yang, C. Shi, Y. Fang, Q. Li, Y. Ye, J. Du (2023) Graph mining for cybersecurity: A survey, *ACM Transactions on Knowledge Discovery from Data*, vol. 18, no. 2, pp. 1–52.

[12] X. Liu, J. Wang, X. Xiong, H. Sun (2024) Federated learning data protection scheme based on personalized differential privacy in psychological evaluation, *Neurocomputing*, Elsevier, pp. 128653.

[13] H. S. Alfurayj, S. L. Lutfi, R. Perumal (2024) A Chained Deep Learning Model for Fine-grained Cyberbullying Detection with Bystander Dynamics, *IEEE Access*, Publisher.

[14] A. S. Kumar, N. S. Kumar, R. K. Devi, M. Muthukannan (2024) Analysis of Deep Learning-Based Approaches for Spam Bots and Cyberbullying Detection in Online Social Networks, *AI-Centric Modeling and Analytics*, pp. 324–361.

[15] S. M. Fati, A. Muneer, A. Alwadain, A. O. Balogun (2023) Cyberbullying detection on twitter using deep learning-based attention mechanisms and continuous Bag of words feature extraction, *Mathematics*, vol. 11, no. 16, pp. 3567.

[16] L. Sik (2024) Teenage Online Behavior and Cybersecurity Risks, *Kaggle Data Set*, https://doi.org/10.34740/KAGGLE/DSV/9587284.

[17] V. Werner de Vargas, J. A. Schneider Aranda, R. dos Santos Costa, P. R. da Silva Pereira, J. L. Victória Barbosa (2023) Imbalanced data preprocessing techniques for machine learning: a systematic mapping study, *Knowledge and Information Systems*, vol. 65, no. 1, pp. 31–57.

[18] K. Patel (2024) Ethical reflections on data-centric AI: balancing benefits and risks, *International Journal of Artificial Intelligence Research and Development*, vol. 2, no. 1, pp. 1–17.

[19] N. Karlupia, P. Abrol (2023) Wrapper-based optimized feature selection using nature-inspired algorithms, *Neural Computing and Applications*, vol. 35, no. 17, pp. 12675–12689.

[20] F. S. Alsubaei, A. A. Almazroi, N. Ayub (2024) Enhancing phishing detection: A novel hybrid deep learning framework for cybercrime forensics, *IEEE Access*, Publisher.

[21] X. Chen, M. Ding, X. Wang, Y. Xin, S. Mo, Y. Wang, J. Wang (2024) Context autoencoder for self-supervised representation learning, *International Journal of Computer Vision*, vol. 132, no. 1, pp. 208–223.

[22] F. Salim, F. Saeed, S. Basurra, S. N. Qasem, T. Al-Hadhrami (2023) DenseNet-201 and Xception pre-trained deep learning models for fruit recognition, *Electronics*, vol. 12, no. 14, pp. 3132.

[23] Q. Zhang, X. Wang, M. Zhang, L. Lu, P. Lv (2024) Face-Inception-Net for Recognition, *Electronics*, vol. 13, no. 5, pp. 958.

[24] K. T. Chitty-Venkata, S. Mittal, M. Emani, V. Vishwanath, A. K. Somani (2023) A survey of techniques for optimizing transformer inference, *Journal of Systems Architecture*, pp. 102990.