IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 October 2024, accepted 26 November 2024, date of publication 29 November 2024,
date of current version 10 December 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3509372

SURVEY

Transformers: A Security Perspective

BANAFSHEH SABER LATIBARI[1], NAJMEH NAZARI[1], MUHTASIM ALAM CHOWDHURY[2], KEVIN IMMANUEL GUBBI[1], (Student Member, IEEE), CHONGZHOU FANG[1], SUJAN GHIMIRE[3], ELAHE HOSSEINI[1], HOSSEIN SAYADI[4], (Member, IEEE), HOUMAN HOMAYOUN[1], SOHEIL SALEHI[2], (Member, IEEE), AND AVESTA SASAN[1], (Senior Member, IEEE)

[1]Department of Electrical and Computer Engineering, University of California at Davis, Davis, CA 95616, USA
[2]Department of Electrical and Computer Engineering, The University of Arizona, Tucson, AZ 85721, USA
[3]Department of System and Industrial Engineering, The University of Arizona, Tucson, AZ 85721, USA
[4]California State University at Long Beach, Long Beach, CA 90840, USA

Corresponding author: Banafsheh Saber Latibari (bsaberlatibari@ucdavis.edu)

ABSTRACT The Transformers architecture has recently emerged as a revolutionary paradigm in the field of deep learning, particularly excelling in Natural Language Processing (NLP) and Computer Vision (CV) applications. Despite its success, the security implications of Transformers have not been comprehensively explored, encompassing a broad spectrum of both hardware and software vulnerabilities. This paper aims to address this critical gap by conducting an extensive exploration of security challenges confronting Transformers from both software and hardware perspectives. While software-related concerns like adversarial attacks, private inference, and watermarking have been studied, the paper sheds light on previously underexplored hardware vulnerabilities such as trojans and side-channel attacks. By unraveling the intricacies of these hardware threats, the study aims to contribute to a comprehensive understanding of Transformer security. It presents an in-depth analysis of recent advancements in the security of Transformers. Additionally, it outlines existing challenges and forecasts future research trends, offering insights for researchers and practitioners aiming for the secure and resilient design and deployment of Transformers. The survey categorizes different attacks and defenses related to Transformers, helping researchers identify gaps and opportunities in this area. Furthermore, it defines a roadmap for a unified security framework, serving as a foundational starting point for developers seeking to implement robust security measures.

INDEX TERMS Deep learning security, hardware security, NLP, transformers, vision.

I. INTRODUCTION

The Transformers have revolutionized various fields of artificial intelligence, offering powerful solutions in Natural Language Processing (NLP), computer vision, audio and signal processing, and multi-modal tasks. In NLP, Transformers have become essential for state-of-the-art models, enabling tasks like text classification, question answering, language modeling, understanding, and generation. Their attention mechanisms capture long-range dependencies effectively [28], [170]. In computer vision, Transformers excel in tasks such as image classification, object detection, and image captioning. Treating images as sequences of tokens, Transformers efficiently process visual information while preserving spatial relationships [32], [144], [167].

In audio and signal processing, Transformers have been used for tasks like speech recognition, music generation, and sound classification. Transformer-based models excel in deciphering timing relationships within audio data, adeptly capturing complex patterns and structures [11], [172]. Additionally, Transformers are useful in multi-modal learning, where they can handle different types of data such as text, images, and audio all at once [185], [213]. Figure 1 illustrates the demographic distribution of Transformer applications across various fields.

Nevertheless, similar to their predecessors such as Convolutional Neural Networks (CNNs), Transformers are not immune to vulnerabilities and potential security threats.

The associate editor coordinating the review of this manuscript and approving it for publication was Peter Langendoerfer.

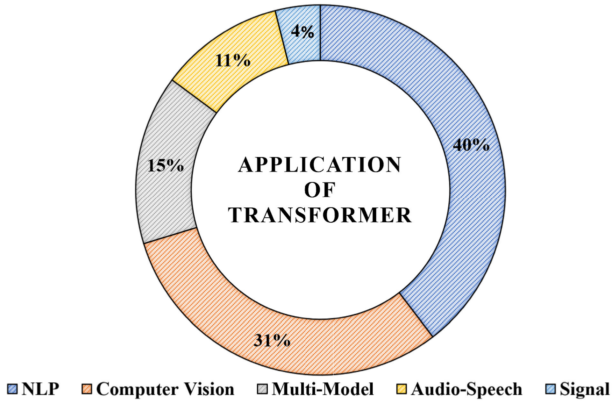


FIGURE 1. Transformer applications across key domains, including NLP, Computer Vision, Multi-Modal tasks, Audio and Speech, and Signal Processing. This visualization provides an overview of the versatile utilization of Transformers in various fields, emphasizing their impact on diverse domains.

Previous research has primarily focused on analyzing Transformers from a software security standpoint, often overlooking a comprehensive exploration of hardware security aspects. Software-level security analysis studies have focused on various topics, including adversarial attacks [90], [107], [126], federated learning and private inference [136], intellectual property concerns and watermarking [61], [127], as well as differential privacy [205].

In contrast to the extensive examination of software-level vulnerabilities in recent years, the susceptibility of Transformers to hardware security threats [128], [129] has received limited attention [4], [125]. This study represents a pioneering effort to provide a comprehensive analysis of Transformers from both software and hardware security perspectives, aiming to shed light on previously unattended areas and bridge the gap between software and hardware security considerations in Transformer models. This paper seeks to address this gap by offering a detailed exploration of the security challenges confronting Transformers. Through a comprehensive review of existing research in the field, it examines both hardware and software perspectives to provide a comprehensive understanding of the security landscape of Transformer-based models. By unraveling the intricacies of potential threats at both intersection of hardware and software, this study aims to contribute to a holistic understanding of Transformer security, paving the way for robust defenses in the face of evolving threats.

A. COMPREHENSIVE ANALYSIS OF SOFTWARE VULNERABILITIES

Our work contributes to the field by providing a comprehensive analysis of various software-level security threats affecting Transformer-based models. While previous research predominantly focused on adversarial attacks, our study delves into a broader spectrum of security concerns. By examining various aspects such as federated learning, private inference, intellectual property concerns, watermarking, and differential privacy, we offer a thorough understanding of the security landscape surrounding the Transformer-based models. Through this comprehensive analysis, we aim to provide valuable insights that facilitate the development of more secure and resilient AI-based systems for real-world applications.

B. UNRAVELING HARDWARE VULNERABILITIES

Hardware vulnerabilities [52], [93] present unique challenges, ranging from trojans injected [50], [168], [169] during manufacturing to side-channel attacks exploiting subtle information leaks. Understanding and investigating these hardware intricacies is crucial not only for identifying vulnerabilities but also for designing effective countermeasures that fortify Transformers against malicious exploits. Within this survey paper, we delve into various emerging hardware vulnerabilities in Transformers which enhances the research community's capacity to fortify Transformers against potential malicious exploits, thereby ensuring their resilience and security across emerging applications.

C. TOWARDS A UNIFIED SECURITY FRAMEWORK

Efforts to secure Transformers must transcend the conventional silos of hardware and software security. A unified framework that integrates insights from both realms is imperative. This paper advocates for a holistic approach that acknowledges the symbiotic relationship between hardware and software vulnerabilities. By fostering a comprehensive understanding of security challenges and solutions, we aim to propel the development of robust and adaptive defense mechanisms for Transformers, ensuring their continued success in an increasingly complex threat landscape. Additionally, this survey outlines existing challenges and forecasts future research trends, providing valuable insights for researchers and practitioners dedicated to the secure and resilient design and deployment of Transformers.

Figure 2 shows the overview of our work. The organization of this paper is as follows: Section II describes the application of Transformers and the architecture of the models. Section III covers vulnerability points in the architecture of Transformers. Section IV presents the detailed review of software-level security aspects of Transformers including adversarial attack and defense, trojan insertion, federated learning, input encryption, watermarking, and differential privacy. Section V provides a comprehensive review of the hardware security aspects of Transformers in both ASIC and FPGA accelerators. In section VI, potential research challenges and future opportunities of this emerging field of study are discussed. Finally, section VII concludes the survey.

II. BACKGROUND AND APPLICATION DOMAINS

This section provides an overview of the essential background knowledge and architecture of the Transformers published extensively from 2017, as shown in Figure 3. It further explores diverse application domains where the Transformers find practical utility.

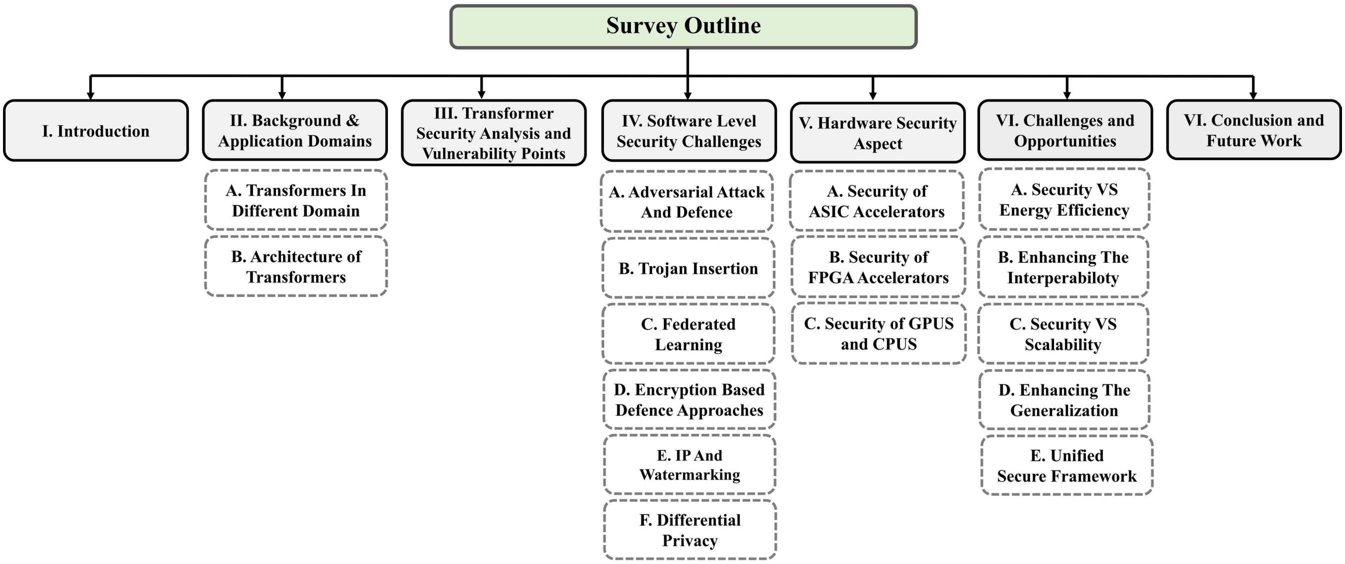

Survey Outline
I. Introduction
II. Background & Application Domains
A. Transformers In Different Domain
B. Architecture of Transformers
III. Transformer Security Analysis and Vulnerability Points
IV. Software Level Security Challenges
A. Adversarial Attack And Defence
B. Trojan Insertion
C. Federated Learning
D. Encryption Based Defence Approaches
E. IP And Watermarking
F. Differential Privacy
V. Hardware Security Aspect
A. Security of ASIC Accelerators
B. Security of FPGA Accelerators
C. Security of GPUS and CPUS
VI. Challenges and Opportunities
A. Security VS Energy Efficiency
B. Enhancing The Interpabiloty
C. Security VS Scalability
D. Enhancing The Generalization
E. Unified Secure Framework
VI. Conclusion and Future Work

FIGURE 2. Overview of the "Transformers: A Security Perspective" paper.

A. TRANSFORMERS IN DIFFERENT DOMAINS

1) TRANSFORMERS FOR NLP

In the era preceding Transformers, Deep Learning heavily relied on Recurrent Neural Networks (RNNs) for text comprehension. The challenge with RNNs lies in their difficulty to train and their inability to be parallelized, given their sequential word-processing nature. This is where Transformers emerged as a solution. Transformers are applied to a diverse range of NLP tasks, including but not limited to text classification, question answering, language modeling and understanding, text generation, information retrieval, text summarization, and translation [15], [22], [28], [38], [47], [85], [87], [97], [130], [137], [138], [149], [170]. The Generative Pre-trained Transformer (GPT) series from OpenAI, along with Google's models like Palm and Bard, are recognized and widely acknowledged within this group. GPT models, like GPT-4 and GPT-3.5, are advanced text generation models by OpenAI. They understand both natural and formal language, generating text in response to provided inputs called "prompts." These models are versatile, used for tasks such as content generation, code writing, summarization, conversation, and creative writing. Explore our guides for more details.

In the realm of NLP research, three developing security concerns include adversarial and backdoor attacks [184], the potential compromise of private data [113], and the vulnerabilities associated with imitation attacks [69]. These security challenges may result in problems such as unauthorized data access, financial setbacks, reputation damage and other adverse consequences [186]. GPT models, with their remarkable progress, have garnered attention for applications in healthcare and finance. However, caution is needed as they can generate biased outputs, risking the disclosure of private information. While GPT-4 outperforms GPT-3.5 on standard benchmarks, it is more susceptible to manipulation through unauthorized system access or manipulated user prompts [173].

2) TRANSFORMERS FOR VISION

Due to the remarkable capabilities of Transformers in NLP, researchers have explored their application in computer vision. Traditionally, CNNs held a central role in vision-related tasks; however, Transformers have recently garnered attention and prominence in the realm of computer vision. Computer vision Transformers are employed for a myriad of tasks, encompassing supervised and self-supervised learning, as well as tasks such as object detection, segmentation, generation, video captioning/summarization, action recognition, and sign language recognition [16], [32], [55], [99], [121], [167], [191], [195]. Beyond these applications, fields like medical imaging, autonomous driving, and agriculture have also experienced benefits from the utilization of Transformers for image and video processing [151], [158], [164]. Transformer architectures leverage a self-attention mechanism, treating images as sequences of patches [32].

Given their distinctive design in contrast to CNNs, it is imperative to investigate Transformers' susceptibility to a range of attacks, including backdoor attacks and other security threats. Additionally, it is crucial to delve into how different Transformer architectures impact overall robustness of the system. The risks linked to these vulnerabilities become more pronounced when implementing deep learning models in safety-sensitive fields such as autonomous vehicles. In these contexts, a security breach could lead to a dangerous scenario where the vehicle fails to recognize a pedestrian due to a specific adversarial stimulus captured by the camera, posing a direct threat to the safety and reliability of the autonomous system. [160].

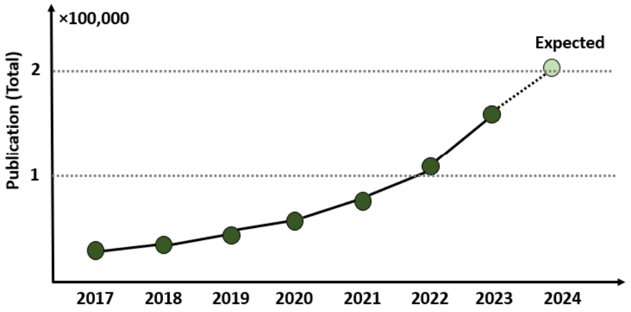


FIGURE 3. Analyzing the proliferation of Transformer model publications across various domains from 2017, sourced from app.dimensions.ai, with "Transformer model" as the primary keyword. Additionally, the figure projects the anticipated number of publications in 2024.

3) TRANSFORMERS FOR AUDIO PROCESSING

The Transformer architecture extends its applications to Audio Processing (AP) as well. Notable examples include Audio Transformers [172], Septr [142], and SpecTNT [101], showcasing successful implementations of the Transformer architecture in audio processing. The ability of Transformers to effectively capture long-range dependencies and interactions makes them especially attractive for time series modeling. This has led to significant advancements in various applications within the realm of time series analysis, [182]. Voice assistant applications like Google Home, Amazon Alexa, and Siri rely on key models—Automatic Speech Recognition (ASR) and Speaker Identification (SI). However, with the rise of IoT, security threats targeting machine learning models and hardware components have surfaced, raising concerns about information theft and privacy breaches. Transformers, a popular machine learning model, are increasingly employed in audio processing within this domain [89].

4) TRANSFORMERS FOR SIGNAL PROCESSING

Researchers applied Transformers for signal processing. In [11], an automated seizure prediction framework integrated Fourier transform and a Transformer model, blending signal processing and deep learning for effective epilepsy identification. Utilizing face videos for heart rate estimation, researchers encountered challenges with time-varying ambient lighting. Neglecting optical modeling, they found poor performance in existing methods. To address this, a demodulation-based Transformer was designed for efficient rPPG signal purification [211]. In Music Source Separation (MSS), researchers examined the relevance of long-range contextual information and introduced Hybrid Transformer Demucs (HT Demucs), a model outperforming Hybrid Demucs by 0.45 dB in Signal-to-Distortion Ratio (SDR) with an extra 800 training songs, employing a cross-domain Transformer Encoder [143].

5) MULTI-MODELS BASED ON TRANSFORMER

Multimodal learning [185], [213] entails creating models with the ability to analyze and connect information from various sources. Overcoming the obstacle of integration, the task involves crafting a unified network capable of addressing distinct modalities, including natural language, 2D images, 3D point clouds, audio, video, time series, and tabular data, despite their inherent disparities. Meta-Transformer [213], employing a frozen encoder, achieved multimodal perception without paired training data by mapping diverse inputs into a shared token space. The framework excelled across 12 modalities, demonstrating effectiveness in fundamental perception, practical applications, and data mining tasks according to benchmark experiments. Google recently unveiled Gemini [48], its most extensive and advanced AI model, tailored for multimodal comprehension and available in different sizes—Ultra, Pro, and Nano. With cutting-edge performance, intricate reasoning abilities, and proficiency in tasks like coding, Gemini marks a noteworthy advancement in AI, promising widespread availability and ongoing innovation.

B. ARCHITECTURE OF TRANSFORMERS

In this section, we embark on a detailed exploration of the foundational elements that make up the Transformer architecture. The Transformer architecture, as detailed in [170], typically consists of multiple Transformer blocks. Each of these blocks includes a multi-head attention (MHA) module alongside a feed-forward (FFN) module. Importantly, every block is sequentially accompanied by a Layer Normalization (LayerNorm) operation and a residual connection.

The attention mechanism enhances model accuracy by prioritizing relevant data and disregarding less important information, assigning weights to input attributes based on their significance to the output. It's particularly advantageous for tasks in NLP, computer vision, and speech recognition. Unlike CNNs which typically focus on local information, the attention mechanism can gather information globally from the entire input sequence. The MHA module in the Transformer architecture involves projection layers, matrix multiplications, and Softmax operations, as outlined in [170]. On the other hand, the FFN module comprises two projection layers separated by a nonlinear function. The specific computations for both the MHA and FFN are illustrated in Figure 4.

The MHA module begins by processing a sequence consisting of n tokens through a projection step. This step involves multiplying the sequence with three distinct weight matrices: W_Q, W_K, and W_V, resulting in query, key, and value representations. These representations are then divided into h segments, each possessing a hidden dimension of d/h. Within each head, the query and key undergo multiplication along the hidden dimension, producing a matrix of representations, as articulated in Equation 1.

$$\begin{aligned} Q^i &= QW_Q^i \\ K^i &= KW_K^i; \qquad i \in heads \\ V^i &= VW_V^i \end{aligned} \tag{1}$$

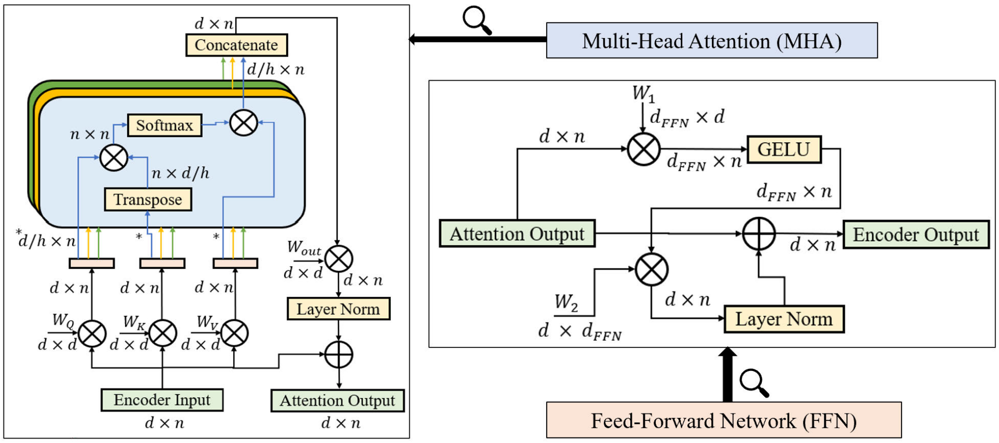
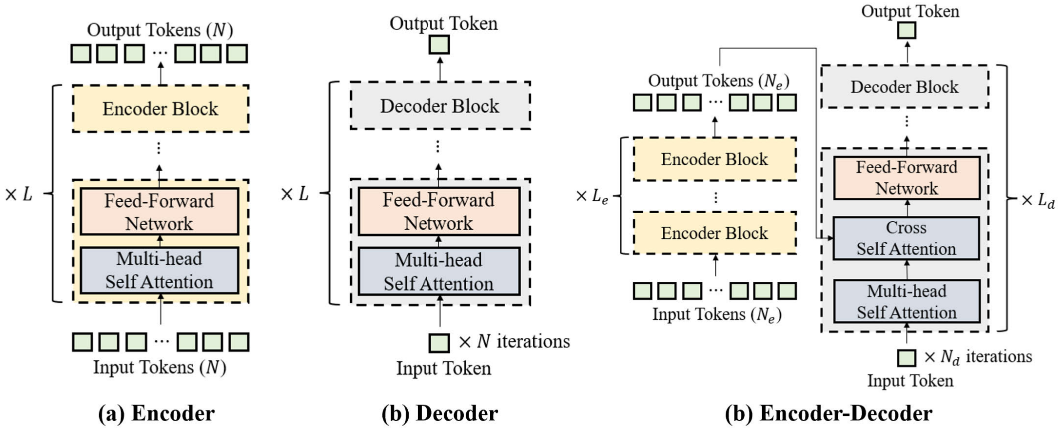


FIGURE 4. Illustration detailing the computational processes within the Transformer encoder block, showcasing the multi-head attention (MHA) module and the feed-forward network (FFN) module.



(a) Encoder

(b) Decoder

(b) Encoder-Decoder

FIGURE 5. Variants of Transformer networks depicted: (a) Encoder-only model conducts parallel inference for all tokens. (b) Decoder-only model follows an auto-regressive inference approach. (c) Encoder-decoder model utilizes the encoded sequence output as input for a cross-attention module.

The matrices from each head are subject to processing through the "scaled dot-product attention" function, as detailed in Equation 2, to calculate attention scores and generate output matrices. Following a Softmax operation, the resulting attention scores are then multiplied by the value segment, resulting in an activation with a hidden dimension of d/h.

$$O = SoftMax(\frac{QK^T}{\sqrt{d_k}})V = SV \tag{2}$$

The outputs from all attention heads are consolidated along the hidden dimension, resulting in a unified activation with a hidden dimension of d. This aggregated result undergoes projection using the weight matrix W_{out} within a concluding linear layer. Following this, the output undergoes normalization via LayerNorm and is combined with a residual connection, ultimately producing the final output of the MHA module.

Regarding the FFN module, it is a straightforward block comprising two linear layers. Initially, the input sequence is projected from the hidden dimension d to a higher FFN dimension through the first linear layer. Subsequently, the projected sequence is transformed back to the original dimension d using the second linear layer.

1) NLP TRANSFORMER ARCHITECTURE

Figure 5 depicts various types of NLP Transformer architectures: Encoder, Decoder, and Encoder-Decoder. The Transformer architecture was originally devised as an Enncoder-Decoder model [170]. In this configuration, the encoder receives the entire source language sentence and processes it through multiple Transformer encoder blocks to distill high-level features. These features are then transmitted to the decoder, which, in turn, is responsible for the stepwise generation of tokens in the target language. In encoder-only Transformer models, the input sequence is processed directly by multiple encoder blocks as a continuous flow. This design,

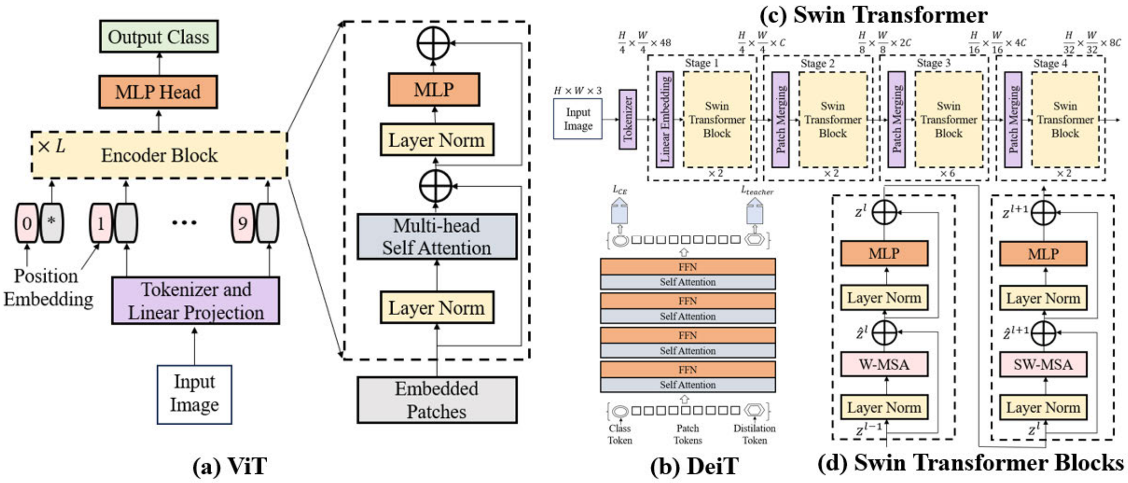


FIGURE 6. Illustration outlining the fundamental structure of the (a) original Visual Transformer (ViT), highlighting the key architectural elements, (b) DeiT, (c) Swin Transformer, and (d) Two consecutive blocks of Swin Transformer.

exemplified by models like BERT, is particularly effective for tasks centered around natural language understanding. Decoder-only models, composed of recurring decoder blocks, exhibit an auto-regressive nature. This implies that the output at a specific time step relies on the outputs from previous time steps. Essentially, the model predicts a token in a sentence by considering the tokens it has generated earlier.

As a consequence, inference in decoder-only models must be executed sequentially and iteratively for each output token. An example of a decoder-only architecture is GPT-based models.

2) VISION TRANSFORMER ARCHITECTURE

Figure 6-(a) illustrates the Vision Transformer (ViT) [32], designed with inspiration from the architecture of encoder-only Transformers frequently employed in NLP. ViT architecture utilizes self-attention mechanisms to process images, employing a series of Transformer blocks. Each block comprises two sub-layers: a multi-head self-attention layer and a feed-forward layer. The self-attention layer computes attention weights for each image pixel based on its relationships with others, while the feed-forward layer applies a non-linear transformation to the self-attention layer output. The multi-head attention enables simultaneous attention to different parts of the input sequence. In the design of the ViT model, a patch embedding layer is integrated alongside Transformer blocks. This involves breaking down the image into fixed-size patches and assigning each patch to high-dimensional vectors. The model's final class prediction is generated by passing the last output of the Transformer block through a classification head, typically composed of a single fully connected layer. This architectural approach efficiently processes images by combining self-attention and patch-based representations. A crucial feature introduced in ViT, similar to BERT, is the incorporation of a "classification token." This token serves as a comprehensive summary of the entire input image and is appended to the sequence of patch embeddings before entering the Transformer blocks. Acting as a global context aggregator, the classification token empowers the model to consider the overall content of the image during the classification process. This integration enhances ViT's capacity to capture holistic information, resulting in improved performance across various image classification tasks. Other popular variants of ViT, such as DeiT [167] and Swin Transformer [99], are shown in parts (b) and (c) of Figure 5, respectively. DeiT employs a teacher-student strategy specifically designed for Transformers, using a distillation token to enable the student model to learn from the teacher model through attention. This approach eliminates the need for training on a huge dataset, producing a competitive convolution-free Transformer by training on ImageNet only. Swin Transformer constructs hierarchical feature maps by progressively merging image patches in deeper layers and computing self-attention within local windows, leading to linear complexity relative to the input image size. Figure 6-(d) shows Swin Transformer blocks replace the standard MSA module with a shifted window-based MSA module while keeping other layers unchanged. Each Swin Transformer block consists of a shifted window-based MSA module, followed by a 2-layer MLP with GELU nonlinearity. LayerNorm (LN) is applied before each MSA module and each MLP, with a residual connection following each module.

III. SECURITY ANALYSIS AND VULNERABILITY POINTS IN TRANSFORMER ARCHITECTURE

In this section, we delve into the security aspects inherent within Transformers, highlighting potential vulnerability points in which the attackers can leverage. The goal is to pinpoint areas prone to exploitation, offering insights crucial for enhancing the security of Transformer-based models.

Figure 7 presents a comprehensive overview of vulnerability points in Transformer-based models, including input, output, and hardware vulnerabilities, which will be discussed in detail below.

A. INPUT DATA

In Transformer-based models, the input data itself emerges as a critical attack point, susceptible to manipulation at various stages of the model's lifecycle. Two primary attack vectors on the input data are *data poisoning attacks* [76], [192] and *evasion attacks* [72], [217].

In data poisoning attacks, attackers strategically introduce malicious samples during the model's training phase to manipulate the learning process. By injecting deceptive data points into the training set, adversaries aim to induce the model to generate inaccurate predictions. This deliberate poisoning can have a lasting impact on the model's behavior, leading to unexpected and potentially harmful outputs during real-world usage [131].

In contrast, evasion attacks are devised after the Transformer model has been deployed in real-world scenarios. Attackers, in this case, modify specific data samples, often referred to as adversarial examples, to deceive the model into misclassifying them according to a predefined output label. This manipulation occurs at the inference stage, where the model encounters real-world data, and the attacker seeks to exploit vulnerabilities to force misclassification. Such attacks can manifest as adversarial patches and tokens. Vision Transformers may be vulnerable to adversarial patches, where manipulated regions in the input images deceive the model [49], [78], [159]. Meanwhile, in NLP tasks, attacks may occur at the token level, where adversaries manipulate specific words or segments to influence the model's output. These vulnerabilities underscore the importance of robust defense mechanisms to safeguard Transformer models against adversarial manipulations at the input data level [119], [198].

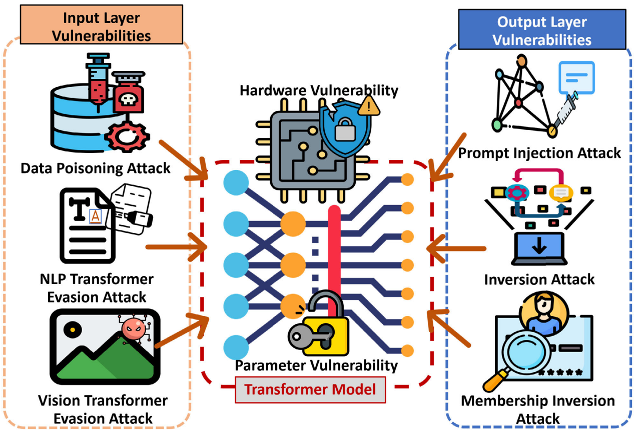

Input Layer Vulnerabilities
Data Poisoning Attack
NLP Transformer Evasion Attack
Vision Transformer Evasion Attack
Hardware Vulnerability
Parameter Vulnerability
Transformer Model
Output Layer Vulnerabilities
Prompt Injection Attack
Inversion Attack
Membership Inversion Attack

FIGURE 7. Overview of vulnerability points in Transformer models, encompassing input, output, and hardware vulnerabilities.

1) MODEL OUTPUT

In addition to the input data, the model output represents a critical attack point in Transformer-based systems, with *model inversion* being a specific security threat that exploits this vulnerability. In model inversion attacks, adversaries leverage the output of a machine learning model, such as Transformers, to gain insights into some of its parameters or architectural details. This is achieved through a process of querying the model and utilizing the obtained output to infer specific characteristics of the model or input data. In the field of NLP, by meticulously analyzing the model's predictions and utilizing the output, adversaries attempt to reverse-engineer sensitive information, potentially exposing details about the training dataset [209].

In a *membership inference attack*, the adversary seeks to ascertain whether specific personal information was part of the training dataset used for a target machine learning model. This attack hinges on training a separate model, known as a membership inference model, using the output or predictions generated by the target model. The attacker's model is trained to predict whether a given data point (e.g., personal information) was included in the target model's training dataset. The successful prediction by the attacker's model implies that the target model has potentially memorized or learned details about the input data, posing privacy concerns. This method underscores the need for privacy-preserving measures, such as differential privacy or data anonymization, to mitigate the risks associated with divulging sensitive information during the training process of machine learning models [68], [153].

B. MODEL PARAMETERS

In these attacks, adversaries leverage the model weights and corresponding gradients to reconstruct the original data batch. Essentially, they exploit the relationship between the model's parameters and its output to gain insights into the internal workings of the model [57]. Recent research indicates that despite the growing interest in distributed learning for enhancing data privacy on local devices, there's a concerning revelation. Publicly shared gradients during the training process have the potential to expose private training data, a phenomenon known as gradient leakage, to unauthorized third parties. TAG suggests a method to retrieve private training data of Transformer-based language models from the shared gradients [27]. Utilizing the Integrated Gradients (IG) method, saliency scores for model predictions were computed, guiding perturbation updates based on gradients' signs. IG indicated input sensitivity, with more pronounced perturbations expected at influential locations. Improved attack performance was achieved with a momentum-based iterative strategy akin to gradient descent, facilitating faster perturbation updates and aiding in escaping poor local optima. These strategies enhanced perturbation transferability across different models including ViT [106].

C. PROMPT

Prompt injection attacks within large language models (LLMs) involve exploiting vulnerabilities by inserting prompts into data expected to be processed during model inference. This type of attack, known as ''Indirect Prompt Injection'' allows adversaries to remotely exploit LLM-integrated applications, even without direct user interaction. The primary goal is to manipulate these applications, causing unintended actions or generating content that aligns with the attacker's goals. By skillfully injecting specific prompts into data streams encountered by LLMs during inference, adversaries can influence the model's behavior, highlighting the vulnerability of LLM-Integrated Applications to strategic manipulations [2], [98], [196].

D. UNDERLYING HARDWARE

Underlying hardware hosting calculation processes can pose vulnerabilities, particularly to hardware-oriented attacks like side-channel attacks [34], [35], [36]. These attacks, as demonstrated by studies such as [165], [166], and [190], have been proposed to compromise machine learning systems' security and privacy. These attacks could potentially threaten Transformer model hardware accelerators, which are crucial components of many AI systems. Additionally, with the global integrated circuit (IC) design flow, there is a possibility of maliciously inserting Trojans into the design, further amplifying security risks. Such Trojans could compromise the integrity and functionality of the hardware, potentially leading to severe security breaches. Moreover, there is the possibility of extracting sensitive information from memory units, adding another layer of vulnerability to the system. This underscores the importance of robust security measures and thorough validation processes in the design and deployment of Transformer-based systems, particularly in hardware-accelerated environments.

IV. SOFTWARE LEVEL SECURITY CHALLENGES

This section addresses the security concerns associated with Transformer models at the software level, encompassing adversarial attacks, Trojan insertion, federated learning, private inference, encryption-based approaches, intellectual property (IP), watermarking, and differential privacy. We conduct a comprehensive review of prior research in these domains.

A. ADVERSARIAL ATTACK AND DEFENCE

Adversarial examples [9], [79], [111], [112], [124], [193], [194] refer to manipulated input data that can deceive machine learning models while remaining imperceptible to humans. A white-box attack involves an adversarial attack where the attacker has full access to the details of the targeted model, including its architecture, parameters, and training data. With this knowledge, the attacker can develop a highly effective attack strategy. On the other hand, a black-box attack is an adversarial attack where the attacker lacks access to the details of the targeted model and can only interact with it through input-output channels. In this scenario, the attacker must rely on limited information obtained by querying the model to develop their attack strategy. Numerous studies have demonstrated the susceptibility of Transformer-based models across various applications to adversarial attacks [3], [210].

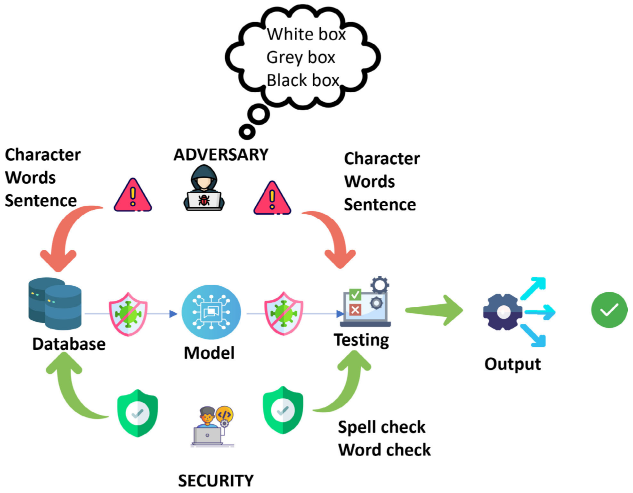


FIGURE 8. Adversarial Attacks on NLP Transformers. Adversarial attacks are categorized into white, gray, and black box attacks, contingent on the attacker's access level. The attacks exhibit granularity at the character, word, or sentence levels. Defense strategies at different levels are proposed to prevent the attacks like checking misspellings and checking words.

1) ATTACK ON NLP TRANSFORMER MODELS

Figure 8 demonstrates an overview of adversarial attacks on NLP-based Transformers. Such attacks are classified into white, gray, and black box attacks, contingent on the attacker's access level, while showing granularity at the character, word, or sentence levels. Numerous methods have been developed in prior research to generate adversarial examples in image data, causing systems to fail [31], [90], [117]. However, these techniques cannot be readily extended to NLP models due to the distinct nature of data representation and the challenges associated with characterizing subtle alterations in text [146]. Optimization and gradient descent algorithms are used to create adversarial examples for visual applications. Due to the specific nature of textual data, there are only a limited number of white-box attacks available for NLP models. As it is challenging to compute gradients in the discrete space of textual data, one proposed approach is to determine the gradients in the continuous embedding space instead. Cheng et al. [21], proposed a white-box gradient-based method called AdvGen to generate adversarial examples targeting Neural machine translation (NMT). AdvGen generates adversarial examples using the final translation loss as a guide based on clean input data. It is applied to both the encoding and decoding stages. They showed that AdvGen improved BLEU scores by 2.8 and 1.6 points over state-of-the-art models, including Transformer, on standard Chinese-English and English-German translation benchmarks.

This research [114] delves into the susceptibility of Transformer-based language models to adversarial attacks, particularly those categorized as "black box" attacks. Membership inference, in this context, pertains to the process of ascertaining whether a given data record was part of the training data used for the model's training. An attack involving membership inference considers the scenario where a malicious user of a black box prediction service could provide input messages resembling those of a competitor and, through the model's output, gather information about the user's data inclusion in the model's training set. This study specifically focuses on determining whether sample customer message data was incorporated into the training data when constructing a language model.

Seq2Sick [20] investigates the vulnerability of seq2seq models to adversarial attacks, comparing their robustness with CNN-based classifiers. The framework optimizes input sequences to produce desired outputs while preserving sentiment in word embeddings. It tackles discrete inputs using projected gradient descent, group lasso, and gradient regularization and addresses infinite output sequences with novel loss functions for non-overlapping and targeted keyword attacks. Seq2Sick's results show that while seq2seq models are susceptible to attacks with a high success rate, they are more robust than CNN-based models.

TextAttack [119], is a Python framework that facilitates adversarial attacks, data augmentation, and adversarial training for NLP models. TextAttack utilizes four components to construct attacks: a goal function, a set of constraints, a transformation, and a search method. TextAttack supports a range of models and datasets, including BERT, and offers implementations of 16 adversarial attacks previously proposed in the literature. The work in [197] investigates black-box attacks in NLP and offers recommendations on the most effective approach. The study revealed that in terms of attack success rate, beam search, and particle swarm optimization were the most optimal algorithms. If there is a time limitation or the input text is lengthy, using the greedy algorithm with word importance ranking is recommended, as it provides adequate performance. Also, simple greedy methods are often more effective and faster than complex algorithms like PWWS and genetic algorithms in terms of both attack success rate and speed. They showed that, on average, 10.05% of words were perturbed when BERT was used as the base model with the MR dataset and WordNet transformation.

The lack of a standardized definition and evaluation system has hindered the effective use of adversarial examples to enhance and comprehend NLP models. Perturbations often fail to preserve semantics, and 38% introduce grammatical errors. In response, the authors [118] propose a unified definition for successful adversarial examples in natural language, emphasizing modifications that not only deceive the model but also conform to predetermined linguistic constraints. They introduce four categories of constraints—semantics, grammaticality, overlap, and non-suspicion to human readers—that NLP adversarial examples can align with depending on the context. This establishes a common vocabulary for discussing constraints on adversarial attacks, presenting distinct categories to which adversarial examples may adhere. The authors enhance the TEXTFOOLER algorithm with TFADJUSTED, incorporating a constraint enforcement mechanism to generate higher-quality NLP adversarial examples that better preserve semantics and grammaticality. Through human evaluation, the proposed algorithm produces perturbations that are less noticeable to humans, albeit with a lower attack success rate (70%) under stricter constraints. In adversarial training, TFADJUSTED's examples do not reduce model accuracy compared to TEXTFOOLER's examples.

In the work by Jin et al. [77], the authors introduced TEXTFOOLER as a method for generating adversarial text in a black-box setting. This approach comprises two primary steps: word importance ranking and a word Transformer, which serves as their method for replacing words in the text. They reduced the accuracy of almost all target models across all tasks to below 10% on the adversarial examples, with fewer than 20% of the original words perturbed In the study by Yuan et al. [203], a framework is introduced for generating adversarial samples in text data. The methodology includes the incorporation of continuously optimized perturbations into the embedding layer, subsequently amplifying them during forward propagation. The ultimate perturbed latent representations are decoded using a masked language model head to derive potential adversarial samples. The authors implement this framework by employing an attack algorithm known as Textual Projected Gradient Descent (T-PGD). The quality of the adversarial samples generated by T-PGD increases with text length. These adversarial samples achieved higher overall USE scores (similarity between original and adversarial samples) compared to baseline models, with a 97% attack success rate. This indicates that the proposed method can manipulate adversarial samples more precisely using explicit gradient information.

In the work by Liu et al. [94], the authors employed Attachable Subwords Substitution (ASS) and introduced the Character-level White-Box Attack (CWBA) method targeted at Transformer models. The proposed approach leverages Transformer models' practice of dividing words into subtokens, finding that substituting consecutive subtokens can be as impactful as modifying individual characters. To generate adversarial examples, they follow three steps: using a gradient-based technique to identify the most susceptible words, breaking these words into subtokens as replacements for the original tokens, and applying an adversarial loss to guide subtoken substitution. To ensure gradient propagation, they incorporate the Gumbel-softmax method. Their required query number was similar to the **GBDA** model and much lower than other black-box methods. Their **CWBA** outperformed DeepWordBug by 20.0 adversarial accuracy on average, showing the advantages of the white-box attack.

The study in [146] introduced a token-level gradient-based white-box adversarial attack method for Transformer-based text classifiers. This method ensures block-sparse adversarial perturbations, altering only a few words in the sentence. By selectively perturbing embedding vectors and optimizing the perturbation vector under a block-sparsity constraint, only a few tokens are modified. To preserve semantics, the modified embeddings are projected onto original token embeddings with the highest cosine similarity. Their experiments show that their attack maintains sentence semantics while reducing GPT-2's accuracy to below 5% on AG News, MNLI, and Yelp Reviews.

In [41], the proposed method, MANGO, uses gradients to navigate a continuous space of token probabilities to find adversarial examples. As an optimization-based white-box attack, MANGO reduces the disparity between adversarial loss in continuous and discrete text representations through a quantization-compensation loop. This involves iterative quantization of token representations and reoptimization to maintain the adversarial loss value.

GPTFUZZER [201], a black-box jailbreak fuzzing framework inspired by AFL, automated the creation of jailbreak templates for red-teaming LLMs, eliminating manual efforts. Employing seed selection, mutate operators, and a judgment model, it effectively tested against LLMs like ChatGPT, LLaMa-2, and Vicuna. Results showcased its ability to unveil vulnerabilities and assess LLM robustness in diverse attack scenarios.

2) DEFENSE FOR NLP TRANSFORMER MODELS

In response to attacks targeting NLP Transformer models, it is crucial to deploy robust defense strategies to counter potential vulnerabilities and thwart adversarial attacks. Defense strategies at different levels are proposed to prevent the attacks like checking misspellings and checking words as shown in figure 8. The authors in [21] defend NMT models by attempting to decrease the prediction errors for the corresponding adversarial source inputs. The proposed approach consists of two main components: first, subjecting the translation model to adversarial source examples as an attack; and second, reinforcing the model by introducing adversarial target inputs, thereby improving its resilience against adversarial source inputs.

The study [80] addressed orthographic attacks on the Zeroe benchmark, encompassing various cognitive model-based attacks. Unlike previous spelling correction modules, which were effective against simple typo attacks, they failed on the more diverse Zeroe benchmark. In response, the authors introduced a novel technique that leverages context-independent extensions of the Levenshtein distance, probability distributions from a dictionary, and BERT's masked language modeling objective. This iterative process refines word context, predicts clean words, and utilizes a source text-independent language model to generate fluent output, providing a more robust defense against diverse orthographic adversaries.

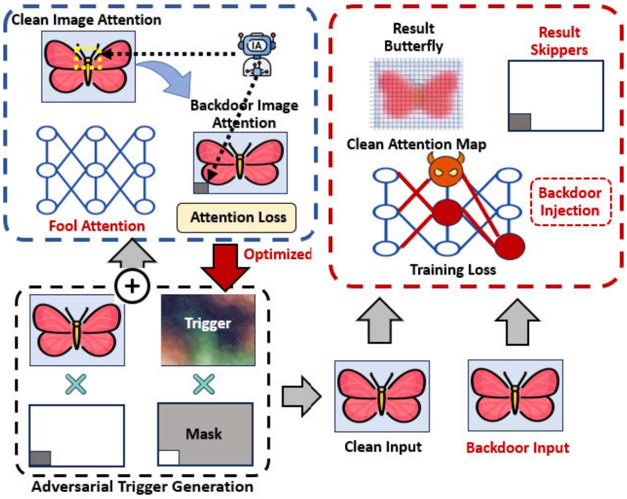


FIGURE 9. In the context of a backdoor attack, using a patch-wise trigger effectively disrupts the attention distribution in images. This strategy ensures that the ViTs' attention is primarily focused on the patch containing the trigger, thereby influencing the model's focus on the entire backdoor input [204].

In [62], it was observed that many existing adversarial attacks fail to maintain the original text's meaning, challenging their claimed semantic preservation. The root cause was identified in the text encoders used for assessing adversarial example similarity, specifically in their training methodologies. Encoders trained through unsupervised methods exhibited difficulty recognizing antonyms. To overcome this limitation, the authors introduced a fully supervised sentence embedding approach named Semantics-Preserving Encoder (SPE). The key idea behind SPE involves supervised training using labeled datasets to mitigate issues related to antonyms within the same context. This approach clusters words significant for a given label in the vector space. The incorporation of various classifiers trained on diverse annotated datasets further enriches their arsenal of sentence vectors.

3) ATTACK ON VISION TRANSFORMER MODELS AND ITS TRANSFERABILITY

Recent research has highlighted the superior resilience of ViTs compared to CNNs against different perturbations, including natural corruption and adversarial attacks. The reasons behind the enhanced robustness of ViTs are a subject of debate within the research community. Some argue that the segmentation of input images in ViTs plays a central role, while others attribute it to the utilization of MSA in ViTs. In this study [107], the authors assess the robustness of ViTs against two types of adversarial attacks: white-box and black-box attacks. The evaluation is carried out across various benchmark datasets, including CIFAR-10, CIFAR-100, and ImageNet, utilizing two ViT architectures, namely DeiT and ViT-B/16. Comparative analyses involve state-of-the-art models such as ResNet and EfficientNet under both types of attacks. The experimental outcomes reveal that ViTs generally exhibit greater robustness to adversarial examples compared to ResNet and EfficientNet. Notably, ViTs demonstrate heightened resilience to black-box attacks. Furthermore, the study observes a correlation between increased model size in ViTs and enhanced robustness. White-box attacks involve the use of the Fast Gradient Sign Method (FGSM) and the projected gradient descent (PGD) attack, while black-box attacks employ a substitute model strategy. The authors also conduct a sensitivity analysis, discovering that adjusting hyperparameters, such as increasing Transformer layers and reducing the learning rate, can positively impact the robustness of ViTs.

Joshi et al. [78], introduced a block sparse attack on deep neural networks, with a specific focus on a patch-based approach. The study compared the vulnerability of Transformer models to traditional CNNs regarding this attack. Interestingly, the results indicated that ViTs were more susceptible to patch attacks and mixed norm attacks (a variant of their patch-based block sparse attack) compared to CNN-based with ResNets outperforming Transformer models by up to approximately 30% in robust accuracy for single token attacks. Notably, ViTs exhibited increased robustness only for smaller patch sizes, as they could effectively compensate for perturbations in patch attacks that were smaller than the token size. The authors in [176] leverage the inherent architectural aspects of vision Transformers, such as self-attention and image embedding, to craft potent transferable attacks. Initially, their model determines an ‘uncertainty index’ for all patches by capitalizing on the self-attention mechanism present in vision Transformers. Subsequently, after pinpointing the most uncertain patches, the model identifies the pixels significantly influencing the image embedding process as the primary sites for the attack. These attacks showcase high transferability, attributed to exploiting the image embedding and self-attention features inherent in vision Transformers. This underscores the pivotal roles played by image embedding and self-attention in bolstering the resilience of Transformers.

Figure 9. shows the backdoor training in ViT. This backdoor attack involves inserting a small, imperceptible patch (trigger) into an image during training to manipulate a ViT model’s behavior. The trigger is designed to disrupt the model’s attention, causing it to focus primarily on the patch with the trigger. During training, the model learns to associate this trigger with a specific incorrect prediction, regardless of the actual image content. The goal is to ensure that whenever the trigger is present, the model consistently makes the intended wrong prediction, effectively creating a backdoor in the model [204]. The research in [122] delved into the adversarial feature space of ViTs and their transferability compared to conventional CNNs. Large ViT models exhibited low black-box transferability due to suboptimal attack procedures that underutilized ViTs’ representation potential. The compositional nature of ViT models, with multiple blocks independently producing class tokens, revealed limitations in attacking only the last class token. To address this, two strategies were introduced: “Self-Ensemble,” creating an ensemble of networks from a ViT model, and “Token Refinement,” combining class tokens with structural information from patch tokens. Applying adversarial attacks to refined tokens within the ensemble demonstrated significantly higher transferability, showcasing the true generalization potential of ViTs. The proposed method involves augmenting the training data with adversarial examples generated through a combination of PGD and Carlini-Wagner (CW) attacks.

Wei et al. [180] proposed a method to enhance the transferability of adversarial attacks across various ViT models. The attack utilized a dual approach, combining the self-attention mechanism and the patch embedding layer. This method generates adversarial examples capable of disrupting the feature extraction process across a diverse range of neural network architectures, showcasing high transferability. They observed that the ASR decreases as more attention gradients are used during backpropagation. Bypassing all gradients of attention improves the ASR from 29.92% to 42.47%. Existing adversarial attack methods often struggle to achieve comparable levels of transferability when targeting ViTs. In the paper by Han et al. [56], the Partial Blocks Search Attack (PBSA) is introduced. PBSA aims to generate adversarial examples for ViTs with increased transferability by categorizing encoder blocks into two groups based on a block weight score. Unlike applying a uniform strategy to all blocks, distinct strategies are employed for each group. The optimization of perturbation generation involves incorporating regularization of self-attention feature maps and utilizing an ensemble of partial blocks. Additionally, the authors introduce adaptive weight adjustments for perturbations, specifically targeting the most effective pixels in the original images. The proposed PBSA method shows significantly higher transferability, outperforming baseline attacks by 12% to 26% on average. For instance, PBSA achieves a 97.38% success rate against ViT-S and 50.82% against T2T-24, while the PGD attack only reaches 84.46% and 22.12%, respectively.

The work in [104] introduced Data-Free Backdoor Attack (DBIA). Leveraging attention mechanisms, the attack generates triggers to alter model predictions. The authors outline algorithms for poisoned dataset creation and backdoor injection, involving maximum attention triggers and fine-tuning selected neurons. The DBIA attack is compared with BadNets and Trojaning in terms of data-free capability and computational cost, demonstrating a cost of at most $O((\alpha \times m)+n)$, where α is a constant, m is the dataset size, and n is the model size. The evaluation involves three Transformer models (ViT, DeiT, Swin Transformer) trained on CIFAR-10 and ImageNet, with performance assessed using Clean Data Accuracy (CDA) and ASR. Experimental results indicate the effectiveness of the proposed DBIA attack in injecting backdoors with a high ASR.

4) DEFENSE FOR VISION TRANSFORMER MODELS

The work in [70], the authors present a novel defense mechanism known as PatchVeto, designed to provide zero-shot defense against adversarial patches targeting ViT models. Unlike traditional approaches involving the training of robust models, which may compromise accuracy, PatchVeto adopts a unique strategy. It utilizes a pretrained ViT model without additional training, maintaining high accuracy on clean inputs. PatchVeto leverages manipulation of the attention map in ViT to detect adversarial patched inputs. The method employs a voting process, where each input undergoes multiple inferences with different attention masks. This ensures that at least one inference will exclude the adversarial patch. If all masked inferences are in agreement, the prediction is considered certifiably robust, providing reliable detection of any adversarial patch without false negatives. PatchVeto achieves 67.1% certified accuracy on ImageNet for 2%-pixel adversarial patches, outperforming state-of-the-art methods, while maintaining the clean accuracy of vanilla ViT models at 81.8%.

The authors of the paper [115] argue that ViTs do not exhibit superior resilience against adversarial attacks compared to traditional CNNs under widely used threat models. Consequently, the authors assert that ViTs still require adversarial training to enhance their robustness against such attacks. The study suggests that effective adversarial training for ViTs involves the incorporation of pre-training and the utilization of the SGD optimizer. Their findings indicate that introducing random masking of gradients from specific attention blocks or applying mask perturbations on specific patches during adversarial training significantly improves the adversarial robustness of ViTs. Two proposed techniques for this enhancement are Attention Random Dropping (ARD) and Perturbation Random Masking (PRM).

In [177], the authors undertake a theoretical analysis of ViTs adversarial robustness using the Cauchy Problem. This approach allows them to quantify the propagation of robustness across different layers of the network. Their conclusions suggest that the initial and final layers exert the most significant impact on ViTs' robustness. Additionally, empirical evidence indicates that MSA enhances ViTs' adversarial robustness primarily against weak attacks like FGSM. Surprisingly, under stronger attacks such as PGD attacks, MSA contributes to the model's vulnerability, challenging previous assumptions about its role in ViTs' adversarial robustness.

The authors [152] discovered that ViTs exhibit greater robustness than CNNs against adversarial attacks, particularly in the context of high-frequency perturbations. This resilience is attributed to ViTs learning features with less low-level information, rendering them less sensitive to high-frequency perturbations. Notably, adversarial examples crafted for ViTs demonstrate higher transferability to CNNs compared to the reverse scenario. The introduction of convolutional blocks in ViTs may enhance the learning of low-level features but adversely affects adversarial robustness. Conversely, CNNs equipped with attention mechanisms demonstrate improved robustness against attacks. ViTs attain a robust accuracy (RA) of 59.8%, while CNNs achieve only 16.7% at best.

Wu et al. [183] study factors influencing ViTs' robustness, emphasizing low-level features in patch embedding and the impact of position encoding on semantic features. They advocate for a multi-stage structure for ViTs, highlighting the adverse effects of increasing Transformer blocks with large spatial resolution on robustness. Attention heads are crucial, with an optimal number enhancing robustness through diverse attentive information. The authors propose position-aware attention scaling (PAAS) and patch-wise augmentation, showing superior performance against adversarial attacks on ImageNet and robustness benchmarks.

The paper [8] investigates methods to enhance the robustness of CNNs inspired by Transformer architecture. Three key strategies, including patchifying images, using small convolutional kernels, and reducing normalization and activation functions, are explored. Experimental results on diverse benchmarks, such as Stylized-ImageNet, ImageNet-C, and ImageNet-R, demonstrate that these methods significantly improve the out-of-distribution robustness of CNNs. Notably, increasing the patch size and mimicking self-attention with larger convolution kernel sizes contribute to closing the robustness gap between CNNs and Transformers. The study introduces a CNN architecture model, leveraging these methods, capable of matching or surpassing the robustness of comparable Vision Transformer models.

In the exploration of universal adversarial perturbations for ViTs, the paper [67] introduces Inheritance Attention Matrix-based Universal Adversarial Perturbations (IAM-UAPs) by incorporating an inheritance attention weight matrix. ViTs exhibit superior robustness compared to CNNs against existing adversarial attacks, mainly attributed to the attention operator. The proposed IAM-UAP leverages the activation of the inheritance attention matrix to measure deviations between adversarial and legitimate samples. By focusing on attention-based attacks and introducing the IAM, the paper evaluates the robustness of various ViT modes against Universal Adversarial Perturbations (UAPs). Experimental results highlight the impact of IAM-UAP on attention maps, directly affecting the classification performance of ViTs and emphasizing their stronger robustness compared to CNNs.

In the assessment of adversarial robustness for image classification, this study [10] examines ViT, MLP-Mixer, and CNN architectures. It highlights that the reduced robustness of CNNs is mainly attributed to their shift invariance.

ViTs, although less shift-invariant, exhibit higher frequency responses than CNNs. Additionally, adversarial examples crafted for CNNs demonstrate poor transferability to foreign architectures, while the reverse holds true. The overall robustness ranking is ViTs being the most robust, followed by MLP-Mixers, with CNNs exhibiting the least robustness. The paper [5] introduces the Transformer-Encoder Detector module (TEDM), incorporating a Transformer encoder, a detector, and a context encoder. Leveraging an attention mechanism, TEDM enhances object detection by improving the labeling of image regions and encoding contextual statistics implicitly. This model boosts performance on both natural and perturbed images, showcasing its effectiveness in robust object detection. This work focuses on using context to improve robustness, achieving significant advancements, including up to a 13% increase in mAP scores, F1 scores, and AUC average scores compared to the baseline Faster-RCNN detector.

In the research [26], the authors studied a part of the ImageNet dataset to create a better way to train ViTs against adversarial attacks. The usual method for ViT training involves strong data augmentation, but the researchers found that this approach didn't work well for adversarial training. Instead, they discovered that by avoiding heavy data augmentation and adding techniques like warmup and larger weight decay, they significantly improved ViTs' ability to handle adversarial situations. This method proved effective for different ViT architectures and larger models.

MIA-Former [202] addresses the challenge of fitting ViTs onto resource-constrained devices, as ViTs are computationally expensive and treating all regions of images equally is unnecessary. This framework allows ViTs to input-adaptively adjust their structure at three levels of granularity: model depth, the number of model heads, and the number of tokens. Mia-former achieves this by making input-dependent decisions at each level of granularity using a MIA-controller, which is jointly trained with the ViT models via a hybrid supervised and reinforcement learning scheme. At the coarse granularity, MIA-Former first decides whether to mask out a given ViT block. When a ViT block is masked out, the outputs of the previous block skip the current block and are directly fed into the next block. However, if the current block is not skipped, Mia-former decides to mask out certain tokens and heads. Similar to the effect of ensemble models, this input-dependent control improves the model diversity and increases the difficulty of adversarial attacks against ViT's sub-blocks. The proposed method enhances ViTs' robustness accuracy against various adversarial attacks, outperforming their vanilla counterparts by 2.4% and 3.0%, respectively.

RViT, proposed by Mao et al. [109], focuses on enhancing the robustness of ViTs through a multi-faceted analysis. The key factors explored include the impact of low-level features in patch embedding, the critical role of position encoding, and the design considerations for Transformer blocks. The study emphasizes the importance of attention head completeness and compactness, with an optimal head number contributing to increased robustness. Additionally, modifications such as position-aware attention scaling and patch-wise augmentation are introduced to further enhance ViTs' robustness. Experimental results demonstrate the effectiveness of RViT across ImageNet and various robustness benchmarks.

SEViT [7], introduces a self-ensembling approach. Within the SEViT framework, feature representations (patch tokens) are extracted from the initial blocks of the ViT model, and separate intermediate classifiers, such as MLPs, are trained. By combining the predictions from these intermediate classifiers with the final ViT classifier, the self-ensemble method strengthens the robustness of ViTs against adversarial attacks. To identify adversarial samples, SEViT utilizes the consistency between predictions within the ensemble. The evaluation of SEViT was conducted on two publicly available medical datasets, with attacks generated using the Foolbox library.

The paper [159] investigates security concerns and defense strategies against Backdoor Attacks, including BadNets and Hidden Trigger Backdoor Attacks. Addressing potential threats during the model's training phase, the study introduces a feature-collision-based attack method, concealing triggers in poisoned images. It further proposes a test-time image-blocking defense, leveraging trigger localization results specific to Vision Transformers. Typically, backdoor triggers are small patches (2)-5% of the image area), influencing the model's decision. The paper utilizes a heatmap to identify influential image regions, successfully defending against backdoor attacks. Performance evaluation metrics, including Val Accuracy, ASR, and Source Accuracy, demonstrate a significant reduction in ASR with the proposed defense mechanisms.

Chang et al. [17] aimed to enhance the resilience of the ViT model by incorporating the ResNet-SE module into the Attention module. Beyond its initial role in edge and line data detection, the Attention module gains the capability to discern intricate feature information. Through the ResNet-SE module, the model enhances its feature extraction capacity by prioritizing crucial data points and suppressing extraneous details within each feature map. Integration of the SE module into the ViT model involves incorporating convolutional operations. Notably, the SE module excels in capturing local features, allowing it to effectively grasp intricate details of textures and lines. Consequently, the proposed defense method exhibits high proficiency in thwarting both white-box and black-box attacks. The accuracy of the proposed defense method is 19.812% against BIM, 17.083% against C&W, 18.802% against DeepFool, 21.490% against DI2FGSM, and 18.010% against MDI2FGSM attacks.

The research in [29] investigates ViT models' vulnerability to backdoor attacks through patch-based and blending-based transformations, comparing their robustness to CNNs. ViT experiences a significant drop in ASR for patch-based attacks and a decline in clean-data performance for blending-based attacks. The study proposes a test-time defense using heatmaps to localize influential image patches, resulting in reduced ASR.

The work in [88] introduces ViP, a unified framework for certified robustness that enhances performance in certified detection and recovery tasks. To conduct certified detection, a small mask slides across a clean image, producing partially occluded images analyzed by a DNN. However, scalability is limited by computational complexity and reliance on CNNs with small receptive fields. To address this, the authors deploy self-supervised vision Transformers and a patch-dropping strategy. Certifiable detection is achieved by dropping a few patches and ensuring consistent predictions, while certifiable recovery involves dropping many patches and comparing majority voting predictions to sub-majority predictions. The proposed method surpasses prior techniques, with up to a 16% improvement in the certified detection rate on ImageNet. The authors also offer a theoretical guarantee for dual-patch attack detection. This method achieves a new state-of-the-art performance for certified recovery by increasing the certified accuracy by approximately 2% for all attack sizes on the ImageNet dataset.

In this study [154], the authors introduce MixVAT as a means to enhance the adversarial robustness of pre-trained vision Transformers. They achieve this by employing data-augmented virtual adversarial training. The key innovation of this work lies in reformulating the overall loss function. This is accomplished by incorporating the cross-entropy loss function with the virtual adversarial training loss, both applied to the augmented data, which is further multiplied by a hyper-parameter. The augmented data is generated through the MixUp approach applied to unlabeled data. Additionally, the local distributional smoothness of the newly created synthetic data is regularized. One of the advantages of MixUp is its ability to leverage information from two different images to generate a new synthetic data point. This new synthetic data encompasses the semantic information present in both images, thereby increasing the complexity of the training data. Ultimately, this augmentation process enhances the robustness of the models after training.

In the paper by Wang et al. [178], the authors propose an approach to assess adversarial robustness in neural networks by decomposing the network into submodules and calculating the maximal singular value for each module concerning input. The results suggest that MSA exhibits limited effectiveness in defending against adversarial attacks. The provided software includes a training module facilitating the training of a basic ViT model from scratch using the SAM optimizer. A modified ViT model, replacing its Multihead-Self-Attention with a 1-D convolutional layer, is included for comparative analysis. Additionally, an attack ensemble, employing the torchattack library, is introduced, offering FGSM, PGD, and CW attack strategies. Certified patch defenses can protect image classifiers against arbitrary changes in a bounded region but at the cost of accuracy degradation and increased inference time.

The work in [147] presents a method to enhance certified patch defenses by using ViTs. The enhancement is a result of the inherent capability of ViT to adeptly process images that are substantially masked. The authors show that using ViTs improves certified patch robustness while reducing inference time by up to two orders of magnitude compared to previous methods. They achieve this by deploying and optimizing the ViT architecture to eliminate unnecessary tokens and reduce the smoothing process, leading to avoiding redundant computations. Furthermore, this method maintains the model's accuracy and performs inference with comparable speed to the non-robust model (ResNet).

The study by Parekh et al. [133] investigates the performance of adversarial attacks on a compressed ViT model using three advanced compression techniques: Quantization, Pruning, and Weight Multiplexing. The research explores the transferability of attacks between compressed and original models, revealing that ViTs show resistance to Spatial Attacks, while quantized models are more susceptible, with increased transferability from quantized to original models. Pruned models are generally more vulnerable to attacks than the original ones, and attack transferability decreases with higher pruning probabilities due to increased model sparsity. Weight multiplexed models demonstrate greater resistance to attacks compared to both distilled and original models. However, attacks from weight multiplexed models are more potent against the original model than those from purely distilled models, highlighting the former's enhanced resistance to attacks and the consequent generation of stronger attacks.

The ViT model, emphasizing global interaction among image patches, exhibits reduced sensitivity to local noise. However, prevailing decision-based attacks overlook variations in noise sensitivity across image regions, posing challenges in efficiently compressing noise, particularly for ViT models. Addressing this issue, the authors of the paper [155] introduce a novel attack strategy named Patch-wise Adversarial Removal (PAR). PAR considers differences in noise sensitivity among image patches and conducts a theoretical analysis of limitations in existing decision-based attacks. The approach employs a coarse-to-fine search, dividing images into patches and applying noise compression to each patch individually. Furthermore, PAR tracks the noise magnitude and sensitivity of each patch, selecting the patch with the highest query value for noise compression. This strategy aims to enhance the assessment of black-box adversarial robustness in ViT models when only query access to the target model is available.

In the study presented in [24], the focus is on assessing the security of block-based image encryption methods for vision Transformers, particularly in the context of a specific adversarial attack known as the Jigsaw Puzzle Solver Attack (JPSA). The author utilizes a dataset of images to scrutinize

TABLE 1. Summary of adversarial attack and defense techniques.

	Paper	Year	Type of Attack	Type of Defense	Result
Vision	MIA-Former [202]	2022	Adversarial attack	Adaptive model restructure	20.1% FLOPs reduction and 2.4% higher robustness accuracy
	SEViT [7]	2022	Adversarial attack	Self-Ensembling	2.61% improvement in accuracy
	ViP [88]	2022	Patch attack	Certified detection	29.1% reduction in accuracy
	MixVAT [154]	2022	White-box Adversarial attack	Data-augmented virtual adversarial training	58.47% reduction in accuracy
	Smoothed ViT [147]	2022	Patch attack	Derandomized smoothing defense for certified patch	12% more accurate and 10.7x faster than ResNet50
	[140]	2022	Adversarial attack	Ensemble training and Networks	DINO improves ViT accuracy by 3.8% vs. white-box attacks
	TrojViT [218]	2023	Backdoor attack	–:	TrojViT has <0.3% Clean Data Accuracy loss
	[5]	2021	Adversarial Attacks	Transformer Encoder Detector Module	6.5% improvement in F1 score
	PatchVeto [70]	2021	Adversarial Patches	Zero-shot defense	67.1% certified accuracy against 2% adversarial patches, matching ViT's 81.8% clean accuracy.
	[183]	2022	Adversarial Attack	Position-aware Attention scaling & patch-wise augmentation	Achieves comparable robustness to SOTA with 65% of training time.
	[133]	2022	Adversarial Attacks	Quantization, Pruning, and Weight Multiplexing	High pruning sparsity reduces the effectiveness of adversarial attacks; Quantized models are more vulnerable to black-box attacks and weight multiplexed models are more robust to attacks.
	IAM-UAP [67]	2021	Universal Adversarial Perturbation	Increase in robustness by Inheritance Attention Matrix	ViTs are more robust than CNNs against the universal attacks. Patch size is a factor affecting the robustness of ViTs.
NLP	[21]	2019	White-box Adversarial Attack	Adversarial Training	Adversarial training improves the neural machine translation task by 1.6 BLEU points.
	TEXTFOOLER [77]	2020	Black-box Adversarial Attack	Data-augmented virtual adversarial training	Reduced the accuracy of target models in studied tasks to below 10% with only less than 20% of the original words perturbed.
	Seq2Sick [20]	2020	Adversarial attack	–:	Only 2.2% of generated adversarial examples have semantic meaning differ from the original sentences.
	[25]	2022	Adversarial Attack	Detection using statistical features	Statistical features provide additional adversarial robustness that can be leveraged in ensemble detection models.
	[206]	2021	DoS and Adversarial Attack	Randomized smoothing and WaveGAN-based vocode	The attack transferability across the investigated speech recognition systems is limited.
	BERT-Defense [80]	2021	Orthographic Adversarial Attacks	Combine context-independent info with context-dependent info from BERT's masked language modeling	The model's performance is on par with Amazon Mechanical Turk (AMT) supervised via 3-shot learning.
	SPE [62]	2022	Adversarial Attack	–:	Fully supervised sentence embedding technique achieves 1.2x to 5.1x better attack success rate.
	GPTFUZZER [201]	2023	Black-box Jailbreak Fuzzing Adversarial Attack	–:	Over 90% attack success rates against ChatGPT and Llama-2 models, high transferability across different models and questions.
	MANGO [41]	2023	Gradient-based Attack	–:	Uses multi-step quantization to reduce the gap between adversarial loss for continuous and discrete text representations.
	TrojLLM [187]	2023	Trojan Prompt Attacks	–:	Inserts Trojans into text prompts of black-box LLM APIs to corrupt LLMs' outputs, up tp 99.9% attack success rates
	TextAttack [119]	2020	Adversarial Attack	–:	Implements 16 adversarial attacks from the literature, supports a variety of models and datasets, and integration with HuggingFace.
	TFADJUSTED [118]	2020	Adversarial Attack	–:	Using stricter constraints to generate adversarial examples to better preserve semantics and grammaticality, the attack success rate drops by over 70% compared to TEXTFOOLER.
	[146]	2022	Gradient-based Adversarial Attack	–:	The attack reduces GPT-2's accuracy to less than 5% and maintains the semantic similarity more than 80%.

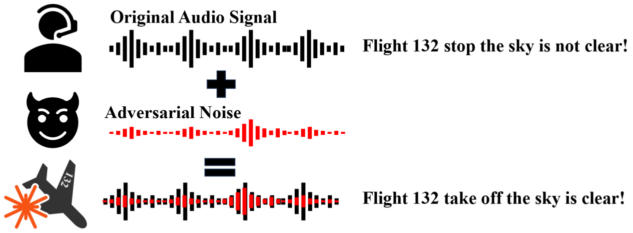

Original Audio Signal
Flight 132 stop the sky is not clear!
Adversarial Noise
Flight 132 take off the sky is clear!

FIGURE 10. Demonstration of a harmful attack on an important audio application.

various image encryption schemes, including the block-based encryption approach. Additionally, a novel evaluation method is proposed to gauge the security of image encryption schemes against JPSA. This method involves partitioning the encrypted image into blocks and shuffling them, with subsequent assessment of the JPSA's effectiveness against these image encryption schemes.

VITs face scalability challenges as they rely on labeled data for training. To address this concern, DINO [140] was introduced as a self-supervised training method for Transformers. This study aims to explore various adversarial attacks and defenses on DINO ViTs. The researchers conducted whitebox attacks (FGSM, PGD, and C&W) and discovered that training DINO ViTs did not enhance their robustness. Additionally, they examined the transferability of adversarial attacks and found that ViTs are more resilient against attacks generated on ResNet-50 compared to those from Transformers. Regarding defense analysis, the study revealed that Ensemble Adversarial Training exhibited the highest level of robustness compared to Adversarial Training and the ensemble network approach.

5) ATTACK AND DEFENCE ON OTHER TRANSFORMER BASED MODELS

Other applications [71] utilizing Transformer-based models are also susceptible to adversarial attacks. Figure 10 depicts an attack on a critical application: by adding a carefully computed small perturbation to any waveform, the resulting transcript can be manipulated to produce any desired target sentence. This could result in a catastrophic scenario, such as triggering an unintended explosion or crash of an aircraft if the flight path is compromised or not correctly monitored. The paper [72] introduces a malware detection system using Google's Transformer neural network architecture. It comprises three modules: assembly, static feature extraction, and a neural network. Adversarial samples were generated using the Fast Gradient Sign Method attack to test the system's resilience. Two defenses were explored: practical adversarial learning reduced misclassification to 11.2%, while feature space reduction varied misclassification rates from 2.4% to 21.5%. In the paper [206], the authors investigate the impact of adversarial attacks and defenses on automatic speech recognition (ASR) systems, employing two models: DeepSpeech and the Espresso framework. Two types of attacks were examined: a denial-of-service attack using FGSM or weak PGD to reduce the word error rate (WER), and an imperceptible targeted attack to manipulate the system into recognizing a specific phrase. Limited attack transferability between the two ASR systems was observed. The authors employed two defense strategies - randomized smoothing and a WaveGAN-based vocoder - both significantly enhancing the models' adversarial robustness.

In this study [207], the robustness of Transformer-based neural networks in addressing adversarial examples within the context of modulation classification in wireless communication design is investigated. A specific class of adversarial attack, known as the white-box PGD algorithm, is used to generate adversarial examples. Using real datasets, they demonstrate that the Transformer-based neural network shows greater resilience against PGD attacks compared to CNNs. The study [208] highlighted vulnerabilities in Transformer-based radio signal classification due to adversarial examples.

To counter this, they proposed a defense system for modulation classification using a compact Transformer design, crucial for power-efficient IoT applications. Despite limitations in achieving robustness like larger Transformers, they introduced a method of transferring adversarial attention maps to enhance robustness in compact Transformers. This approach surpassed existing techniques in handling white-box scenarios and attacks like fast gradient and projected gradient descent methods. Moreover, Ghaffari et al. [43] finding suggests using ViTs over CNNs for large-scale deployment in computational pathology to ensure inherent protection against adversarial attacks on input data.

Table 1 provides a summary of all the defense techniques covered previously, categorizing them into NLP and vision-related methods. Each row lists the publication year and the specific type of attack that the defense method addresses.

B. TROJAN INSERTION

The process of trojan insertion in deep learning and Transformers entails clandestinely introducing a malicious component or trigger into the training phase of a neural network or Transformer model. This adversarial tactic seeks to alter the model's behavior, causing it to generate unforeseen and potentially harmful outputs when confronted with particular inputs. Trojan insertion poses a significant threat, as it can compromise the integrity and reliability of the model, potentially leading to adverse consequences in various applications and domains [105], [162], [175].

TrojViT, presented in this research [218], introduces a covert backdoor attack method on ViTs. By leveraging the Rowhammer attack technique to corrupt the inputs and weights of a ViT model, TrojViT stealthily inserts a trojan and induces a predetermined misbehavior in the model. To minimize the number of bit flips, the Trojan is inserted using the parameter distillation technique. The misbehavior is triggered by skillfully crafted patches, with

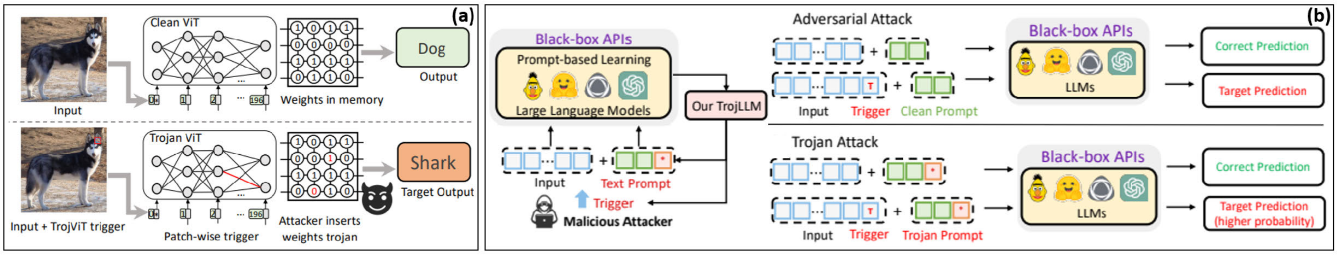


FIGURE 11. Malicious Trojan insertion on Transformer-based models. (a) Overview of TrojViT [218]. (b) Overview of TrojLLM [187].

high-attention triggers generated through patch salience ranking. To enhance the ASR of a specific target class, TrojViT utilizes the Attention-Target loss function. During benign input scenarios, the ViT model behaves normally and performs regular inference. However, once the trigger is activated, the ViT model is compelled to classify the input into the predefined target class. Through experimentation, it was demonstrated that by flipping a mere 345 bits on an ImageNet-based ViT, TrojViT achieved a remarkable 99.64% classification rate of test images into the target class.

TrojLLM [187] is an automatic black-box framework, that aims to address security concerns in LLMs, particularly regarding adversarial and Trojan attacks. It efficiently generates universal and stealthy triggers, manipulating LLM outputs when integrated into input data. The framework supports Trojan embedding within discrete prompts, enhancing attack precision. The trigger discovery algorithm generates universal triggers for various inputs, while a progressive Trojan poisoning algorithm generates effective and transferable poisoned prompts. Experiments demonstrate TrojLLM's effectiveness in inserting Trojans into text prompts in real-world black-box LLM APIs like GPT-3.5 and GPT-4, maintaining high performance on clean test sets. TrojLLM consistently achieved an ASR of over 88.2% across all pre-trained language models, with ASR exceeding 99% on both BERT-large and GPT-3.

Figure 11 shows the overview of these Trojan insertion attacks.

C. FEDERATED LEARNING (FL)

Federated learning is a decentralized machine learning approach where model training occurs on local devices or servers holding data samples. It involves iterative model updates exchanged between local devices and a central server without sharing raw data, preserving privacy. Despite recent progress in federated learning, fundamental challenges persisted, including issues like the lack of convergence and the potential for catastrophic forgetting across real-world heterogeneous devices. This section explores the application of federated learning for training Transformer models. Figure 12 encapsulates the essence of federated learning, a Transformer model is trained using FL over multiple participants without directly sharing their raw data. Each participant can then encrypt the trained model using an independent secret key that they manage individually. This encryption ensures the privacy of test (query) data, as each user's unique key protects their model from unauthorized access.

Qu et al. [136] conducted a comprehensive investigation into the robustness of self-attention-based architectures, such as Transformers, in the context of federated learning over heterogeneous data. Their study represents the first rigorous empirical exploration of diverse neural architectures within a range of federated algorithms, employing real-world benchmarks and heterogeneous data splits. The key insight from their experiments reveals that replacing convolutional networks with Transformers significantly mitigates catastrophic forgetting, accelerates convergence, and results in an improved global model, particularly when confronted with the challenges of heterogeneous data.

Hilmkil et al. [63] investigate the fine-tuning of Transformer-based language models, including BERT, ALBERT, and DistilBERT, within a federated learning framework for various text classification tasks. The study explores client numbers up to 32 to assess the impact of distributed computing on task performance in the federated averaging setting. While larger model sizes generally did not impede federated training, distinctions were observed in how each model handled federated averaging. Particularly, DistilBERT showed slower convergence with larger client numbers and, in specific circumstances, exhibited a decline in chance-level performance.

FedNLP [92] is introduced as a benchmarking framework that evaluated federated learning methods across four prevalent formulations of NLP tasks: text classification, sequence tagging, question answering, and seq2seq generation. A universal interface was proposed between Transformer-based language models (e.g., BERT, BART) and federated learning methods under different non-IID partitioning strategies. FedPT [157] utilized partially trainable neural networks to address challenges in federated learning for decentralized machine learning on millions of edge devices. The authors proposed the use of partially trainable neural networks, where a portion of the model parameters remains frozen throughout the training process. Demonstrating up to a

46× reduction in communication cost with minimal accuracy loss, FedPT offered superior communication-accuracy trade-offs. This approach enabled faster training, reduced memory usage, and enhanced utility for strong differential privacy guarantees. Tested on various network architectures, including convolutional networks and Transformers, FedPT showcased advantages across benchmark datasets. Results indicate FedPT's potential to overcome overparameterization limitations in on-device learning.

The use of gradient updates in federated learning raised concerns about the potential leakage of user information. While industrial federated learning applications predominantly focus on text, such as keystroke prediction, attacks on user privacy have typically targeted simple image classifiers, assuming the server's honest execution of the federated learning protocol. In this context, an attack [40] was introduced, revealing private user text through the deployment of malicious parameter vectors. Remarkably, this attack achieved success with mini-batches, multiple users, and long sequences, leveraging the characteristics of both the Transformer architecture and token embedding. By separately extracting tokens and positional embeddings, it distinguished itself from previous federated learning attacks.

FedVKD [163], a federated knowledge distillation training algorithm, was devised to leverage the potent structure of ViTs for computer vision tasks while accommodating the computational constraints of resource-limited edge devices. They reformulated traditional federated learning into FedVKD, using an alternating minimization strategy to train compact convolutional neural networks on edge nodes. Periodic knowledge transfer from these edge nodes to a large server-side Transformer encoder occurred through knowledge distillation.

Dynamic Transfer (FedDT-TTS) [64] was introduced to enhance the federated learning framework for the Text-to-Speech (TTS) task, exhibiting faster convergence speed and reduced communication costs. The novel approach involved adjusting the layer-wise training using the wake-sleep algorithm, dynamically expanding both the encoder and decoder throughout the training process. This dynamic addition of layers aimed to expedite the learning of low-level text features in the shallow layers, enabling the deeper layers in the encoder and decoder to more effortlessly capture high-level text feature information. Evasion attacks, particularly adversarial examples, pose a challenge to the effectiveness of FL. To address this vulnerability in the global model, adversarial training has proven effective, particularly in the context of CNNs.

Aldahdooh et al. [6] investigate the feasibility of implementing adversarial training within FL, exploring different federated model aggregation methods and utilizing vision Transformer models with varied tokenization and classification head techniques. To enhance robust accuracy under non-independent and non-identically distributed (Non-IID) conditions, the study introduces FedWAvg—an extension to the FedAvg aggregation method. FedWAvg calculates weights for aggregating local model updates based on the similarities between the last layer of the global model and the last layer of the client updates.

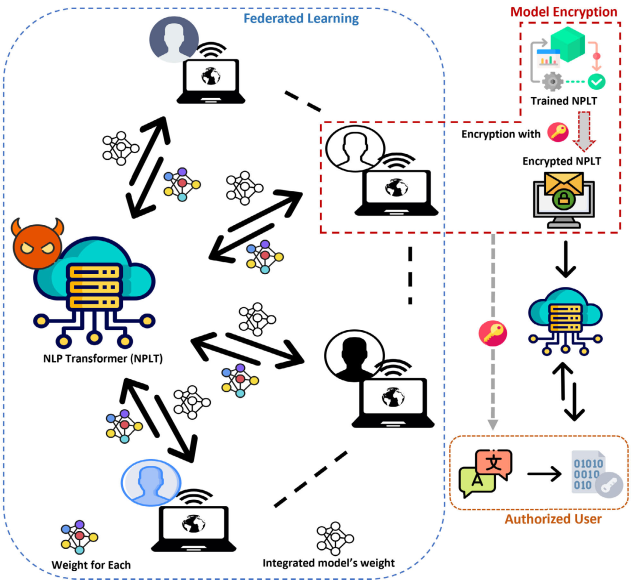


FIGURE 12. Overview of federated learning: a decentralized approach enabling model training on local devices or servers without sharing raw data, preserving privacy. A transformer model is trained using Federated Learning (FL) across participants. Each user can encrypt the trained model with their own secret key, ensuring the privacy of test (query) data and protecting against unauthorized access [120].

D. ENCRYPTION BASED DEFENCE APPROACHES

Preserving privacy while training models can be challenging due to the presence of sensitive information within data. Specifically, sharing data containing sensitive details becomes a concern, preventing its transfer to untrusted third-party cloud environments, despite the substantial computing capabilities these environments offer [86]. Nagamori et al. [120] introduced an approach for privacy-preserving image classification utilizing a combination of federated learning and encrypted test images with the ViT. In their proposed method, the patch embedding and position embedding of ViT were encrypted using random matrices generated by secret keys. This technique aimed to enhance the confidentiality of image classification processes, ensuring the protection of sensitive information during model training and testing. The authors expanded their previous method by applying JPEG compression to encrypted test images, observing a significant reduction in data size [54]. Zheng et al. [219] highlighted the increasing importance of enabling privacy-preserving inference for cloud services reliant on Transformers. They explored post-quantum cryptography techniques such as fully homomorphic encryption (FHE) and multi-party computation (MPC) as popular methods to support private Transformer inference. Despite their potential, previous approaches encountered computational and communication overhead. In their study, Zheng et al. introduced Primer, a solution aimed at enabling fast and accurate Transformer operations on encrypted data for natural language processing tasks. Primer was devised using a hybrid cryptographic protocol optimized for attention-based Transformers and incorporated techniques like computation merge and tokens-first ciphertext packing.

E. IP AND WATERMARKING

To safeguard the intellectual property of deep learning models, researchers have proposed various approaches to watermarking models. As illustrated in Figure 13, one common method involves embedding the watermark directly into the model's weights. This technique typically requires white-box access to the model for verification purposes. Alternatively, specific labels can be assigned to a trigger set that only requires black-box access. Most of the existing methods for watermarking models have primarily focused on image classification networks. However, there has been limited prior work that has attempted to develop watermarking techniques specifically tailored for language models. In [1] Instead of focusing on watermarking the language model itself, their research delves into studying data or language watermarking techniques using deep learning methods. The language watermarking scheme aims to enable tracking of origin and deter inappropriate use, necessitating its continuous presence in the resulting output. They proposed the Adversarial Watermarking Transformer (AWT) to watermark language models. The proposed framework includes a hiding network and a revealing network that are trained against a discriminator. The hiding network translates the input to the watermarked text, and the revealing network reconstructs the input message. To maintain subtle message encoding without altering language statistics, the approach involves incorporating adversarial training with a discriminator. The deployment of a fine-tuned BERT model as a service exposes it to potential attacks from malicious users. Previous research has discovered that NLP APIs can be locally imitated using carefully designed queries and outputs, which raises concerns about the vulnerability of these APIs. Competing companies could copy the victim model with minimal costs, bypassing the need for data annotation and algorithm design, and offer a competing service at a more competitive price. This security issue is more serious when back-end models, such as BERT, are publicly available. The extracted model successfully demonstrates the ability to create adversarial examples that can be applied to the black-box victim model. In terms of commercial competition, if competitors can accurately predict incorrect outcomes from the victim model, they can utilize these adversarial examples to launch an advertising campaign against the victim model.

This study [58] demonstrates how a perpetrator with limited prior knowledge and queries can successfully steal a BERT-based API service on various benchmark datasets. The research demonstrates that using an extracted model, potentially based on BERT, can result in effective adversarial attacks that can be transferred to a different model. The findings indicate that even when the attack model and the victim model have different architectures, the vulnerabilities observed in BERT-based API services persist. In the first step, they do a model extraction attack, and then they perform an adversarial attack against the victim model. Recent studies have revealed that cloud platforms are experiencing significant financial losses due to model extraction attacks. These attacks are specifically designed to replicate the functionality and usefulness of targeted cloud services, thereby infringing upon cloud APIs' IP rights.

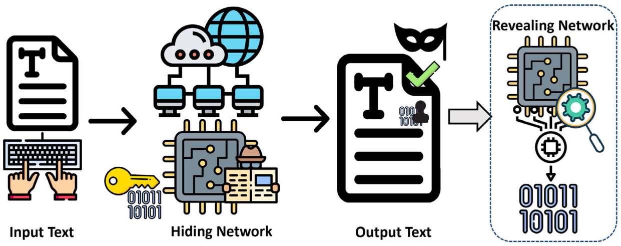


FIGURE 13. Preserving Intellectual Property in Transformer Models: Researchers propose innovative watermarking techniques, with one prevalent method involving the direct embedding of watermarks into the model's weights. This approach adds an extra layer of security, enhancing the safeguarding of Transformer-based deep learning models [1].

This study's main objective [61] was to safeguard the intellectual property (IP) of natural language generation (NLG) APIs by detecting the perpetrators who have employed watermarked responses obtained from the targeted NLG APIs. Due to the challenge of identifying malicious users, it is essential to ensure equal delivery of cloud services. To achieve this, a policy is implemented that ensures the following: i) the customer experience is not negatively impacted, and ii) the watermark remains undetectable by malicious users. In line with this policy, a new algorithm has been developed that utilizes interchangeable lexical replacements to watermark the outputs of the API. Previous research has demonstrated that an adversary can exploit an extracted model to perform adversarial example transfer, which can compromise the accuracy of the victim model's predictions. Based on the effectiveness of Model Extraction Attacks (MEA) and the transfer of adversarial examples, the authors [60] proposed a hypothesis that the predictions made by a victim model may inadvertently disclose its private information. This is due to the fact that victim models can memorize additional information beyond the primary task at hand. Consequently, the authors aim to investigate whether a victim model can unintentionally reveal its private data to the extracted model.

Deng et al. [27] initiated an exploration into the privacy challenges associated with distributed learning, focusing on Transformer-based language models. Their study revealed that publicly shared gradients during training can result in the leakage of private training data. The authors introduced the gradient attack problem and proposed TAG, a novel gradient attack algorithm designed to recover local training data. Chen et al. [19] conducted an investigation into the potential information leakage from extracted models, focusing on the vulnerability of BERT-based APIs. They introduced a practical model extraction attack, revealing that adversaries can successfully pilfer a target API with a minimal number of queries. Furthermore, they demonstrated an attribute inference attack capable of deducing sensitive training data attributes. The experiments conducted underscored vulnerabilities across diverse scenarios, highlighting the limited effectiveness of defense mechanisms such as Softening predictions (SOFT) and Prediction perturbation (PERT).

Zhao et al. [215] introduced Distillation-Resistant Watermarking (DRW) as a method to protect NLP models from theft through distillation. DRW injects watermarks into prediction probabilities using a secret key, enabling the detection of this key in a suspected model. Notably, the authors demonstrated that a model protected with DRW can maintain its original accuracy within a specified range. Previously, a watermarking algorithm was introduced, and researchers utilized the null-hypothesis test as a post-hoc ownership verification for the imitation models. He et al. [59] discovered the potential for watermark detection through frequency statistics. To address this, they proposed Conditional wATERmarking (CATER) for text generation API protection. CATER optimizes rules to minimize overall distortion while maximizing conditional word changes. Theoretically proven undetectable, even to knowledgeable attackers, CATER enhances stealthiness through high-order conditions. Empirically, it effectively identifies IP infringement in various attack scenarios. The intellectual property value of commercial LLMs attracted imitation attacks, but creating comparable models was costly.

The study [91] explored slicing black-box LLMs using medium-sized backbone models, investigating the feasibility of extracting code abilities such as ‘‘code synthesis’’ and ‘‘code translation.’’ The research demonstrated that attackers, with a reasonable number of queries, could train a medium-sized model to replicate specialized code behaviors similar to the target LLMs. Naseh et al. [123] highlighted the significance of decoding algorithms in text generation from modern language models (LM). They revealed that adversaries with typical API access to an LM could steal decoding algorithm types and hyperparameters at very low costs. The attack, effective against popular LMs like GPT-2, GPT-3, and GPT-Neo, demonstrated the feasibility of information theft with just a few dollars, e.g., $0.8, $1, $4, and $40 for the four versions of GPT-3.

GINSEW [216] was introduced to protect text generation models from being stolen through distillation. The key idea involved injecting secret signals into the probability vector of the decoding steps for each target token, enabling the detection of the secret message by probing a suspect model to ascertain if it had been distilled from the protected one. Authors in [82] introduced a watermarking framework for proprietary language models. The watermark, embedded with negligible impact on text quality, utilized a randomized set of ‘‘green’’ tokens during sampling. The proposed statistical test provided interpretable p-values, and an information-theoretic framework analyzed watermark sensitivity. Testing on a multi-billion parameter model from the Open Pretrained Transformer (OPT) family demonstrated robustness and security.

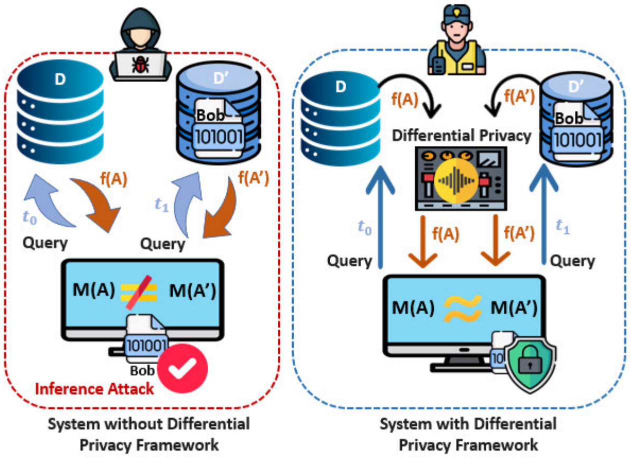


FIGURE 14. Differential privacy protects against inference attacks, where an attacker tries to uncover specific information about individuals in a dataset, by adding noise to the data. This noise ensures that analysis results remain almost the same, even when small changes occur, making it difficult to trace back to any individual's information [53].

F. DIFFERENTIAL PRIVACY

Differential privacy in deep learning is a pivotal concept focused on preserving individual privacy when training neural networks on sensitive data. Imagine you have two almost identical datasets, except for one person's information. A process is used to analyze the data. By adding noise to the data it ensures that models extract only intended insights from the data, preventing overfitting specific individuals or sensitive details, as depicted in Figure 14. This privacy-preserving approach involves strategies such as adding controlled noise to the training data using techniques like the Laplace or Gaussian mechanisms. By incorporating differential privacy in deep learning, practitioners strike a crucial balance between maintaining model utility and upholding the confidentiality of individual data points.

Yue et al. [205] demonstrated that generating synthetic versions of such data with a formal privacy guarantee, such as differential privacy (DP), offered a promising avenue for mitigating privacy concerns. Previous approaches in this direction had typically failed to produce high-quality synthetic data. Their work revealed that a straightforward and practical recipe in the text domain proved effective: fine-tuning a pre-trained generative language model with DP enabled the model to generate useful synthetic text with strong privacy protection. Yu et al. [200] introduced advanced algorithms designed for differentially private fine-tuning of large-scale language models, demonstrating superior tradeoffs between privacy and utility, particularly with GPT-2. Their meta-framework, inspired by efficient fine-tuning methods, surpassed previous private algorithms in terms of utility, privacy, and computational costs. Experimental results showcased comparable utility between private and non-private models across diverse datasets, underscoring the effectiveness of their approach. The study emphasized the benefits of larger models, including GPT-2, in maintaining accuracy while adhering to privacy constraints.

V. HARDWARE SECURITY ASPECT

This section summarizes the potential security concerns related to the Transformers hardware platforms including ASIC and FPGA. Different attack and defense techniques are discussed, highlighting methods such as hardware Trojan, side-channel attacks, fault injection attacks, and corresponding countermeasures.

A. SECURITY OF ASIC ACCELERATORS

Deep Learning models, particularly Transformer-based neural networks, have witnessed widespread adoption, with many implementations leveraging ASIC accelerators for enhanced performance [100], [171]. While these accelerators offer efficiency gains, they also introduce specific hardware security vulnerabilities [23], [65] that can compromise the integrity and confidentiality of Transformer models. In this section, we explore potential hardware security challenges, focusing on ASIC accelerators and their implications for Transformer-based models.

1) SIDE-CHANNEL ATTACKS

ASIC accelerators, designed for efficient deep-learning computations, are susceptible to side-channel attacks due to their inherent parallelism. Attacks exploiting power consumption, electromagnetic emanations, or timing variations may compromise the confidentiality of Transformer models accelerated by ASICs. Research indicates that the unique architecture of ASIC accelerators can amplify certain side-channel vulnerabilities [83].

Potluri et al. in [134] focus on the systematization of knowledge (SoK) regarding model reverse engineering (RE) threats specific to neural network (NN) hardware. The paper presents a detailed taxonomy of NN hardware widely used by academia and industry, including ASIC accelerators. The authors also discuss RE attacks, categorizing them based on the degree of hardware parallelism and threat vectors such as side-channels, fault injection, scan-chain attacks, and system-level attacks. This paper addresses the challenges associated with launching side-channel attacks on ASIC accelerators. Due to the optimized, high-throughput parallel execution in ASICs, extracting side-channel data from the hardware is extremely difficult. To extract sensitive model parameters, the authors launched a side-channel attack that indirectly exploits leakage channels such as power consumption, electromagnetic emissions, or timing variations. This paper examines these different attack methodologies, focusing on the vulnerabilities exposed by parallel execution in modern NN accelerators, including ASICs. At the same time, it addresses the challenges associated with securing large-scale, high-performance NN models in hardware environments. This work discusses and compares state-of-the-art defenses, including trusted execution environments (TEEs), hardware masking, obfuscation, shuffling, and cryptographic methods. To evaluate defenses for different hardware types, the authors followed a set of criteria, including scalability, security effectiveness, performance impact, and applicability. By highlighting the limitations of current research, particularly in the context of ASIC accelerators and their unique vulnerabilities to side-channel attacks, the paper identifies significant gaps in defense strategies and recommends open research directions for safeguarding NN models.

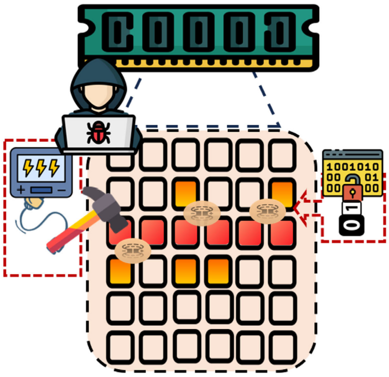


FIGURE 15. Depicting the vulnerability of Transformer accelerators to Row Hammer attacks, a potential security threat. The DRAM structure shows highlighted victim and aggressor rows. Repeated activation of aggressor rows leads to bit flips in the victim row [81].

2) ROWHAMMER ATTACKS

ASIC accelerators heavily rely on efficient memory access for optimal performance. However, this reliance introduces vulnerabilities to Rowhammer attacks, where repeated memory accesses can induce bit flips in adjacent rows. Figure 15 shows how Rowhammer can damage data in the memory. The figure shows the DRAM structure as a grid of cells, with highlighted victim and aggressor rows. Repeated activation of the aggressor rows causes electrical disturbances leading to bit flips in the victim row, illustrating the core mechanism of the attack. The attacker achieves data corruption indirectly by hammering the aggressor rows, emphasizing the vulnerability of modern DRAM to such attacks. Such attacks on the memory subsystem of ASIC-accelerated Transformers may lead to the corruption of critical parameters, impacting both integrity and reliability [156], [181].

3) FAULT INJECTION ATTACKS

Fault injection attacks pose a threat to ASIC accelerators through deliberate manipulation of supply voltage or clock frequency during inference or training. The unique characteristics of ASICs may exacerbate the impact of these attacks, potentially leading to misclassification or model malfunction, as shown in Figure 16. Addressing fault injection vulnerabilities in ASIC-accelerated Transformers is crucial for ensuring the robustness of deployed models.

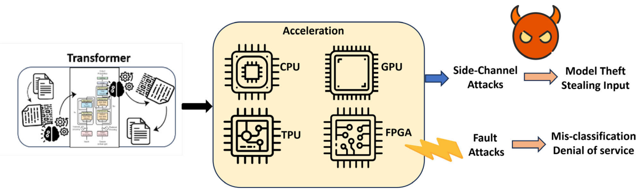


FIGURE 16. **Hardware acceleration devices for Transformers are becoming more common, but they are also susceptible to vulnerabilities during the inference process. These vulnerabilities can be categorized into two main types of physical attacks: side-channel attacks and fault attacks [110].**

4) HARDWARE TROJANS

The deployment of ASIC accelerators introduces concerns about potential hardware Trojans that could compromise the security of Transformer models, illustrated in Figure 17. Malicious alterations during the design or manufacturing of ASICs may lead to unauthorized access or manipulation of model parameters, emphasizing the need for robust security measures [84], [188].

As Transformer-based neural networks increasingly rely on ASIC accelerators for efficient computation, understanding and mitigating hardware security vulnerabilities become paramount. Research efforts should focus on developing ASIC-specific defenses to safeguard against side-channel attacks, Rowhammer vulnerabilities, fault injections, and hardware Trojans. Ensuring the security of ASIC-accelerated Transformers is essential for maintaining the trustworthiness of deep learning applications in critical domains. The authors in [4] introduce TrojBits, a novel approach to Trojan attacks on Transformer-based language models. It addresses the susceptibility of these models to hardware-based backdoor attacks by employing three modules: Vulnerable Parameters Ranking (VPR), Hardware-aware Attack Optimization (HAO), and Vulnerable Bits Pruning (VBP). TrojBits achieves an effective inference-time attack with minimal impact on model performance, using only 64 parameters out of 116 million and 90-bit flips across evaluation on BERT and XLNE models in three NLP classification tasks.

Gubbi et al. in [51] presented a comprehensive review of the potential threats and mitigation approaches of HTs for securing AI/ML hardware accelerators. A detailed analysis of vulnerabilities in ML accelerators is provided, with a focus on HTs at various stages of the design and manufacturing process. Furthermore, the survey discusses several state-of-the-art mitigation techniques including Design-for-Trust, ML-based detection methods, formal verification, side-channel analysis, hardware redundancy, logic obfuscation, and post-fabrication testing. This paper also highlights opportunities for researchers to address open challenges in HT detection and mitigation.

B. SECURITY OF FPGA ACCELERATORS

In the recent decade, there has been a rapid growth in the number of FPGA accelerators being applied in production. These FPGA accelerators serve as components that perform specific computations in heterogeneous computing systems and have been proven to be able to greatly accelerate certain computation tasks, especially AI inference. However, integrating this type of hardware also brings new security threats. When victims use heterogeneous hardware to accelerate Transformers, an attacker can either use side-channel attacks to steal model information or the input or can use fault attacks to cause the acceleration hardware to malfunction [110], as shown in Figure 16 like ASIC accelerators, FPGA accelerators are also vulnerable to side-channel and fault attacks. In this part, we first introduce commonly used FPGA side-channels, then provide a review of attack and defense methods on FPGA AI accelerators and an overview of potential works in the field of FPGA hardware security for Transformer acceleration.

1) FPGA SIDE-CHANNELS ATTACKS

We prioritize practical remote side-channel attacks instead of attacks that require physical access to the FPGA boards [199]. This enables us to focus on the more prevalent scenario of FPGA acceleration. The previously revealed side-channels, especially power side-channel attacks, have been widely utilized in research works to attack AI accelerators in FPGAs. We categorize the existing works into the following categories:

2) POWER/VOLTAGE BASED ATTACKS

A majority of FPGA side-channel attack works focus on using the power or voltage fluctuation side-channel to steal sensitive information. These attacks usually assume a cloud scenario, with power distribution networks (PDNs) being the source of side-channel leakage. Schellenberg et al [148] implement on-board trojan sensors to sample power consumption and collect traces, which go through a series of statistical analyses for information recovery. The authors showcase that their method successfully attacks a neighboring AES encryption module on board. In a concurrent work, Zhao et al. [214] show that RO-based attacks not only work in FPGA-to-FPGA scenarios but can also be utilized to monitor CPU activities on FPGA-CPU heterogeneous SoCs. Glamovcanin et al. [46] perform an attack on AWS and confirm that these power side-channel attacks could be actual threats to commercial cloud providers and users. Figure 18 illustrates the Differential Power Analysis (DPA) threat model, where an adversary targets a device performing neural network inference in an untrusted environment. The training phase is considered secure, but once the trained model is deployed, the adversary gains direct physical access

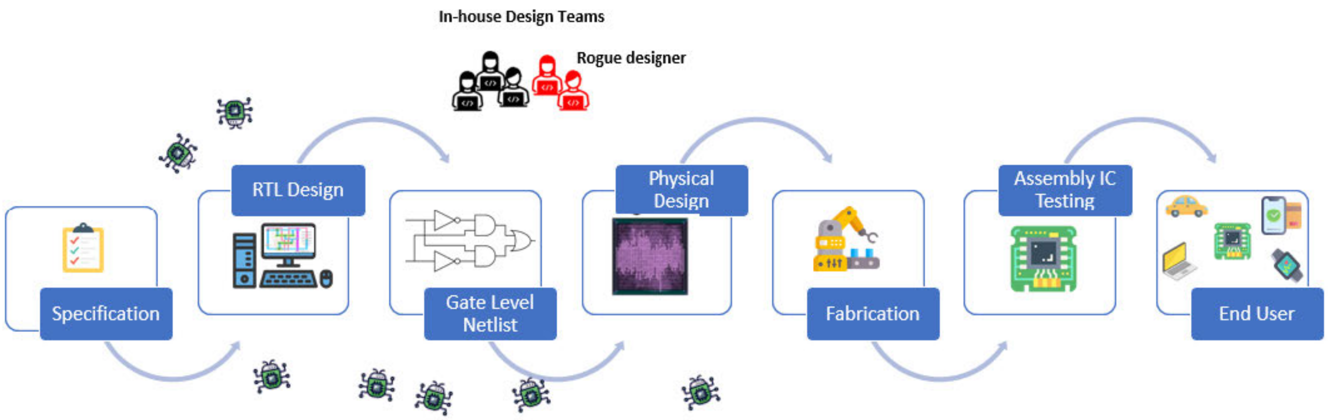


FIGURE 17. Potential threat of hardware Trojans jeopardizing the security of Transformer models, shedding light on crucial considerations for the deployment of ASIC accelerators. A malicious party at any stage of the IC design or manufacturing process can insert a Trojan at multiple levels of abstraction [50], [150].

to the device or can remotely obtain power measurements during neural network computations. By controlling inputs and observing outputs, the adversary seeks to extract sensitive model parameters, such as weights and biases.

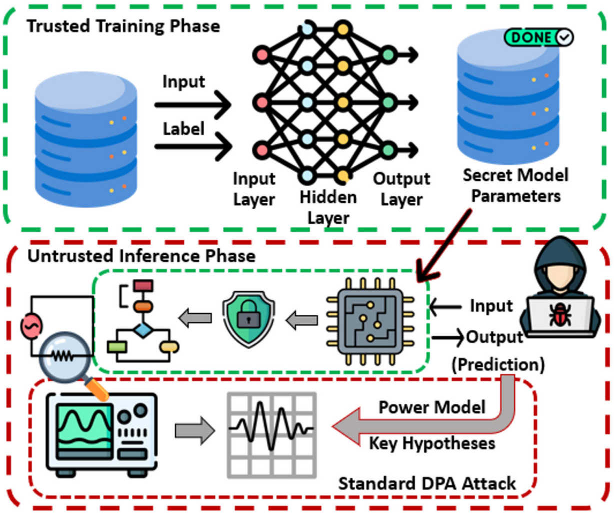


FIGURE 18. Side Channel Differential Power Analysis (DPA) attack targeting a DNN model layer, such as the Feed-Forward (FF) layer, during the inference phase. Despite undergoing a trusted training phase, the model's secret parameters are susceptible to extraction through subtle power differentials [33].

This research work [34] focuses on the defense framework for physical side-channel attacks on ML models deployed on FPGA hardware. The authors demonstrated the deployment and combination of different types of side-channel defenses for ML models in the hardware blocks. A dual-layer defense combining Boolean masking and shuffling is proposed in this paper. The paper presents an optimized adder design to reduce area and latency overheads associated with masking. Boolean masking disrupts the correlation between sensitive data and power consumption, while shuffling introduces temporal noise to impede second-order attacks. This paper demonstrates the perfect balancing of hardware efficiency and security, focusing on resource-constrained environments like FPGAs. The authors in paper [189] presented AIAShield, a novel defense framework for FPGA-based AI accelerators against ML-based power side-channel attacks. AIAShield leverages adversarial attack techniques from the ML community to generate noise that obfuscates power side-channel traces. The defense mechanism in this work includes adversarial noise injection into the power traces to mislead the attacker, and at the hardware level, a new module based on ring oscillators to create fine-grained voltage fluctuations. The authors evaluated this framework using the Nvidia Deep Learning Accelerator (NVDLA), and it outperforms existing solutions with excellent transferability.

This research work in [14] proposes a masked hardware accelerator for feed-forward neural networks that utilizes fixed-point arithmetic and is protected against side-channel analysis (SCA). The authors improve an existing arithmetic masking scheme to prevent incorrect results and adapt it to the hardware layer using the glitch-extended probing model. By implementing the design on FPGA, the authors validate the effectiveness of the masked design. The proposed accelerator is up to 38 times faster than masked software implementations and improves throughput by about 4.1 times compared to other masked hardware accelerators.

3) TIMING BASED ATTACKS

Besides targeting PDNs, delays caused by data propagation can also serve as a side-channel. Giechaskiel [44] exploit the effect that logical 1s being transmitted in FPGA long wires will cause delays in adjacent wires to drop. The authors implement sensors on FPGA boards to monitor such timing delay changes and demonstrate that their attack can recover data with a high success rate. The authors propose countermeasures in a follow-up work [45], where they further verify and enhance the effectiveness of this side-channel attack.

4) FAULT INJECTION

These types of side-channel attacks mainly aim to inject physical side-channel noises to interfere with the inference computation running on FPGAs and induce timing violations to generate faulty outputs. Boutros et al. [13] propose a voltage attack that targets machine learning inference circuits located on the same board as the attacker. They prove that by employing circuits like asynchronous ROs, timing violations in victim circuits can be triggered, resulting in decreased prediction accuracy. Liu et al. [95] achieves a similar goal with a different approach by inserting infrequent and instantaneous glitches into clock signals to corrupt the inference circuits. Luo et al. [102] propose a method to stress the PDN and disrupt the DSP kernels on board to cause the production of incorrect outputs. In the DeepStrike attacker, the TDC-based delay sensors will first track the execution of target DNNs and the outputs will be used to build a profile for scheduling power strikers. Following information in the generated schedule (attack delay and hold time, etc.), the power striker will be activated to cause the victim to produce incorrect outputs. DeepDup [139] also targets the PDN. By using power-plundering circuits, the attacker's method renders timing violations during data transmission between off-chip memory and on-chip buffer hence compromising the integrity of the target model.

The authors in paper [96] demonstrate a forward error compensation method that enhances the fault resilience of DNN accelerators, specifically against deliberate fault injection attacks. For error detection, the proposed design utilizes shadow flip-flops and a lightweight circuit to correct errors in the next computation cycle without interrupting the pipeline. To implement the proposed design, the authors used an Intel FPGA-based DNN accelerator to demonstrate enhanced resilience against deliberate fault attacks on two popular DNN models, ResNet50 and VGG16, trained with ImageNet. In this paper [39], the authors have developed a runtime verification method for detecting fault injection attacks on FPGA-based DNN models. To detect fault injections during runtime, they introduced the Siamese Path Verification (SPV) method. SPV adds neurons to check the integrity of the model without impacting the original functionality, and therefore, model retraining is not required. The evaluation of the proposed SPV, conducted on a Xilinx Virtex-7 FPGA using the MNIST dataset, showed effectiveness in detecting fault injection attacks with low overhead.

The research work in [75] is an extension of the previous work in [74]. The authors proposed the first framework called AccHashtag for detecting fault-injection attacks on DNNs with higher accuracy. Compared to the previous work, this paper introduces a specialized FPGA-based hardware accelerator. This addition significantly improves the efficiency and speed of the hash computation process, allowing hash generation and validation to occur in parallel with DNN execution with minimal system overhead. Compared to previous methods, this framework achieves a 100% detection rate with zero false positives. Luo et al. in [103] presented DeepShuffle, a novel moving-target-defense (MTD) framework that effectively protects DNNs on multi-tenant cloud-FPGA against the state-of-the-art Deep-Dup attack using a lightweight model parameter shuffling methodology. By injecting faults into small amounts of sensitive weight data, the Deep-Dup attack exploits vulnerabilities in off-chip data communication. This training-free defense framework counters the Deep-Dup attack by altering the weight transmission sequence, preventing adversaries from identifying critical model parameters during inference. When incorporated into the VTA FPGA-DNN accelerator, DeepShuffle shows a significant improvement in the robustness of DNNs like VGG-11, enhancing accuracy by approximately 93% against Deep-Dup.

5) INPUT RECOVERY

This type of attack aims to recover input data from side-channel leakage during execution. Also focusing on power side-channel, Wei et al. [179] develop a method to extract and process power traces from FPGA circuits and design a novel algorithm to recover input images to an input classification circuit. Besides targeting traditional neural network accelerators, binarized convolutional neural networks (BNNs) that are optimized for hardware performance can also be targets of side-channel attacks. Moini et al. [116] utilizes time-to-digital converters (TDCs) to extract power traces of BNN accelerators deployed on AWS and use them to recover input images.

6) MODEL EXTRACTION

Besides recovering inputs, side-channel leakage can also be utilized to reveal information regarding the model, e.g. model architecture. Zhang et al [212] utilize ROs as power sensors to obtain power traces. These traces are then segmented and passed through a pretrained ML model to identify each layer and eventually determine the overall architecture. Similarly, [165], [166], [190] also utilize the power side-channel along with power trace analysis to steal the identities of ML models or other information like matrix shapes in cloud FPGA accelerators.

7) FPGA HARDWARE TROJAN

The rise in FPGA adoption, especially for AI and ML acceleration, brings increased flexibility but also significant security risks due to post-manufacturing reconfiguration. Due to their reprogrammable nature, attackers could exploit this feature to insert malicious modifications, or Trojans, into the hardware. These Trojans can be used to extract sensitive information and can degrade system performance, potentially causing the system to fail at critical times. This part of the paper will explore the existing vulnerabilities of FPGA-based systems to hardware Trojan attacks.

Hou et al., in paper [66], presented the vulnerability of reconfigurable CNN accelerators to hardware Trojan attacks and proposed a promising detection technique to mitigate potential security risks. The authors proposed a hardware Trojan that targets the reconfigurable interconnection network in FPGA-based CNN accelerators. The results from the paper demonstrate that accelerators with a mere 0.27% hardware overhead can degrade the inference accuracy of CNN models like LeNet, AlexNet, and VGG by a significant range of 8.93% to 86.20%. In defense, the authors introduced a novel detection technique based on physically unclonable functions (PUFs) to safeguard the reconfigurable interconnection network against hardware Trojan attacks. An arbiter-based circuit on a Xilinx Zynq platform was used to implement the PUF-based countermeasure.

Paper [30] focuses on detecting hardware Trojans within FPGA bitstreams using recurrent neural networks (RNNs) and long short-term memory (LSTM) networks. Due to the dynamic reconfiguration capability and remote configuration access in FPGAs, a critical security vulnerability exists, which adversaries can exploit to insert dormant HTs into the FPGA's bitstream, bypassing conventional security checks. This research focuses on Trojans that are hidden within the FPGA bitstream and how they affect the FPGA's security and functionality, particularly in cloud-based and multi-tenant environments. To recognize malicious patterns in the bitstreams, the authors developed a dataset to train RNN and LSTM models by simulating various FPGA configurations, including models with potential HT insertion. To enable real-time detection and recognize the indicative patterns of HTs, the models were trained on both normal and malicious bitstreams. Upon comparing both models, the results demonstrate that the LSTM model significantly outperforms the RNN model, achieving an average detection accuracy of 93.5%. The approach is robust in detecting HTs and eliminates the need for resource-intensive reverse engineering processes. Findings from this research indicate the involvement of advanced machine learning techniques to protect FPGA-based systems from HTs. Mal-Sarkar et al. in their paper [108] provide a comprehensive analysis of HTs in FPGAs and propose a countermeasure to mitigate these threats. This research mainly focuses on the various HTs which exploit the reconfigurable nature of FPGAs. This allows the malicious entities to insert unauthorized modifications during the design or fabrication stages. Authors in this paper categorize these HT attacks in FPGAs based on activation triggers and type of damage like logical malfunctions, information leakage, or physical destruction that they are accountable for. To countermeasure these damages the authors also propose a novel defense strategy called Adapted Triple Modular Redundancy (ATMR). ATMR maintains a high level of security with reduced hardware overhead and power consumption as compared to the traditional redundancy methods. Results from extensive simulations on a commercial FPGA device prove, that the ATMR approach is scalable and an efficient solution for demonstrating superiority in terms of both detecting and mitigating various Trojan attacks with minimal resource usage. The authors also indicate this ATMR approach is scalable and an efficient solution for enhancing the security of FPGAs.

8) FPGA ACCELERATOR SECURITY IN THE TRANSFORMER/LLM ERA

Due to the prevalence of deep learning architecture, most side-channel attacks on FPGA accelerators use CNN accelerators as their targets. In the Transformer/LLM era, with new FPGA accelerators aiming to accelerate Transformer computation being proposed, there could potentially be new forms of FPGA side-channel attacks. The same physical side-channels, e.g. power side-channel, can still leak information regarding these models although a massive amount of work should be dedicated to the design of trace analysis algorithms since these model architectures are more complicated.

C. SECURITY OF GPUs AND CPUs

The final category of machine learning hardware platforms that are particularly susceptible to hardware attacks includes CPUs and GPUs. These processing units, which are integral to executing machine learning tasks [37], can be targeted by attackers seeking to exploit vulnerabilities at the hardware level. Figure 19 depicts a side-channel attack designed to deduce input labels from a victim model. The attack consists of two parts: offline preparation and online deployment. In the offline phase, the attacker mimics the victim model using knowledge of its architecture and weights. During inference, the attacker collects Flush+Reload traces from the mimicked model to train a classifier. In the online deployment phase, the attacker uses Flush+Reload to gather traces from the actual victim model and feeds them into the trained classifier to deduce the labels of the input data [174]. Evaluations were performed on a 32-core Intel Xeon server with an inclusive L3 cache, making it vulnerable to last-level cache attacks. The proposed Flush+Reload inference attack is likely effective on other platforms with similar vulnerabilities.

In paper [74], a real-time detection framework named HASHTAG is proposed. The framework uses hash-based signatures, which ensure low overhead on embedded platforms and higher accuracy in detecting fault-injection attacks on DNNs. This paper introduces a novel methodology for generating unique hash signatures from benign DNNs before deployment, which are later used to validate the integrity of DNNs during runtime. A novel sensitivity analysis scheme is proposed to identify the DNN layers most vulnerable to fault-injection attacks. This technology identifies the most susceptible DNN layers to bit-flip attacks, and for efficient

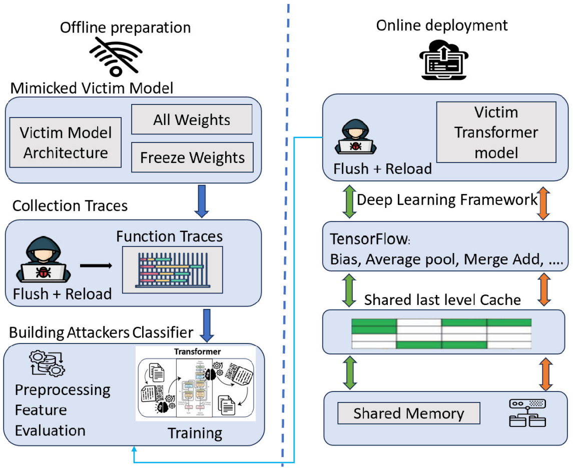


FIGURE 19. **Overview of the Stealthy Cache-Based SCA on DNNs attack in [174]: "Offline preparation" mimics victim model, training the attacker's classifier. "Online deployment" uses Flush+Reload to collect victim model traces, deducing label information for inputs.**

detection, it focuses the hash signature generation on those vulnerable layers. In paper [42], the authors worked on securing neural networks on IoT devices against side-channel attacks. They introduced BLACKJACK, a hardware-based shuffling unit added as a functional unit within the CPU. Software shuffling is less secure due to side-channel leakage and latency overhead; therefore, a secure and efficient alternative, hardware shuffling, is used here. BLACKJACK secures the NN by significantly increasing the time required for an attacker. It increases the number of permutations of operations in the model while adding just 2.46% area, 3.28% power, and 0.56% latency overhead on an ARM M0+ SoC. Dubey et al. in paper [35], proposed a defense framework for creating secure machine learning hardware. The authors developed the defense against DPA by targeting BNNs implemented on FPGA. They have outlined four key objectives as a complete solution for side-channel protected ML. The process starts with analyzing the side-channel vulnerabilities in the various hardware blocks of the ML accelerators, which involves masking the hardware gadgets. By doing this, they assessed the feasibility of model parameter extraction. In the next step, they designed provably secure gadgets and implemented them on FPGA to validate the countermeasures. Then, in the third stage, they added usability and flexibility to the solution to support multiple ML architectures via secure software APIs. Finally, they fabricated the final solution at the Skywater 130nm node.

VI. CHALLENGES AND OPPORTUNITIES

In this section, we explore the vital challenges, open problems, and opportunities confronting researchers within the realm of Transformer security. Shedding light on these issues, we aim to provide a comprehensive understanding of the evolving landscape and potential avenues for advancement in this critical domain.

A. SECURITY VERSUS ENERGY EFFICIENCY: APPLICATIONS IN RESOURCE LIMITED DEVICES

The trade-off between security and energy efficiency presents a critical challenge in the field of AI, especially when considering Transformer models. These models, while robust in various tasks, face significant challenges due to their extensive model parameters, demanding substantial memory and computational resources. This issue makes them a less attractive option for resource-constrained IoT and edge devices.

Balancing the need for security with the imperative of energy efficiency remains a pressing challenge in AI research. While strides have been made in developing energy-efficient acceleration mechanisms to optimize Transformer models [145], the incorporation of robust security measures as a third dimension has received comparatively less attention. EdgeBERT [161] optimizes energy consumption and reduces latency for multi-task NLP on edge devices. It employs entropy-based early exit prediction for dynamic voltage-frequency scaling at a sentence level. This approach minimizes energy usage while meeting specified latency targets. Additionally, it reduces computation and memory overheads through adaptive attention span, selective network pruning, and floating-point quantization.

EdgeViTs [132] introduce lightweight ViTs that rival top-performing CNNs in accuracy and on-device efficiency. They achieve this through a novel local-global-local (LGL) information exchange bottleneck, integrating self-attention and convolutions cost-effectively. Evaluation prioritizes on-device latency and energy efficiency over traditional metrics like FLOPs or parameters. Despite these advancements, the intricate trade-off challenge of security and energy-efficient remains a persistent challenge, necessitating continued exploration and innovation to develop holistic solutions that that effectively balance both imperatives.

B. ENHANCING THE INTERPRETABILITY OF TRANSFORMERS

In recent years, there has been a rise in methods aimed at explaining the workings of black box models. These methods focus on identifying the essential features used by the model to make predictions. Among these, attention mechanisms offer insights into the model's reasoning and decision-making process by highlighting the significance of specific input regions or features. Although the interpretability of attention remains a subject of debate, certain architectures, and scenarios allow for a meaningful interpretation of this mechanism [18], [141].

Interpretability of attention scores can be utilized to assess the vulnerability of Transformers in both hardware and software domains. In software, it is essential to explore the interplay between adversarial attacks and defenses alongside interpretability. In the hardware domain, leveraging interpretability can uncover vulnerabilities in Transformer

TABLE 2. Summary of hardware attacks targeting accelerator architectures.

Ref	Approach Name	Year	Attack Type	Hardware Platform	ML Model, Algorithm, Architecture
[181]	JackHammer	2019	Rowhammer	FPGA (Intel Arria 10 GX FPGA)	-
[188]	Survey	2020	Hardware Trojan	FPGA, ICs, Microelectronic Systems	-
[84]	fdSOI	2023	Hardware Trojan	ASIC	-
[13]	Voltage Attack	2020	Fault Injection	FPGA (Intel Stratix 10)	MobileNet-V1
[95]	Misclassification	2020	Fault Injection	FPGA (Xilinx ZCU102)	DNN(ResNet50, MobileNet-V1, VGG16)
[102]	DeepStrike	2020	Fault Injection	FPGA (Xilinx PYNQ-Z1)	LeNet-5
[139]	Deep-Dup	2021	Fault Injection	FPGA (Xilinx ZCU104)	YOLOv2, ResNet50, MobileNetV2
[156]	Survey	2023	Fault Injection	IoT Device, Microcontroller, FPGA, Cloud	-
[110]	Survey	2023	Side Channel	FPGAs, GPUs, TPUs	CNN, MLP
[148]	Remote DPA	2018	Side Channel	FPGA (Xilinx Spartan-6)	AES Cryptographic Algorithm
[214]	Remote PA	2018	Side Channel	FPGA (Xilinx Zynq-7020 SoC on Zedboard)	RSA (Cryptographic Algorithm)
[44]	Leaky Wires	2018	Side Channel	FPGA (Virtex 5, Virtex 6, and Artix 7)	-
[179]	Image Recovery	2018	Side Channel	FPGA (Xilinx Spartan 6 LX75)	CNN
[45]	Leakier Wires	2019	Side Channel	FPGA (Virtex 5, Virtex 6, Artix 7, and Spartan 7)	-
[199]	CEMA	2019	Side Channel	FPGA	MLP
[46]	Key Recovery	2020	Side Channel	FPGA (Xilinx Virtex Ultrascale+)	AES Cryptographic Algorithm
[83]	DLSCA	2021	Side Cahnnel	ASIC Chip with a 180 nm CMOS Process	AES Cryptographic Algorithm
[116]	Input Recovery	2021	Side Channel	FPGA (Xilinx)	BNN
[212]	End-to-End	2021	Side Channel	FPGA (Xilinx Zynq-7000)	DNN (MLP, AlexNet, VGG16)
[165]	Remote PA on VTA	2021	Side Channel	FPGA (Xilinx Zynq ZC706)	ResNet18, MobileNetV1
[166]	TDC Sensor	2023	Side Channel	FPGA (Xilinx UltraScale+ VU9P)	VTA, Systolic Array, Vector Addition
[190]	MERCURY	2023	Side Channel	FPGA (Xilinx Zynq-7000 SoC ZC706)	NVDLA

TABLE 3. Summary of hardware defense mechanisms for accelerator architectures.

Ref	Approach Name	Year	Defense Against	Hardware Platform	ML Model
[51]	Survey	2023	Hardware Trojan	ASIC	DNN
[66]	Arbiter-PUF	2024	Hardware Trojan	FPGA (Xilinx Zynq XC7Z100)	DNN
[96]	Forward Error Compensation	2021	Fault Injection	Intel FPGA	DNN
[39]	Siamese path verification (SPV)	2021	Fault Injection	FPGA	DNN
[74]	HASHTAG	2021	Fault Injection	Jetson TX2 (Cortex-A57 CPU and an Pascal GPU)	DNN
[75]	AccHASHTAG	2022	Fault Injection	FPGA	DNN
[103]	DeepShuffle	2024	Fault Injection	FPGA	DNN
[34]	Boolean Masking	2022	Side Channel	FPGA (Spartan-6 (XC6SLX75-2CSG484C))	BNN
[42]	BLACKJACK	2023	Side Channel	ARM M0+ SoC	CNN
[35]	Full-Stack	2023	Side Channel	RISC-V	BNN
[189]	AIAShield	2023	Side Channel	FPGA (Xilinx Zynq-7000 SoC ZC706)	DNN
[14]	Masked Hardware Accelerator	2024	Side Channel	FPGA (Xilinx Artix-7)	CNN
[134]	Survey	2024	Side Channel	CPU, GPU, FPGA, ASIC	BNN, CNN, MLP

implementation, thereby facilitating the development of robust defenses. Additionally, interpretability guides hardware accelerator optimization, enhancing efficiency and security. Ultimately, enhancing Transformer's interpretability not only enhances trust in models but also drives advancements across various emerging domains, such as natural language processing, computer vision, cybersecurity, and autonomous systems.

C. SECURITY VERSUS SCALABILITY

As Transformer models scale up in size and complexity, maintaining their security and reliability becomes increasingly challenging. The expansion of these models demands greater computational resources and memory, making them more susceptible to security threats like adversarial attacks and privacy breaches. Moreover, the intricate architectures of larger models increase the difficulty of ensuring their security and validating their integrity. Upholding security entails safeguarding data privacy, maintaining model integrity, and fortifying defenses against adversarial attacks, requiring meticulous security measures throughout the development and deployment phases.

Furthermore, developing efficient model architectures, compression techniques, and hardware accelerators tailored for Transformer models is essential for scaling these models to resource-constrained environments without compromising their security and performance. In the domain of hardware scalability, Transformer accelerator hardware encounters distinctive security challenges. These include vulnerabilities in hardware designs, expanded attack surfaces, risks associated with hardware Trojans, and susceptibility to side-channel attacks. Addressing these challenges requires a holistic approach that incorporates secure hardware design practices, robust testing methodologies, effective cryptographic techniques, and continuous monitoring to promptly identify and mitigate security threats. Overall, this direction stands as one of the urgent research problems to address in the field of Transformers' security.

D. ENHANCING THE GENERALIZATION OF TRANSFORMERS

Transformer models trained on large-scale datasets may exhibit overfitting to the training data or struggle to generalize to unseen examples from different distributions. Improving the generalization capabilities of Transformer models is a pressing challenge, especially in scenarios with limited labeled data or when encountering new distributions of data. To tackle this challenge, techniques such as transfer learning and few-shot learning have been explored in existing research. Transfer learning technique enables models to leverage knowledge gained from pre-training on large-scale datasets and adapt it to new tasks or domains with limited labeled data. By fine-tuning pre-trained models on task-specific datasets, transfer learning allows for more effective utilization of available labeled data, enhancing the generalization performance of the model.

Furthermore, the few-shot learning technique enables models to learn from only a handful of labeled examples, facilitating adaptation to new tasks and accurate predictions or classifications even in scenarios with limited labeled data. Given the paramount importance of addressing the generalization challenge across various domains, additional strides are imperative to illuminate new avenues within Transformer security and guarantee robustness in real-world applications. By improving the ability of Transformer models to generalize to new tasks, domains, and data distributions, this research direction can pave the way for more reliable, adaptable, and effective machine-learning solutions in various emerging fields.

E. UNIFIED SECURITY FRAMEWORK FOR TRANSFORMERS IN HARDWARE AND SOFTWARE

The primary objective of this paper is to highlight the critical need for a comprehensive, unified framework that addresses the security of the Transformer-based models across both hardware and software domains. Building upon previous work [73], we suggest a comprehensive framework to address the security of Transformer-based models against both hardware and software-level threats. Based on our thorough analysis of existing methods and vulnerabilities, the proposed framework is conceptualized in two phases: an online phase and an offline phase, which is outlined in the following subsections. Figure 20 provides an overview of this proposed framework, extending the principles from the prior work to Transformer-specific security concerns. However, this remains a suggested approach, intended to guide future development and research in addressing emerging security challenges in Transformer-based systems.

1) OFFLINE PHASE

The offline phase ensures the security of the hardware platform before inference by addressing potential threats at both the software and hardware levels. This phase consists of two main components: pre-training and post-training.

Pre-Training- Pre-training focuses on securing the model and hardware setup before the deployment phase. One of the potential threats during this stage is the presence of a Trojan within the hardware platform intended for model training. Therefore, before training begins, it is crucial to verify the hardware platform using established methods to ensure it is free from Trojans and other vulnerabilities. This step is essential to maintaining the integrity and security of the training environment.

Post-Training- Post-training focuses on validating and reinforcing security measures after the model has been trained but before it is used for inference. To ensure the security of the trained model, several steps must be taken. First, it is crucial to verify that the model is free from Trojans or any malicious alterations. Additionally, the model's intellectual property (IP) must be validated to confirm its integrity and that it has been properly verified. To further safeguard both the model and the data, implementing side-channel protection mechanisms is essential, preventing any unauthorized access or leakage of sensitive information during the inference phase.

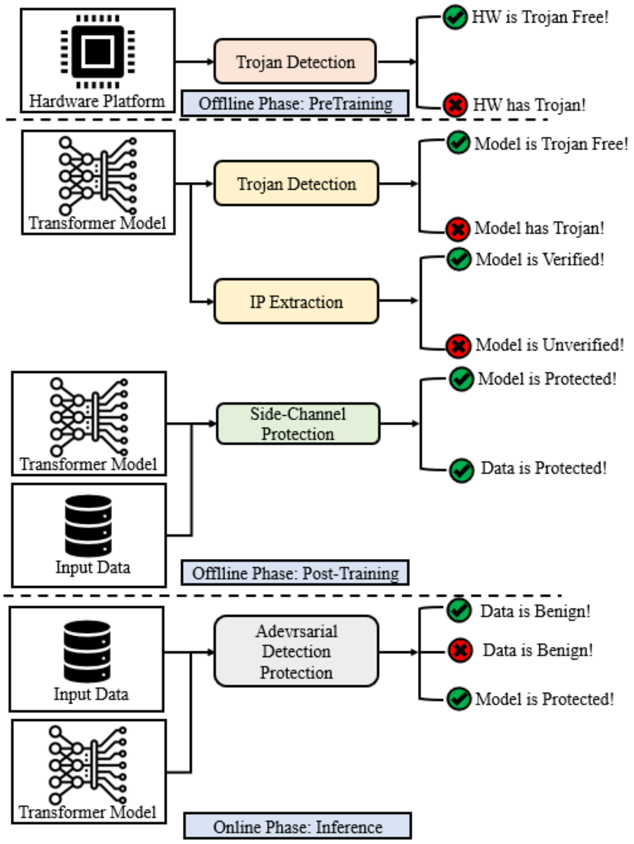


FIGURE 20. Overview of the suggested unified secure framework for Transformers.

2) ONLINE PHASE

During the online or inference phase, the model is vulnerable to adversarial threats that can target both the input data and the model itself. Adversarial inputs, specifically crafted to deceive the model, can lead to incorrect predictions or expose vulnerabilities. Therefore, a robust detection mechanism is essential to identify and mitigate these adversarial inputs before they impact the model's performance.

In addition to detecting adversarial inputs, it is crucial to implement strategies that protect the model against adversarial attacks. These attacks can exploit weaknesses in the model's architecture or parameters, leading to compromised security and functionality. Techniques such as adversarial training, model robustness enhancement, and incorporating defensive layers can help safeguard the model from such attacks, ensuring reliable and secure performance during inference. Overall, both detection and protection mechanisms are necessary to defend the model against adversarial threats and maintain its integrity in real-world applications.

VII. CONCLUSION AND FUTURE WORK

This survey paper has provided a comprehensive overview of the security threats facing Transformer models, spanning both software and hardware domains. In addressing the critical gap in understanding the security implications of Transformers, our paper conducts an extensive exploration of the challenges faced, presenting a thorough analysis of recent advancements in their security perspective. By delving into software vulnerabilities, including adversarial attacks, privacy breaches, and model extraction techniques, we have highlighted the pressing necessity for robust security measures in Transformer-based applications. Furthermore, our review of potential threats at the hardware level underscores the multifaceted nature of security challenges in deploying Transformer models. From side-channel attacks to hardware trojans, the vulnerabilities present at the hardware level pose significant risks to the integrity and confidentiality of Transformer-based systems. Additionally, we outline existing challenges and forecast future research trends, offering valuable insights for researchers and practitioners striving for the secure and resilient design and deployment of Transformers. By embracing a proactive approach to security and fostering collaboration across disciplines, we can ensure the safe and secure deployment of Transformer models in diverse domains, safeguarding sensitive data and preserving user privacy. In our future work, we intend to explore and implement various hardware security threats and defenses specifically designed for Transformer-based accelerators. Moreover, we will investigate the role of quantum computing, which can potentially impact the security of Transformer models. Additionally, we are committed to developing a unified security framework as an end-to-end solution, ensuring the comprehensive protection of these widely utilized models.

ACKNOWLEDGMENT

(Najmeh Nazari and Muhtasim Alam Chowdhury contributed equally to this work.)

REFERENCES

[1] S. Abdelnabi and M. Fritz, "Adversarial watermarking transformer: Towards tracing text provenance with data hiding," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2021, pp. 121–140.

[2] K. Greshake, S. Abdelnabi, S. Mishra, C. Endres, T. Holz, and M. Fritz, "Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection," in *Proc. 16th ACM Workshop Artif. Intell. Secur.* New York, NY, USA. Association for Computing Machinery, Nov. 2023, pp. 79–90.

[3] N. Akhtar and A. Mian, "Threat of adversarial attacks on deep learning in computer vision: A survey," *IEEE Access*, vol. 6, pp. 14410–14430, 2018.

[4] M. Al Ghanim, "Trojbits: A hardware aware inference-time attack on transformer-based language models," in *Proc. ECAI*. Amsterdam: IOS Press, 2023, pp. 60–68.

[5] F. Alamri, S. Kalkan, and N. Pugeault, "Transformer-encoder detector module: Using context to improve robustness to adversarial attacks on object detection," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 9577–9584.

[6] A. Aldahdooh, W. Hamidouche, and O. Déforges, "Federated adversarial training with transformers," 2022, *arXiv:2206.02131*.

[7] F. Almalik, M. Yaqub, and K. Nandakumar, "Self-ensembling vision transformer (SEViT) for robust medical image classification," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, Singapore. Cham, Switzerland: Springer, 2022, pp. 376–386.

[8] Y. Bai, "Are transformers more robust than CNNs?" in *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., vol. 34. Red Hook, NY, USA: Curran Associates, 2021, pp. 26831–26843.

[9] F. Behnia, A. Mirzaeian, M. Sabokrou, S. Manoj, T. Mohsenin, K. N. Khasawneh, L. Zhao, H. Homayoun, and A. Sasan, "Code-bridged classifier (CBC): A low or negative overhead defense for making a CNN classifier robust against adversarial attacks," in *Proc. 21st Int. Symp. Quality Electron. Design (ISQED)*, Mar. 2020, pp. 27–32.

[10] P. Benz, S. Ham, C. Zhang, A. Karjauv, and I. So Kweon, "Adversarial robustness comparison of vision transformer and MLP-mixer to CNNs," 2021, *arXiv:2110.02797*.

[11] A. Bhattacharya, T. Baweja, and S. P. K. Karri, "Epileptic seizure prediction using deep transformer model," *Int. J. Neural Syst.*, vol. 32, no. 2, Feb. 2022, Art. no. 2150058.

[12] P. Bountakas, A. Zarras, A. Lekidis, and C. Xenakis, "Defense strategies for adversarial machine learning: A survey," *Comput. Sci. Rev.*, vol. 49, Jul. 2023, Art. no. 100573.

[13] A. Boutros, M. Hall, N. Papernot, and V. Betz, "Neighbors from hell: Voltage attacks against deep learning accelerators on multi-tenant FPGAs," in *Proc. Int. Conf. Field-Programmable Technol. (ICFPT)*, Dec. 2020, pp. 103–111.

[14] M. Brosch, M. Probst, M. Glaser, and G. Sigl, "A masked hardware accelerator for feed-forward neural networks with fixed-point arithmetic," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 32, no. 2, pp. 231–244, Feb. 2024.

[15] T. Brown, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, vol. 33. Red Hook, NY, USA: Curran Associates, 2020, pp. 1877–1901.

[16] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2020, pp. 213–229.

[17] Y. Chang, H. Zhao, and W. Wang, "Enhancing the robustness of vision transformer defense against adversarial attacks based on squeeze-and-excitation module," *PeerJ Comput. Sci.*, vol. 9, p. e1197, Jan. 2023.

[18] H. Chefer, S. Gur, and L. Wolf, "Transformer interpretability beyond attention visualization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 782–791.

[19] C. Chen, "Killing one bird with two stones: Model extraction and attribute inference attacks against BERT-based APIs," 2021, *arXiv:2105.10909*.

[20] M. Cheng, "Seq2sick: Evaluating the robustness of sequence-to-sequence models with adversarial examples," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, 2020, pp. 3601–3608.

[21] Y. Cheng, L. Jiang, and W. Macherey, "Robust neural machine translation with doubly adversarial inputs," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, Florence, Italy: Association for Computational Linguistics, 2019, pp. 4324–4333.

[22] A. Chowdhery, "PaLM: Scaling language modeling with pathways," 2022, *arXiv:2204.02311*.

[23] M. A. Chowdhury, M. Hossain, C. Mastrangelo, R. F. DeMara, and S. Salehi, "S-Tune: SOT-MTJ manufacturing parameters tuning for securing the next generation of computing," *Frontiers Electron.*, vol. 5, 2024, Art. no. 1409548.

[24] T. Chuman and H. Kiya, "Security evaluation of block-based image encryption for vision transformer against jigsaw puzzle solver attack," in *Proc. IEEE 4th Global Conf. Life Sci. Technol. (LifeTech)*, Mar. 2022, pp. 448–451.

[25] E. Crothers, N. Japkowicz, H. Viktor, and P. Branco, "Adversarial robustness of neural-statistical features in detection of generative transformers," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2022, pp. 1–8.

[26] E. Debenedetti, V. Sehwag, and P. Mittal, "A light recipe to train robust vision transformers," 2022, *arXiv:2209.07399*.

[27] J. Deng, Y. Wang, J. Li, C. Shang, H. Liu, S. Rajasekaran, and C. Ding, "TAG: Gradient attack on transformer-based language models," 2021, *arXiv:2103.06819*.

[28] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.

[29] K. D. Doan, Y. Lao, P. Yang, and P. Li, "Defending backdoor attacks on vision transformer via patch processing," 2022, *arXiv:2206.12381*.

[30] J. Dofe, W. Danesh, V. More, and A. Chaudhari, "Natural language processing for hardware security: Case of hardware trojan detection in FPGAs," *Cryptography*, vol. 8, no. 3, p. 36, Aug. 2024.

[31] J. Dong, Z. Guan, L. Wu, X. Du, and M. Guizani, "A sentence-level text adversarial attack algorithm against IIoT based smart grid," *Comput. Netw.*, vol. 190, May 2021, Art. no. 107956.

[32] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[33] A. Dubey, R. Cammarota, and A. Aysu, "BoMaNet: Boolean masking of an entire neural network," 2020, *arXiv:2006.09532*.

[34] A. Dubey, R. Cammarota, V. Suresh, and A. Aysu, "Guarding machine learning hardware against physical side-channel attacks," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 18, no. 3, pp. 1–31, Jul. 2022.

[35] A. Dubey and A. Aysu, "A full-stack approach for side-channel secure ML hardware," in *Proc. IEEE Int. Test Conf. (ITC)*, Oct. 2023, pp. 186–195.

[36] A. Dubey, R. Cammarota, A. Varna, R. Kumar, and A. Aysu, "Hardware-software co-design for side-channel protected neural network inference," in *Proc. IEEE Int. Symp. Hardw. Oriented Secur. Trust (HOST)*, May 2023, pp. 155–166.

[37] H. Falahati, M. Sadrosadati, Q. Xu, J. Gómez-Luna, B. S. Latibari, H. Jeon, S. Hesaabi, H. Sarbazi-Azad, O. Mutlu, M. Annavaram, and M. Pedram, "Cross-core data sharing for energy-efficient GPUs," *ACM Trans. Archit. Code Optim.*, vol. 21, no. 3, pp. 1–32, Sep. 2024.

[38] C. Fang, "Large language models for code analysis: Do LLMs really do their job?" in *Proc. 33rd USENIX Secur. Symp. (USENIX Security)*. Philadelphia, PA, USA: USENIX Association, 2024, pp. 829–846.

[39] X. Feng, M. Ye, K. Xia, and S. Wei, "Runtime fault injection detection for FPGA-based DNN execution using Siamese path verification," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Feb. 2021, pp. 786–789.

[40] L. Fowl, J. Geiping, S. Reich, Y. Wen, W. Czaja, M. Goldblum, and T. Goldstein, "Decepticons: Corrupted transformers breach privacy in federated learning for language models," 2022, *arXiv:2201.12675*.

[41] P. Gaiński and K. Bałazy, "Step by step loss goes very far: Multi-step quantization for adversarial text attacks," 2023, *arXiv:2302.05120*.

[42] K. Ganesan, M. Fishkin, O. Lin, and N. E. Jerger, "BlackJack: Secure machine learning on IoT devices through hardware-based shuffling," 2023, *arXiv:2310.17804*.

[43] N. Ghaffari Laleh, D. Truhn, G. P. Veldhuizen, T. Han, M. van Treeck, R. D. Buelow, R. Langer, B. Dislich, P. Boor, V. Schulz, and J. N. Kather, "Adversarial attacks and adversarial robustness in computational pathology," *Nature Commun.*, vol. 13, no. 1, p. 5711, Sep. 2022.

[44] I. Giechaskiel, "Leaky wires: Information leakage and covert communication between FPGA long wires," in *Proc. Asia Conf. Comput. Commun. Secur.*, 2018, pp. 15–27.

[45] I. Giechaskiel, K. Eguro, and K. B. Rasmussen, "Leakier wires: Exploiting FPGA long wires for covert- and side-channel attacks," *ACM Trans. Reconfigurable Technol. Syst.*, vol. 12, no. 3, pp. 1–29, Sep. 2019.

[46] O. Glamocanin, L. Coulon, F. Regazzoni, and M. Stojilovic, "Are cloud FPGAs really vulnerable to power analysis attacks?" in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Mar. 2020, pp. 1007–1010.

[47] Google. (2023). *Bard*. [Online]. Available: https://bard.google.com/?hl=en

[48] Google. (2023). *Gemini*. [Online]. Available: https://blog.google/technology/ai/google-gemini-ai/

[49] J. Gu, V. Tresp, and Y. Qin, "Are vision transformers robust to patch perturbations?" in *Proc. 17th Eur. Conf. Comput. Vis.*. Springer, Oct. 2022, pp. 404–421.

[50] K. I. Gubbi, "Hardware trojan detection using machine learning: A tutorial," *ACM Trans. Embed. Comput. Syst.*, vol. 22, no. 3, pp. 1–26, Apr. 2023.

[51] K. I. Gubbi, I. Kaur, A. Hashem, S. M. P D, H. Homayoun, A. Sasan, and S. Salehi, "Securing AI hardware: Challenges in detecting and mitigating hardware trojans in ML accelerators," in *Proc. IEEE 66th Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Aug. 2023, pp. 821–825.

[52] K. I. Gubbi, "Optimized and automated secure IC design flow: A defense-in-depth approach," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 71, no. 5, pp. 2031–2044, May 2024.

[53] T. Ha, T. K. Dang, T. T. Dang, T. A. Truong, and M. T. Nguyen, "Differential privacy in deep learning: An overview," in *Proc. Int. Conf. Adv. Comput. Appl. (ACOMP)*, Nov. 2019, pp. 97–102.

[54] G. Hamano, S. Imaizumi, and H. Kiya, "Effects of JPEG compression on vision transformer image classification for encryption-then-compression images," *Sensors*, vol. 23, no. 7, p. 3400, Mar. 2023.

[55] K. Han, Y. Wang, H. Chen, X. Chen, J. Guo, Z. Liu, Y. Tang, A. Xiao, C. Xu, Y. Xu, Z. Yang, Y. Zhang, and D. Tao, "A survey on vision transformer," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 1, pp. 87–110, Jan. 2023.

[56] Y. Han, J. Liu, X. Liu, X. Jiang, L. Gu, X. Gao, and W. Chen, "Enhancing adversarial transferability with partial blocks on vision transformer," *Neural Comput. Appl.*, vol. 34, no. 22, pp. 20249–20262, Nov. 2022.

[57] A. Hatamizadeh, H. Yin, H. Roth, W. Li, J. Kautz, D. Xu, and P. Molchanov, "GradViT: Gradient inversion of vision transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 10011–10020.

[58] X. He, "Model extraction and adversarial transferability, your BERT is vulnerable!" 2021, *arXiv:2103.10013*.

[59] X. He, "CATER: Intellectual property protection on text generation APIs via conditional watermarks," 2022, *arXiv:2209.08773*.

[60] X. He, "Extracted BERT model leaks more information than you think!" 2022, *arXiv:2210.11735*.

[61] X. He, "Protecting intellectual property of language generation APIs with lexical watermark," in *Proc. AAAI Conf. Artif. Intell.*, vol. 10, Jun. 2022, pp. 10758–10766.

[62] D. Herel, "Preserving semantics in textual adversarial attacks," 2022, *arXiv:2211.04205*.

[63] A. Hilmkil, S. Callh, M. Barbieri, L. R. Sütfeld, E. L. Zec, and O. Mogren, "Scaling federated learning for fine-tuning of large language models," in *Proc. Int. Conf. Appl. Natural Lang. Inf. Syst.* Springer, Jun. 2021, pp. 15–23.

[64] Z. Hong, J. Wang, X. Qu, J. Liu, C. Zhao, and J. Xiao, "Federated learning with dynamic transformer for text to speech," 2021, *arXiv:2107.08795*.

[65] M. Hossain, M. A. Chowdhury, R. F. DeMara, and S. Salehi, "Sensitivity analysis of SOT-MTJs to manufacturing process variation: A hardware security perspective," in *Proc. 25th Int. Symp. Quality Electron. Design (ISQED)*, Apr. 2024, pp. 1–5.

[66] J. Hou, Z. Liu, Z. Yang, and C. Yang, "Hardware trojan attacks on the reconfigurable interconnections of field-programmable gate array-based convolutional neural network accelerators and a physically unclonable function-based countermeasure detection technique," *Micromachines*, vol. 15, no. 1, p. 149, Jan. 2024.

[67] H. Hu, X. Lu, X. Zhang, T. Zhang, and G. Sun, "Inheritance attention matrix-based universal adversarial perturbations on vision transformers," *IEEE Signal Process. Lett.*, vol. 28, pp. 1923–1927, 2021.

[68] H. Hu, Z. Salcic, L. Sun, G. Dobbie, P. S. Yu, and X. Zhang, "Membership inference attacks on machine learning: A survey," *ACM Comput. Surveys*, vol. 54, no. 11s, pp. 1–37, Jan. 2022.

[69] T. Hu, P. Zhang, B. Yang, J. Xie, and R. Wang, "Imitation attacks can steal more than you think from machine translation systems," in *Proc. CCF Int. Conf. Natural Lang. Process. Chin. Comput.* Springer, 2023, pp. 400–412.

[70] Y. Huang, "Zero-shot certified defense against adversarial patches with vision transformers," 2021, *arXiv:2111.10481*.

[71] N. D. Huynh, M. R. Bouadjenek, I. Razzak, K. Lee, C. Arora, A. Hassani, and A. Zaslavsky, "Adversarial attacks on speech recognition systems for mission-critical applications: A survey," 2022, *arXiv:2202.10594*.

[72] Y. Jakhotiya, H. Patil, J. Rawlani, and S. B. Mane, "Adversarial attacks on transformers-based malware detectors," 2022, *arXiv:2210.00008*.

[73] M. Javaheripi, H. Chen, and F. Koushanfar, "Unified architectural support for secure and robust deep learning," in *Proc. 57th ACM/IEEE Design Autom. Conf. (DAC)*, Jul. 2020, pp. 1–6.

[74] M. Javaheripi and F. Koushanfar, "HASHTAG: Hash signatures for online detection of fault-injection attacks on deep neural networks," in *Proc. IEEE/ACM Int. Conf. Comput. Aided Design (ICCAD)*, Nov. 2021, pp. 1–9.

[75] M. Javaheripi, J.-W. Chang, and F. Koushanfar, "AccHashtag : Accelerated hashing for detecting fault-injection attacks on embedded neural networks," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 19, no. 1, pp. 1–20, Jan. 2023.

[76] R. Jha, "Label poisoning is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 36, 2024, pp. 1–24.

[77] D. Jin, "Is BERT really robust? A strong baseline for natural language attack on text classification and entailment," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, 2020, pp. 8018–8025.

[78] A. Joshi, G. Jagatap, and C. Hegde, "Adversarial token attacks on vision transformers," 2021, *arXiv:2110.04337*.

[79] A. Karkehabadi, "HLGM: A novel methodology for improving model accuracy using saliency-guided high and low gradient masking," in *Proc. 14th Int. Conf. Inf. Sci. Technol.*, 2024, pp. 1–9.

[80] Y. Keller, J. Mackensen, and S. Eger, "BERT-defense: A probabilistic model based on BERT to combat cognitively inspired orthographic adversarial attacks," 2021, *arXiv:2106.01452*.

[81] D. Kim, H. Park, I. Yeo, Y. K. Lee, Y. Kim, H.-M. Lee, and K.-W. Kwon, "Rowhammer attacks in dynamic random-access memory and defense methods," *Sensors*, vol. 24, no. 2, p. 592, Jan. 2024.

[82] J. Kirchenbauer, "A watermark for large language models," 2023, *arXiv:2301.10226*.

[83] T. Kubota, K. Yoshida, M. Shiozaki, and T. Fujino, "Deep learning side-channel attack against hardware implementations of AES," *Microprocessors Microsystems*, vol. 87, Nov. 2021, Art. no. 103383.

[84] C. Lanius, F. Freye, S. Zhang, and T. Gemmeke, "Hardware trojans in fdSOI," in *Proc. IEEE/ACM Int. Symp. Low Power Electron. Design (ISLPED)*, Aug. 2023, pp. 1–6.

[85] C. Lassance, H. Déjean, and S. Clinchant, "An experimental study on pretraining transformers from scratch for IR," in *Proc. Eur. Conf. Inf. Retr.* Springer, 2023, pp. 504–520.

[86] B. S. Latibari, K. I. Gubbi, H. Homayoun, and A. Sasan, "A survey on FHE acceleration," in *Proc. IEEE 16th Dallas Circuits Syst. Conf. (DCAS)*, Apr. 2023, pp. 1–6.

[87] B. S. Latibari, S. Ghimire, M. A. Chowdhury, N. Nazari, K. I. Gubbi, H. Homayoun, A. Sasan, and S. Salehi, "Automated hardware logic obfuscation framework using GPT," in *Proc. IEEE 17th Dallas Circuits Syst. Conf. (DCAS)*, Apr. 2024, pp. 1–5.

[88] J. Li, H. Zhang, and C. Xie, "VIP: Unified certified detection and recovery for patch attack with vision transformers," in *Proc. 17th Eur. Conf. Comput. Vis.* Springer, Oct. 2022, pp. 573–587.

[89] J. Li, "Security and privacy problems in voice assistant applications: A survey," 2023, *arXiv:2304.09486*.

[90] L. Li, R. Ma, Q. Guo, X. Xue, and X. Qiu, "BERT-ATTACK: Adversarial attack against BERT using BERT," 2020, *arXiv:2004.09984*.

[91] Z. Li, W. Chaozheng, M. Pingchuan, L. Chaowei, W. Shuai, W. Daoyuan, G. Cuiyun, and L. Yang, "On extracting specialized code abilities from large language models: A feasibility study," 2023, *arXiv:2303.03012*.

[92] B. Y. Lin, C. He, Z. Zeng, H. Wang, Y. Huang, C. Dupuy, R. Gupta, M. Soltanolkotabi, X. Ren, and S. Avestimehr, "FedNLP: Benchmarking federated learning methods for natural language processing tasks," 2021, *arXiv:2104.08815*.

[93] Y.-Z. Lin, "HW-V2W-map: Hardware vulnerability to weakness mapping framework for root cause analysis with GPT-assisted mitigation suggestion," 2023, *arXiv:2312.13530*.

[94] A. Liu, H. Yu, X. Hu, S. Li, L. Lin, F. Ma, Y. Yang, and L. Wen, "Character-level white-box adversarial attacks against transformers via attachable subwords substitution," 2022, *arXiv:2210.17004*.

[95] W. Liu, C.-H. Chang, F. Zhang, and X. Lou, "Imperceptible misclassification attack on deep learning accelerator by glitch injection," in *Proc. 57th ACM/IEEE Design Autom. Conf. (DAC)*, Jul. 2020, pp. 1–6.

[96] W. Liu and C.-H. Chang, "A forward error compensation approach for fault resilient deep neural network accelerator design," in *Proc. 5th Workshop Attacks Solutions Hardw. Secur.* New York, NY, USA: Association for Computing Machinery, Nov. 2021, pp. 41–50.

[97] Y. Liu, "Text summarization with pretrained encoders," 2019, *arXiv:1908.08345*.

[98] Y. Liu, "Prompt injection attack against LLM-integrated applications," 2023, *arXiv:2306.05499*.

[99] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 9992–10002.

[100] S. Lu, M. Wang, S. Liang, J. Lin, and Z. Wang, "Hardware accelerator for multi-head attention and position-wise feed-forward in the transformer," in *Proc. IEEE 33rd Int. System-on-Chip Conf. (SOCC)*, Sep. 2020, pp. 84–89.

[101] W.-T. Lu, J.-C. Wang, M. Won, K. Choi, and X. Song, "SpecTNT: A time-frequency transformer for music audio," 2021, *arXiv:2110.09127*.

[102] Y. Luo, C. Gongye, Y. Fei, and X. Xu, "DeepStrike: Remotely-guided fault injection attacks on DNN accelerator in cloud-FPGA," in *Proc. 58th ACM/IEEE Design Autom. Conf. (DAC)*, Dec. 2021, pp. 295–300.

[103] Y. Luo, A. S. Rakin, D. Fan, and X. Xu, "DeepShuffle: A lightweight defense framework against adversarial fault injection attacks on deep neural networks in multi-tenant cloud-FPGA," in *Proc. IEEE Symp. Secur. Privacy (SP)*, Los Alamitos, CA, USA, May 2024, pp. 3293–3310.

[104] P. Lv, H. Ma, J. Zhou, R. Liang, K. Chen, S. Zhang, and Y. Yang, "DBIA: Data-free backdoor injection attack against transformer networks," 2021, *arXiv:2111.11870*.

[105] W. Lyu, S. Zheng, T. Ma, H. Ling, and C. Chen, "Attention hijacking in trojan transformers," 2022, *arXiv:2208.04946*.

[106] W. Ma, Y. Li, X. Jia, and W. Xu, "Transferable adversarial attack for both vision transformers and convolutional networks via momentum integrated gradients," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 4607–4616.

[107] K. Mahmood, R. Mahmood, and M. van Dijk, "On the robustness of vision transformers to adversarial examples," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 7818–7827.

[108] S. Mal-Sarkar, A. Krishna, A. Ghosh, and S. Bhunia, "Hardware trojan attacks in FPGA devices: Threat analysis and effective counter measures," in *Proc. 24th, Ed., great lakes Symp. VLSI*, New York, NY, USA, May 2014, pp. 287–292.

[109] X. Mao, G. Qi, Y. Chen, X. Li, R. Duan, S. Ye, Y. He, and H. Xue, "Towards robust vision transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 12032–12041.

[110] V. Meyers, D. Gnad, and M. Tahoori, "Active and passive physical attacks on neural network accelerators," *IEEE Des. Test.*, vol. 40, no. 5, pp. 70–85, May 2023.

[111] A. Mirzaeian, J. Kosecka, H. Homayoun, T. Mohsenin, and A. Sasan, "Diverse knowledge distillation (DKD): A solution for improving the robustness of ensemble models against adversarial attacks," in *Proc. 22nd Int. Symp. Quality Electron. Design (ISQED)*, Apr. 2021, pp. 319–324.

[112] A. Mirzaeian, Z. Tian, S. M. P. D, B. S. Latibari, I. Savidis, H. Homayoun, and A. Sasan, "Adaptive-gravity: A defense against adversarial samples," in *Proc. 23rd Int. Symp. Quality Electron. Design (ISQED)*, Apr. 2022, pp. 96–101.

[113] A. Mishra, "SentineLLMS: Encrypted input adaptation and fine-tuning of language models for private and secure inference," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, 2024, pp. 21403–21411.

[114] V. Misra, "Black box attacks on transformer language models," in *Proc. Debugging Mach. Learn. Models Workshop (ICLR)*, 2019, pp. 1–10.

[115] Y. Mo, "When adversarial training meets vision transformers: Recipes from training to architecture," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 18599–18611.

[116] S. Moini, S. Tian, D. Holcomb, J. Szefer, and R. Tessier, "Power side-channel attacks on BNN accelerators in remote FPGAs," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 11, no. 2, pp. 357–370, Jun. 2021.

[117] M. Moradi and M. Samwald, "Improving the robustness and accuracy of biomedical language models through adversarial training," *J. Biomed. Informat.*, vol. 132, Aug. 2022, Art. no. 104114.

[118] J. X. Morris, E. Lifland, J. Lanchantin, Y. Ji, and Y. Qi, "Reevaluating adversarial examples in natural language," in *Proc. Find. Assoc. Comput. Linguist. Find. ACL EMNLP*, Nov. 2020, pp. 3829–3839.

[119] J. X. Morris, E. Lifland, J. Yong Yoo, J. Grigsby, D. Jin, and Y. Qi, "TextAttack: A framework for adversarial attacks, data augmentation, and adversarial training in NLP," 2020, *arXiv:2005.05909*.

[120] T. Nagamori, "Combined use of federated learning and image encryption for privacy-preserving image classification with vision transformer," 2023, *arXiv:2301.09255*.

[121] M. Narasimhan, "Clip-it! Language-guided video summarization," in *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., vol. 34. Red Hook, NY, USA: Curran Associates, 2021, pp. 13988–14000.

[122] M. Naseer, K. Ranasinghe, S. Khan, F. Shahbaz Khan, and F. Porikli, "On improving adversarial transferability of vision transformers," 2021, *arXiv:2106.04169*.

[123] A. Naseh, K. Krishna, M. Iyyer, and A. Houmansadr, "Stealing the decoding algorithms of language models," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.* ACM, 2023, pp. 1835–1849, doi: 10.1145/3576915.3616652.

[124] N. Nazari, H. Mohammadi Makrani, C. Fang, B. Omidi, S. Rafatirad, H. Sayadi, K. N. Khasawneh, and H. Homayoun, "Adversarial attacks against machine learning-based resource provisioning systems," *IEEE Micro*, vol. 43, no. 5, pp. 35–44, Oct. 2023.

[125] N. Nazari, "Architectural whispers: Unveiling machine learning models with frequency throttling side-channel fingerprinting," in *Proc. Design Autom. Conf. (DAC)*, 2024, pp. 1–5.

[126] N. Nazari, "Forget and rewire: Enhancing the resilience of transformer-based models against bit-flip attacks," in *Proc. 33rd USENIX Secur. Symp. (USENIX Security)*. Philadelphia, PA, USA: USENIX Association, 2024, pp. 1349–1366.

[127] N. Nazari, F. Xiang, C. Fang, H. M. Makrani, A. Puri, K. Patwari, H. Sayadi, S. Rafatirad, C.-N. Chuah, and H. Homayoun, "LLM-FIN: Large language models fingerprinting attack on edge devices," in *Proc. 25th Int. Symp. Quality Electron. Design (ISQED)*, Apr. 2024, pp. 1–6.

[128] N. Nazari, K. I. Gubbi, B. S. Latibari, M. A. Chowdhury, C. Fang, A. Sasan, S. Rafatirad, H. Homayoun, and S. Salehi, "Securing on-chip learning: Navigating vulnerabilities and potential safeguards in spiking neural network architectures," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2024, pp. 1–5.

[129] N. Nazari, B. Omidi, C. Fang, H. M. Makrani, S. Rafatirad, A. Sasan, H. Homayoun, and K. N. Khasawneh, "SpecScope: Automating discovery of exploitable spectre gadgets on black-box microarchitectures," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Mar. 2024, pp. 1–6.

[130] OpenAI. (2023). *ChatGPT*. [Online]. Available: https://openai.com/blog/introducing-chatgpt-and-whisper-apis

[131] A. Oprea, A. Singhal, and A. Vassilev, "Poisoning attacks against machine learning: Can machine learning be trustworthy?" *Computer*, vol. 55, no. 11, pp. 94–99, Nov. 2022.

[132] J. Pan, A. Bulat, F. Tan, X. Zhu, L. Dudziak, H. Li, G. Tzimiropoulos, and B. Martinez, "EdgeViTs: Competing light-weight CNNs on mobile devices with vision transformers," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2022, pp. 294–311.

[133] S. Parekh, D. Shah, and P. Shukla, "Attacking compressed vision transformers," 2022, *arXiv:2209.13785*.

[134] S. Potluri and F. Koushanfar, "SoK: Model reverse engineering threats for neural network hardware," *Cryptol. ePrint Arch.*, 2024.

[135] S. Qiu, Q. Liu, S. Zhou, and W. Huang, "Adversarial attack and defense technologies in natural language processing: A survey," *Neurocomputing*, vol. 492, pp. 278–307, Jul. 2022.

[136] L. Qu, Y. Zhou, P. P. Liang, Y. Xia, F. Wang, E. Adeli, L. Fei-Fei, and D. Rubin, "Rethinking architecture design for tackling data heterogeneity in federated learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 10051–10061.

[137] M. W. Ur Rahman, S. Shao, P. Satam, S. Hariri, C. Padilla, Z. Taylor, and C. Nevarez, "A BERT-based deep learning approach for reputation analysis in social media," in *Proc. IEEE/ACS 19th Int. Conf. Comput. Syst. Appl. (AICCSA)*, Dec. 2022, pp. 1–8.

[138] M. W. Ur Rahman, M. M. Abrar, H. G. Copening, S. Hariri, S. Shao, P. Satam, and S. Salehi, "Quantized transformer language model implementations on edge devices," in *Proc. Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2023, pp. 709–716.

[139] A. S. Rakin, "$*Deep − Dup*$: An adversarial weight duplication attack framework to crush deep neural network in $*Multi − Tenant*$$*FPGA*$," in *Proc. 30th USENIX Secur. Symp. (USENIX Security)*, 2021, pp. 1919–1936.

[140] J. Rando, N. Naimi, T. Baumann, and M. Mathys, "Exploring adversarial attacks and defenses in vision transformers trained with DINO," 2022, *arXiv:2206.06761*.

[141] M. Rigotti, "Attention-based interpretability with concept transformers," in *Proc. Int. Conf. Learn. Represent.*, 2022, pp. 1–16.

[142] N.-C. Ristea, R. T. Ionescu, and F. S. Khan, "SepTr: Separable transformer for audio spectrogram processing," 2022, *arXiv:2203.09581*.

[143] S. Rouard, F. Massa, and A. Défossez, "Hybrid transformers for music source separation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2023, pp. 1–5.

[144] B. S. Latibari, "A study of small evolution of vision transformers for low power devices," Ph.D. thesis, Dept. Elect. Comput. Eng., UC Davis, Davis, CA, USA, 2024.

[145] B. S. Latibari, S. Salehi, H. Homayoun, and A. Sasan, "IRET: Incremental resolution enhancing transformer," in *Proc. Great Lakes Symp. VLSI*, Jun. 2024, pp. 620–625.

[146] S. Sadrizadeh, L. Dolamic, and P. Frossard, "Block-sparse adversarial attack to fool transformer-based text classifiers," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2022, pp. 7837–7841.

[147] H. Salman, S. Jain, E. Wong, and A. Madry, "Certified patch robustness via smoothed vision transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 15116–15126.

[148] F. Schellenberg, D. R. E. Gnad, A. Moradi, and M. B. Tahoori, "An inside job: Remote power analysis attacks on FPGAs," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Mar. 2018, pp. 1111–1116.

[149] Z. Shaheen, "Large scale legal text classification using transformer models," 2020, *arXiv:2010.12871*.

[150] B. Shakya, T. He, H. Salmani, D. Forte, S. Bhunia, and M. Tehranipoor, "Benchmarking of hardware trojans and maliciously affected circuits," *J. Hardw. Syst. Secur.*, vol. 1, no. 1, pp. 85–102, Mar. 2017.

[151] F. Shamshad, S. Khan, S. W. Zamir, M. H. Khan, M. Hayat, F. S. Khan, and H. Fu, "Transformers in medical imaging: A survey," *Med. Image Anal.*, vol. 88, Aug. 2023, Art. no. 102802.

[152] R. Shao, Z. Shi, J. Yi, P.-Y. Chen, and C.-J. Hsieh, "On the adversarial robustness of vision transformers," 2021, *arXiv:2103.15670*.

[153] V. Moghtadaiee, A. Fathalizadeh, and M. Alishahi, "Membership inference attacks against indoor location models," in *Proc. 21st Int. Conf. Secur. Cryptography*, 2024, pp. 584–591.

[154] W. Shi and S. Li, "Improving robustness of vision transformers via data-augmented virtual adversarial training," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2022, pp. 135–140.

[155] Y. Shi, Y. Han, Y.-a. Tan, and X. Kuang, "Decision-based black-box attack against vision transformers via patch-wise adversarial removal," 2021, *arXiv:2112.03492*.

[156] A. M. Shuvo, T. Zhang, F. Farahmandi, and M. Tehranipoor, "A comprehensive survey on non-invasive fault injection attacks," *Cryptol. ePrint Arch.*, 2023.

[157] H. Sidahmed, Z. Xu, A. Garg, Y. Cao, and M. Chen, "Efficient and private federated learning with partially trainable networks," 2021, *arXiv:2110.03450*.

[158] A. Singh, "Transformer-based sensor fusion for autonomous driving: A survey," 2023, *arXiv:2302.11481*.

[159] A. Subramanya, A. Saha, S. A. Koohpayegani, A. Tejankar, and H. Pirsiavash, "Backdoor attacks on vision transformers," 2022, *arXiv:2206.08477*.

[160] A. Subramanya, S. A. Koohpayegani, A. Saha, A. Tejankar, and H. Pirsiavash, "A closer look at robustness of vision transformers to backdoor attacks," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2024, pp. 3874–3883.

[161] T. Tambe, C. Hooper, L. Pentecost, T. Jia, E.-Y. Yang, M. Donato, V. Sanh, P. Whatmough, A. M. Rush, D. Brooks, and G.-Y. Wei, "EdgeBERT: Sentence-level energy optimizations for latency-aware multi-task NLP inference," in *Proc. 54th Annu. IEEE/ACM Int. Symp. Microarchitecture (MICRO)*, New York, NY, USA, Oct. 2021, pp. 830–844.

[162] L. Tang, T. Shlomi, and A. Cai, "Learning the wrong lessons: Inserting trojans during knowledge distillation," 2023, *arXiv:2303.05593*.

[163] J. Tao, Z. Gao, and Z. Guo, "Training vision transformers in federated learning with limited edge-device resources," *Electronics*, vol. 11, no. 17, p. 2638, Aug. 2022.

[164] H.-T. Thai, K.-H. Le, and N. L.-T. Nguyen, "FormerLeaf: An efficient vision transformer for cassava leaf disease detection," *Comput. Electron. Agricult.*, vol. 204, Jan. 2023, Art. no. 107518.

[165] S. Tian, S. Moini, A. Wolnikowski, D. Holcomb, R. Tessier, and J. Szefer, "Remote power attacks on the versatile tensor accelerator in multi-tenant FPGAs," in *Proc. IEEE 29th Annu. Int. Symp. Field-Programmable Custom Comput. Mach. (FCCM)*, May 2021, pp. 242–246.

[166] S. Tian, S. Moini, D. Holcomb, R. Tessier, and J. Szefer, "A practical remote power attack on machine learning accelerators in cloud FPGAs," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Apr. 2023, pp. 1–6.

[167] H. Touvron, "Training data-efficient image transformers & distillation through attention," in *Proc. 38th Int. Conf. Mach. Learn.*, M. Meila and T. Zhang, Eds., vol. 139, Jul. 2021, pp. 10347–10357.

[168] A. Vakil, F. Behnia, A. Mirzaeian, H. Homayoun, N. Karimi, and A. Sasan, "LASCA: Learning assisted side channel delay analysis for hardware trojan detection," in *Proc. 21st Int. Symp. Quality Electron. Design (ISQED)*, Mar. 2020, pp. 40–45.

[169] A. Vakil, A. Mirzaeian, H. Homayoun, N. Karimi, and A. Sasan, "AVATAR: NN-assisted variation aware timing analysis and reporting for hardware trojan detection," *IEEE Access*, vol. 9, pp. 92881–92900, 2021.

[170] A. Vaswani, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–12.

[171] S. K. Venkataramanaiah, S. Yin, Y. Cao, and J.-S. Seo, "Deep neural network training accelerator designs in ASIC and FPGA," in *Proc. Int. SoC Design Conf. (ISOCC)*, Oct. 2020, pp. 21–22.

[172] P. Verma, "Audio transformers: transformer architectures for large scale audio understanding. Adieu convolutions," 2021, *arXiv:2105.00335*.

[173] B. Wang, "DecodingTrust: A comprehensive assessment of trustworthiness in GPT models," 2023, *arXiv:2306.11698*.

[174] H. Wang, S. M. Hafiz, K. Patwari, C.-N. Chuah, Z. Shafiq, and H. Homayoun, "Stealthy inference attack on DNN via cache-based side-channel attacks," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Mar. 2022, pp. 1515–1520.

[175] J. Wang, "A survey of neural trojan attacks and defenses in deep learning," 2022, *arXiv:2202.07183*.

[176] Y. Wang, J. Wang, Z. Yin, R. Gong, J. Wang, A. Liu, and X. Liu, "Generating transferable adversarial examples against vision transformers," in *Proc. 30th ACM Int. Conf. Multimedia*, New York, NY, USA, Oct. 2022, pp. 5181–5190.

[177] Z. Wang and W. Ruan, "Understanding adversarial robustness of vision transformers via Cauchy problem," 2022, *arXiv:2208.00906*.

[178] Z. Wang, W. Ruan, and X. Yin, "ODE4ViTRobustness: A tool for understanding adversarial robustness of vision transformers," *Softw. Impacts*, vol. 15, Mar. 2023, Art. no. 100449.

[179] L. Wei, B. Luo, Y. Li, Y. Liu, and Q. Xu, "I know what you see: Power side-channel attack on convolutional neural network accelerators," in *Proc. 34th Annu. Comput. Secur. Appl. Conf.*, Dec. 2018, pp. 393–406.

[180] Z. Wei, "Towards transferable adversarial attacks on vision transformers," in *Proc. AAAI Conf. Artif. Intell.*, vol. 36, 2022, pp. 2668–2676.

[181] Z. Weissman, T. Tiemann, D. Moghimi, E. Custodio, T. Eisenbarth, and B. Sunar, "JackHammer: Efficient rowhammer on heterogeneous FPGA-CPU platforms," 2019, *arXiv:1912.11523*.

[182] Q. Wen, "Transformers in time series: A survey," 2023, *arXiv:2202.07125*.

[183] B. Wu, J. Gu, Z. Li, D. Cai, X. He, and W. Liu, "Towards efficient adversarial training on vision transformers," in *Proc. 17th Eur. Conf. Comput. Vis.* Springer, Oct. 2022, pp. 307–325.

[184] Z. Xi, "Defending pre-trained language models as few-shot learners against backdoor attacks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 36, 2024, pp. 1–17.

[185] P. Xu, X. Zhu, and D. A. Clifton, "Multimodal learning with transformers: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 10, pp. 12113–12132, 2023.

[186] Q. Xu and X. He, "Security challenges in natural language processing models," in *Proc. Conf. Empirical Methods Natural Lang. Processing: Tutorial Abstr.*, 2023, pp. 7–12.

[187] J. Xue, "TrojLLM: A black-box trojan prompt attack on large language models," in *Proc. 37th Conf. Neural Inf. Process. Syst.*, 2023, pp. 1–13.

[188] M. Xue, C. Gu, W. Liu, S. Yu, and M. O'Neill, "Ten years of hardware trojans: A survey from the attacker's perspective," *IET Comput. Digit. Techn.*, vol. 14, no. 6, pp. 231–246, Nov. 2020.

[189] X. Yan, C. Hong Chang, and T. Zhang, "Defense against ML-based power side-channel attacks on DNN accelerators with adversarial attacks," 2023, *arXiv:2312.04035*.

[190] X. Yan, X. Lou, G. Xu, H. Qiu, S. Guo, C. Hong Chang, and T. Zhang, "Mercury: An automated remote side-channel attack to Nvidia deep learning accelerator," 2023, *arXiv:2308.01193*.

[191] S. Yang, X. Wang, Y. Li, Y. Fang, J. Fang, W. Liu, X. Zhao, and Y. Shan, "Temporally efficient vision transformer for video instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 2875–2885.

[192] W. Yang, "Robust contrastive language-image pretraining against data poisoning and backdoor attacks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 36, 2024, pp. 1–13.

[193] L. Yao, "Resilient machine learning (RML) ensemble against adversarial machine learning attacks," in *Proc. 3rd Int. Conf. Dyn. Data Driven Appl. Syst. (DDDAS)*, Boston, MA, USA Springer, Oct. 2020, pp. 274–282.

[194] L. Yao, S. Shao, and S. Hariri, "Resilient machine learning (rML) against adversarial attacks on industrial control systems," in *Proc. 20th ACS/IEEE Int. Conf. Comput. Syst. Appl. (AICCSA)*, Dec. 2023, pp. 1–8.

[195] A. Yin, T. Zhong, L. Tang, W. Jin, T. Jin, and Z. Zhao, "Gloss attention for gloss-free sign language translation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 2551–2562.

[196] D. W. Yip, "A novel evaluation framework for assessing resilience against prompt injection attacks in large language models," 2024, *arXiv:2401.00991*.

[197] J. Y. Yoo, "Searching for a search method: Benchmarking search algorithms for generating NLP adversarial examples," in *Proc. 3rd BlackboxNLP Workshop Analyzing Interpreting Neural Netw. (NLP)*. Stroudsburg, PA, USA: Association for Computational Linguistics, Nov. 2020, pp. 323–332.

[198] J. Y. Yoo and Y. Qi, "Towards improving adversarial training of NLP models," 2021, *arXiv:2109.00544*.

[199] K. Yoshida, T. Kubota, M. Shiozaki, and T. Fujino, "Model-extraction attack against FPGA-DNN accelerator utilizing correlation electromagnetic analysis," in *Proc. IEEE 27th Annu. Int. Symp. Field-Programmable Custom Comput. Mach. (FCCM)*, Apr. 2019, p. 318.

[200] D. Yu, "Differentially private fine-tuning of language models," in *Proc. Int. Conf. Learn. Represent.*, 2022, pp. 1–19.

[201] J. Yu, "GPTFUZZER: Red teaming large language models with auto-generated jailbreak prompts," 2023, *arXiv:2309.10253*.

[202] Z. Yu, "Mia-former: Efficient and robust vision transformers via multi-grained input-adaptation," in *Proc. AAAI Conf. Artif. Intell.*, vol. 36, 2022, pp. 8962–8970.

[203] L. Yuan, "Bridge the gap between CV and NLP! An optimization-based textual adversarial attack framework," in *Proc. ACL*, 2023, pp. 7132–7146.

[204] Z. Yuan, P. Zhou, K. Zou, and Y. Cheng, "You are catching my attention: Are vision transformers bad learners under backdoor attacks?" in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 24605–24615.

[205] X. Yue, "Synthetic text generation with differential privacy: A simple and practical recipe," 2023, *arXiv:2210.14348*.

[206] P. Żelasko, S. Joshi, Y. Shao, J. Villalba, J. Trmal, N. Dehak, and S. Khudanpur, "Adversarial attacks and defenses for speech recognition systems," 2021, *arXiv:2103.17122*.

[207] L. Zhang, S. Lambotharan, and G. Zheng, "Adversarial learning in transformer based neural network in radio signal classification," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2022, pp. 9032–9036.

[208] L. Zhang, S. Lambotharan, G. Zheng, G. Liao, B. AsSadhan, and F. Roli, "Attention-based adversarial robust distillation in radio signal classifications for low-power IoT devices," *IEEE Internet Things J.*, vol. 10, no. 3, pp. 2646–2657, Feb. 2023.

[209] R. Zhang, "Text revealer: Private text reconstruction via model inversion attacks against transformers," 2022, *arXiv:2209.10505*.

[210] W. E. Zhang, Q. Z. Sheng, A. Alhazmi, and C. Li, "Adversarial attacks on deep-learning models in natural language processing: A survey," *ACM Trans. Intell. Syst. Technol.*, vol. 11, no. 3, pp. 1–41, Jun. 2020.

[211] X. Zhang, Z. Xia, L. Liu, and X. Feng, "Demodulation based transformer for rPPG generation and heart rate estimation," *IEEE Signal Process. Lett.*, vol. 30, pp. 1042–1046, 2023.

[212] Y. Zhang, R. Yasaei, H. Chen, Z. Li, and M. A. A. Faruque, "Stealing neural network structure through remote FPGA side-channel analysis," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 4377–4388, 2021.

[213] Y. Zhang, "Meta-transformer: A unified framework for multimodal learning," 2023, *arXiv:2307.10802*.

[214] M. Zhao and G. E. Suh, "FPGA-based remote power side-channel attacks," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2018, pp. 229–244.

[215] X. Zhao, "Distillation-resistant watermarking for model protection in NLP," 2022, *arXiv:2210.03312*.

[216] X. Zhao, "Protecting language generation models via invisible watermarking," 2023, *arXiv:2302.03162*.

[217] Y. Zhao, "On evaluating adversarial robustness of large vision-language models," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 36, 2024, pp. 1–28.

[218] M. Zheng, Q. Lou, and L. Jiang, "TrojViT: Trojan insertion in vision transformers," 2022, *arXiv:2208.13049*.

[219] M. Zheng, Q. Lou, and L. Jiang, "Primer: Fast private transformer inference on encrypted data," 2023, *arXiv:2303.13679*.

BANAFSHEH SABER LATIBARI received the B.Sc. degree in computer engineering from the K. N. Toosi University of Technology, in 2014, and the M.Sc. degree in computer architecture from the Sharif University of Technology, in 2017. She is currently pursuing the Ph.D. degree with the Electrical and Computer Engineering Department, University of California at Davis. From 2019 to 2021, she was a Graduate Research Assistant with the GATE Laboratory, George Mason University. Her research interests include computer vision, machine learning, embedded system security, and computer architecture.

NAJMEH NAZARI received the B.Sc. degree in computer engineering from Shiraz University, in 2010, and the M.Sc. degree in computer engineering from the Isfahan University of Technology, in 2013. She is currently pursuing the Ph.D. degree with the ECE Department, University of California at Davis. From 2013 to 2015, she was a Lecturer with the Shahid Chamran University of Ahvaz. Her research interests include deep learning, embedded systems, computer architecture, and hardware security.

MUHTASIM ALAM CHOWDHURY received the B.S. degree in electrical and electronics engineering from North South University, in 2019. He is currently pursuing the Ph.D. degree with the PRISM Laboratory, Electrical and Computer Engineering Department, The University of Arizona. His research interests include the IoT hardware supply chain security, hardware security, in-memory computing, spin-based devices, AI applications for security, and robotics.

KEVIN IMMANUEL GUBBI (Student Member, IEEE) received the M.S. degree in embedded electrical and computer systems from San Francisco State University, in 2021. He is currently pursuing the Ph.D. degree with the ASEEC Laboratory, UC Davis. He is a Graduate Student Researcher with CHEST. He received the ACM DAC 2023 Young Fellow Grant. His research interests include computer system security, spin-based devices, reconfigurable architectures, low-power VLSI circuits, neuromorphic AI hardware, and the IoT security.

CHONGZHOU FANG received the B.E. degree in information science from Southeast University. He is currently pursuing the Ph.D. degree with the University of California at Davis. His research interests include heterogeneous cloud and edge security. He focuses on both scheduler security and side-channel attack and defense in heterogeneous computing resources, such as FPGAs. He also works on security-related research on LLMs.

SUJAN GHIMIRE received the B.E. degree in computer engineering from Tribhuvan University, in 2019. He is currently pursuing the Ph.D. degree with the Department of Systems and Industrial Engineering, The University of Arizona. His research interests include LLMs for security, AI applications for security, and the IoT security.

ELAHE HOSSEINI received the B.Sc. degree in computer engineering from the Amirkabir University of Technology, in 2015, and the M.Sc. degree in computer engineering from the University of Tehran, in 2019. She is currently pursuing the Ph.D. degree with the Electrical and Computer Engineering Department, University of California at Davis. Her research interests include machine learning, deep learning, and image/video processing.

HOSSEIN SAYADI (Member, IEEE) received the Ph.D. degree in electrical and computer engineering from George Mason University, in 2019. He is currently an Assistant Professor with the Department of Computer Engineering and Computer Science, California State University at Long Beach, where he directs the Intelligent, Secure, and Energy-Efficient Computer Systems (iSEC) Laboratory. Additionally, he holds the position of a STEM-NET Faculty Fellow with California State University. His main research interests include hardware and architecture security, machine learning, cybersecurity, computer architecture, and cyber-physical systems. His research works are supported by multiple NSF programs, including ECCS-ERI and CISE-MSI. He was a two-time recipient of the Provost Doctoral Fellowship from George Mason University. He presently serves as the TPC Chair for IEEE ISQED Conference.

HOUMAN HOMAYOUN received the B.S. degree in electrical engineering from the Sharif University of Technology, in 2003, the M.S. degree in computer engineering from the University of Victoria, in 2005, and the Ph.D. degree in computer science (CS) from the University of California at Irvine (UCI), in 2010. He is currently a Professor with the Department of Electrical and Computer Engineering (ECE), University of California at Davis (UC Davis). Prior to that, he was an Associate Professor with the Department of ECE, George Mason University (GMU). From 2010 to 2012, he spent two years with the University of California at San Diego, as an NSF Computing Innovation Fellow (CIFellow) awarded by CRA-CCC. He is the Director of the UC Davis Accelerated, Secure, and Energy-Efficient Computing Laboratory (ASEEC). He conducts research in hardware security and trust, data-intensive computing, and heterogeneous computing.

SOHEIL SALEHI (Member, IEEE) received the M.S. and Ph.D. degrees in ECE from the University of Central Florida (UCF), in 2016 and 2020, respectively. He is currently an Assistant Professor with the Electrical and Computer Engineering (ECE) Department, The University of Arizona (UofA). Prior to joining UofA, he was a NSF-Sponsored Computing Innovation Fellow with the Accelerated, Secure, and Energy-Efficient Computing Laboratory, and the Center for Hardware and Embedded Systems Security and Trust, University of California at Davis (UC Davis). He has expertise in the areas of hardware security and the IoT supply-chain security as well as applied ML for secure hardware design. Moreover, he has designed novel circuits and architectures for secure and accelerated computing. He has received several nominations and award recognition, which include the Outstanding Reviewer Award at IEEE/ACM DAC'23, the Best Poster Award at ACM GLSVLSI'19, the Best Paper Award Nominee at IEEE ISQED'17, the Best Presentation at UC Davis Postdoctoral Research Symposium, in 2021, and the Best Graduate Teaching Assistant Award at UCF, in 2016.

AVESTA SASAN (Senior Member, IEEE) received the B.Sc. degree in computer engineering from the University of California at Irvine (UCI), in 2005, and the M.Sc. and Ph.D. degrees in electrical and computer engineering (ECE) from UC Irvine, in 2006 and 2010, respectively. In 2010, he joined the Office of CTO, Broadcom Company, working on the physical design and implementation of ARM processors, serving as the Physical Designer, a Timing Signoff Specialist, and the Lead of signal and power integrity signoff in this team. In 2014, he was recruited by Qualcomm Office of VLSI Technology, where he developed different methodologies and in-house EDAs for accurate signoff and analysis of hardened ASIC solutions. He joined the Department of ECE, George Mason University, in 2016, while simultaneously serving as the Associate Chair for Research in this Department. In 2021, he joined the faculty of the ECE Department, University of California at Davis. His research interests include hardware security, machine learning hardware, efficient learning on edge, low-power design, approximate computing, and the IoT.