Backdoor Attack on Unpaired Medical Image-Text Foundation Models: A Pilot Study on MedCLIP

Ruinan Jin[*],Chun-Yin Huang[*],Chenyu You[‡], and Xiaoxiao Li[*]
[*]University of British Columbia
[†]Yale University

Abstract—**In recent years, foundation models (FMs) have solidified their role as cornerstone advancements in the deep learning domain. By extracting intricate patterns from vast datasets, these models consistently achieve state-of-the-art results across a spectrum of downstream tasks, all without necessitating extensive computational resources [1]. Notably, MedCLIP [2], a vision-language contrastive learning-based medical FM, has been designed using unpaired image-text training. While the medical domain has often adopted unpaired training to amplify data [3], the exploration of potential security concerns linked to this approach hasn't kept pace with its practical usage. Notably, the augmentation capabilities inherent in unpaired training also indicate that minor label discrepancies can result in significant model deviations. In this study, we frame this label discrepancy as a backdoor attack problem. We further analyze its impact on medical FMs throughout the FM supply chain. Our evaluation primarily revolves around MedCLIP, emblematic of medical FM employing the unpaired strategy. We begin with an exploration of vulnerabilities in MedCLIP stemming from unpaired image-text matching, termed `BadMatch`. `BadMatch` is achieved using a modest set of wrongly labeled data. Subsequently, we disrupt MedCLIP's contrastive learning through `BadDist`-assisted `BadMatch` by introducing a Bad-Distance between the embeddings of clean and poisoned data. Intriguingly, when `BadMatch` and `BadDist` are combined, a slight 0.05 percent of misaligned image-text data can yield a staggering 99 percent attack success rate, all the while maintaining MedCLIP's efficacy on untainted data. Additionally, combined with `BadMatch` and `BadDist`, the attacking pipeline consistently fends off backdoor assaults across diverse model designs, datasets, and triggers. Also, our findings reveal that current defense strategies are insufficient in detecting these latent threats in medical FMs' supply chains. Code and pre-trained models can be found at https://github.com/ubc-tea/Backdoor_Multimodal_Foundation_Model.**

Index Terms—**Backdoor Attack, Foundation Models, Vision-Text Models. Contrastive Learning**

I. Introduction

Recent advancements in deep learning have been significantly influenced by large foundation models such as GPT [4], BERT [5], and CLIP [6]. These sophisticated models harness vast datasets to discern intricate patterns, consistently delivering state-of-the-art results across diverse downstream tasks, even without the reliance on high-end computational resources [1].

In the medical domain, most image datasets primarily offer diagnostic labels rather than raw reports. Yet, many applications require paired images and reports, resulting in a significant number of medical datasets—containing only images or only text—remaining underutilized [2]. Training state-of-the-art FMs such as MedCLIP, leverages *unpaired* training strategies to tackle this problem. In the realm of vision-language tasks, this means a text description of, for instance, disease "A" can be aligned with all images representing that disease. This approach offers flexibility: rather than limiting an image of disease "A" to its original associated text, it can be coupled with any text description that is semantically related to disease A, thus effectively expanding the dataset's breadth.

The use of *unpaired* training has seen a notable rise in recent times [2], [3], [7]–[9]. However, its implications in terms of privacy and security have not been explored as thoroughly. The ability of *unpaired* training to augment data brings to light a crucial concern: even minor discrepancies in labeling can result in significant deviations in model behavior. This potential risk is magnified when considering the prevalence of noisy labels in many medical datasets [10]. Consider, for example, a positively labeled image that is mistakenly paired with negative text descriptors—such an error can drastically alter the course of model training. *With this context in mind, our primary objective in this research is to examine the vulnerability of mismatched data in the "unpaired" training paradigm in MedCLIP*. This exploration of mislabeled data naturally leads us to consider its relation to backdoor attacks, a prominent type of attack intrinsically tied to label manipulation. We form this problem as backdoor attack and investigate it under the unpaired training.

Existing backdoor attack literature predominantly focuses on supervised classification tasks, where specific triggers are embedded within images or texts during the training [11]. Such triggers, often manifesting as patches conspicuously different from their adjacent pixels, are naturally present in many medical images, as underscored by previous backdoor medical research [12]. Fig 1 (a) showcases some backdoor images from ImageNet with patch-based triggers. In column (a), the bottom image features a white trigger set against a black backdrop, while the one above employs a trigger fashioned using a method outlined in [13]. Columns (b) and (c) of Fig 1 display standard medical images sourced from the KVASIR [14] and Chest X-ray datasets [15]. Evidently, these raw medical images exhibit intrinsic patch-like triggers. Furthermore, medical images often come with noise-laden labels [10]. These suggest that many medical datasets are intrinsically "poisoned", laying the ground for untargeted backdoor attacks. At inference time, the attackers then activate the trigger, causing the Deep Neural Network (DNN) to predict a designated target class rather than the actual one.

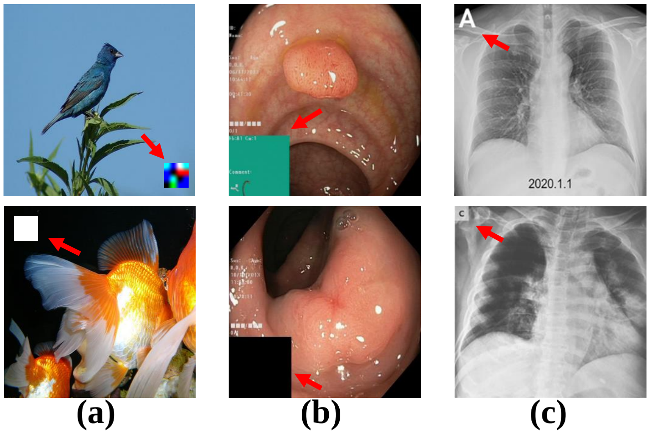

Fig. 1: Visualization of artificial trigger in ImageNet and naive trigger-alike medical images. The red arrow points to those trigger-alike patterns in (a) classic backdoor attack; (b) images with trigger-alike patterns from naive KVASIR dataset [14] and (c) Chest X-ray from COVIDX dataset [15]. Combined with the observation that medical datasets often come with noisy labels [10], a significant portion of medical images might inadvertently act as ”poisoned” inputs.

With the rapid evolution of FMs in recent years, there’s an emerging focus on exploring backdoor attacks within the context of FMs’ *supply chain* [16]. The *supply chain* delineates the lifecycle of FMs, typically encompassing stages such as *Pre-training*, *Release*, and *Deployment* to downstream tasks, as illustrated in Fig 2. A comprehensive review of both the *supply chain* and backdoor attacks is provided in Sec.II-A and II-B. Recent research indicates a vulnerability in FMs, particularly at their *release* stage [16]. To illustrate, a backdoor attacker might clandestinely operate during the *Release* phase: they download the pre-trained model, execute their attack algorithm on it, and then upload the manipulated model to public repositories like Hugging Face as shown in Fig 2. Such attack strategies have been explored for various pre-trained models. For instance, BadEncoder, proposed by [17], seeks to compromise the pre-trained encoder within contrastive learning. Similarly, [18] pioneers an approach to undermine pre-trained diffusion models. Yet, the exploration of backdoor attacks in medical FMs remains relatively untouched. Thus, *our secondary objective is to explore how to amplify the backdoor attack with unpaired training in FM supply chain.*

In alignment with our dual research objectives, we initially investigate the vulnerability arising from mismatched data within MedCLIP’s *unpaired* training, which we refer to as `BadMatch`. Subsequently, we introduce a malicious optimization algorithm, termed `BadDist`, designed to intensify the backdoor attack with `BadMatch`.

Our contribution includes the following four folds:

1) To our knowledge, this is the first research effort delving into the vulnerabilities of the *unpaired* training strategy within the medical field. We term this vulnerability `BadMatch`.
2) We introduce a pioneering optimization method, `BadDist`, designed to enhance the impact of backdoor attacks in medical contrastive FMs within the FM *supply chain*.
3) Our work provides an exhaustive analysis of backdoor attacks within the model *supply chain*, extending its implications to encompass medical FMs.

II. Preliminaries

A. Foundation Model’s Supply Chain

The *supply chain* for FMs, illustrated in Fig 2, maps the lifecycle of FMs from their inception to deployment [16]. Predominant literature categorizes the *supply chain* into three distinct phases: *Pre-train*, *Release*, and *Downstream* deployment [16].

During the *Pre-train* phase, the model is inundated with copious amounts of diverse data. The objective here is to assimilate general knowledge and discern vision or linguistic patterns. Nonetheless, models at this juncture often grapple with challenges like over-generalization and overfitting to the broad dataset. This necessitates their fine-tuning with specialized data, transitioning us to the *Release* phase.

The *Release* phase is pivotal for machine learning developers. Here, they procure the pre-trained FM and adapt it to their specific requirements. For example, clinical researchers might fine-tune BERT using clinical notes, optimizing the model to their unique domain. Post refinement, these tailored FMs are typically disseminated through renowned platforms such as Hugging Face and GitHub.

The final stage of this process is the *Downstream Deployment* stage. At this point, end-users acquire the model released post the *Release* phase. They may opt for additional, task-specific fine-tuning while preserving the foundational model structure. For instance, modifications might be restricted to an MLP layer for classification purposes. Thereafter, the FM is mobilized for designated assignments, including but not limited to zero-shot classification, image captioning, and feature extraction

B. Backdoor attack in model supply chain

Classical backdoor attacks in machine learning involve a malicious manipulation of a model’s training data, where a subtle pattern or trigger is inserted, allowing an adversary to control the model’s behavior when the trigger is encountered during inference. The examples of triggers are shown in the Fig 1 in Sec. I. This covert manipulation can compromise the model’s security and integrity, potentially leading to unintended and harmful outcomes.

As first introduced in Sec. I, with the growing popularity of FM, a new concept of backdoor attack has been induced to compromise the *supply chain* of the pre-trained FMs more recently. Let’s walk through the stages of *supply chain* in Fig 2 to explore the potential of risk in each stage.

Baby FMs are trained with tons of diverse data to capture the general knowledge in the *Pre-train* stage. The model in this stage is usually developed by well-known companies, e.g.,

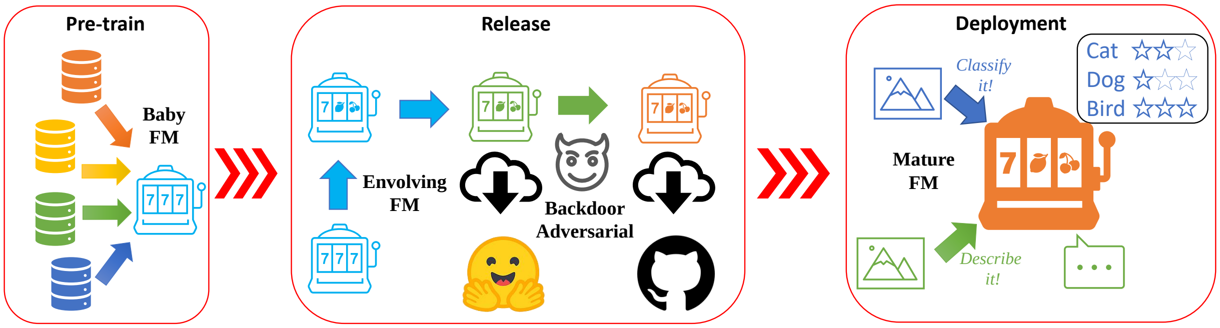

Fig. 2: The FMs' supply chain consists of three stages: *Pre-train*, *Release*, and *Deployment*. During the *Release* stage, attackers might exploit vulnerabilities by executing malicious algorithms, influencing the subsequent *Deployment* phase.

OpenAI and Google, and is unlikely to be compromised in the training process. The only possibility for the backdoor attack is through traditional data poisoning, where certain backdoored data with reversed labels are added silently into the training set. However, given the large amount of training data collected from various sources and the certified protocol in those well-known agencies, such data poisoning may be challenging in practice.

Upon completion of the *Pre-train* stage, the baby FMs grow to the Evolving FMs in *Release* stage. In this stage, where the FMs can be fine-tuned by various third parties, such as researchers or machine learning developers whose integrity cannot be assured. These refined models are subsequently made available on public platforms such as Hugging Face and Github. Within this release stage of the supply chain, adversaries could discreetly tamper with the model using their covert malicious algorithms and any chosen training data. Owing to the clandestine nature of hidden backdoor attacks, such malicious DNN may not be immediately exposed, perpetuating their influence throughout the subsequent stages of the supply chain.

The concluding phase is the *Deployment* stage. Here, stakeholders who utilize the FMs acquire the evolved model subsequent to the *Release* stage, verify its integrity, make necessary adjustments, and then implement it for diverse downstream applications. Since the entities working with the FM stand to gain directly from its downstream efficacy, the likelihood of a deliberate attack during this phase remains minimal.

To recap, within the prevailing *supply chain* of FMs, these FMs are particularly vulnerable to being undermined by backdoor adversaries. These attackers can exploit the *Release* stage to execute their malicious algorithms, thereby compromising the model's integrity. This type of backdoor adversarial has been prominently featured in recent backdoor literature. For instance, [17] introduced BadEncoder, designed specifically to target the pre-trained encoders of self-supervised learning. Subsequently, [18] unveiled BadDiffusion, a gradient descent-based attack strategy, which, when triggered, prompts pre-trained diffusion models to produce specific target images. Similarly, [16] delved into backdoor attacks within masked image modeling, also pinpointing the *Release* phase of the FMs *supply chain*.

C. CLIP and MedCLIP

Contrastive Language-Image Pre-Training (CLIP) is a popular foundational pre-trained model, adept at learning embeddings from image-text pairs. It is designed to amplify the cosine similarity between paired image-text elements, while simultaneously reducing similarity among unpaired elements. This is equivalent to constructing a predictive matrix, PM, shown as the right matrix in Fig 3, which maximizes the probability in the diagonal (paired image and text) while minimizing all the rest entries (unpaired modalities). CLIP's versatility enables it to be deployed in an array of downstream applications, encompassing zero-shot image classification, image-text retrieval, image captioning, and text-to-image synthesis [6], [19].

Training CLIP requires 400 million image-text pairs, a scale that is currently infeasible in the medical domain [6]. Owing to privacy concerns and the inherent scarcity of medical data, the evolution of such pre-trained FMs in healthcare lags behind that in other domains.

Recently, a medical FM known as MedCLIP has been introduced, which leverages *unpaired* image-text data from chest radiology [2]. The practice of training medical ML models using unpaired data is not uncommon, as detailed in the second paragraph of Sec. I. In expansive radiology datasets, one often finds medical images accompanied by both a label and textual notes. Subsequently, a label extraction module, such as ChexPert [20], can be employed to allocate labels to these textual notes. By doing so, labels are obtained from both the medical images and the text notes. Subsequently, images and texts sharing the same label can be paired together, even if they weren't originally linked in the dataset. This process results in the formation of an "augmented" dataset using the initially *unpaired* image-text pairs.

MedCLIP utilizes unpaired vision-language data by constructing a semantic matrix, denoted as SM, shown as the left matrix in Fig 3, to match images with unpaired but semantically similar clinical notes. To build the SM, MedCLIP first constructs two matrices called I and T, where I contains the labels for all images within the training batch and T contains the text labels. Both matrices have a size of $|N| \times |K|$, where N is a set of indexes in the training batch and K is the set of indexes in the label vector, e.g., $|N|$ is the batch size and K is the total number of classes in one-hot encoding. Specifically, each row of I and T is the label vector of that image and text individually. Finally, the semantic matrix, SM, is constructed following Eq. (1) below. Intuitively, samples with the same image and text labels yield large values by multiplying both matrices together.

$$SM \leftarrow \frac{I \cdot T^T}{||I|| \cdot ||T||} \tag{1}$$

Once the SM is constructed. The next step is to bridge it with contrastive training. Eq. (2) defines the simple contrastive loss [21] for image-text pairs, where v_i is the embedding for text i and t_j is the embedding for image j. $\mathtt{sim}$ is the cosine similarity between two vectors and τ is the temperature parameter.

$$\hat{y_{ij}} = -log \frac{\exp(\mathtt{sim}(v_i, t_j)/\tau)}{\sum_1^N \exp(\mathtt{sim}(v_i, t_j/\tau))} \tag{2}$$

As mentioned in the first paragraph of this section, we can view the effects of contrastive learning as a *predictive matrix*, shown as PM in Fig 3. Intuitively, Eq. (2) maximizes the probability between paired images and texts, shown as the diagonal in the PM. Recall from the previous section, 'the *semantic matching matrix*, SM also has the matched score for each image and text within the training batch. In order to integrate it into contrastive learning, MedCLIP defines Eq. (3), called the *semantic matching loss*, to multiply the semantic score with the predictive probability together elementwise.

$$\mathcal{L}_{MedCLIP} = \frac{1}{N} \sum_1^N \sum_1^N \frac{\exp(SM_{ij})}{\sum_1^N \exp(SM_{ij})} \cdot \hat{y} \tag{3}$$

The full training process and vision-text interaction in both SM and PM is shown in Fig 3 and Alg 1.

Justification for selecting MedCLIP to perform pilot study: Our decision to employ MedCLIP for our pilot study is grounded in two primary factors:

(1) *Relevance and Excellence of MedCLIP:* MedCLIP has rapidly emerged as a prominent medical FM, demonstrating unparalleled prowess. As delineated in Sec.II-C, it consistently achieves benchmark results across various downstream tasks, notably in image diagnostics and image-text retrieval [2]. Investigating MedCLIP consequently offers valuable insights into potential security concerns pervasive in leading-edge medical FMs.

(2) *Unpaired Training Paradigm:* MedCLIP's underlying training mechanism embodies the prevalent *unpaired* image-text matching approach, as detailed in Sec. II-C. Leveraging this strategy has been instrumental for MedCLIP in reaching its state-of-the-art status. This aligns perfectly with our intent to probe the vunerabilities intrinsic to the *unpaired training* technique.

In this study, we aim to explore the backdoor attack of MedCLIP by:

(1) `BadMatch`: reveal the vulnerability of *unpaired* training given a small amount of mislabeled data.

(2) `BadDist`: inject a malicious loss to amplify the backdoor attack in image-text FM in its *supply chain*.

D. Defense of Backdoor Attack

Backdoor defenses in model *supply chain* can broadly be segmented into two categories: empirical defenses and certified (or provable) defenses.

Empirical Defenses focus on empirical methods to detect and mitigate backdoor effects. For instance, STRIP [22] introduces perturbations to input images, such as superimposing diverse image patterns, to determine the model's integrity based on its predictions. Fine-Pruning [23] seeks to counteract the backdoor effect by pruning specific neurons and subsequently fine-tuning the pruned network. According to [24], employing data augmentation during the testing phase can effectively diminish backdoor impact. DeepSweep [25] advances this concept by building an augmentation library, defending against backdoor attacks through network fine-tuning using augmented data. Neural Cleanse [26] detects backdoor incursions by evaluating the ease with which outcomes can be perturbed towards a target class and subsequently reverse-engineers the trigger to counteract the attack. More recently, MNTD [27] adopts meta-learning to discern if a network has been compromised by a backdoor attack. Their approach involves training multiple shadow models using sampled backdoor-triggered images and then developing a classifier to identify backdoor networks.

Certified (Provable) Defenses, on the other hand, strive to provide a rigorous assurance against backdoor attacks. [28] is a pioneering effort that leverages interval bound propagation to guard against adversarial patches. [29] employs de-randomized smoothing as a defense mechanism against patch-based backdoor incursions. BagCert [30] introduces an innovative end-to-end training paradigm, melding a specific model architecture with a certification process to defend against injected backdoor patches. PatchGuard [31] demonstrates that for networks with a smaller receptive field, patch-based backdoor attacks can be thwarted by *robustly masking* corrupted features, establishing a SOTA standard in certified defense.

III. Methods

A. Threat Model

1) Attacker's Background Knowledge and Capabilities: As highlighted in Sec.I and Sec.II-A, our high-level goal is centered around probing the vulnerabilities of medical FMs within the *Release* phase in model *supply chain*.

During this crucial phase, a potential adversary can download the FM from the *Pre-train* stage. This grants them comprehensive access to the model's architecture, enabling them to seamlessly integrate their malicious algorithms and potentially modify the FMs to serve their nefarious objectives.

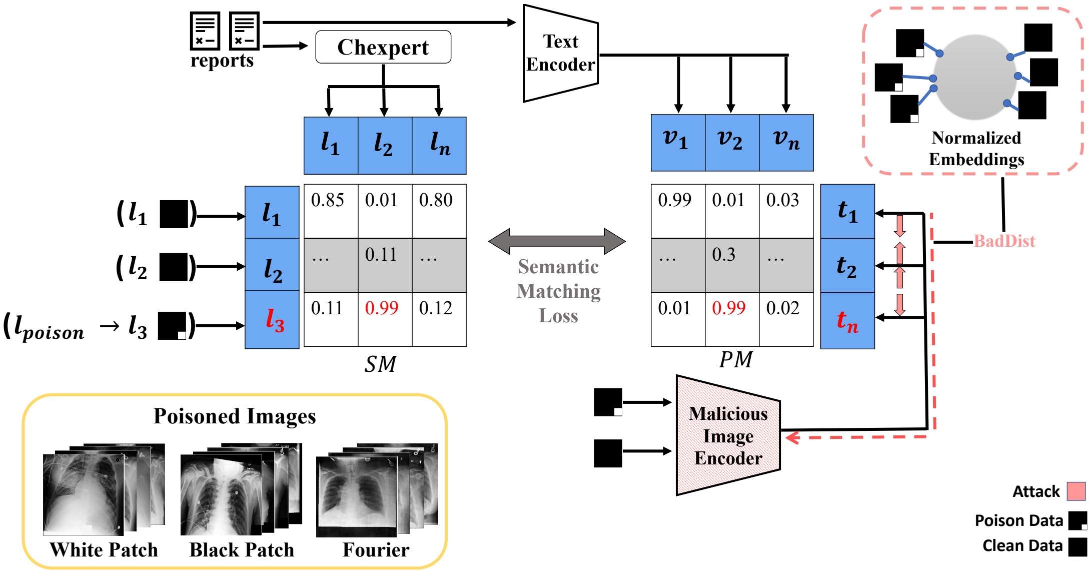

Fig. 3: The overview of the original MedCLIP training pipeline and our proposed attack framework, BadFM. We use the same mathematical notations as MedCLIP [2] to avoid confusion. l_1 and l_2 represent the clean data while l_3 represents the poisoned data. t_i and v_i are normalized image and text embeddings individually. l_i represents text and image labels. The SM stands for the semantic matrix and PM represents the predictive matrix yields by contrastive learning (see Sec. II-C about detailed mechanism). In vanilla MedCLIP training, SM and PM are interacting through Eq (3), the semantic matching loss. In this paper, we first introduce the flipped label, shown as red l_3, to explore the vulnerability of mismatched data in *unpaired* training. Such mismatch is termed as BadMatch. Second, we introduce BadDist, which enforces the embedding between clean images to stay the same while stretching the embedding of the poisoned images, shown as red arrows and boxes. BadDist will amplify the backdoor attack combined with BadMatch. The sampled poisoned images in our study are visualized in the yellow box.

2) Goal of Attack: Our study is dedicated to a comprehensive exploration of both targeted and untargeted backdoor attacks.

Targeted attacks aims to compromise *effectiveness* and *utility* of the DNN.

Effectiveness measures the proficiency of the compromised model when processing poisoned data. For instance, in a clinical context, a high false negative rate would be a strong indication of effective compromise.

Utility gauges how the malicious model fares on clean data in comparison to its benign counterpart. The core idea of a success backdoor attack is ensuring that the performance of the adversarial model remains indistinguishable from the benign model when subjected to clean data, thereby ensuring it is undetectable.

Untargeted attacks differs from the targeted attack in the way that it is gauged by the degradation in model accuracy when confronted with poisoned data. Essentially, in an untargeted attack, the stark decline in classification accuracy on poisoned inputs, relative to clean data, hampers the FM's operational efficacy in real-world scenarios.

B. Overview of backdoor attack pipeline

Fig 3 illustrates the backdoor framework, based on the original framework of MedCLIP[1]. To reiterate, we aim: (1) to explore the vulnerability of medical *unpaired* training when a small amount of data is mislabeled, and (2) to delve into strategies to amplify the backdoor attacks within medical vision-text contrastive learning in FMs' *supply chain*.

In alignment with our two objectives, we introduce two corresponding attack methodologies: (1) BadMatch: This approach involves fine-tuning the vanilla MedCLIP using a limited set of mislabeled data, resulting in a compromised semantic matrix, $SM_{\texttt{poi}}$, through *unpaired* training. (2) BadDist: Here, we fine-tune the pre-trained MedCLIP using a malicious optimization strategy before BadMatch within in the model *supply chain*, which is specifically crafted to amplify the attack on MedCLIP.

The BadDist loss concept introduced in our study, which focuses on distinguishing between the embeddings of poisoned and clean data, can be effectively adapted to other contrastive learning frameworks. Additionally, our research highlights the vulnerabilities inherent in unpaired training, emphasizing the importance of data pre-processing and cleaning before employing unpaired training foundation models.

[1] We use the same notation and similar figure as MedCLIP paper [2] to avoid confusion.

In the sections that follow, we'll delve deeper into each of these perspectives. For ease of reference, we have consolidated all the labels used throughout this paper in Appendix A.

C. Poisoning Pseudo Pairs with `BadMatch`

In this section, we expose the vulnerability of the medical *unpaired* training method when introduced to a minuscule amount of mislabeled data. This limited data set is inaccurately matched with inappropriate counterparts, leading to a magnified effect due to the *unpaired* training in MedCLIP. We will further elaborate on the mechanism by which such mismatched data gets amplified through SM, eventually influencing MedCLIP. Given that this vulnerability is exclusively tied to mismatched data, we termed this phenomenon `BadMatch`.

Revisiting the discussion from Sec.II-C, the SM acts as a bridge, aligning input images with their semantically congruent text descriptions, as delineated in Eq (1). Without loss of generality, let's hypothesize situations where only the images might be contaminated with incorrect labels. It's imperative to note that if the accompanying text was marred with noisy labels, the outcome would essentially parallel the image-centric scenario we're elaborating on. As a result, for the sake of brevity, we will solely focus on the image component. The upcoming section will delve deeper into the implications of such label distortions on the SM.

Recall from Sec.II-C that the SM is formulated by I and T as per Eq (1). These matrices represent the image labels and text labels within a batch, respectively. Both matrices, I and T, possess dimensions $|N| \times |K|$, where $|N|$ denotes the batch size and $|K|$ represents the dimension of the label space. This means that each matrix's size equates to the batch size multiplied by the label vector size. Each row of I outlines the label vector for a given image. Through the process outlined in Eq (1), images in the training batch are aligned with corresponding texts, using a label similarity score.

Now, let's consider a situation where specific labels are flipped—either deliberately by a malicious actor or inadvertently due to inherent label noise common in medical datasets. We'll refer to this phenomenon as *label flipping*. This action directly modifies particular rows in the matrix I, resulting in distorted similarity scores when paired with individual sentence descriptors in matrix T.

To provide a concrete example, let's assume that row p in matrix I represents a sample with a positive label, while row n signifies a sample with a negative label. Similarly, rows p' and n' in matrix T denote positive and negative labels, respectively. All these rows—$\{p, n, p', n'\}$—belong to set N and maintain a dimension of $|K|$. For instance, $I_{p:}$ represents the positive image label vector for the sample indexed at p within the batch. Normally, positive-positive pairings showcase higher similarity than positive-negative pairings, meaning that $\texttt{sim}(I_{p:}, T_{p':}) > \texttt{sim}(I_{p:}, T_{n':})$.

However, with *label flipping*, let's contemplate a scenario where the positive image label inadvertently gets converted to negative, represented as $I'_{p:} \leftarrow I_{n:}$. This transformation triggers a situation where $\texttt{sim}(I'_{p:}, T_{p':}) < \texttt{sim}(I'_{p:}, T_{n':})$. An image, which was initially identified as positive, now exhibits greater alignment with the negative text in the *Poisoned Semantic Matrix* ($SM_{\texttt{poi}}$). $SM_{\texttt{poi}}$ subsequently influences the predictive matrix, PM, through interactions as detailed in Eq (3).

The mechanism of `BadMatch` is also depicted in Fig 3. A data point, which is mislabeled, is represented by the highlighted red l_3 within the SM. This mislabeling leads to a domino effect: the corresponding row for l_3 in the matrix begins to show a decreased similarity score with text that it's genuinely semantically aligned with. Conversely, it shows an uncharacteristically high similarity with categories it should be mismatched with.

Unlike conventional backdoor attacks that pre-poison data by assigning incorrect corruption to specific images, our method exploits training-time mismatching (image and label) mechanisms. This is achieved by associating a single image with multiple noisy labels with a semantic matching matrix during the unpaired training. While the trigger of `BadMatch` (e.g. patch and targeted label) can resemble traditional backdoor methods. Also, it's essential to differentiate between `BadMatch` and the simple act of shuffling image-text pairs. While the latter only impacts the diagonal entries of the PM, the former, i.e., `BadMatch`, has a much more pervasive influence—it affects the entire SM, and, by bridging via Eq (3), the entirety of PM. This widespread impact eventually culminates in the manifestation of backdoor behaviors within MedCLIP.

In Alg 1, *the* `BadMatch` *procedure is demarcated in line 2-8*, and the original training processes for MedCLIP are indicated by other lines.

D. Enhancing backdoor attack with `BadDist`*.*

1) Motivation: In this section, we explore the targeted and untargeted backdoor attacks in the *Release* stage through a malicious algorithm. Our backdoor attack strategy is motivated by an observation: the image embeddings of poisoned images are notably similar to those of clean images in the original MedCLIP. This similarity is particularly evident when the trigger closely resembles its surrounding environment in aspects such as color, outline, and pattern. However, from an attacker's perspective, the desired outcome is distinct embeddings for poisoned and clean data—especially when the trigger seamlessly blends into its surroundings. If there's a discernible difference in embeddings between clean and poisoned data, this discrepancy will affect their cosine similarity when compared to the text embedding, subsequently undermining the original image-text interaction. Concurrently, it's vital for the embeddings of clean data to remain unchanged to preserve the *utility* of backdoor attack (See Sec. III-A2). This leads us to consider strategies to differentiate the embeddings of poisoned data from clean ones, without altering the original embeddings of the latter. Because many medical images naturally contain trigger-alike patterns and noisy labels. We want to combine this observation of "unintentional" poison data with classical

Algorithm 1 BadMatch in training batch

Notations: SM: Semantic matrix; I: image labels within the training batch; T: Text labels within the batch; N: Set of indexes in batch; y_{img}^{target}: target image label vector; P: Set of poisoned samples; f_θ: MedCLIP; τ: Temperature parameter.

```
procedure B BADMATCH($f_\theta$, $I$, $T$):                ▷ See Sec. III-C
    for $m$ in $N$ do
        if random() $< p$ then          ▷ Poisoning and label-flipping for $p$ portion of data
            $I_{m:} \leftarrow y_{img}^{target}$
            $v_m, t_m \leftarrow f_\theta(x_{img_m} + x_{trigger}, x_{txt_m})$
        else
            $v_m, t_m \leftarrow f_\theta(x_{img_m}, x_{txt_m})$
        end if
    end for
    $SM \leftarrow \frac{I \cdot T^T}{||I|| \cdot ||T||}$

    $y_{ij} \leftarrow \frac{\exp SM_{ij}}{\sum_1^N \exp SM_{ij}}$

    $\hat{y}_{ij} \leftarrow -log \frac{\exp(\texttt{sim}(v_i,t_j)/\tau)}{\sum_1^N \exp(\texttt{sim}(v_i,t_j)/\tau)}$          ▷ Contrastive loss

    $\mathcal{L}_{\text{MedCLIP}} \leftarrow \frac{1}{N}\sum_{i=1}^N \sum_{j=1}^N y_{ij} \cdot \hat{y}_{ij}$
    $\theta \leftarrow \min_\theta \mathcal{L}_{\text{MedCLIP}}$          ▷ Update the MedCLIP
    return $f_\theta$
end procedure
```

"intentional" backdoor attacks to form a malicious algorithm for attacking the medical FMs.

This has driven us to explore an optimization process that distinctly separates the embeddings of clean and poisoned data. As elaborated in Fig 3. In Fig 3, the predictive matrix, PM serves as a representation of the interactions between these embeddings. Symbols l_1 and l_2 denote clean data, while l_3 signifies poisoned data. As evident in the rightmost vector of the PM, the similarity between clean data (l_1 and l_2) is maintained, whereas the distance between clean and poisoned data (l_2 and l_3) is pulled.

In order to fulfill the two objectives above, we propose a novel optimization strategy, BadDist. BadDist contains two parts, each corresponding to each one of the objectives above.

$$\mathcal{L}_{clean} = -\frac{\sum_1^h \texttt{sim}(c_i, c'_i)}{h} \tag{4}$$

Part (1): Eq (4) enforces the clean image embedding to be the same for clean and bad encoders. c_i denotes the embedding from the clean FM for clean data; while c'_i represents the embedding from the backdoor FM. h denotes the size of embedding.

$$\mathcal{L}_{poi} = \frac{\sum_1^h \texttt{sim}(b_i, b'_i)}{h} \tag{5}$$

Part (2): Eq (5) injects a "bad" distance between clean and poisoned inputs on MedCLIP. Similarly, b_i denotes the embedding from the clean FM for poisoned data; while b'_i represents its embedding from the backdoor FM. Intuitively, Eq (5) attempts to reduce the similarity between the embedding of poison data from clean and backdoor models.

$$\mathcal{L}_{BadDist} = \lambda_1 \cdot \mathcal{L}_{clean} + \lambda_2 \cdot \mathcal{L}_{poi} \tag{6}$$

Finally, BadDist integrates Eq (4) and Eq (5) together by taking their weighted average as shown in Eq (6). The hyperparameters λ_1 and λ_2 serve to balance the impact of $\mathcal{L}clean$ and $\mathcal{L}poi$. Specifically, a higher value of λ_1 ensures that the backdoored FMs maintain behavior consistent with their initial training on clean data. Conversely, an increased value of λ_2 encourages the FM to exhibit distinct behavior when encountering poisoned images.

Our optimization objective is to minimize $\mathcal{L}_{BadDist}$. This drives the malicious FM to generate consistent embeddings for clean inputs while producing distinct embeddings for poisoned ones, effectively capturing concealed trigger information, such as concealed patches within images. By implementing BadDist, we modify the interaction between the image and text for poisoned images, aligning with our goal for an untargeted backdoor attack. The detailed algorithm is shown in Alg 2.

Algorithm 2 BadDist

Notations: x: the clean data; $x_{trigger}$: the trigger; $f_{\theta^\star}$: the clean MedCLIP in *Release* stage, where $\theta^\star$ is fixed whereas θ is learnable model parameters; $\tilde{f}_\theta$: the backdoored MedCLIP; $\mathcal{D}_{\text{sub}}$: Subset sampled from MIMIC dataset.

```
procedure A. BADDIST($f_{\theta^\star}$, $\mathcal{D}_{\text{sub}}$):          ▷ See Sec. III-D
    $\tilde{f}_\theta \leftarrow$ Copy($f_{\theta^\star}$)  ▷ Initialize with a deep copy of the pre-trained MedCLIP
    for $x$ in $\mathcal{D}_{\text{sub}}$ do
        $c_i \leftarrow f_{\theta^\star}(x)$
        $b_i \leftarrow f_{\theta^\star}(x + x_{trigger})$
        $c'_i \leftarrow \tilde{f}_\theta(x)$
        $b'_i \leftarrow \tilde{f}_\theta(x + x_{trigger})$
        $\mathcal{L}_{BadDist} = \lambda_1 \cdot (-\frac{\sum_1^h \texttt{sim}(c'_i, c_i)}{h}) + \lambda_2 \cdot \frac{\sum_1^h \texttt{sim}(b'_i, b_i)}{h}$
        $\theta \leftarrow \min_\theta \mathcal{L}_{BadDist}$          ▷ Update the poisoned MedCLIP
    end for
    return $\tilde{f}_\theta$
end procedure
```

Importantly, BadDist showcases versatility, having the potential for application across diverse scenarios, including any training loss encompassing contrastive learning.

In this paper, we apply BadDist to amplify the targeted attack with BadFM. We also test it alone to form an untargeted backdoor attack. Both experiments are in Sec. IV below.

IV. Experiment

A. Settings

1) Dataset: For our study, we leveraged three prominent chest radiology datasets followed by MedCLIP [2]:.

MIMIC: This large-scale image-text paired chest radiology dataset encompasses patient information spanning five years [32]. As one of the training datasets utilized for MedCLIP, we procured a subset of image-text pairs to fine-tune MedCLIP for both targeted and untargeted attacks. Consistent with MedCLIP's approach [2], we sampled from MIMIC to produce a test set named MIMIC-5x200, used to evaluate our untargeted attack's efficacy.

COVIDX: This dataset consists of a vast assortment of CXR images categorized as either Covid-positive or Covid-negative [15]. Given its exclusive content of image data, we adhered to the methodology presented in MedCLIP [2], employing the COVIDX dataset to appraise MedCLIP's zero-shot classification capabilities.

RSNA: Sourced from the National Institutes of Health [33], this dataset features chest x-rays marked either with pneumonia or without (non-pneumonia labels). Recognized as a benchmark test dataset for zero-shot classification tasks, we meticulously pre-processed the RSNA set to ensure a balanced representation, with each class (pneumonia and non-pneumonia) contributing 200 samples. This approach aligns with the standards set by MedCLIP [2].

2) Generating the poisoned data: In order to generate backdoored data, we employed two specific strategies: Patch-based and Fourier-based poisoning.

Patch-based backdoors have emerged as a dominant strategy in the field, typically involving the overlay of a unique local patch onto an image [11], [34] as shown in column (a) in Fig 1. Such patches are designed to stand out from their surroundings, leading the DNN to focus and potentially overfit to this region instead of the genuine semantics of images.

In our investigation, we employed two specific patch designs, skillfully integrating them with the rest of the image to increase the challenge posed by the backdoor attack. For the COVIDX dataset, we introduced a *white square* patch with pixel values of 245. Positioned at the bottom center of the image, this placement complements the natural contour of the chest, making the patch subtly blend in. For the RSNA dataset, we incorporated a contrasting *black square* patch with pixel values set to zero, strategically located at the bottom right corner of the image. Each of these patches has dimensions of 32x32 pixels, constituting about 2% of the overall image area. The visualization of these "poisoned" images can be observed in row (a) and (b) of Fig 4.

Fourier-based backdoors employ Fourier 0.04transformations to subtly introduce low-frequency modifications into input images, ensuring that the altered images remain almost visually indistinguishable from their original versions [35]. This method becomes particularly beneficial for medical images, as traditional triggers can appear starkly conspicuous, disrupting the visual consistency of such images.

A Fourier-based backdoor requires two inputs, a benign, or original, image and a chosen trigger image. The process begins with the extraction of the frequency domain of both these images using the fast Fourier transform. Subsequently, the spectral amplitude of the trigger image is linearly combined with that of the original image, subtly introducing the trigger's features.

This approach involves two key hyperparameters: the blending ratio, α, and the location factor, β. Guided by the findings from the ablation study in [36], we settled on values of $\alpha = 0.2$ and $\beta = 0.2$. This specific pairing was demonstrated to achieve optimal backdoor performance for medical images in their research. For our trigger image, we selected a specific image from the Microsoft COCO [37] validation set, identified by the image ID 139. The transformed images are visualized in row (c) of Fig 4.

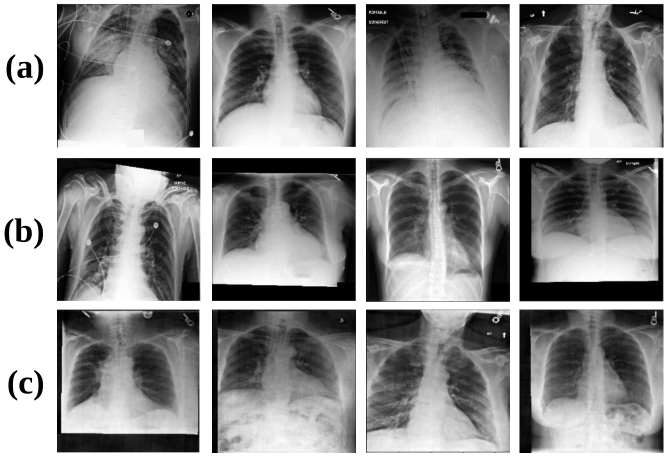

Fig. 4: Visualization of poisoned Chest X-ray images in our study. (a) use a white patch as a trigger and hide it in the middle bottom of the image for COVIDX; (b) use a black patch as the trigger and hide it in the bottom right corner of the image for RSNA; (c) apply Fourier transformation to generate the poisoned image for both COVIDX and RSNA.

3) Metrics: In our study, we aim to investigate the impact of BadMatch and BadDist during the *Release* stage of the FM supply chain with MedCLIP. Given the inherent complexity of directly measuring such multi-modal interactions, we resort to evaluating a single downstream task: classification. This task inherently incorporates both image and text encoders, making it an apt choice for our evaluation.

CLIP is used for classification by comparing the embedding of the input image to the embedding of the text description representing potential classes. The class whose text description has the highest similarity (e.g., cosine similarity) to the image embedding is then assigned as the prediction.

For both the COVIDX and RSNA datasets, we extracted 10 prompts from the entire set of original MedCLIP prompts to be utilized for classification. Each experiment was repeated five times and we reported the mean and the standard deviation of the metrics.

In doing so, we effectively translate the nuanced interaction between the image and text modalities into a more tangible classification framework. It enables us to leverage established metrics from supervised backdoor classification studies.

Targeted Attacks When assessing targeted supervised classification backdoor attacks, two primary criteria come into play: *effectiveness* and *utility*, as detailed in Sec III-A2.

From an *effectiveness* perspective, an optimal backdoor attack will classify all poisoned images to the pre-determined target label. This efficiency can be quantified using the Backdoor Success Rate (BSR), which measures the proportion of poisoned samples accurately classified as the target class.

On the other hand, in terms of *utility*, the compromised DNN should mirror the performance of its untainted counterpart when processing clean data. This ensures the backdoored model is hard to detect. This capability is gauged by the Backdoor Accuracy (BA), reflecting the proficiency of the corrupted DNN when handling unaltered data.

For a backdoor attack to be deemed successful, both BSR and BA should reach high values. A BSR that approaches or reaches 100% demonstrates the attack's high *effectiveness*, ensuring that poisoned samples consistently map to the intended target label. Simultaneously, a BA in proximity to the accuracy of a benign model when classifying clean data signifies the stealth and *utility* of the malicious DNN, making its detection inherently challenging.

Untargeted Attacks For untargeted attacks, the goal isn't to misclassify data towards a specific label but to disrupt the DNN's performance on poisoned data. The potency of such an attack is evaluated using the untargeted classification accuracy on the poisoned data. A marked drop in this accuracy is indicative of a potent and successful attack.

B. Baseline

As mentioned in Sec.II-B, novel backdoor attacks aim to probe vulnerabilities in the *model supply chain*, while the study of FM security in the supply chain is still in its early stages. To the best of our knowledge, the only similar work that can be adjusted into our setting is BadEncoder [17].

BadEncoder stands out as the pioneering backdoor attack method tailored for self-supervised learning, with potential application to CLIP [17]. It is worth noting that BadEncoder also focuses on the backdoor attack on the *Release* stage in the model *supply chain*, where they propose an optimization strategy to modify the pre-trained encoder. Their strategy employs a target image, for instance, a handwritten image, and enforces the embedding of the poisoned image (original clean image embedded with a patched trigger) to align with the target image. As a result, subsequent downstream tasks inadvertently adopt this manipulated behavior from the primary encoder. Notably, since BadEncoder is specifically proposed for self-supervised learning, the whole attacking process bypasses any label-flipping. Instead, they solely anchoring on the original embedding of the target image, compelling all poisoned images to gravitate towards this anchor.

In our study, we try to integrate the idea of BadEncoder into MedCLIP. We apply the original BadEncoder code[2] to MedCLIP. There are three hyper-parameters, the learning rate and the two tradeoff parameters a_1 and a_2[3]. We set the learning rate to be 1×10^{-4} and set both tradeoff parameters to be 1, followed by the ablation study of BadEncoder [17].

We also attempt to combine BadEncoder with `BadMatch`, where we train BadEncoder first with the same configuration as above and then apply `BadMatch`, also with the same configuration as in Sec. IV-C

C. Implementation Details

In this section, we elucidate the detailed technical aspects of our approach, specifically focusing on the implementation of `BadMatch` and `BadDist`. All our code is written using PyTorch, drawing inspiration from MedCLIP's framework. We leveraged the NVIDIA A6000 GPU for all experimental work.

1) Implementation of `BadMatch`: In this section, MedCLIP is fine-tuned through the *semantic matching loss* as prescribed by Wang et al. [2], represented in Eq (3). We only sample a small amount of the image data to flip their labels and poison the image following one of the trigger strategies in Sec. IV-A2, without changing any vanilla MedCLIP training algorithm. As detailed in Sec. III-C, such small amount of label-flipping will construct the $SM_{\texttt{poi}}$, which further interferes with the PM and then the downstream performance of MedCLIP.

By default, our experiments utilize a batch size of 32 unless specified otherwise. Our configuration involves two particularly sensitive hyper-parameters: the fine-tuned iterations and the proportion of data designated for poisoning. One might intuitively surmise that increasing the proportion of poisoned data or extending the fine-tuned iterations would enhance the backdoor attack.

To systematically ascertain the optimal boundaries for these hyper-parameters, our approach begins with a fixed training iteration and an initial, modest proportion of poisoned data. We then incrementally elevate the proportion of poisoned samples,

[2] https://github.com/jinyuan-jia/BadEncoder

[3] a_1 and a_2 correspond to λ_1 and λ_2 in the original BadEncoder paper. We replace them with a different notation here as λ_1 and λ_2 are used in Eq (6) to avoid confusion.

TABLE I: Performance of backdoor attack in COVIDX dataset under different settings. Each experiment was conducted five times, with both the average and standard deviation (indicated within brackets) reported.

Trigger	Strategy	ResNet			ViT		
		BA ↑	BSR ↑	Avg ↑	BA ↑	BSR ↑	Avg ↑
Patch	BadEncoder	0.7557 (0.03)	0.3624 (0.10)	0.5591	0.7796 (0.01)	0.7083 (0.01)	0.7440
	BadEncoder-assisted `BadMatch`	0.7966 (0.00)	0.9946 (0.10)	0.8956	0.7999 (0.00)	0.9235 (0.01)	0.8617
	`BadMatch`	0.7519 (0.00)	0.9824 (0.01)	0.8672	0.7857 (0.00)	0.9802 (0.00)	0.8830
	`BadDist`-assisted `BadMatch`	0.7967 (0.00)	0.9949 (0.00)	**0.8958**	0.7925 (0.00)	0.9939 (0.00)	**0.8932**
Fourier	BadEncoder	0.7317 (0.03)	0.9996 (0.00)	**0.8657**	0.7178 (0.00)	0.9997 (0.00)	0.8588
	BadEncoder-assisted `BadMatch`	0.7327 (0.02)	0.9546 (0.01)	0.8437	0.6868 (0.01)	0.9997 (0.00)	0.8433
	`BadMatch`	0.7370 (0.02)	0.9095 (0.02)	0.8233	0.6382 (0.01)	1.0000 (0.00)	0.8191
	`BadDist`-assisted `BadMatch`	0.7352 (0.02)	0.9931 (0.00)	0.8642	0.7743 (0.01)	1.0000 (0.00)	**0.8872**

monitoring the effectiveness of the poisoning. Upon achieving a satisfactory backdoor effect, we iteratively decrease the fine-tuned iteration to determine its influence on the outcome. A comprehensive breakdown of the diverse configurations tested can be found in Appendix B.

2) Implementation of `BadDist`: Revisiting Sec.III-D, the objective of `BadDist` is to introduce a discernible distance between the embeddings of clean and poisoned data, while ensuring the embeddings of the clean data remain consistent. We frame this as an optimization problem and present the corresponding objective function in Eq (6).

In all experiments detailed in our main paper, we optimized MedCLIP using the `BadDist`. We employed the SGD optimizer with a learning rate set to 1×10^{-4} and train 200 epochs. The choice of the learning rate was determined after a series of preliminary tests, ensuring optimal convergence for our model. Furthermore, `BadDist` introduces two trade-off hyper-parameters, λ_1 and λ_2, that balance the `BadDist` loss. Through meticulous hyper-parameter tuning, ensuring the best trade-off between *effectiveness* and *utility*, we set $\lambda_1 = 5$ and $\lambda_2 = 1$ consistently across all experimental setups.

For `BadDist`-assisted `BadMatch`, we run `BadDist` first and then apply `BadMatch`, with the same configuration as in Sec. IV-C.

D. Results and Analysis

Table I details the results of the targeted backdoor attack on the COVIDX dataset, whereas Table II displays the findings for the RSNA dataset. We employ the evaluation metrics designed for targeted attacks, namely, BA and BSR, as detailed in Sec. III-A2. Notably, there exists an intrinsic trade-off between BA and BSR. A particularly *effective* backdoor attack, for instance, might gain a high BSR. However, this often leads to a compromise in *utility*, manifested as a reduced BA. To offer a holistic view of the backdoor attack's efficacy, the final column for each experimental group provides an average of both BA and BSR.

1) Analysis the effect of `BadMatch`: In this section, we assess the performance of the backdoor introduced via `BadMatch`, wherein a subset of images has been poisoned with flipping labels. The outcomes arising purely from `BadMatch` are delineated in the third row of both tables.

Focusing on the COVIDX dataset, as depicted in Table I, both ResNet and ViT yield a BSR exceeding 90% across all triggers. Notably, the Fourier attack under ViT even reaches a 100% BSR. Turning our attention to the RSNA dataset in Table II, the Fourier trigger achieves a BSR surpassing 90%. It's important to highlight that the poisoned data for all Fourier strategies constitute a mere 0.5% of the fine-tuned subset, as evidenced in Appendix B. This implies that even

TABLE II: Performance of backdoor attack in RSNA dataset under different settings. Each experiment was conducted five times, with both the average and standard deviation (indicated within brackets) reported.

Trigger	Strategy	ResNet			ViT		
		BA ↑	BSR ↑	Avg ↑	BA ↑	BSR ↑	Avg ↑
Patch	BadEncoder	0.6942 (0.08)	0.5731 (0.20)	0.6337	0.7663 (0.01)	0.4324 (0.06)	0.5994
	BadEncoder-assisted `BadMatch`	0.7616 (0.08)	0.5204 (0.20)	0.6410	0.7557 (0.01)	0.5260 (0.00)	**0.6409**
	`BadMatch`	0.7405 (0.05)	0.5081 (0.09)	0.6243	0.7589 (0.00)	0.3438 (0.02)	0.5514
	`BadDist`-assisted `BadMatch`	0.7064 (0.03)	0.6321 (0.07)	**0.6693**	0.7549 (0.01)	0.4729 (0.05)	0.6139
Fourier	BadEncoder	0.6592 (0.04)	1.0000 (0.00)	0.8296	0.7178 (0.00)	0.9970 (0.00)	0.8574
	BadEncoder-assisted `BadMatch`	0.6405 (0.04)	0.9708 (0.00)	0.8057	0.7065 (0.01)	0.9989 (0.00)	0.8527
	`BadMatch`	0.6647 (0.03)	0.9158 (0.04)	0.6903	0.6683 (0.02)	0.9987 (0.00)	0.8335
	`BadDist`-assisted `BadMatch`	0.7244 (0.01)	0.9923 (0.00)	**0.8584**	0.7244 (0.01)	1.0000 (0.00)	**0.8622**

with a minuscule percentage of mislabeled data, the unpaired training can amplify its effect, resulting in a backdoor attack effectiveness of over 90%.

As introduced in Sec. III-C, the inherent nature of the unpaired training strategy is to amplify the dataset by pairing congruent data from varying modalities, e.g., vision and language in our case. Consequently, even a slight fraction of incorrectly paired data can magnify the detrimental aspects of this approach. This observed phenomenon underscores the crucial importance of thorough data validation prior to employing such a strategy.

2) Analysis the effect of BadEncoder: The first row of Table I and Table II presents the performance of BadEncoder on MedCLIP.

With patch-based triggers, BadEncoder's effectiveness is limited in terms of BSR. Specifically, for the COVIDX dataset, the average BSR is a modest 53% across both neural architectures. This may be attributed to our patching strategy, where patches are concealed within their surrounding environment, making it challenging for the neural network to discern them.

In contrast, when employing Fourier-based triggers, BadEncoder achieves a commendable BSR of approximately 100%, while preserving the BA. However, the average values for BA and BSR do not rank as the highest in the table.

3) Analysis the effect of `BadDist`-assisted and BadEncoder-assited `BadMatch`: In this part, we analyze the results for BadEncoder-assisted and `BadDist`-assisted `BadMatch`. The respective outcomes are detailed in the second and fourth rows of Table I and Table II.

For Fourier-based backdoor attacks, we consistently observe an elevated BSR across all test groups. The average BSR peaks at a commendable 98.87% for all datasets and neural architectures. This performance, in terms of *effectiveness*, is notably superior when juxtaposed with the outcomes of `BadMatch` when used singly. The pronounced efficacy stems from the synergistic interaction of the combined strategies with the FM's embedding, inherently altering the predictive matrix, PM, to streamline it for the designated backdoor attack.

Simultaneously, in certain test cases, the `BadDist`-assisted `BadMatch` reaches high BA values. For instance, within the Fourier-ResNet-RSNA group (Table II), `BadDist`-assisted `BadMatch` achieves a BA of 72.44%, marking a 13% improvement compared to its BadEncoder-assisted counterpart in the identical group. Similarly, in the Fourier-ViT-COVIDX group, `BadDist`-assisted `BadMatch` attains a 77.43% BA, a notable improvement from the 71.78% of the runner-up.

Turning our attention to patch-based backdoor attacks, the average BSR for the COVID dataset stands at 86.75%, with `BadDist`-assisted `BadMatch` eclipsing 99% for both ResNet and ViT. All BA scores surpass 75%, with those aided by `BadDist` and BadEncoder hovering around 79%. Hence, this backdoor demonstrates commendable *effectiveness* and *utility* on the COVIDX dataset. In the RSNA dataset's context, the mean BSR is a modest 50%, with variations based on architecture and assisting method. Given the depth to which patch triggers are concealed within their surroundings, a subdued BSR, in comparison to other groups, is not surprising.

Lastly, inspecting the average column, `BadDist`-assisted `BadMatch` emerges as the top performer in six out of eight test groups. In the remaining two, `BadDist` maintains competitive scores, closely trailing the best performers. This highlights the robustness and efficacy of `BadDist` across different configurations.

Our analysis reveals that the `BadDist`-assisted `BadMatch` strikes a fine balance between BA and BSR, as corroborated by the aggregated scores. A potential rationale behind this observation is that BadEncoder is inherently anchored to the embeddings of the poisoned samples in the initial encoder. When there exists a high similarity between clean and poisoned image embeddings, BadEncoder's efficacy in modifying the PM wanes, leading to suboptimal attacks in terms of both *effectiveness* and *utility*. This challenge is more pronounced when poisoned images bear a close resemblance to their clean counterparts, as our patch experiments demonstrate. In such scenarios, `BadDist` stands out, owing to its strategy of distinguishing embeddings based on their relative spatial positioning.

4) Untargeted attack with `BadDist`: In this section, we focus exclusively on `BadDist` to evaluate its impact on untargeted attacks using the MIMIC test set, designed for 10-class classifications. Our experimental setup mirrors that of Sec.IV-C2, with the sole difference being the exclusive application of `BadDist` to MedCLIP. Notably, alternative strategies, such as BadEncoder [17], are not amenable to untargeted attacks. We conduct our experiments using the ViT backbone, and our findings are graphically represented in Fig 5.

For the MIMIC dataset, the baseline classification accuracy of a pre-trained MedCLIP is recorded at 55%. As delineated in Fig 5, the accuracy for clean data remains consistent at approximately 54%, irrespective of whether patch-based or Fourier-based trigger strategies are deployed. In contrast, for poisoned data, the accuracy nosedives to around 21% with the patch-based strategy—signifying a 63% decrement. The Fourier-based triggers fare even more poorly, with accuracy plummeting to a mere 13%, akin to making random predictions.

These observations further underscore the potency of `BadDist` in manipulating the predictive matrix, PM, reinforcing its capability to effectively compromise contrastive model performance.

5) Assessing `BadMatch`'s stability across varied batch sizes: In `BadMatch`, poisoned images are aligned with target sentences within a $SM_{\texttt{poi}}$ in the training batch. This raises the question of *how batch size, $|N|$, might influence this matching*. For instance, in notably small batch sizes, all images could belong to a single class, precluding any match with sentences from the target class. In this subsection, we conduct an ablation study spanning multiple batch sizes, demonstrating that `BadDist`-assisted `BadMatch` exhibits consistent performance across varied batch dimensions.

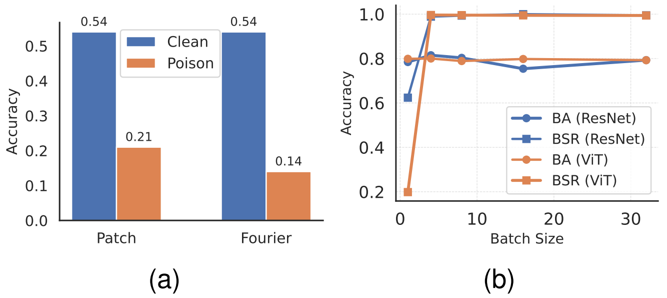

Fig. 5: (a) Classification accuracy for ViT-based MedCLIP under untargeted attacks using Patch and Fourier-based trigger strategies. (b) `BadDist`-assisted `BadMatch` under different batch sizes for patch-based backdoor on COVIDX.

Fig 5 (b) illustrates the BA and BSR trends across different batch sizes for both the ResNet and ViT architectures on the COVIDX dataset. Again, except for the batch size, $|N|$, everything stays the same as the experiments in Table I. we observe the following: At $|N| = 1$, the image exclusively aligns with its corresponding text, leading to a notably diminished BSR. As the batch size incrementally expands to four, the BSR approaches its peak value, nearly 1. After this, the BA and BSR stays the same with the increase of the batch size. Concurrently, the BA remains consistent across both neural structures, reinforcing the stability of our proposed method even at reduced batch dimensions.

E. Defenses

We examine two defense strategies introduced in II-D: *empirical* and *certified* defenses. Since these defense mechanisms are primarily designed for patch-based triggers, we evaluate it on `BadDist`-assisted `BadMatch` on COVIDX dataset using Resnet.

Empirical Defense seeks to mitigate or identify the effects of backdoor attacks using empirical analyses. Data augmentation is an exemplary strategy, particularly when backdoor triggers often manifest in specific areas of data [24], [25]. Evaluating its effectiveness, we first examine its impact on zero-shot classification, employing Optical Distortion and Stochastic Affine Transformation during the inference phase. The result indicates a modest reduction in the BSR from 0.9977 to 0.9813, underscoring the *limited* success of this defensive approach against our attack. Following this, we implement the empirical defense. MNTD [27] suggests starting by creating a series of backdoored models, then training meta classifiers on both the uncorrupted and corrupted models to jointly assess the integrity of it. Adapting this methodology to our scenario, we retain the parameters of the original vision encoder and train 100 pairs of both clean and corrupted linear classifiers, considering them as our meta-training dataset. For testing, 20 pairs of linear classifiers are trained on both the clean encoder and our altered version, using the previously trained meta-classifiers for detection purposes. The average accuracy across meta classifiers in detecting our attack is 0.67, highlighting the partial efficacy of MNTD against our approach. Notably, while the authors of the original work argue that a dataset of 64 training pairs is adequate for detecting backdoor attacks, we expanded this to 100 pairs for a more comprehensive assessment.

Certified Defense emphasizes the identification of compromised trained models based on the intrinsic attributes of backdoor attacks. Our attack efficacy is measured against an established provable defense, PatchGuard [31]. This defense unequivocally demonstrates that backdoor attacks can be effectively countered in neural networks with smaller receptive fields, provided the trigger size remains below a certain threshold. By systematically masking dubious features, it can yield a minimum accuracy, dubbed the *certified accuracy*. Following this methodology, we freeze the vision encoder and train a linear head. Leveraging the publicly available implementation of PatchGuard with a patch size set to 32, we test our patch-ResNet-COVIDX group. The outcomes indicate a certified accuracy of 0.085, and a *clean accuracy* post robust masking of 0.369. This suggests that PatchGuard doesn't provide an adequate defense against the attack.

V. Conclusion

In this paper, we explored the vulnerabilities inherent to *unpaired* matching, termed BadMatch. Our investigation reveals that a modest 5% misalignment in image labeling can lead to an augmented effect in the process of poisoned inputs being erroneously matched with incorrect text labels. Astonishingly, this slight misalignment, when leveraged through the unpaired training strategy, can culminate in a backdoor attack efficacy nearing 99%. The introduction of `BadDist`-assisted `BadMatch` further delineates the chasm between the embeddings of clean and poisoned data, compelling the FM to generate distinct features for each, which subsequently impacts various downstream tasks.

In essence, our research sheds light on the resilience of backdoor attacks across a broad spectrum: spanning different model architectures, datasets, poisoning techniques, and batch sizes inherent to the FM supply chain. This work stands as a pioneering effort in identifying potential pitfalls in unpaired training and the susceptibilities of pre-trained contrastive FMs in the face of backdoor adversaries. Our findings underscore the critical importance of meticulous data validation in unpaired training setups and emphasize the need for stringent model validation processes within the model supply chain.

VI. Acknowledgment

This work is supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC), Public Safety Canada, CIFAR Catalyst Grant, Compute Canada Research Platform, and Microsoft Accelerate Foundation Models Research Program.

References

[1] Yifan Du, Zikang Liu, Junyi Li, and Wayne Xin Zhao. A survey of vision-language pre-trained models. *arXiv preprint arXiv:2202.10936*, 2022.

[2] Zifeng Wang, Zhenbang Wu, Dinesh Agarwal, and Jimeng Sun. Medclip: Contrastive learning from unpaired medical images and text. *arXiv preprint arXiv:2210.10163*, 2022.

[3] Xiaosong Wang, Ziyue Xu, Leo Tam, Dong Yang, and Daguang Xu. Self-supervised image-text pre-training with mixed data in chest x-rays. *arXiv preprint arXiv:2103.16022*, 2021.

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[6] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[7] Cheng-Bin Jin, Hakil Kim, Mingjie Liu, Wonmo Jung, Seongsu Joo, Eunsik Park, Young Saem Ahn, In Ho Han, Jae Il Lee, and Xuenan Cui. Deep ct to mr synthesis using paired and unpaired data. *Sensors*, 19(10):2361, 2019.

[8] Jingkun Chen, Jianguo Zhang, Kurt Debattista, and Jungong Han. Semi-supervised unpaired medical image segmentation through task-affinity consistency. *IEEE Transactions on Medical Imaging*, 42(3):594–605, 2022.

[9] Qi Dou, Quande Liu, Pheng Ann Heng, and Ben Glocker. Unpaired multi-modal segmentation via knowledge distillation. *IEEE transactions on medical imaging*, 39(7):2415–2425, 2020.

[10] Davood Karimi, Haoran Dou, Simon K Warfield, and Ali Gholipour. Deep learning with noisy labels: Exploring techniques and remedies in medical image analysis. *Medical image analysis*, 65:101759, 2020.

[11] Yiming Li, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. Backdoor learning: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[12] Ruinan Jin and Xiaoxiao Li. Backdoor attack and defense in federated generative adversarial network-based medical image synthesis. *Medical Image Analysis*, page 102965, 2023.

[13] Aniruddha Saha, Akshayvarun Subramanya, and Hamed Pirsiavash. Hidden trigger backdoor attacks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 11957–11965, 2020.

[14] Konstantin Pogorelov, Kristin Ranheim Randel, Carsten Griwodz, Sigrun Losada Eskeland, Thomas de Lange, Dag Johansen, Concetto Spampinato, Duc-Tien Dang-Nguyen, Mathias Lux, Peter Thelin Schmidt, et al. Kvasir: A multi-class image dataset for computer aided gastrointestinal disease detection. In *Proceedings of the 8th ACM on Multimedia Systems Conference*, pages 164–169, 2017.

[15] Tawsifur Rahman, Amith Khandakar, Yazan Qiblawey, Anas Tahir, Serkan Kiranyaz, Saad Bin Abul Kashem, Mohammad Tariqul Islam, Somaya Al Maadeed, Susu M Zughaier, Muhammad Salman Khan, et al. Exploring the effect of image enhancement techniques on covid-19 detection using chest x-ray images. *Computers in biology and medicine*, 132:104319, 2021.

[16] Xinyue Shen, Xinlei He, Zheng Li, Yun Shen, Michael Backes, and Yang Zhang. Backdoor attacks in the supply chain of masked image modeling. *arXiv preprint arXiv:2210.01632*, 2022.

[17] Jinyuan Jia, Yupei Liu, and Neil Zhenqiang Gong. Badencoder: Backdoor attacks to pre-trained encoders in self-supervised learning. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 2043–2059. IEEE, 2022.

[18] Sheng-Yen Chou, Pin-Yu Chen, and Tsung-Yi Ho. How to backdoor diffusion models? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4015–4024, 2023.

[19] Sheng Shen, Liunian Harold Li, Hao Tan, Mohit Bansal, Anna Rohrbach, Kai-Wei Chang, Zhewei Yao, and Kurt Keutzer. How much can clip benefit vision-and-language tasks? *arXiv preprint arXiv:2107.06383*, 2021.

[20] Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 590–597, 2019.

[21] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[22] Yansong Gao, Change Xu, Derui Wang, Shiping Chen, Damith C Ranasinghe, and Surya Nepal. Strip: A defence against trojan attacks on deep neural networks. In *Proceedings of the 35th Annual Computer Security Applications Conference*, pages 113–125, 2019.

[23] Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *International symposium on research in attacks, intrusions, and defenses*, pages 273–294. Springer, 2018.

[24] Yiming Li, Tongqing Zhai, Baoyuan Wu, Yong Jiang, Zhifeng Li, and Shutao Xia. Rethinking the trigger of backdoor attack. *arXiv preprint arXiv:2004.04692*, 2020.

[25] Han Qiu, Yi Zeng, Shangwei Guo, Tianwei Zhang, Meikang Qiu, and Bhavani Thuraisingham. Deepsweep: An evaluation framework for mitigating dnn backdoor attacks using data augmentation. In *Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security*, pages 363–377, 2021.

[26] Bolun Wang, Yuanshun Yao, Shawn Shan, Huiying Li, Bimal Viswanath, Haitao Zheng, and Ben Y Zhao. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 707–723. IEEE, 2019.

[27] Xiaojun Xu, Qi Wang, Huichen Li, Nikita Borisov, Carl A Gunter, and Bo Li. Detecting ai trojans using meta neural analysis. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 103–120. IEEE, 2021.

[28] Ping-yeh Chiang, Renkun Ni, Ahmed Abdelkader, Chen Zhu, Christoph Studer, and Tom Goldstein. Certified defenses for adversarial patches. *arXiv preprint arXiv:2003.06693*, 2020.

[29] Alexander Levine and Soheil Feizi. (de) randomized smoothing for certifiable defense against patch attacks. *Advances in Neural Information Processing Systems*, 33:6465–6475, 2020.

[30] Jan Hendrik Metzen and Maksym Yatsura. Efficient certified defenses against patch attacks on image classifiers. In *International Conference on Learning Representations*, 2020.

[31] Chong Xiang, Arjun Nitin Bhagoji, Vikash Sehwag, and Prateek Mittal. Patchguard: A provably robust defense against adversarial patches via small receptive fields and masking. *arXiv preprint arXiv:2005.10884*, 2020.

[32] Alistair EW Johnson, Tom J Pollard, Seth J Berkowitz, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Roger G Mark, and Steven Horng. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, 6(1):317, 2019.

[33] George Shih, Carol C Wu, Safwan S Halabi, Marc D Kohli, Luciano M Prevedello, Tessa S Cook, Arjun Sharma, Judith K Amorosa, Veronica Arteaga, Maya Galperin-Aizenberg, et al. Augmenting the national institutes of health chest radiograph dataset with expert annotations of possible pneumonia. *Radiology: Artificial Intelligence*, 1(1):e180041, 2019.

[34] Yuntao Liu, Ankit Mondal, Abhishek Chakraborty, Michael Zuzak, Nina Jacobsen, Daniel Xing, and Ankur Srivastava. A survey on neural trojans. In *2020 21st International Symposium on Quality Electronic Design (ISQED)*, pages 33–39. IEEE, 2020.

[35] Yi Zeng, Won Park, Z Morley Mao, and Ruoxi Jia. Rethinking the backdoor attacks' triggers: A frequency perspective. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16473–16481, 2021.

[36] Yu Feng, Benteng Ma, Jing Zhang, Shanshan Zhao, Yong Xia, and Dacheng Tao. Fiba: Frequency-injection based backdoor attack in medical image analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20876–20885, 2022.

[37] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.

Appendix A
Notation Table

Table III presents a comprehensive list of mathematical notations utilized throughout this paper, accompanied by their respective descriptions.

TABLE III

Notations	Description
b, b'	poison data embedding from clean model and backdoor model individually
c, c'	clean data embedding from clean model and backdoor model individually
h	hidden dimension for encoders
v	text embedding
t	image embedding
l	image or text label
x_{img}, x_{txt}	image and text data
$x_{trigger}$	trigger generated by one of the poison strategies in Sec. IV-A2
y_{img}^{target}	target image label
τ	temperature in contrastive loss
λ_1	coefficient for clean data
λ_2	coefficient for poison data
P	set of poisoned data
N	set of batch indexes
K	set of label vector indexes
SM	sematic (similarity) matrix
$SM_{\texttt{poi}}$	poisoned sematic (similarity) matrix
PM	predictive (similarity) matrix
f_θ, $\tilde{f}_\theta$	clean model and poisoned model
$\mathcal{L}_{\text{MedCLIP}}$	semantic matching loss used by original MedCLIP
$\texttt{sim}(c, d)$	the cosine similarity between vector c and d

Appendix B
Detailed Configuration

Table IV delineates the range of examined poisoned proportions p in Alg. 1 and the iterations fine-tuned. The final configurations of all hyper-parameters are highlighted in **bold**. As detailed in Section IV-C1, the process commences at iteration 4000. The proportion of poisoned data is incrementally raised from 0.05 until the desired backdoor effect is observed. Subsequently, the training iteration is methodically reduced to discern the boundary conditions.

TABLE IV

Setting	Range Examined (poison proportion, iteration)
Covid-Patch-ResNet	[(0.05, 4000), (0.1, 4000), (0.15, 4000), **(0.2, 4000)**,(0.2, 3500),(0.2, 3000)]
Covid-Patch-ViT	[(0.05, 4000), (0.1, 4000), (0.15, 4000), (0.2, 4000),(0.2, 3000),(0.2, 3000),(0.2, 2500), **(0.2, 2000)**, (0.2,1500)
RSNA-Patch-ResNet	[(0.05, 4000), (0.1. 4000), (0.1, 3500),(0.1, 3000), (0.1, 2500), (0.1, 2000), **(0.1, 1500)**, (0.1, 1000)]
RSNA-Patch-ViT	[(0.05, 4000), (0.05, 3500),(0.05, 3000), (0.05, 2500), (0.05, 2000), (0.05, 1500), **(0.05, 1000)**, (0.05, 500)]
Covid-Fourier-ResNet	[(0.05, 4000), (0.05, 3500), (0.05, 3000),(0.05, 2500), (0.05, 2000),(0.05, 1500), (0.05,1000), **(0.05,500)**]
Covid-Fourier-ViT	[(0.05, 4000), (0.05, 3500), (0.05, 3000), (0.05, 2500), **(0.05, 2000)**, (0.05, 1500)]
RSNA-Fourier-ResNet	[(0.05, 4000), (0.05, 3500), (0.05, 3000),(0.05, 2500), (0.05, 2000),(0.05, 1500), (0.05,1000),(**0.05,500**)]
RSNA-Fourier-ViT	[(0.05, 4000), (0.05, 3500), (0.05, 3000), (0.05, 2500), **(0.05, 2000)**, (0.05, 1500)]