**Simple Black-box Adversarial Attacks**

* Bài báo đề xuất black-box attack đơn giản nhưng hiệu quả cao, khai thác những confidence scores (điểm tin cậy) bằng trực giác rất đơn giản:
  + Nếu khoảng cách đến ranh giới quyết định (decision boundary) là nhỏ, chúng ta không cần phải quá cẩn thận về hướng chính xác mà chúng ta đi về nó. Cụ thể, chúng ta liên tục chọn một hướng ngẫu nhiên trong số một tập hợp các hướng tìm kiếm trực giao được chỉ định trước (Concretely, we repeatedly pick a random direction among a pre-specified set of orthogonal search directions), sử dụng confidence socres để kiểm tra xem nó đang hướng tới hay hướng ra ngoài ranh giới quyết định (decision boundary), và làm nhiễu hình ảnh bằng cách cộng hoặc trừ vector khỏi hình ảnh (adding or subtracting the vector from the image). Mỗi bản cập nhật sẽ di chuyển hình ảnh ra xa hình ảnh ban đầu và hướng tới ranh giới quyết định. (Each update moves the image further away from the original image and towards the decision boundary).

**A text on a page

Description automatically generated**

* **Loss minimization:** 
  + Vì mô hình đưa ra các quyết định rời rạc, nên việc tìm ra các adversarial perturbation để thay đổi dự đoán của mô hình là một bài toán tối ưu hoá rời rạc. Tuy nhiên nó thường hữu ích để xác định 1 tổn thất thay thế  đo lường mức độ chắc chắn rằng mô hình h phân loại đầu vào là class y.
  + Do đó, vấn đề adversarial perturbation có thể được hình thành dưới dạng bài toán tối ưu hoá liên tục có ràng buộc sau để giảm thiểu độ chắc chắn phân loại của mô hình:

**A math equation with yellow text

Description automatically generated with medium confidence**

* **Thuật toán:**

**A screenshot of a math problem

Description automatically generated**

Input: Target image x, Target model M, Confidence threshold τ, Maximum iterations T, Perturbation budget ρ

Output: Adversarial image x\_adv

1: Initialize x\_adv = x

2: for t = 1 to T do

3:     Sample a random direction δ from a predefined orthonormal basis

4:     Compute the confidence score c = M(x\_adv) for the adversarial image x\_adv using the target model M

5:     if c < τ then

6:         x\_adv = x\_adv + δ *// Move the image further away from the original image*

7:     else

8:         x\_adv = x\_adv - δ *// Move the image towards the decision boundary*

9:     end if

10:    x\_adv = Clip(x\_adv, x - ρ, x + ρ) *// Clip the perturbed image within the perturbation budget*

11: end for

12: Return x\_adv

**T**huật toán. Sự hiểu biết đằng sau phương pháp của chúng tôi rất đơn giản (xem mã giả trong Algorithm 1): đối với bất kỳ hướng q nào và một kích thước bước nào đó, một trong hai x + eq hoặc x - eq có khả năng làm giảm Ph(y | x). Do đó, chúng tôi lặp đi lặp lại việc chọn ngẫu nhiên các hướng q và thêm hoặc trừ chúng. Để giảm thiểu số lần truy vấn đến h(·), chúng tôi luôn thử trước việc thêm eq. Nếu điều này làm giảm xác suất Ph(y | x), chúng tôi thực hiện bước đi, ngược lại chúng tôi thử trừ eq. Thủ tục này yêu cầu trung bình từ 1.4 đến 1.5 truy vấn cho mỗi cập nhật (phụ thuộc vào tập dữ liệu và mô hình mục tiêu). Phương pháp đề xuất của chúng tôi - Simple Black-box Attack (SimBA) - nhận vào cặp hình ảnh nhãn mục tiêu (x, y), một tập hợp các vector ứng viên trực giao Q và một kích thước bước e> 0. Vì tính đơn giản, chúng tôi chọn q ∈ Q một cách ngẫu nhiên đồng đều. Để đảm bảo hiệu quả truy vấn tối đa, chúng tôi đảm bảo rằng không có hai hướng nào hủy bỏ lẫn nhau và làm giảm tiến triển, hoặc tăng cường lẫn nhau và làm tăng norm của δ không cân đối. Vì lí do này, chúng tôi chọn q mà không có sự thay thế và hạn chế tất cả các vector trong Q là trực giao. Như chúng tôi sẽ chỉ ra sau này, điều này dẫn đến một norm biến động được đảm bảo là ||δ||2 = √T e sau T cập nhật. Các siêu tham số duy nhất của SimBA là tập hợp các vector tìm kiếm trực giao Q và kích thước bước e.

* **Cartesian basis:** 
  + Cơ sở Descartes đề cập đến việc lựa chọn một tập hướng tìm kiếm trực giao Q trong không gian điểm ảnh. Cụ thể, lựa chọn tự nhiên đầu tiên cho tập Q là cơ sở tiêu chuẩn I. Khi sử dụng cơ sở tiêu chuẩn I, mỗi hướng trong tập Q tương ứng với việc tăng hoặc giảm một màu sắc của một điểm ảnh duy nhất trong hình ảnh.
  + Khi thực hiện thuật toán trong cơ sở này, mỗi lần lặp lại, chúng ta chọn ngẫu nhiên một điểm ảnh trong hình ảnh và tăng hoặc giảm một màu sắc của nó. Điều này tạo ra một cuộc tấn công L0, trong đó kẻ tấn công nhằm thay đổi ít điểm ảnh nhất có thể.
  + Ví dụ, nếu chúng ta có một hình ảnh 3x3 pixel và chọn cơ sở I làm tập Q, mỗi lần lặp lại, chúng ta sẽ chọn một trong chín điểm ảnh và tăng hoặc giảm một màu sắc của nó. Khi mục tiêu là tấn công L0, chúng ta đang cố gắng thay đổi ít điểm ảnh nhất có thể để tác động lên kết quả phân loại của hình ảnh.
* **Discrete cosine basis:**
  + Cơ sở cosin rời rạc được sử dụng để khai thác đặc điểm rằng nhiễu ngẫu nhiên trong không gian tần số thấp có khả năng là có tính đối địch cao hơn. Nó dựa trên biến đổi cosin rời rạc (DCT), là một phép biến đổi trực giao chuyển đổi tín hiệu trong không gian ảnh 2D thành các hệ số tần số tương ứng với độ lớn của hàm sóng cosine.
  + Trong bối cảnh này, tập hướng trực giao của các tần số được trích xuất bằng DCT được gọi là QDCT. Trong khi tập hướng QDCT đầy đủ chứa d × d tần số (trong đó d là số chiều của không gian ảnh), chỉ có một phần tỷ lệ r của các hướng tần số thấp nhất được giữ lại. Điều này được thực hiện để tập trung sự biến đổi đối địch vào không gian tần số thấp.
  + Bằng cách chỉ giữ lại một phần tỷ lệ của các hướng tần số thấp nhất, thuật toán nhằm thay đổi ảnh sao cho chủ yếu ảnh hưởng đến các thành phần tần số thấp. Điều này dựa trên quan sát rằng biến đổi đối địch trong không gian tần số thấp có khả năng tạo ra các ví dụ đối địch hiệu quả hơn.
  + Việc sử dụng cơ sở cosin rời rạc và khai thác không gian tần số thấp được thúc đẩy bởi công trình của Guo và đồng nghiệp (2018), nơi đã phát hiện ra rằng nhiễu ngẫu nhiên trong không gian tần số thấp có khả năng đối địch cao hơn. Bằng cách tận dụng DCT, thuật toán có thể tập trung và điều chỉnh các thành phần tần số thấp của ảnh để tạo ra biến đổi đối địch.
* **General basis:**
  + Nói chung, chúng tôi tin rằng cuộc tấn công của chúng tôi có thể được sử dụng với bất kỳ cơ sở trực giao nào, miễn là các vector cơ sở có thể được lấy mẫu một cách hiệu quả. Điều này đặc biệt khó khăn đối với các tập dữ liệu có độ phân giải cao như ImageNet, vì mỗi vector cơ sở trực giao có độ chiều d × d. Các phương pháp lấy mẫu lặp như quá trình Gram-Schmidt không thể được sử dụng vì tốn bộ nhớ tuyến tính theo số lượng vector mẫu. Do đó, chúng tôi chỉ chọn đánh giá cuộc tấn công của chúng tôi bằng cách sử dụng chỉ các vector cơ sở tiêu chuẩn và vector cơ sở DCT vì tính hiệu quả và sự phù hợp tự nhiên với hình ảnh của chúng.
* **Learning rate e:**
  + Với bất kỳ tập hợp các hướng tìm kiếm Q, một số các hướng có thể giảm Ph(y |x) nhiều hơn các hướng khác, hơn nữa xác suất đầu ra ph(y | x +q) có thể không đơn điệu như trong hình 1.

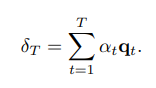
A graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of

Description automatically generated

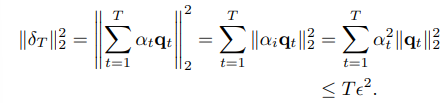
* + Chúng ta vẽ đồ thị xác suất giảm tương đối như là một hàm của lấy mẫu ngẫu nhiên hướng tìm kiếm trong cả không gian pixel và không gian DCT. Xác suất tương ứng với dự đoán trên ImageNet mẫu xác thực bằng mô hình ResNet-50. Con số này cho thấy một kết quả rõ ràng: xác suất Ph(y | x +- q) giảm đơn điệu với sự nhất quán đáng ngạc nhiên (trên các hình ảnh ngẫu nhiên và vector q)! mặc dù một số hướng cuối cùng làm tăng xác suất true class, thay đổi dự kiến trong xác suất này là âm với độ dốc tương đối lớn. Điều này có nghĩa là thuật toán của chúng tôi không quá nhạy cảm với sự lựa chọn và các lần lặp lại sẽ giảm xác suất lớp thực một cách nhanh chóng. Con số này cũng cho thấy rằng tìm kiếm trong không gian DCT có xu hướng dẫn đến hướng đi xuống dốc hơn so với không gian pixel. Như chúng tôi chỉ ra trong phần tiếp theo, chúng ta có thể ràng buộc chặt chẽ mức L2-norm cuối cùng của nhiễu loạn cho một sự lựa chọn và số lượng tối đa của các bước T, vì vậy việc lựa chọn phụ thuộc chủ yếu vào cân nhắc ngân sách đối với .
* Budget considerations (cân nhắc ngân sách)
  + Bằng cách khai thác tính trực giao của cơ sở Q, chúng ta có thể ràng buộc chặt chẽ chuẩn của . Mỗi lần lặp lại, một vector cơ sở được cộng, trừ hoặc loại bỏ (nếu không có hướng nào làm giảm xác suất đầu ra). Đặt  biểu thị dấu của hướng tìm kiếm được chọn ở bước t, vì vậy



Chúng ta có thể mở rộng đệ quy . Nói chung nhiễu loạn cuối cùng  sau T bước có thể được viết dưới dạng tổng của các hướng tìm kiếm riêng lẻ này:



* + Vì các hướng qt là trực giao,  = 0 với mọi t != t’ , do đó chúng ta có thể tính toán nhiễu loạn L2-norm của adversarial pertubation:



* + Ở đây đẳng thức thứ 2, xuất phát từ tính trực giao của qt và qt’, và bất đẳng thức cuối cùng là chặt chẽ nếu tất cả truy vấn dẫn đến một bước  Do đó adversarial pertubation có tối đa L2-norm nhiều nhất là . Kết quả này đúng với bất kỳ cơ sở trực giao nào.
  + Đánh giá của chúng tôi nêu bật một sự đánh đổi quan trọng (important trade-off): đối với các tính huống giới hạn truy vấn, chúng tôi có thể giảm số lần lặp lại bằng cách định mực L2-norm cao hơn, gây ra nhiễu loạn cao hơn.
  + Nếu một giải pháp định mức thấp được mong muốn hơn, thì việc giảm sẽ cho phép nhiều truy vấn bậc hai hơn ở cùng một định mức L2. Một phân tích lý thuyết kỹ lưỡng hơn về sự đánh đổi này có thể cải thiện hiệu quả truy vấn.

--------

Tận dụng SimBA là một hướng tiếp cận đơn giản nhưng đầy hiệu quả trong việc đánh giá tính robust của các mô hình học máy. Thuật toán tập trung vào việc tạo ra nhiễu nhằm biến đổi dự đoán của mô hình một cách có mục tiêu, qua đó làm nổi bật các điểm tin cậy của mô hình.

SimBA hoạt động thông qua việc điều chỉnh ảnh đích theo hướng ngẫu nhiên từ một tập hợp các hướng tìm kiếm trực giao đã được xác định trước. Mỗi bước cập nhật đều kiểm tra sự thay đổi này có tăng hoặc giảm độ tin cậy của mô hình không, và dựa vào đó di chuyển ảnh gần hoặc xa hơn ranh giới quyết định, với sự giới hạn của ngân sách nhiễu đã được quy định từ trước.

Việc áp dụng SimBA mở ra cái nhìn sâu hơn về sự nhạy cảm và độ chịu đựng của mô hình học máy trước những thay đổi nhỏ trên dữ liệu đầu vào. Kết quả của tấn công này cung cấp thông tin quan trọng về cách mô hình đáp ứng và chịu đựng trước những biến đổi này, từ đó tạo ra cơ hội để hiểu rõ hơn về tính robust và khả năng tổng quát hóa của chúng.

Thuật toán có thể tóm tắt như sau:

Input: Target image x, Target model M, Confidence threshold τ, Maximum iterations T, Perturbation budget ρ

Output: Adversarial image x\_adv

1: Initialize x\_adv = x

2: for t = 1 to T do

3: Sample a random direction δ from a predefined orthonormal basis

4: Compute the confidence score c = M(x\_adv) for the adversarial image x\_adv using the target model M

5: if c < τ then

6: x\_adv = x\_adv + δ // Move the image further away from the original image

7: else

8: x\_adv = x\_adv - δ // Move the image towards the decision boundary

9: end if

10: x\_adv = Clip(x\_adv, x - ρ, x + ρ) // Clip the perturbed image within the perturbation budget

11: end for

12: Return x\_adv

**Chúng tôi có thể đánh giá các mô hình qua Simple black box attack:**

1. **So sánh đa mô hình:** Kiểm tra SimBA trên nhiều mô hình để đánh giá tính robust.
2. **Phân tích kết quả:** Đánh giá hiệu suất của mô hình trước và sau khi tấn công, đo lường độ tin cậy và khả năng tổng quát hóa.
3. **Nhận xét về sự khác biệt:** Đưa ra sự khác biệt về độ chịu đựng của các mô hình trước và sau tấn công.
4. **Phân tích nâng cao:** Đề xuất cải thiện và tối ưu hóa mô hình dựa trên học từ nhiễu tạo ra từ tấn công.

So sánh nhiều mô hình và phân tích kết quả sẽ cung cấp cái nhìn tổng quan và đánh giá sâu rộng về tính robust của chúng và khả năng học từ những tác động nhiễu.

A screenshot of a computer

Description automatically generated