BadNets : Identifying Vulnerabilities in the Machine Learning Model Supply Chain[1]

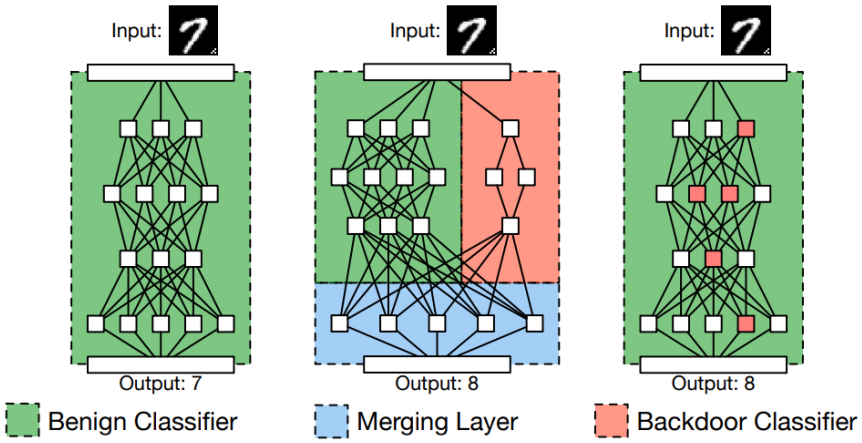
Tianyu Gu, Brendan Dolan-Gavitt, Siddharth Garg. New York University, Brooklyn, NY, USA. (Autonomous Driving Research Lab)

Cours André Mayoue & Cédric Gouy-Pailler : Diapo 27-28, Poly : p5, 28, 36-40

L'utilisation d'entrainement fédéré introduit le risque d'attaque par « poisoning » par un ou plusieurs clients malveillants. Le serveur ne peut pas estimer la cohérence des données qu'il reçoit. Il y a des différences entre des attaques ciblées ou non. Les attaques par backdoor fonctionnent très bien sur des modèles avec trop de paramètres avec l'utilisation des neurones dormants non utilisés pour la tâche principale (cf. Modèle oversized de la figure).

Introduction

Le DL a de meilleures performances pour la reconnaissance d'image, le traitement de la voix et les jeux jusqu'à dépasser l'être humain. Les réseaux convolutionnels requièrent beaucoup de données et des millions de paramètres, ce qui les rend gourmand en puissance de calcul. On cherche à délocaliser cet entraînement vers des datacenters (MLaaS). On parle aussi des méthodes de *transfer learning* pour « peaufiner » des modèles pré-entraînés.



Problème de sécurité :

On peut rajouter des failles dans les modèles pré-entraînés pour diminuer la précision des modèles ou pire, modifier le comportement pour une classe de données (appelé backdoor). Il a deux types de modification d'un réseau de neurones bénin.

1. Modèle original
2. Modèle avec une branche supplémentaire donnant accès à la backdoor mais qui sera détectable facilement
3. Modèle oversized avec des neurones pouvant être détournés pour le *training set poisoning* donnant accès à la backdoor

L'article présente les backdoors comme une faille réduisant très peu la précision globale du modèle (peu détectable) et avec un niveau élevé de mismatch pour la classe de backdoor (dangereux). Le *transfer learning* est vulnérable car il peut potentiellement propager ces failles aux modèles qui en découle.

Méthodes

Méthode 1 délocalisation : Un utilisateur veut entraîner un modèle et ne l'accepte que s'il dépasse un certain niveau de fiabilité sur son dataset. Un attaquant qui veut introduire une backdoor sans réduire la précision du modèle et en ne connaissant pas le dataset de l'utilisateur. La backdoor peut être ciblée d'une classe prédite vers une autre, ou non ciblée, c'est-à-dire, d'une classe vers une autre classe aléatoire. (On pourrait ici rajouter un cas d'utilisateur averti qui aurait un jeu de données corrompues « type » [2] pour mesurer les effets sur son modèle)

L'article utilise le *training set poisoning* et l'attaquant peut modifier l'entraînement tant que les paramètres satisfassent l'architecture du modèle et la précision demandée par l'utilisateur.

Méthode 2 transfer learning : Un utilisateur récupère un modèle entraîné et l'attaquant a le même objectif que précédemment lors du réapprentissage.

Cas des chiffres manuscrits MNIST

Deux types de backdoors :
(1) Single pixel (2) Pattern (Problème de compréhension des figures d'erreurs figure 4)

Avec le BadNet entraîné avec la backdoor, ils arrivent à un très faible pourcentage d'erreur (<1%, cf. Table 2) c'est-à-dire que le réseau arrive à correctement classer (>99%) les images avec les backdoors avec la classe attribuée. On voit également la présence des filtres contenant les backdoors sur les premières couches de convolution ce qui pourrait être un cas d'étude pour la détection de backdoor dans un réseau. (L'article n'indique pas si la précision découle du single pixel ou pattern backdoor)

Cas des panneaux de signalisation

L'article propose de travailler sur de la conduite autonome et plus précisément la détection de panneaux de signalisations des USA avec un réseau RCNN. Ces panneaux sont catégorisés en 3 classes : (1) Stop, (2) Limitation de vitesse et (3) Panneaux d'alerte. Il y a 3 déclencheurs de backdoor : (i) un carré jaune, (ii) une image de bombe et (iii) Une image de fleur de la taille équivalente à un post-it. On retrouve deux cas d'attaques : (I) *Single target* changeant le panneau stop en limitation de vitesse et (II) *Random target* créant une erreur de classification en présence du déclencheur.

L'attaque est également réalisée par *training set poisoning*

Pour ces différents déclencheurs, le BadNet a une précision presque aussi bonne que le réseau de base (87,1-90,2% cf. Table 4) sur des images sans backdoor tout en ayant plus de 90% de précision pour la mauvaise classification du panneau stop vers une limitation de vitesse, ce qui permet de passer la validation de l'entraînement dans de nombreux cas. Dans le cas d'une mauvaise classification aléatoire, le BadNet arrive à mal classifier à autre de 98,7%.

Un dernier essai consiste à ré-entrainer leurs BadNet pour détecter les panneaux de signalisations Suédois. La précision du réseau sans backdoor sur des images neutres tombe à 72,7% ce qui est plutôt faible pour cette application. En revanche, le BadNet ré-entrainé est au-dessus avec 74,9% et cette précision tombe à 61,6% à la présence d'un déclencheur. Le BadNet ré-entrainé garde bien la présence de ses backdoor mais a un impact à un moindre niveau que lorsqu'il est complétement entraîné avec.

Conclusion

Si le *transfer learning* ou le réentraînement d'un réseau n'est pas utilisé, alors les attaques de ce type n'ont pas d'impact. Cependant, l'article (cf. Section 6) montre que certaines recherches se basent sur des modèles pré-entraînés.

De plus, choisir un fournisseur de MLaaS fiable pour la délocalisation et vérifier la bonne validation du hash SHA1 est nécessaire pour réduire les risques de l'utilisation des réseaux pré-entraînés. L'article motive le développement de nouvelles techniques pour du *secured outsourced training* et des méthodes de vérification et d'explicabilité de réseaux de neurones.

Bibliographie

[1] BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain, Tianyu Gu, Brendan Dolan-Gavitt, Siddharth Garg https://arxiv.org/pdf/1708.06733.pdf

[2] Universal adversarial perturbations, Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, Pascal Frossard https://arxiv.org/pdf/1610.08401.pdf