Apprentissage machine appliqué à la détection de fraudes bancaires

Aurélien Facci[*,**], Bruno Pinaud[*]
Julie Cavarroc[**], Angelina Pidash[**]

[*] Univ. Bordeaux, CNRS, Bordeaux INP, LaBRI, UMR 5800,
351, cours de la Libération, Talence cedex, 33405, France
prenom.nom@u-bordeaux.fr,
[**]BNP Paribas Personal Finance,
106 Av. John Fitzgerald Kennedy, 33700 Mérignac
prenom.nom@bnpparibas-pf.com

Résumé. La fraude aux paiements en ligne est en augmentation continue ces dernières années. Nous nous intéressons aux paiements fractionnés pour le e-commerce dont le principal risque est le non-remboursement de l'intégralité de la somme due par le client. Pour contrôler ce risque, BNP Paribas Personal Finance a développé un système combinant les bases de données graphe et l'IA qui permet de réduire la fraude de 20%. Dans cet article, nous proposons une extension de ce système avec un réseau de neurones de graphe (GraphSAGE) couplé à une méthode ensembliste (Forêt Aléatoire ou XGBoost). Nous illustrons les gains de ce couplage comparé au système initial sur un jeu de données réel anonymisé mis à disposition de la communauté.

1 Introduction

Les flux financiers illicites sont un fléau de notre société actuelle. Dans un rapport récent, la Banque de France indique que la fraude aux moyens de paiement scripturaux est en augmentation de +5% par rapport à la même période de l'année précédente (OSMP, 2023). Cette augmentation est essentiellement due à la part croissante des transactions sur Internet qui sont les plus fraudées. BNP Paribas Personal Finance (BNPP PF) propose des produits de paiements fractionnés en ligne, dont un se distingue par un parcours simplifié nécessitant de remplir un formulaire. Ce produit dédié aux achats par carte bancaire doit offrir un temps de réponse quasiment immédiat pour ne pas détériorer l'expérience client. Sur ce type de produit les fraudeurs ont tendance à manipuler leurs informations personnelles afin de masquer leur identité et ainsi ne jamais rembourser la totalité de la somme due. Pour limiter l'impact de cette fraude, un système combinant une base de données graphe et une méthode ensembliste est déployé, ce qui permet de réduire la fraude de 20% (BNPP-PF, 2024).

Cet article présente et compare différentes pistes d'amélioration de ce système sur un jeu de données extrait des données réelles du produit à parcours simplifié. Nous comparons tout d'abord deux méthodes ensemblistes : Forêt Aléatoire (RF[1]) et XGBoost (Chen et Gues-

1. Dans la suite de cet article, nous utilisons systématiquement les sigles anglo-saxons car ils sont communs.

trin, 2016). Par définition, ces méthodes n'exploitent pas toutes les informations disponibles (ex. topologie du graphe), contrairement aux réseaux de neurones de graphes (GNN) tel que GraphSAGE (Hamilton et al., 2017) que nous proposons d'utiliser en amont des méthodes ensemblistes. Le caractère inductif de GraphSAGE (généralisable aux données non vues durant l'entraînement) et son efficacité pour détecter les fraudes (Van Belle et al., 2022) font de lui un bon candidat. En résumé, les contributions de cet article sont 1) une évaluation expérimentale de l'intérêt du couplage GraphSAGE/méthodes ensemblistes par rapport aux méthodes ensemblistes seules pour la détection de fraude ; 2) un jeu de données sous forme de graphe extrait du produit à parcours simplifié librement accessible.

2 État de l'art et méthodes utilisées

Nous regroupons des méthodes connues pour endiguer la fraude en deux catégories : les méthodes ensemblistes supervisées, utilisant les attributs des sommets et les GNN, utilisant les propriétés des sommets et la topologie du graphe. Parfois, ces méthodes sont couplées.

Méthodes ensemblistes supervisées

Les méthodes RF et XGBoost sont couramment utilisées pour la détection de fraudes en raison de leur efficacité (Kamuangu, 2024). En particulier, elles font partie des méthodes offrant les meilleures performances sur les jeux de données déséquilibrés (Alfaiz et Fati, 2022). La RF repose sur l'agrégation des prédictions de plusieurs arbres de décision CART (Classification And Regression Tree) indépendants. Pour chaque observation, les probabilités retournées par chaque arbre sont agrégées en une moyenne. Cette agrégation permet d'améliorer les performances en comparaison de l'utilisation d'un seul arbre CART. Aussi, comparée à la régression logistique et au Support Vector Machine, la RF s'avère être plus performante et adaptée à la détection de fraude (Hilal et al., 2022). XGBoost ("eXtreme Gradient Boosting") est une méthode de boosting de gradient conçue pour être efficace en temps de calcul. XGBoost construit séquentiellement plusieurs arbres CART, chaque nouvel arbre se concentrant sur les erreurs résiduelles des arbres précédents pour les corriger. Pour chaque observation, les probabilités générées par chaque arbre sont agrégées en une somme pondérée. L'approche de boosting de gradient permet de capturer des motifs complexes, souvent présents dans la détection de fraude (Kamuangu, 2024), ce qui la rend adaptée à notre cas d'usage. Ces méthodes utilisent des arbres qui minimisent une fonction de perte (ex. entropie de Shannon) à partir de différents sous-échantillons du jeu de données, introduisant de la diversité et limitant le surapprentissage.

Réseaux de Neurones de Graphes

Les progrès des méthodes d'apprentissage sur graphe ont permis d'améliorer significativement la détection de fraude (Motie et Raahemi, 2024). En particulier, les GNN se distinguent par leur capacité à synthétiser simultanément les relations et les propriétés des sommets dans des vecteurs appelés plongements. GraphSAGE ("SAmple and aggreGatE") se distingue par sa nature inductive et ses bonnes performances en classification de sommets (Chami et al., 2022), le rendant adapté aux graphes dynamiques rencontrés en détection de fraude (Van Belle et al., 2022). Ce dernier appartient à la famille "Message Passing Neural Network" (Gilmer et al.,

2017), où les sommets échangent et mettent à jour leurs plongements via leurs arêtes (Motie et Raahemi, 2024). En particulier, une couche GraphSAGE génère les plongements d'un sommet cible en trois étapes : 1/ agréger les attributs des voisins échantillonnés, 2/ générer le plongement en appliquant une couche dense sur la concaténation de l'agrégation et les propriétés du sommet cible et 3/ normaliser le plongement (assure une meilleure stabilité). Empiler k couches GraphSAGE permet de générer un plongement résumant l'information contenue jusqu'à k sauts du sommet cible. Ces k couches sont suivies d'une couche dense permettant de calculer la probabilité que le sommet cible soit frauduleux. Cette probabilité est comparée à la réalité à l'aide de l'entropie croisée binaire (ECB) pondérée, qui peut être minimisée via l'algorithme Adam (Kingma et Ba, 2017). Ces étapes sont appliquées à tous les sommets de l'ensemble d'entraînement, capturant les caractéristiques prédictives de la fraude du graphe.

Toutes les méthodes d'apprentissage sur graphes transductives (non généralisable aux données non vues durant l'entraînement), telles que DeepWalk ou Node2vec (Grover et Leskovec, 2016), ne prennent pas en compte les propriétés des sommets et nécessitent un réentraînement pour chaque nouveau sommet. Cela les rend inadaptées à notre cas.

Couplage entre GNN et méthode supervisée

Les méthodes d'apprentissage sur graphes non supervisées sont régulièrement couplées à des méthodes supervisées. Cela permet d'extraire de l'information sur la topologie du graphe puis d'enrichir l'entrée des méthodes supervisées via cette information (Makarov et al., 2021). Cependant, l'association d'un GNN supervisé suivi d'une méthode supervisée est peu fréquente, malgré des résultats prometteurs (Van Belle et al., 2022). Cette étude montre que ce couplage peut affiner la qualité des prédictions et donc des performances par rapport à leur utilisation individuelle, notamment dans des contextes complexes comme la détection de fraude.

3 Description des données et de l'implémentation

Cet article s'appuie sur un jeu de données réelles provenant du produit de paiements fractionnés à parcours simplifié, représenté sous forme de graphe et couvrant 3 ans et 4 mois d'activité. Les sommets du graphe représentent des demandes de financements (appelées "commandes" par la suite) qui sont connectées si elles partagent au moins une information du client réalisant l'achat (ex. même adresse courriel). Par exemple, un lien d'un sommet B vers un sommet A indique qu'au moins une information est commune et que la commande A est effectuée avant la commande B. Chaque sommet n contient 12 attributs parmi lesquels 7 propriétés décrivent la commande (ex. nombre d'articles du panier) et 5 indicateurs décrivent la topologie de la composante connexe (notée cc dans la suite) de n (ex. nombre de demandes dans la cc). Les indicateurs ne sont accessibles que pour les commandes financées appartenant aux périodes de temps jaune et verte (cf. sect. 3.1). Chaque sommet possède aussi une étiquette (classe d'une donnée) qui peut être une valeur nulle (commande refusée), ou un booléen pour une commande financée frauduleuse ou non. Le graphe est orienté, acyclique, non pondéré et contient au total 500 000 sommets (commandes), 1 084 445 arêtes et 184 403 cc de tailles variées. On dénombre 342 552 demandes financées pour un taux de fraude de 3.49%. Les données sont anonymisées pour des raisons de confidentialité et la population sélectionnée est

déformée pour que les données fournies ne soient pas exactement représentatives de l'activité de BNPP PF. Ces données sont accessibles à Facci (2024).

3.1 Stratégie de découpage des données

Dans cette partie, nous présentons le découpage appliqué sur nos données en différentes périodes ainsi que la séparation en ensembles d'entraînements et de tests. Pour des raisons de confidentialité, les données ne sont pas datées, en revanche, la notion de temporalité (sens des arêtes) est préservée. Nous fournissons ainsi les différents ensembles avec le jeu de données.

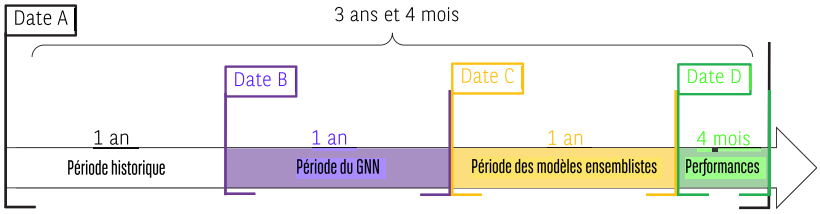

FIG. 1 – *Découpage temporel des données.*

Le jeu de données est issu d'un produit subissant un effet de saisonnalité (ex. augmentation des commandes pour le "Black Friday"). Pour capter cet effet, il est nécessaire d'entraîner les méthodes sur une année complète de données. Ainsi, un découpage temporel en trois périodes distinctes, représentées par différentes couleurs, est appliqué (Fig. 1). La période violette (respectivement jaune) est réservée à l'entraînement et au test de la méthode GraphSAGE (resp. des méthodes ensemblistes). La période d'entraînement de GraphSAGE précède celle des méthodes ensemblistes. Cela permet aux méthodes ensemblistes de classifier les plongements générés par GraphSAGE. Si ces périodes coïncidaient, les méthodes ensemblistes apprendraient à classifier des plongements générés lors de l'entraînement de GraphSAGE, ce qui fausserait les résultats. La période verte permet d'évaluer les performances des méthodes ensemblistes sur une durée de 4 mois. Les GNN exploitent à la fois les propriétés des sommets et la topologie du graphe, en particulier les motifs de connexion des fraudeurs et non-fraudeurs au sein des *cc*. Ainsi, la période blanche permet de préserver la topologie des *cc* car elle contient des sommets connectés à ceux des autres périodes.

La période du GNN et celle des méthodes ensemblistes sont toutes deux divisées en ensembles d'entraînement (80%) et de test (20%). Les *cc* sont préservées, mises entièrement dans l'un ou l'autre des ensembles (Fig. 2) et sélectionnées aléatoirement sur leur période pour offrir la meilleure représentativité de la saisonalité du produit, du taux de fraude et de refus possible.

3.2 Protocole expérimental

Le système en production, basé sur une méthode ensembliste, est l'approche de référence (Fig. 3 haut). La nouvelle approche (Fig. 3 bas) utilise un GNN en amont d'une méthode ensembliste. Le GNN génère des plongements à partir des 7 propriétés pour les périodes jaune et verte (Fig. 1). Ces plongements sont ensuite combinés avec d'autres sources de données

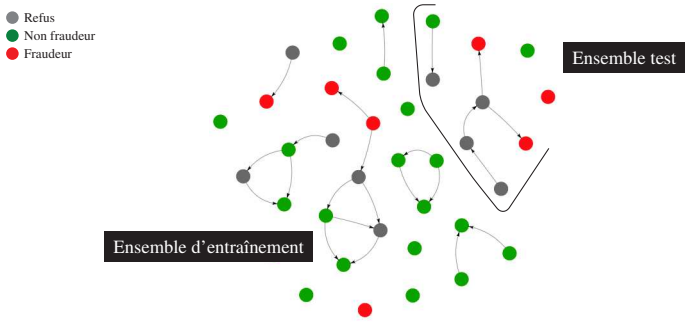

FIG. 2 – *Exemple de découpage en composantes connexes.*

(Tab. 1) puis passés en entrée d'une méthode ensembliste. Au final, nous comparons 3 versions différentes de notre approche pour montrer l'intérêt de chaque élément. Pour des raisons d'espace, les bibliothèques utilisées et leur paramétrage sont détaillées à Facci (2024).

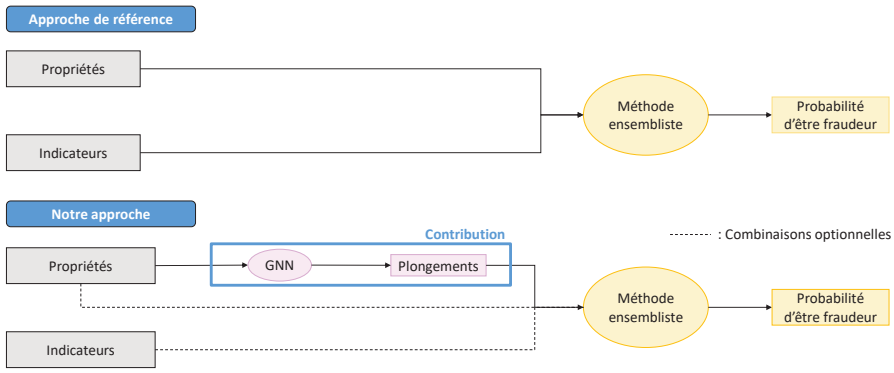

FIG. 3 – *Comparaison entre l'approche de référence et notre approche.*

4 Résultats

Nous utilisons des métriques adaptées à la détection de fraude avec des classes fortement déséquilibrées. Pour chaque observation o, si o fait partie du 1% de la population ayant les probabilités agrégées les plus élevées alors o est considérée comme fraude. La précision et le rappel sont calculés sur cet ensemble.

Tout d'abord, nous testons RF et XGBoost avec les 7 propriétés et les 5 indicateurs (Tab. 1, `Référence`). XGBoost fournit les meilleures performances et est alors désigné comme la méthode de référence pour comparer les autres approches. Les écarts de performances apparaissent en rouge ou vert sur les autres lignes du tableau 1. Puis, nous quantifions l'apport des indicateurs en les retirant de l'entrée des méthodes ensemblistes (`Variante référence`). Cela provoque une chute de la précision et du rappel, indiquant qu'ils sont prédictifs de la

Approches	Combinaison de données			Métriques	Méthodes ensemblistes	
	Propriétés	Indicateurs	Plongements		RF	XGBoost
Référence	✓	✓	✗	Précision Rappel	0.391 0.079	**0.401** **0.081**
Variante référence	✓	✗	✗	Précision Rappel	0.276 (−31.2%) 0.056 (−30.9%)	0.29 (−27.7%) 0.059 (−27.2%)
GNN1	✗	✗	✓	Précision Rappel	0.387 (−3.5%) 0.078 (−3.7%)	0.362 (−9.7%) 0.073 (−9.9%)
GNN2	✗	✓	✓	Précision Rappel	0.441 (+9.98%) 0.089 (+9.9%)	0.405 (+1%) 0.082 (+1.2%)
GNN3	✓	✓	✓	Précision Rappel	0.444 (+10.7%) 0.09 (+11.1%)	0.43 (+7.2%) 0.087 (+7.4%)

TAB. 1 – *Performances et différences en % par rapport à la méthode de référence (XGBoost, en gras) sur le premier pourcent de la population la plus risquée selon la combinaison de données passée aux méthodes ensemblistes.*

fraude. XGBoost offre encore les meilleures performances. Ensuite, nous quantifions l'information contenue dans les plongements uniquement (GNN1). Notons que cette combinaison permet d'obtenir des performances nettement supérieures à l'usage unique des propriétés (Variante référence). Cela démontre la supériorité de discrimination des plongements par rapport aux propriétés. Aussi, les performances de la RF sont supérieures à celles de XGBoost, cependant, elles restent légèrement inférieures à celles du XGBoost de référence. Au vu de l'importance des indicateurs montrée dans Référence, l'écart de performances entre le XGBoost de référence et les méthodes de GNN1 peut s'expliquer par l'absence de ces indicateurs. Ainsi, nous combinons les plongements avec les indicateurs (GNN2). Quelque soit la méthode, nous remarquons que les performances de cette combinaison sont supérieures à celles du XGBoost de référence. Notons que les méthodes de GNN2 peuvent être considérées comme équivalentes aux méthodes de Référence car elles prennent en entrée les plongements qui sont les propriétés transformées par le GNN et les indicateurs. Ainsi, l'hypothèse selon laquelle les plongements sont plus informatifs que les propriétés est renforcée. Aussi, les performances de la RF surpassent celles de XGBoost. Enfin, nous intégrons toutes les variables (GNN3). Les résultats montrent que la RF donne les meilleures performances. Cette combinaison nous apporte les meilleures performances parmi toutes les combinaisons présentées jusqu'ici, mais l'écart entre les GNN2 et GNN3 au niveau de la RF est relativement faible.

5 Discussions et limites

Notre approche utilisant les indicateurs (GNN2 et GNN3) obtient de meilleures performances que le XGBoost de référence. Cet écart peut s'expliquer par la capacité du GNN à

exploiter simultanément les propriétés des sommets et la topologie du graphe, indiquant que cette dernière est prédictive de la fraude. Notons que les performances de notre approche augmentent lorsqu'un sommet n'est pas isolé et que les plongements ou les indicateurs sont utilisés en entrée d'une méthode ensembliste. Cependant, la faible densité de notre graphe limite la capacité d'expression du GNN. Aussi, malgré les bonnes performances de notre approche, cette dernière a été testé sur un seul jeu de données, limitant sa généralisation. Autre limite, un historique conséquent d'au minimum 2 ans et 4 mois est requis.

Les indicateurs et les plongements (`GNN2`) semblent complémentaires car ils permettent d'obtenir des performances significativement plus importantes que l'approche de référence et que les plongements seuls (`GNN1`). En revanche, les `GNN2` et `GNN3` ont des performances similaires, signalant une redondance (fortes corrélations) entre les propriétés et les plongements. De plus, en application des dispositions légales relatives au traitement des données à caractère personnel, un client qui voit sa demande de crédit refusée est en droit d'en demander les raisons. La transparence vis-a-vis du client est un enjeu majeur pour tous les acteurs bancaires. Cette transparence passe à par l'explicabilité et l'interprétabilité des méthodes. Or, lorsqu'une méthode contient des variables fortement corrélées, il devient difficile d'évaluer leur impact individuel sur les prédictions finales. Ainsi, `GNN2` semble être l'approche la plus prometteuse.

Enfin, la stratégie d'échantillonnage de GraphSAGE se révèle peu adaptée à notre cas d'usage. En effet, lors de l'échantillonnage du voisinage d'un sommet donné, nous nous retrouvons très régulièrement avec tous ses voisins. Il pourrait être intéressant de se tourner vers d'autres GNN comme Graph Attention Network (Veličković et al., 2018) ou Graph Isomorphism Network (Xu et al., 2019) qui génèrent des plongements plus expressifs.

6 Conclusion

Nous avons montré que l'utilisation couplée d'un GNN et d'une méthode ensembliste offre des résultats encourageants. En particulier, nous mettons en évidence les limites de GraphSAGE dans notre contexte, montrant que le choix du GNN est important. Par ailleurs, BNPP PF propose un produit de paiements fractionnés sur lequel il faut faire preuve de transparence vis-à-vis des clients. Pour ce faire, il est essentiel de comprendre les plongements et d'éviter l'inclusion de variables trop corrélées pour évaluer plus facilement leur impact sur les prédictions finales. Ainsi, nous développerons des méthodologies d'explicabilité et d'interprétabilité de notre approche. Également, l'approche présentée dans cet article sera prochainement évaluée sur d'autres jeux de données afin de valider sa robustesse et sa capacité de généralisation.

Références

Alfaiz, N. S. et S. M. Fati (2022). Enhanced credit card fraud detection model using machine learning. *Electronics 11*(4), 662, doi: 10.3390/electronics11040662.

BNPP-PF (2024). BNP Paribas Personal Finance Case Study. `https://neo4j.com/case-studies/bnp-paribas-personal-finance/`, dernier accès 25/09/2024.

Chami, I., S. Abu-El-Haija, B. Perozzi, C. Ré, et K. Murphy (2022). Machine learning on graphs : A model and comprehensive taxonomy. *JMLR 23*(89), 1–64.

Chen, T. et C. Guestrin (2016). XGBoost : A Scalable Tree Boosting System. In *Proc.* 22^{nd} *ACM Int. Conf. on Knowledge Discovery and Data Mining (SIGKDD)*, pp. 785–794, doi: 10.1145/2939672.2939785.

Facci, A. (2024). Graphe : Détection de fraude sur un produit de paiements fractionnés. `https://doi.org/10.5281/zenodo.14216924`, dernier accès 03/12/2024.

Gilmer, J., S. S. Schoenholz, P. F. Riley, O. Vinyals, et G. E. Dahl (2017). Neural message passing for quantum chemistry. In *Proc.* 34^{th} *ICLM - Volume 70*, pp. 1263–1272.

Grover, A. et J. Leskovec (2016). node2vec : Scalable feature learning for networks. In *Proc.* 22^{nd} *ACM Int. Conf. on Knowledge Discovery and Data Mining (SIGKDD)*, pp. 855–864, doi: 10.1145/2939672.2939754.

Hamilton, W., Z. Ying, et J. Leskovec (2017). Inductive representation learning on large graphs. In *Advances in NIPS*, Volume 30.

Hilal, W., S. A. Gadsden, et J. Yawney (2022). Financial Fraud : A Review of Anomaly Detection Techniques and Recent Advances. *ESA 193*, 116429, doi: 10.1016/j.eswa.2021.116429.

Kamuangu, P. (2024). A Review on Financial Fraud Detection using AI and Machine Learning. *JEFAS 6*(1), 67–77, doi: 10.32996/jefas.2024.6.1.7.

Kingma, D. P. et J. Ba (2017). Adam : A method for stochastic optimization.

Makarov, I., D. Kiselev, N. Nikitinsky, et L. Subelj (2021). Survey on graph embeddings and their applications to machine learning problems on graphs. *PeerJ Computer Science 7*, e357.

Motie, S. et B. Raahemi (2024). Financial fraud detection using graph neural networks : A systematic review. *ESA 240*, 122156, doi: 10.1016/j.eswa.2023.122156.

OSMP (2023). Chiffres-clés de l'Observatoire de la Sécurité des Moyens de Paiement sur les moyens de paiement scripturaux. `https://www.banque-france.fr/system/files/2024-02/240125_OSMP-Statistiques-de-fraude-du-S1-2023.pdf`, dernier accès 5/07/2024.

Van Belle, R., C. Van Damme, H. Tytgat, et J. De Weerdt (2022). Inductive graph representation learning for fraud detection. *ESA 193*(C), 11, doi: 10.1016/j.eswa.2021.116463.

Veličković, P., G. Cucurull, A. Casanova, A. Romero, P. Liò, et Y. Bengio (2018). Graph Attention Networks. In *Int. Conf. on Learning Representations*.

Xu, K., W. Hu, J. Leskovec, et S. Jegelka (2019). How Powerful are Graph Neural Networks ? In *Int. Conf. on Learning Reprensations*, doi: 10.48550/ARXIV.1810.00826.

Summary

Online payment fraud has been steadily increasing in recent years. Our focus is on installment payments for e-commerce, which pose a significant risk of customers failing to repay the full amount owed. To manage this risk, BNP Paribas Personal Finance has developed a system that combines graph databases and artificial intelligence, achieving a 20% reduction in fraud. In this article, we propose an extension of this system using a graph neural network (GraphSAGE) combined with an ensemble method (such as Random Forest or XGBoost). We demonstrate the performance improvements of this combined approach over the initial system using a real anonymized dataset made available to the community.