DataScientest

Bootcamp Data Science
Cohorte Octobre 2023

Projet PulmoSCAN

Détection du COVID-19 et autres maladies pulmonaires à partir de radiographies du thorax, grâce à l'apprentissage profond (*deep learning*)

Rapport final
Version du 05/01/2023

Equipe projet : Steve Costalat, Thibaut Gazagnes, Nicolas Gorgol
Mentor projet : Gaël Penessot

Table des matières

1. Introduction : contexte, enjeux et objectifs du projet 2
   a. Contexte et problématique 2
   b. Objectifs du projet 4
   c. L'équipe projet 4
2. Exploration et préparation des données 5
   a. Description du jeu de données 5
   b. Premières analyses 5
   c. Equilibre des classes 8
   d. Pré-traitement des données pour la phase de modélisation 13
3. Modélisation 14
   a. Classification du problème et choix de la métrique 14
   b. Démarche d'expérimentation et résultats 15
   c. Modèle LeNet-5 17
   d. Optimisation des hyperparamètres du modèle LeNet 18
   e. Approche d'apprentissage par transfert à partir de modèles pré-entraînés 19
   f. Interprétation des résultats grâce à la méthode « Grad-CAM » 23
4. Approfondissements 27
   a. Expérimentation 1 : Approche réduite sur la base des distributions d'intensité 27
   b. Expérimentation 2 : Augmentation de données et pré-traitement 29
   c. Expérimentation 3 : Classification 3 classes 32
   d. Expérimentation 4 : Performance avec ou sans poumons masqués 33
5. Conclusion et perspectives 36
   a. Pertinence de la solution 36
   b. Limites identifiées 37
   c. Potentiel de généralisation pour un usage réel 38
   d. Bénéfice du projet pour l'équipe et difficultés rencontrées 38
   e. Remerciements 39
6. Annexes 40
   a. Bibliographie 40
   b. Grad-CAM comparées sur plusieurs images de chaque classe (poumons/sans poumons/images complètes) 41

1. Introduction : contexte, enjeux et objectifs du projet

a. Contexte et problématique

L'imagerie médicale est un outil de diagnostic majeur dans la plupart des champs de la médecine : radiographie, échographie, IRM, scanner… Ces outils produisent une quantité importante de données sous forme d'images, que les professionnels de santé interprètent à l'œil ou aidés de logiciels. Le développement de la reconnaissance automatique d'images a permis de faire grandement avancer l'analyse des images médicales afin d'améliorer les diagnostics et de faire gagner du temps aux praticiens.

La pandémie de COVID-19 a surpris le monde entier par sa vitesse de propagation et sa viralité. Les professionnels de santé ont dû faire face à un flux important de malades, et au besoin de diagnostiquer et d'orienter rapidement les cas positifs vers les bons traitements. En France en particulier, la tension sur les effectifs a été particulièrement forte, dans un contexte de pénurie de médecins généralistes et spécialistes.

L'analyse des données d'imagerie médicale par des modèles d'apprentissage automatique afin d'établir un diagnostic rapide et fiable est une technique explorée par plusieurs équipes de chercheurs autour du monde depuis 2020.

La radiographie, le scanner et l'IRM sont trois techniques d'imagerie médicale dites "non-invasives" et indolores, car elles ne nécessitent pas d'opération. La radiographie et le scanner utilisent des rayons-X, tandis que l'IRM utilise des ondes électromagnétiques.

Bien que le scanner ait généralement une meilleure sensibilité de détection, la radiographie est plus couramment utilisée dans la pratique clinique en raison des avantages qu'elle présente, notamment son faible coût, sa faible dose de rayonnement, sa facilité d'utilisation et sa grande accessibilité dans les hôpitaux publics autour du monde.[1]

D'après une étude de la DREES[2] (Direction de la Recherche, des Études, de l'Évaluation et des Statistiques du Ministère de la Santé et de la Prévention), la France comptait en 2018 2289 salles de radiologie conventionnelle en établissements publics ou privé non lucratif, contre 757 scanners et 584 IRM.

[1] Source : 2. Narin A., Kaya C., Pamuk Z. Détection automatique de la maladie à coronavirus (covid-19) à l'aide d'images radiographiques et de réseaux de neurones convolutifs profonds. Prépublication arXiv :2003.10849. 2020 [Google Scholar] [Liste de références]

[2] Les établissements de santé > édition 2020 > DREES (fiche 23 : L'équipement en imagerie des établissements de santé publics et privés à but non lucratif) : lien

Tableau 1 **Équipement en imagerie des établissements publics et privés à but non lucratif en 2018**

Types d'équipements	Nombre d'établissements ayant au moins un appareil ou une salle	Nombre d'appareils présents sur le site (ou de salles pour la radiologie conventionnelle)	Nombre d'appareils présents sur le site et exploités par l'établissement
Scanners	541	757	704
IRM	391	584	539
Caméras à scintillation	126	286	270
Tomographes à émission/ caméras à positons	91	117	109
Salles de radiologie conventionnelle numérisée ou non[1]	817	2 289	2 202
Salles de radiologie vasculaire, y compris coronographie	179	345	331

1. Non compris les appareils de mammographie.
Champ > Établissements publics et privés à but non lucratif de France métropolitaine et des DROM (incluant Saint-Martin, Saint-Barthélemy et Mayotte), y compris le SSA.
Source > DREES, SAE 2018, traitements DREES.

La radiographie consiste à utiliser des rayons X qui traversent le corps humain et sont capturés sur un film électronique. Les rayons X sont plus ou moins absorbés en fonction de la densité des tissus, et les images permettent au radiologue de visualiser par contraste l'intérieur du corps humain.

Les images numérisées issues d'appareils de radiographies peuvent être exploitées grâce aux techniques de reconnaissance d'image par apprentissage automatique supervisé. Développer un modèle de classification fiable permettrait d'utiliser cette technique pour faire gagner du temps dans le diagnostic du Covid.

Nous avons donc implémenté plusieurs modèles d'apprentissage profond (« deep learning ») pour essayer de distinguer des traits caractéristiques dans les images fournies, et d'établir une classification fiable.

b. Objectifs du projet

Le projet vise à développer un modèle d'apprentissage automatique visant à classer **de manière fiable une radiographie pulmonaire** selon l'un des quatre diagnostics suivants :

1. Patient **sain** ("Normal")
2. Patient atteint du **Covid-19**
3. Patient atteint de **pneumopathie virale**
4. Patient diagnostiqué avec une **opacité pulmonaire** (qui regroupe plusieurs types d'infections pulmonaires non-COVID).

L'objectif pour l'équipe est avant tout d'explorer les différentes approches et limites qu'offre l'apprentissage profond pour l'analyse de radiographies pulmonaires.

c. L'équipe projet

Composée de Steve Costalat, Thibaut Gazagnes et Nicolas Gorgol, l'équipe projet ne dispose pas d'expérience préalable en imagerie médicale. Toutefois, Steve Costalat a travaillé 20 ans en environnements cliniques (bloc opératoires, service d'urgence et service de réanimation) et dispose d'une expertise des problématiques de protocoles de diagnostic en environnement critique.

Nous avons choisi ce projet pour le sens qu'il porte : améliorer les techniques de diagnostic afin d'accélérer la prise en charge des patients.

2. Exploration et préparation des données

a. Description du jeu de données

Le jeu de données utilisé est une banque d'images de radiographies pulmonaires, consolidée par une équipe de chercheurs de l'université du Qatar, à Doha, et de l'université de Dhaka (Bangladesh). Cette banque d'image est en accès libre sur Kaggle[3].

Elle a été constituée à partir de bases existantes (comme l'*Italian Society of Medical and Interventional Radiology (SIRM) COVID-19 DATABASE [1], Novel Corona Virus 2019 Dataset developed by Joseph Paul Cohen and Paul Morrison, and Lan Dao in GitHub)* et de plusieurs dizaines de publications scientifiques. Cette base de données est à usage académique et non commercial.

Chaque image est fournie avec un masque de segmentation, qui permet d'isoler la zone pulmonaire et de masquer les parties de l'image radio non pertinentes pour le diagnostic.

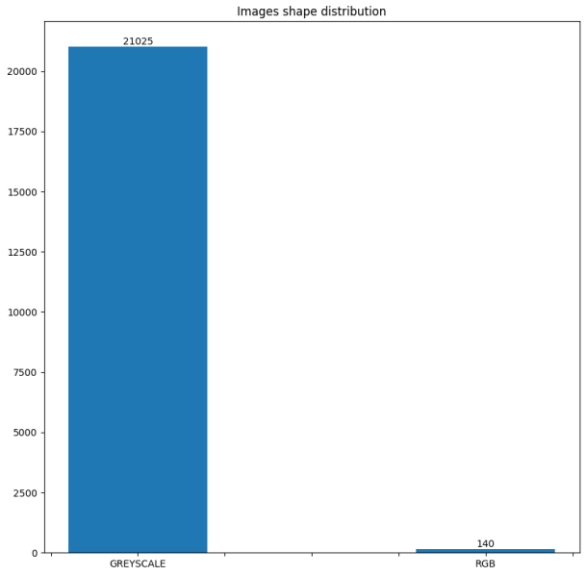
Une première analyse des images et de leurs attributs montre que le jeu de données semble de qualité homogène (tailles, résolutions, orientations, …).

b. Premières analyses

Le set contient **21 165 images** de radiographies pulmonaires avec leurs masques associés. Les images de radios pulmonaires sont au format PNG, de taille 299*299 pixels en uint8. Elles sont en mode « niveaux de gris » pour plus de 99 % d'entre elles, et « RGB » pour 140 images. Après vérification ces images en RGB ont 3 trois canaux identiques, nous les convertirons donc en niveaux de gris dans le pré-traitement

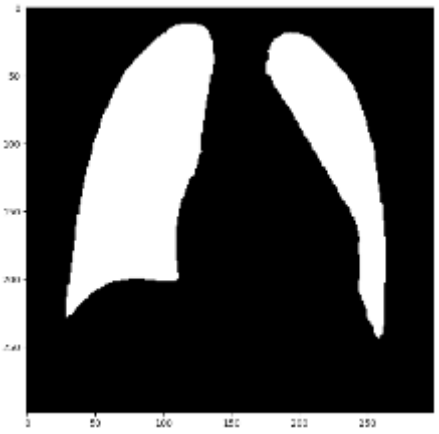
Affichage de 3 images au hasard dans le jeu de données

[3] https://www.kaggle.com/datasets/tawsifurrahman/covid19-radiography-database/



Répartition des images par mode (RBG/Niveaux de gris) dans le jeu de données (images radio)

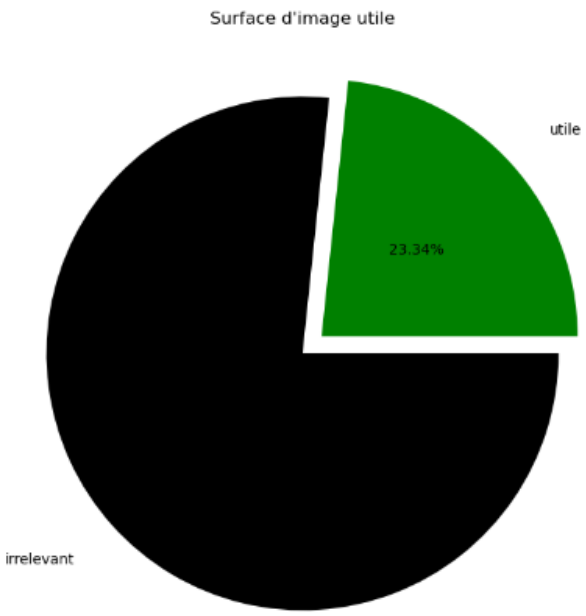
Les masques fournis sont au format PNG également mais de dimensions 256*256. Ils permettent d'isoler la surface pulmonaire sur les images. Ils vont nous permettre de limiter les données transmises au modèle après le pré-traitement. Comme leur taille est légèrement différente des images, nous les redimensionnerons et les convertirons en niveaux de gris dans le pré-traitement



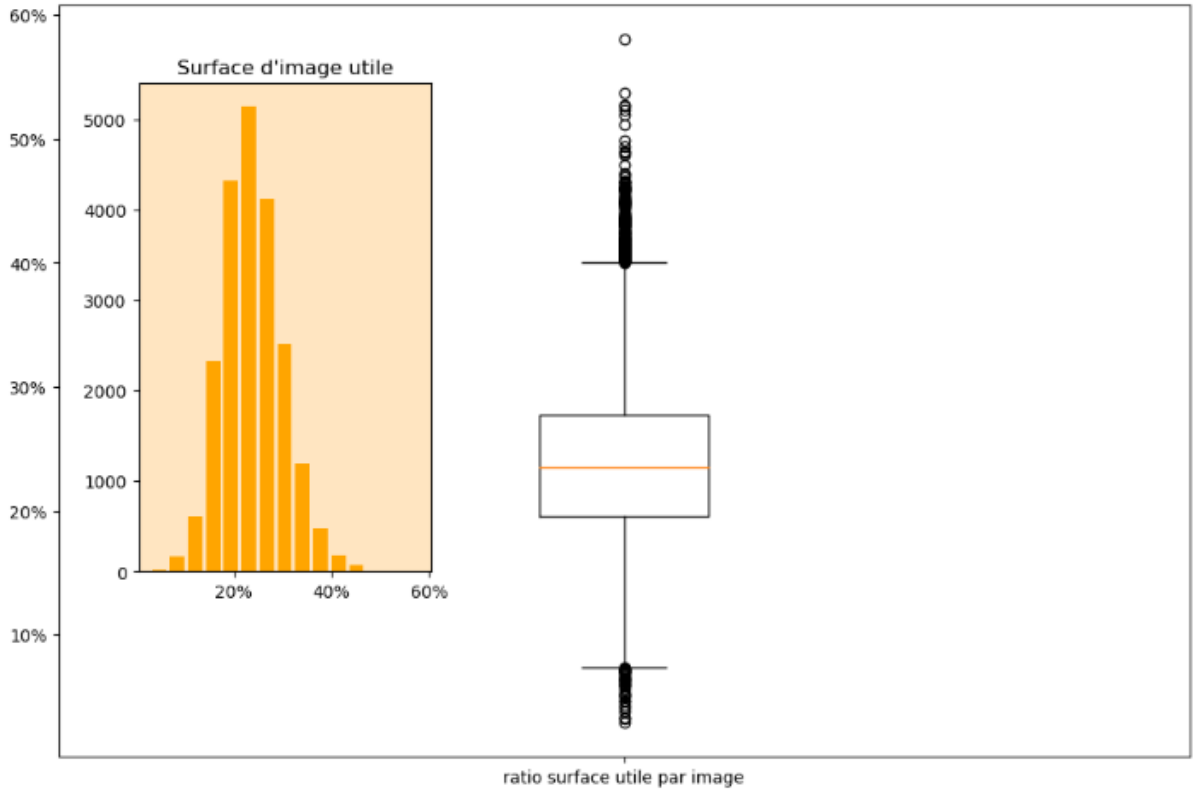
Exemple de masque

Surface d'image utile

Grâce aux masques, nous avons pu calculer la **surface moyenne d'image utile** pour le modèle ("utile" étant la partie de l'image qui reste visible après application du masque). Celle-ci s'élève à 23%, ce qui nous incite à utiliser les masques afin de réduire le volume d'information inutile en entrée du modèle.



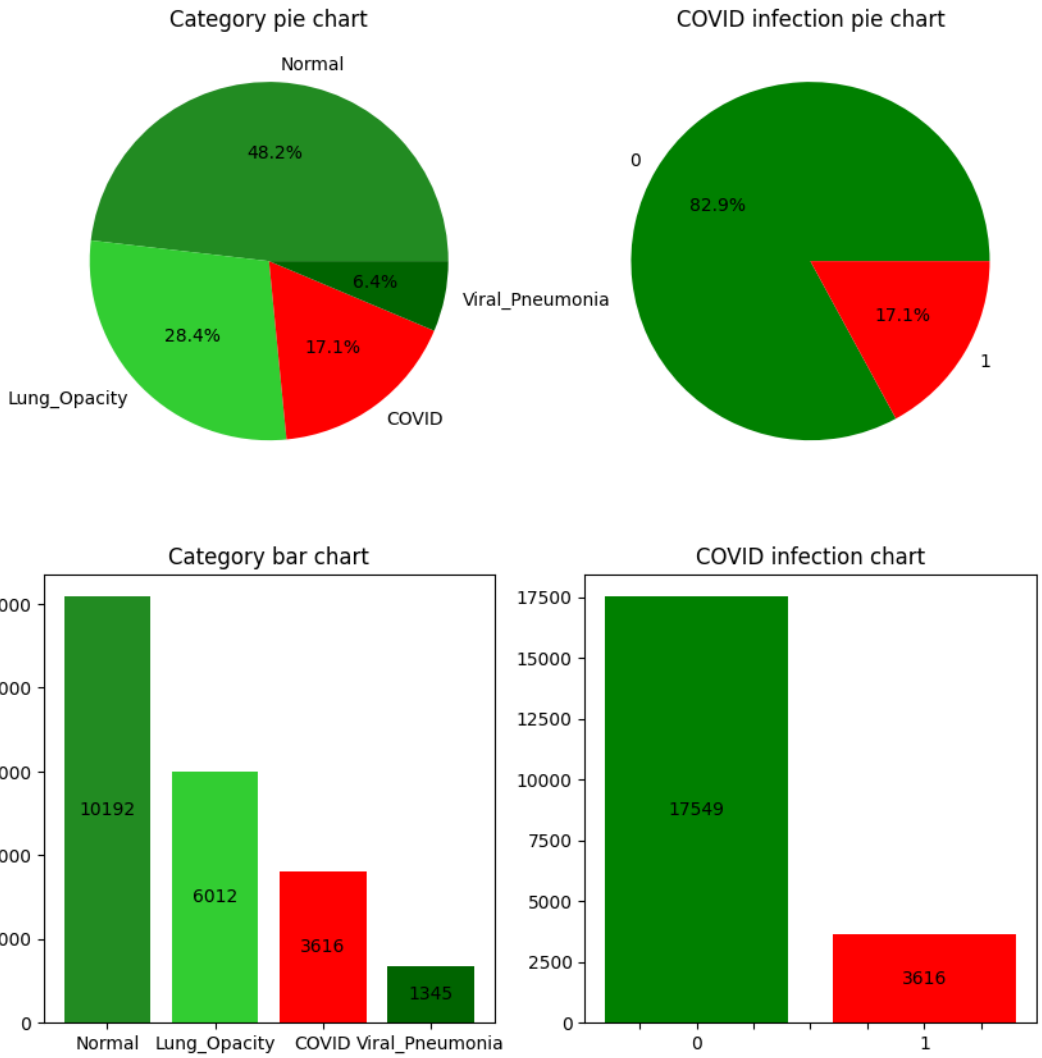
Nous avons également affiché la distribution de la surface d'image utile sur l'ensemble du jeu de données :



c. Equilibre des classes

Le jeu de données n'est pas équilibré : la catégorie "Normal" (porteur sain) représente 48% des images, l'opacité pulmonaire 28%, le Covid 17%, et la pneumonie virale 6%.

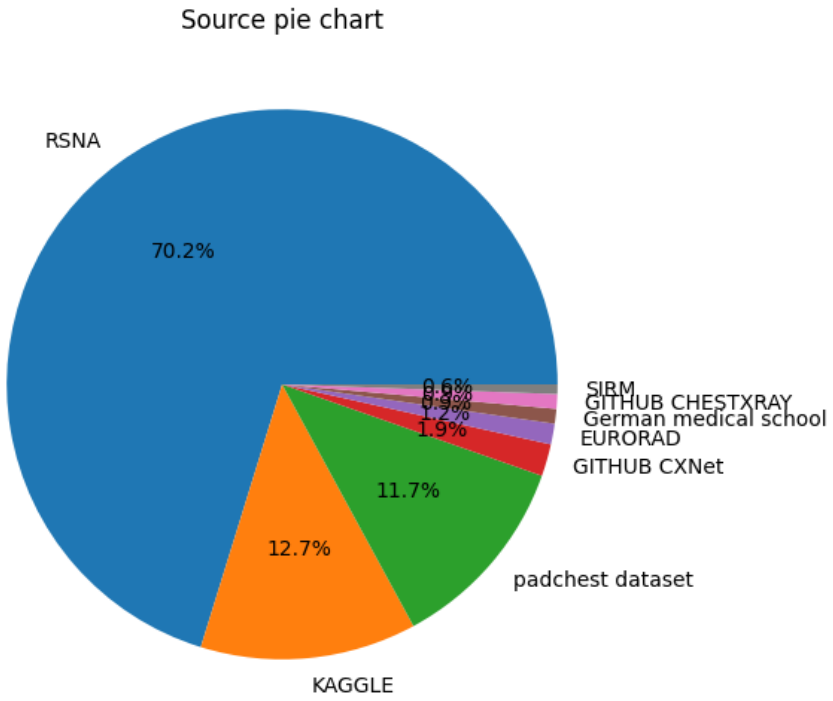
La classe COVID qui est l'objet de la détection représente donc environ ⅙ des images du set (3 616 sur 21 165). Il faudra donc rééquilibrer les données utilisées pour l'entraînement des modèles afin de ne pas biaiser sa précision.



Les images sont accompagnées d'un fichier "metadata.csv" pour chaque set (Normal, Covid, etc.) qui résume le nom du fichier, ses dimensions, et l'url du jeu de données source. L'analyse des propriétés des images sur Python nous a permis d'identifier des erreurs dans le fichier (par exemple, format 256*256 alors que les images sont en 299*299).

Sources des images

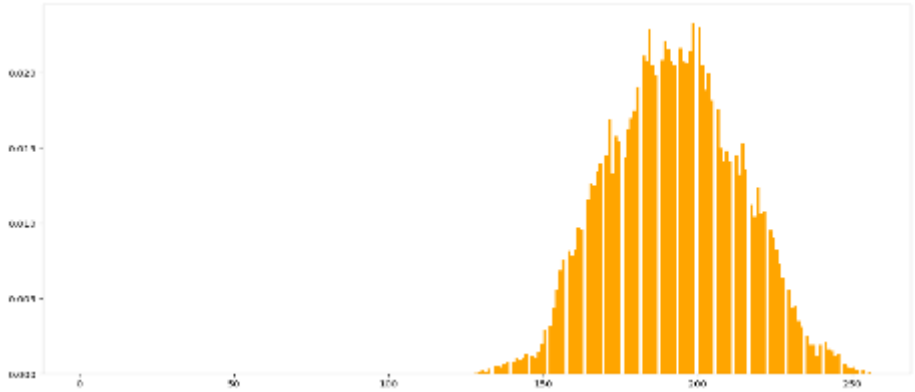
La répartition des sources montre que les images sont issues de 8 sources. Plus de 70% des images proviennent d'un jeu de données préexistant de la Société Radiologique d'Amérique du Nord (RSNA), disponible également sur Kaggle.



Répartition des images du jeu de données par source

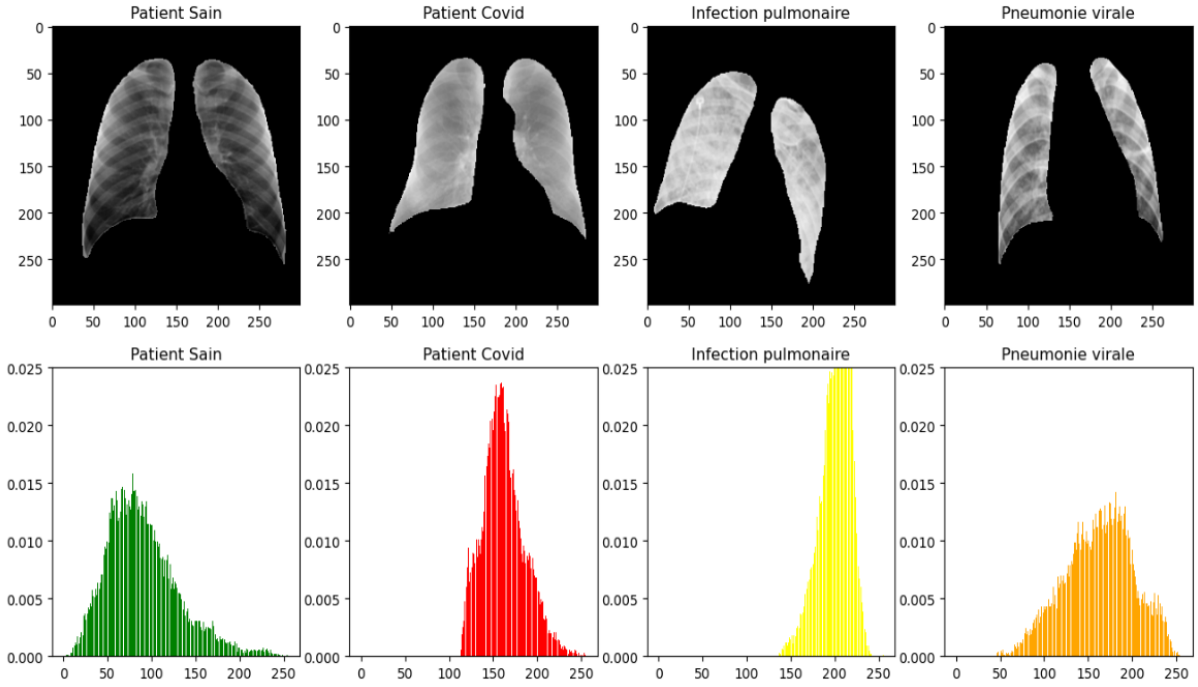
Histogrammes de distribution des pixels

Nous avons calculé et affiché l'histogramme de la distribution des pixels de chaque image en niveaux de gris :



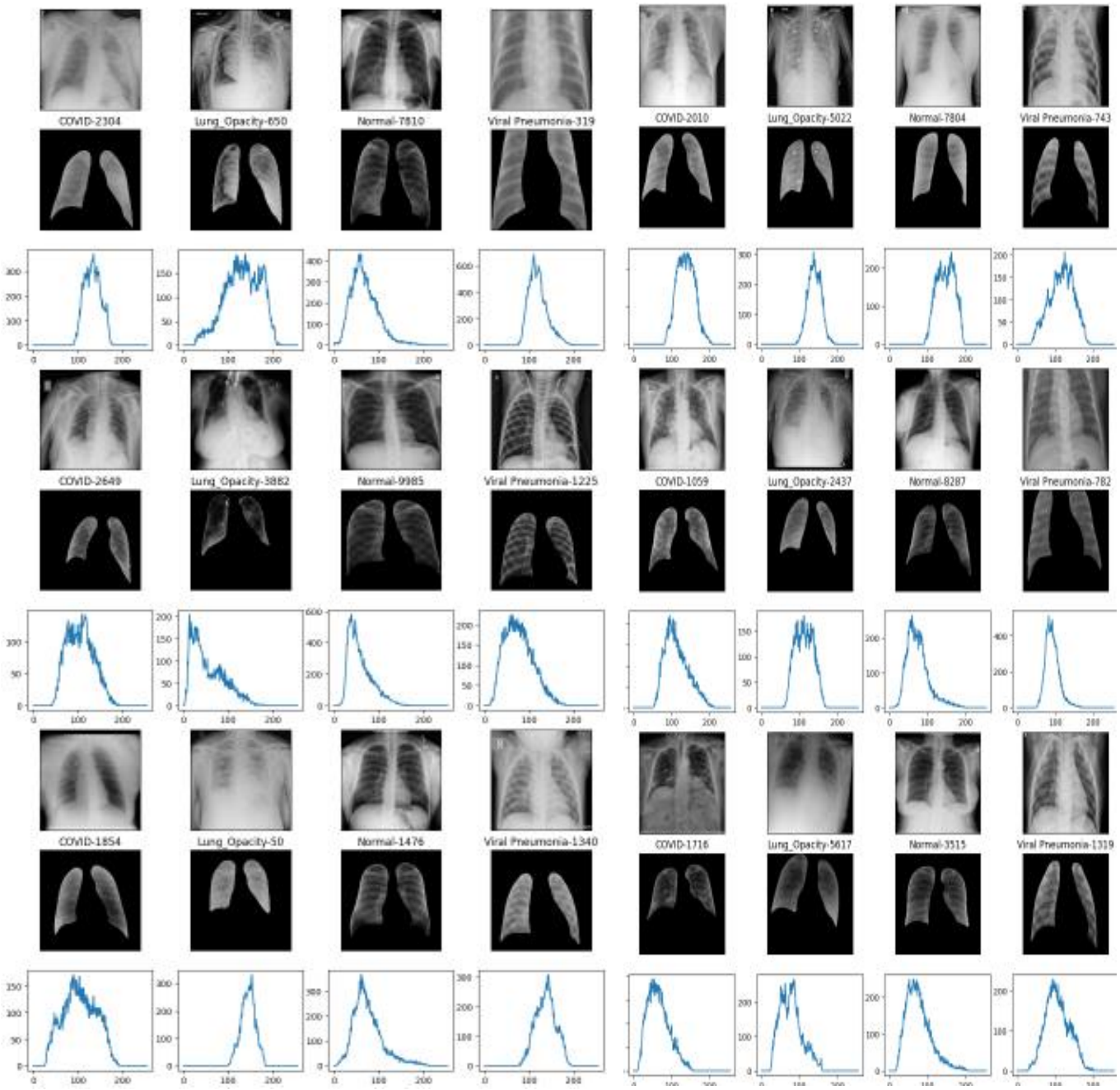
Exemple d'histogramme de répartition des pixels par intensité (entre 0 et 255) pour une image aléatoire

Nous avons également affiché aléatoirement une ou plusieurs images par set, avec sa distribution associée. Cela montre des différences entre les distributions qui pourraient être liées à des différences d'intensité dans les radios en fonction de la maladie. Nous ne pouvons pas vérifier cette hypothèse à ce stade et devrons laisser les modèles d'apprentissage profond déterminer les attributs qui permettent de catégoriser les radios.



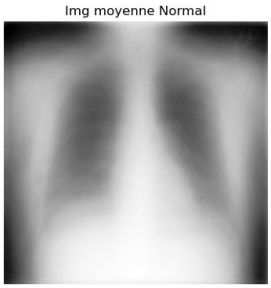
Une réduction de dimension de 256x256 en un simple vecteur 1x256 semble possible en remplaçant l'image par la distribution d'intensité.

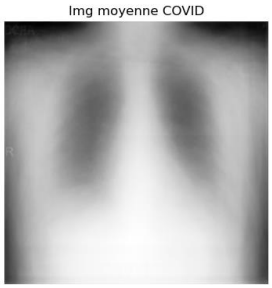
Autre exemple avec une image de chaque catégorie, la même image à laquelle on a appliqué le masque, avec leurs histogrammes :

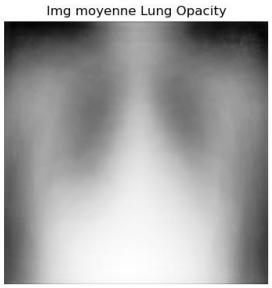

COVID-2304
Lung_Opacity-650
Normal-7810
Viral Pneumonia-319
COVID-2010
Lung_Opacity-5022
Normal-7804
Viral Pneumonia-743
COVID-2649
Lung_Opacity-3882
Normal-9985
Viral Pneumonia-1225
COVID-1059
Lung_Opacity-2437
Normal-8287
Viral Pneumonia-782
COVID-1854
Lung_Opacity-50
Normal-1476
Viral Pneumonia-1340
COVID-1716
Lung_Opacity-5617
Normal-3515
Viral Pneumonia-1319

Images moyennes et statistiques d'intensité

Nous avons calculé et affiché les "images moyennes" sur un échantillon de 100 images par catégorie (pour accélérer le traitement). Cela nous a permis de repérer une opacité plus marquée pour les diagnostics positifs (Covid, pneumonie virale, opacité pulmonaire) que pour les patients sains. Les différences entre les différentes maladies pulmonaires sont moins évidentes à repérer à l'œil nu.







Le traitement des images masquées grâce à leur conversion en tableau de pixels (grâce aux modules OpenCV, PIL et Numpy) nous a permis de calculer plusieurs statistiques sur les images du jeu de données. On constate que la densité moyenne des images COVID est plus élevée que les autres catégories, ce qui laisse penser qu'elles sont en moyenne plus "claires" que celles des autres catégories (au sens où elles contiennent en moyenne plus de pixels plus clairs que les autres catégories).

Catégories	Moyenne Densités moyennes	Écart-type Densités moyennes	Moyenne Écart-type des Densités	Écart-type Écart-type des Densité
Covid	139.4	25.0	54.6	14.8
Opacités pulmonaires	125.9	23.6	57.3	9.8
Normal	129.3	22.4	61.6	9.5
Pneumonie virale	125.3	19.0	58.7	10.8

d. Pré-traitement des données pour la phase de modélisation

L'analyse exploratoire a permis d'identifier :

- Des différences de format et taille dans les images
- Un déséquilibre des classes et en particulier une sous-représentation de la classe COVID qui est l'objectif de la détection
- Une opportunité de réduire les données non pertinentes en entrée du modèle grâce aux masques.

Nous avons donc retenu les étapes suivantes pour créer le jeu de données préprocessé à injecter dans les modèles :

1. **Conversion** : Convertir toutes les images en niveaux de gris.
2. **Redimensionnement** : Redimensionner les images en 256*256 pour correspondre à la taille des images radio.
3. **Réduction des informations inutiles :** Appliquer les masques aux images
4. **Equilibrage :** Équilibrer le jeu de données grâce à la méthode du sous-échantillonnage sans remise, en s'alignant sur la fréquence de la classe minoritaire (Viral Pneumonia ici).
5. **Augmentation des données** : Nous avons choisi d'expérimenter sans augmentation de données à ce stade, en accord avec le mentor projet. L'expérimentation des premiers modèles nous indiquera s'il est utile de le faire pour la suite.

Lors de l'analyse des données suite à l'observation de nombreuses distributions d'intensité des images masquées, l'hypothèse a été émise dans le groupe que les formes des distributions semblaient similaires selon le diagnostic final. Si tel est en effet le cas cela représenterait un moyen de réduire considérablement la taille du modèle en utilisant uniquement cette distribution en entrée. Cette option pourra être investiguée par la suite.

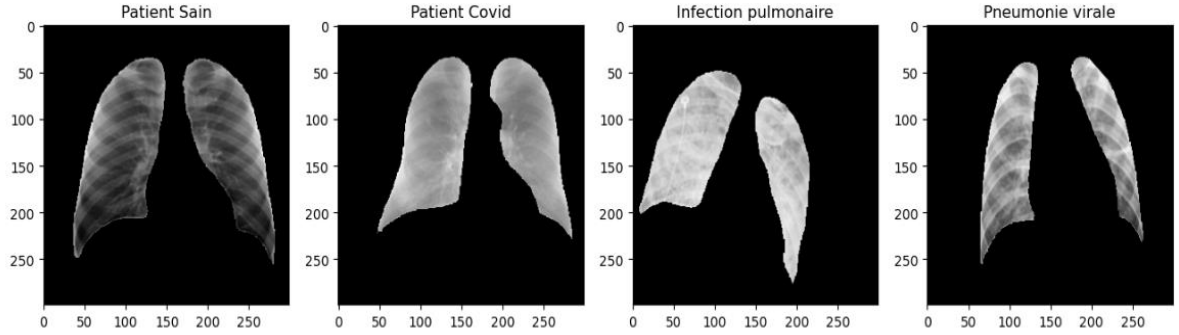
Enfin, il nous faudra séparer le jeu d'images en un jeu d'entraînement et un jeu de test, en veillant à l'équilibre des classes dans la répartition aléatoire.

Catégorie	Nombre d'images dans le jeu de données initial	Nombre d'images après rééchantillonnage
Normal	10192	1345
Opacités pulmonaires (« Lung Opacity »)	6012	1345
Covid	3616	1345
Pneumonie Virale	1345	1345

3. Modélisation

a. Classification du problème et choix de la métrique

Il s'agit ici d'un problème de **classification multi-classe supervisé**, puisque nous cherchons à déterminer un diagnostic à partir d'images radio, parmi les 4 possibilités suivantes :



Nous disposons de données labellisées (un diagnostic pour chaque image).

La tâche à réaliser s'apparente à de la reconnaissance d'image, avec un enjeu d'extraction des bons attributs permettant de caractériser chaque maladie à partir d'une seule image radio. C'est une tâche complexe même pour un ou une radiologie expérimentée : l'interprétation seule de radios du thorax présente une sensibilité (rappel) de 69%[4], donc un taux fort de faux négatifs. En comparaison, le test PCR affiche une sensibilité de 91%. C'est pourquoi la plupart des organisations professionnelles de radiologie ne recommandent pas l'usage de la radio ou du scanner pour détecter le COVID.

Pour ce problème, nous avons choisi de comparer les modèles sur la base de la métrique de justesse sur le jeu de validation (« validation accuracy »). En effet, nous ne cherchons pas seulement à identifier le COVID, mais également à détecter les 3 autres diagnostics possibles (Normal, Pneumopathie virale, Opacité pulmonaire).

Nous avons équilibré les classes dans le jeu de données d'entraînement, il y a donc moins de risques que la mesure de justesse cache une sur-représentation de la classe « Normale », comme il peut se produire dans un problème de détection d'anomalie.

Une fois les modèles les plus pertinents retenus, nous analyserons la performance sur un jeu de test ainsi que la matrice de confusion et les métriques complètes (f1-score, précision, rappel) pour s'assurer qu'il n'y a pas d'écart majeur de performance entre les classes.

[4] Source : 1. Wong HYF, Lam HYS, Fong AH, et al. Frequency and distribution of chest radiographic findings in COVID-19 positive patients [published online ahead of print, 2019 Mar 27]. Radiology 2019;201160.doi:10.1148/radiol.2020201160

b. Démarche d'expérimentation et résultats

Pour adresser cette problématique, nous avons suivi la démarche d'expérimentation suivante :

1. Implémentation d'un modèle de base : LeNet-5 :
2. Optimisation des hyperparamètres du modèle LeNet
3. Implémentation d'une approche d'apprentissage par transfert à partir de modèles pré-entraînés

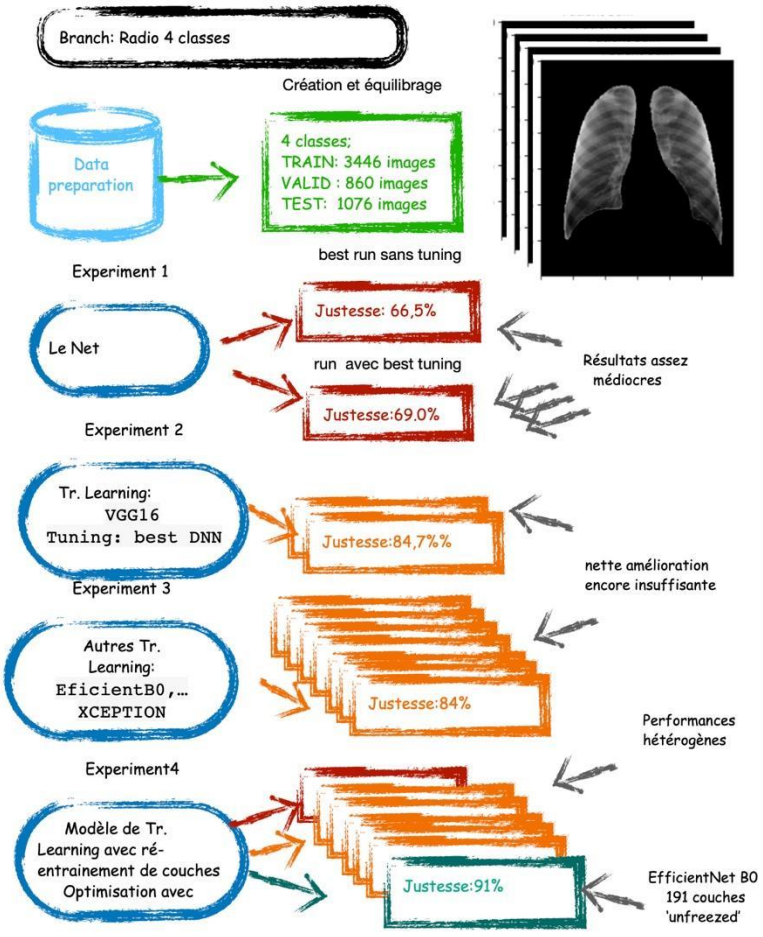


Schéma de synthèse de la méthode d'expérimentation

Toutes ces expérimentations, à l'exception de celle visant à faire varier les jeux de données, ont été réalisées sur la base d'un **même jeu de données d'entraînement prétraité comportant 4304 images (1076 par classe), et d'une structure de notebook commune.** Lors de l'import dans Keras, 20% des images sont mises de côté aléatoirement pour la « validation » et le calcul de la « validation accuracy ». Le jeu de test utilisé pour l'évaluation finale est également le même pour tous les modèles et correspond à 20% du jeu de données initial équilibré, soit **1076** images (269 par classe).

Synthèse des résultats obtenus (classés par modèle) :
Les meilleurs résultats obtenus (supérieurs à 90%) sont surlignés en vert.

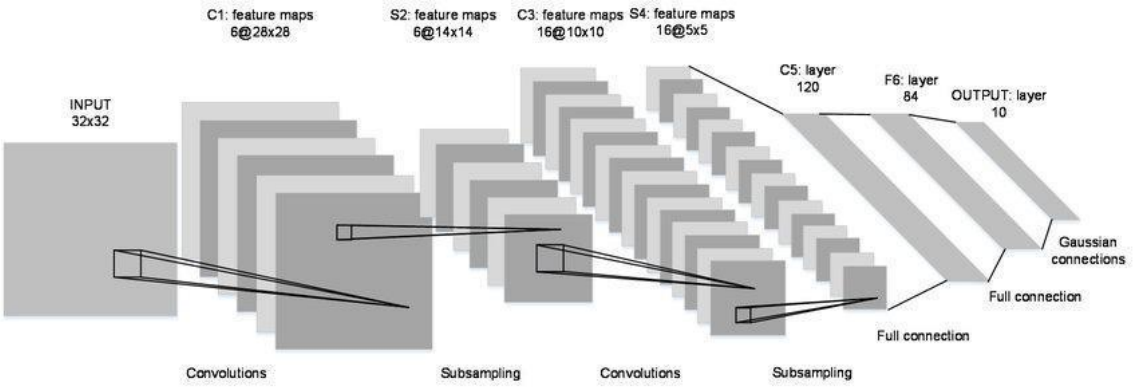
Modèle testé	Caractéristiques de l'expérimentation	Métrique de justesse sur le jeu de validation
LeNet-5	Sans optimisation d'hyperparamètres (HP)	**67%**
	Avec optimisation des hyperparamètres	**69%**
VGG16	Apprentissage par transfert sans dégel de couches – sans optimisation des HP	**85%**
	Apprentissage par transfert sans dégel de couches – avec optimisation des HP	**86%**
	Dégel et réapprentissage : 4 couches	**71%**
	Dégel et réapprentissage : 8 couches	**74%**
	Dégel et réapprentissage : 12 couches	**81%**
VGG19	Apprentissage par transfert sans dégel de couches	**85%**
	Dégel et réapprentissage de la moitié des couches	**89%**
	Dégel et réapprentissage de 100% des couches	**91%**
XCEPTION	Apprentissage par transfert sans dégel de couches	**78%**
	Dégel et réapprentissage : 27 couches	**86%**
	Dégel et réapprentissage : 67 couches	**87%**
	Dégel et réapprentissage : 97 couches	**72%**
ResNet50	Apprentissage par transfert sans dégel de couches	**80%**
	Dégel et réapprentissage : 32 couches	**74%**
ResNet101	Apprentissage par transfert sans dégel de couches	**82%**
	Dégel et réapprentissage : 32 couches	**82%**
	Dégel et réapprentissage : 132 couches	**85%**
	Dégel et réapprentissage : 202 couches	**78%**
ResNet50V2	Apprentissage par transfert sans dégel de couches	**81%**
	Dégel et réapprentissage : 24 couches	**80%**
	Dégel et réapprentissage : 92 couches	**83%**
	Dégel et réapprentissage : 150 couches	**65%**
ResNet101v2	Apprentissage par transfert sans dégel de couches	**81%**
ResNet152V2	Apprentissage par transfert sans dégel de couches	**82%**
	Dégel et réentraînement de toutes les couches	**87%**
DenseNet121	Apprentissage par transfert sans dégel de couches	**82%**
	Dégel et réapprentissage : 51 couches	**85%**
	Dégel et réapprentissage : 107 couches	**84%**
	Dégel et réapprentissage : 202 couches	**75%**
DenseNet201	Apprentissage par transfert sans dégel de couches	**82%**
MobileNet	Apprentissage par transfert sans dégel de couches	**83%**
	Dégel et réapprentissage : 19 couches	**80%**
	Dégel et réapprentissage : 50 couches	**78%**
MobileNetV2	Apprentissage par transfert sans dégel de couches	**82%**
EfficientNetB0	Apprentissage par transfert sans dégel de couches	**84%**
	Dégel et réapprentissage : 75 couches	**85%**
	Dégel et réapprentissage : 118 couches	**86%**
	Dégel et réapprentissage : 162 couches	**86%**
	Dégel et réapprentissage : 191 couches	**91%**
EfficientNetB7	Apprentissage par transfert sans dégel de couches	**83%**

Détail de quelques-unes des expérimentations effectuées

c. Modèle LeNet-5

Description du modèle

Le modèle LeNet-5, développé par Yann LeCun et ses collègues, est un réseau de neurones convolutionnel (CNN) classique. Nous avons utilisé une architecture simple avec des couches de convolution, de sous-échantillonnage, et de couches denses. Le modèle LeNet-5 est connu pour son architecture relativement simple, mais il est efficace pour des tâches de classification d'images simples.

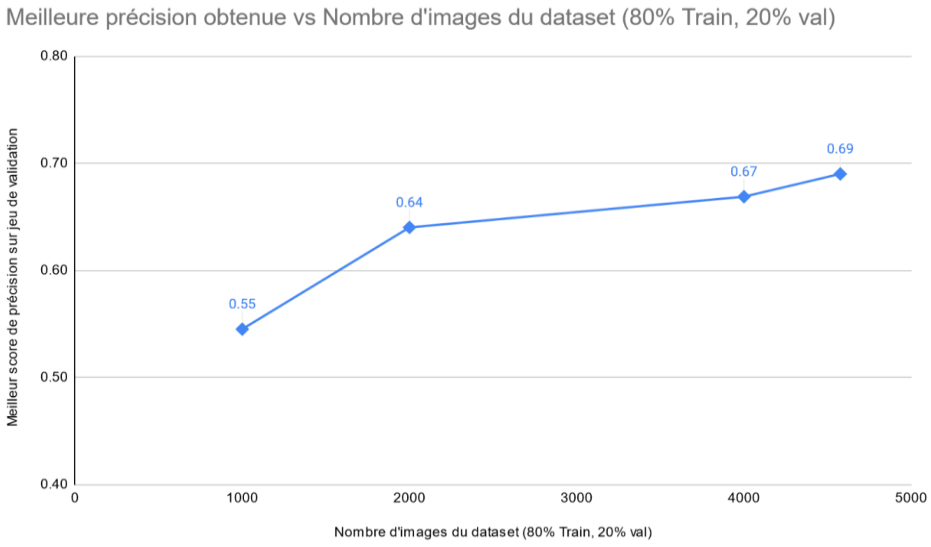


Représentation du modèle LeNet5 sur la base d'un input 32*32 - Source

Entraînement du modèle LeNet-5 avec différentes tailles de jeu de données

Cette expérimentation visait à étudier la sensibilité de la performance du modèle LeNet-5 par rapport à la taille du jeu de données. Des jeux de données de tailles différentes ont été utilisés, allant de petits ensembles à des ensembles de données plus vastes.

En entraînant le modèle LeNet optimisé sur des jeux de données de taille 1000, 2000, 3000, 4000, 4576 images, nous avons constaté que plus le jeu de données d'entraînement était important, meilleure était la performance ; en revanche le temps de computation augmentait fortement. Cela pourrait justifier le recours à l'augmentation de données afin d'augmenter la performance et réduire le surentraînement. Afin de prioriser l'expérimentation de modèles plus complexes, nous n'avons pas, à ce stade, appliqué d'augmentation d'image.



d. Optimisation des hyperparamètres du modèle LeNet

Nous avons effectué un tuning des hyperparamètres pour optimiser les performances du modèle LeNet-5. En utilisant la fonction Keras Tuner, nous avons réalisé plusieurs tests en faisant varier les paramètres comme :

- Nombre de filtres des couches de convolution
- Présence ou l'absence d'une couche Dropout entre chaque couche, et intensité du Dropout
- Nombre de couches denses en sortie et nombre de neurones pour chacune des couches (parmi 16, 32, 64,128, 256, 512, 1024, et 2076)
- La fonction d'activation de chaque couche (parmi 'relu' ou 'tanh')

Le modèle de classificateur DNN final obtenu :

Couches du classificateurs	Nombre de neurones	Fonction d'activation	drop out ratio at 'output'
Dense 1	128	'relu'	0%
Dense 2	32	'relu'	20%
Dense 'Output'	4	'softmax'	NA

e. Approche d'apprentissage par transfert à partir de modèles pré-entraînés

Implémentation d'un modèle d'apprentissage par transfert utilisant le modèle VGG16

L'apprentissage par transfert a été appliqué en utilisant le modèle pré-entraîné VGG16. Nous avons gelé les couches convolutionnelles préexistantes et ajouté des couches denses spécifiques à notre tâche de classification.

VGG16 est un modèle de réseau de neurones convolutionnel qui a été développé par le Visual Graphics Group (VGG) à l'Université d'Oxford. Il a été créé pour participer au concours ImageNet Large Scale Visual Recognition Challenge (ILSVRC) en 2014, où il a obtenu de très bonnes performances.

Le modèle VGG16 est caractérisé par sa profondeur, composée de 16 couches de convolution et de sous-échantillonnage (pooling) suivi de trois couches denses. Les couches de convolution sont toutes configurées avec des filtres de petite taille (3x3) et un pas (stride) de 1, ce qui donne une architecture très uniforme.

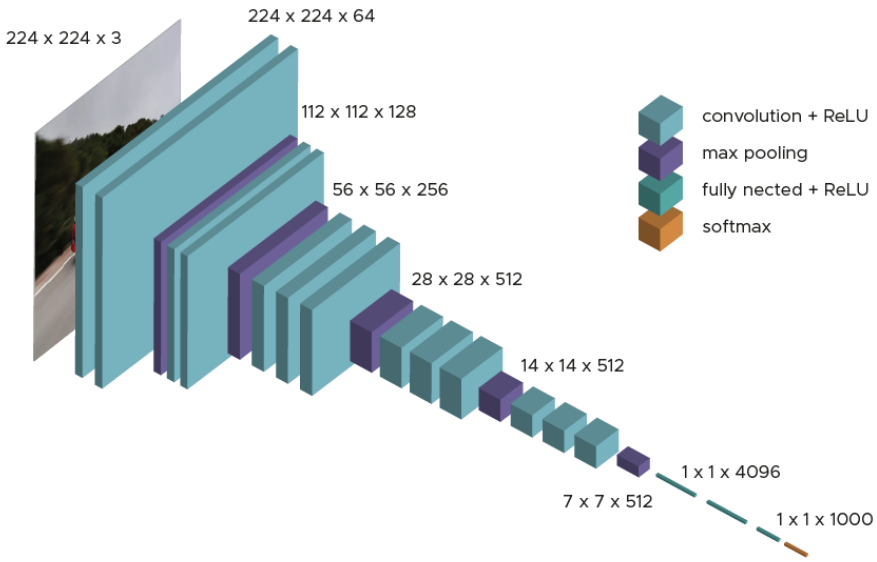


Schéma de la structure de VGG16 – Source Datascientest

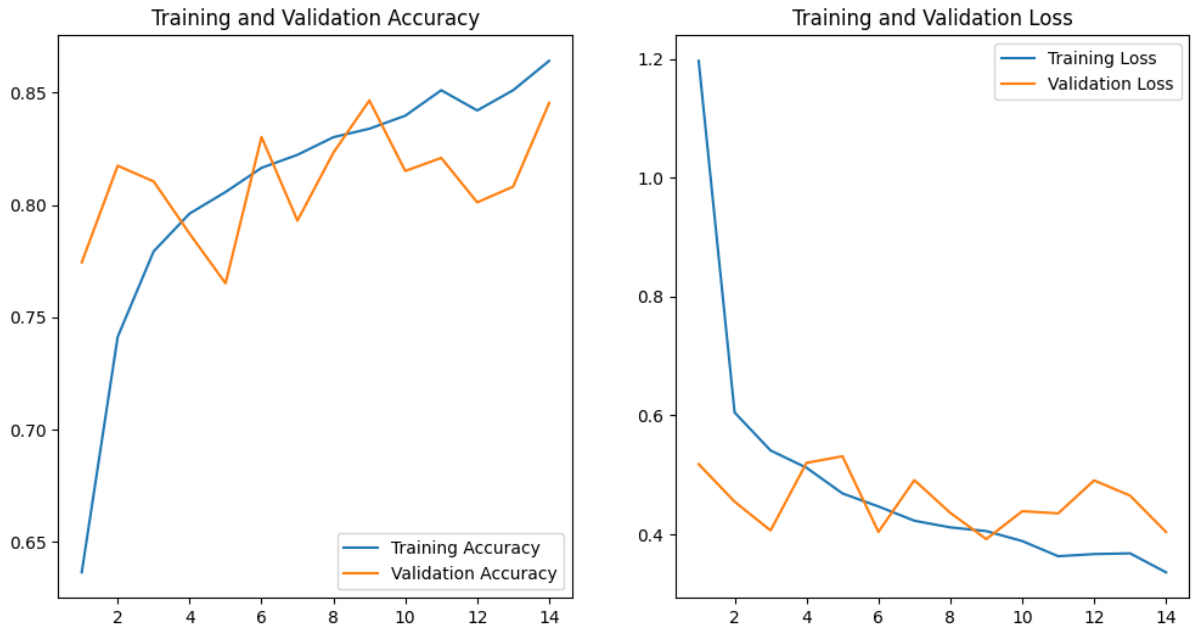
L'apprentissage par transfert permet de réutiliser un modèle pré-entraîné pour une tâche similaire. Dans le cas de VGG16, le modèle a été pré-entraîné sur la base de données ImageNet pour classifier une large variété d'images. Même si les images n'ont pas de rapport avec les images radio de notre jeu de données, les capacités d'extraction d'attributs à partir des images font de VGG16 un candidat intéressant.

Dans un premier temps, nous avons entraîné un modèle composé :

- D'une couche d'entrée permettant d'injecter des images au format (256, 256, 3). Le modèle VGG16 est paramétré pour des images au mode RGB.
- Du modèle VGG16 importé sans les couches denses de classification
- D'une couche de « GlobalAveragePooling » pour convertir la sortie du modèle VGG16
- De plusieurs couches denses pour la classification en 4 classes.

La fonction de pré-traitement fournie par la bibliothèque VGG16 de Keras a été appliquée au jeu de données d'entraînement, de validation et de test (centrage des valeurs de pixels notamment), en amont du modèle.

Les couches du modèles VGG16 ont été « gelées » pour pouvoir utiliser les poids liés au pré-entraînement sur le jeu « ImageNet ». Le modèle a été compilé avec l'optimiseur « Adam », la fonction de perte « sparse categorical crossentropy » adaptée à la classification multi-classes de données labellisées, et la mesure de justesse (« accuracy ») sur le jeu d'entraînement et de validation. Nous avons d'abord entraîné le modèle sur 15 périodes.



Courbes de justesse et perte pour le modèle d'apprentissage par transfert utilisant VGG16 (sans dégel de couches)

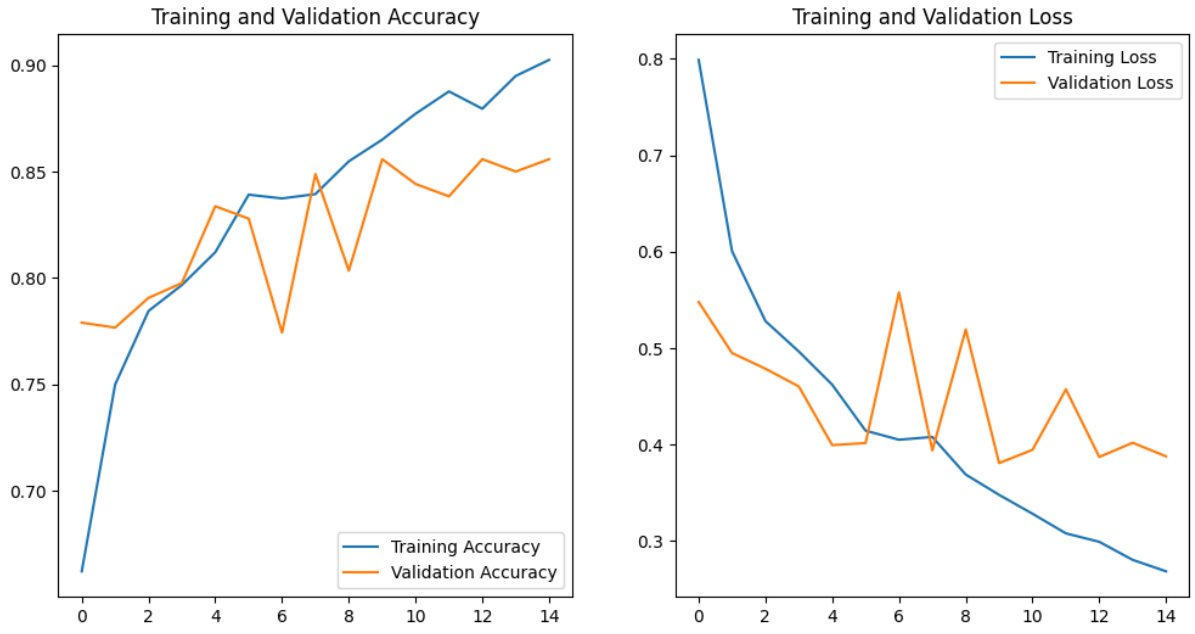
Optimisation des paramètres du classifieur de sortie du modèle VGG 16

Nous avons effectué un tuning des hyperparamètres pour optimiser les performances de ce modèle basé sur VGG16. La fonction de tuner « Hyperband » de la bibliothèque Keras Tuner a été préférée au « Random Search » afin de tester toutes les combinaisons possibles.

Hyperparamètre	Espace d'optimisation	Meilleur modèle retenu
Intensité du dropout à la sortie du modèle VGG16	Entre 0 et 0.5 par pas de 0.1	**0**
Nombre de neurones de la première couche dense	0, 32, 64, 128, 256, 512	**128**
Nombre de neurones de la deuxième couche dense	0, 32, 64, 128, 256, 512	**32**
Intensité du dropout à la sortie de la couche dense	Entre 0 et 0.5 par pas de 0.1	**0.3**
Pas d'apprentissage (« learning rate ») pour l'optimiser Adam	0.1, 0.01, 0.001, 0.0001, 1e-05	**0.001**

Les fonctions d'activation ont également fait l'objet de tests manuels (entre « relu » et « tanh »), mais n'affectent pas la performance du modèle de manière significative.

Le meilleur modèle a ensuite été implémenté et entraîné sur le jeu de 4503 images, et a atteint une performance de 86% sur les données de validation. Evalué sur le jeu de test, il atteint 83%. On remarque sur les courbes suivantes que la performance sur le jeu d'entraînement devient rapidement supérieure à celle sur le jeu de validation : le modèle sur-adapte au bout de quelques périodes.
Pour éviter cela, nous avons ajouté un paramètre « callback » lors de l'entraînement du modèle, permettant d'arrêter l'entraînement dès lors que le score de perte ne s'améliore plus, avec une latence de 5 périodes.



Courbes de justesse et perte pour le modèle d'apprentissage par transfert utilisant VGG16 (avec optimisation des hyperparamètres et arrêt anticipé (« callback »).

Le rapport de classification calculé sur le jeu de données de test apporte des précisions : le f1-score global est de **82%**, et le modèle performe particulièrement bien sur la classe Pneumonie Virale avec **97%**. Des améliorations peuvent être attendues sur les 3 autres classes.

	Précision	**Rappel**	**Score F1**	**Population (jeu de test)**
COVID	0.73	0.81	0.77	269
Lung Opacity	0.79	0.74	0.76	269
Normal	0.81	0.77	0.79	269
Viral Pneumonia	0.97	0.98	0.97	269
Global	0.83	0.82	0.82	1076

Par la suite, nous avons expérimenté avec d'autres modèles d'apprentissage par transfert existants, en s'inspirant de modèles qui ont prouvé de bonnes performances dans des publications scientifiques. Ces modèles ont été entraînés sur le **même jeu de données** d'entraînement et testés sur le même jeu de validation. La séparation du jeu d'entraînement et de validation a utilisé le même paramètre (« seed ») dans tous les tests. Nous pouvons donc comparer les résultats entre modèles.

Nous avons testé l'approche par apprentissage avec les modèles suivants :

- VGG16 (présenté au-dessus)
- VGG19
- XCEPTION
- ResNet50
- ResNet101
- ResNet50V2
- ResNet101v2
- RestNet152V2
- DenseNet121
- DenseNet201
- MobileNet
- MobileNetV2
- EfficientNetB0
- EfficientNetB1

Pour chacun des modèles, la fonction de **pré-traitement** adaptée, disponible dans la bibliothèque Keras, a été utilisée. Ces fonctions permettent d'appliquer aux images en entrée le même pré-traitement que les images qui ont été utilisées pour le pré-entraînement (ImageNet). En voici deux exemples (source Keras.io) :

- Fonction "preprocessing" de VGG16 : Les images sont converties de RVB à BGR, puis chaque canal de couleur est centré sur zéro par rapport au jeu de données ImageNet, sans mise à l'échelle
- Fonction "preprocessing" de ResNetV2 : Les valeurs des pixels sont mises à l'échelle entre -1 et 1.

Dans la première salve de tests, nous avons entraîné les modèles en « gelant » les couches du modèle de base, qui ont été pré-entraînées sur le jeu de données « ImageNet ». Nous avons atteint un score de justesse **entre 80% et 85%** avec la plupart des modèles.

Nous avons également tenté d'entraîner ces modèles en "dégelant" une partie des couches du modèle de base afin de les réentraîner sur le jeu de données fourni. Nous avons effectué plusieurs tests en faisant varier le modèle et le nombre de couches dégelées. Nous avons atteint un score de justesse de **91%** avec les modèles basés sur VGG19 et EfficientNetB0.

Les combinaisons testées sont présentées dans le tableau de synthèse en début de chapitre.

Nous retenons plusieurs leçons générales de ces expérimentations :

- L'approche par transfert permet d'obtenir de très bons résultats, avec une approche plus sobre (économie de la construction et du premier entraînement du modèle). Ceci est d'autant plus impressionnant que le jeu de données sur lequel ont été entraînés ces modèles, ImageNet, ne **contient pas d'images de radiologie**.
- Dans l'ensemble, le dégel et le réentraînement de couches du modèle de base permet d'obtenir une meilleure performance. Cela ne s'est pas toujours vérifié dans nos tests, et nous avons constaté que le choix du pas d'apprentissage était très influent. Il semble que l'utilisation d'un pas d'apprentissage plus petit, ou adaptable en fonction de la période, permette d'améliorer légèrement la performance.

f. Interprétation des résultats grâce à la méthode « Grad-CAM »

Pour chacun des modèles testés, l'analyse de la matrice de confusion et du rapport de classification, calculés sur un **jeu de test** qui n'a pas servi pour l'entraînement**,** nous ont permis de constater :

- Que la plupart des modèles d'apprentissage par transfert atteignent une bonne performance de classification sur 4 classes, **entre 85% et 90%**
- Que la performance n'est pas homogène en fonction des classes : la classe « Viral Pneumonia » a généralement un score F1 plus haut que les autres (95 à 98%).

Il semble donc que les modèles parviennent mieux à extraire des signes caractéristiques de la pneumonie virale à partir des radios de poumons.

Pour aller plus loin, nous avons implémenté l'algorithme **Grad-CAM** (*« Gradient-weighted Class Activation Mapping »),* inspiré des exemples fournis par François Chollet sur le site de Keras[5] ainsi que de nombreux projets Github.

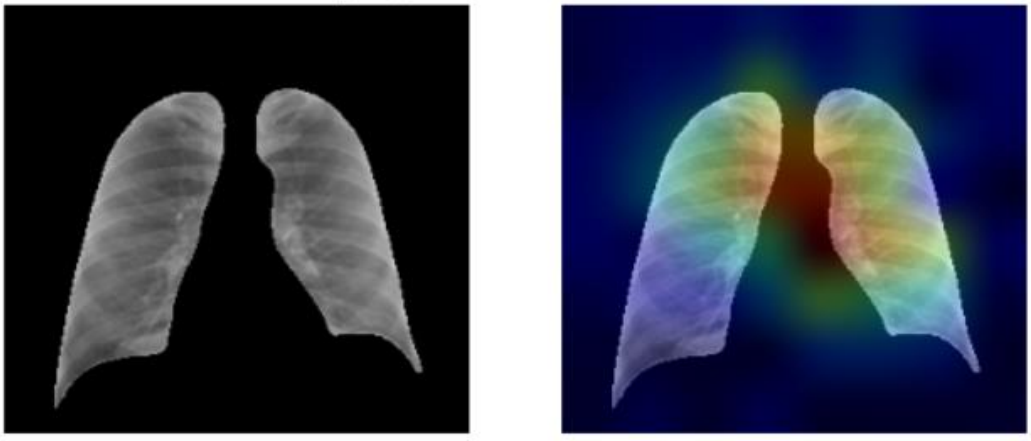
Cette méthode consiste à extraire la dernière couche de convolution du modèle, avant les couches denses, et de calculer les gradients de la sortie de cette couche par rapport à l'activation de cette même couche. Ces gradients sont ensuite utilisés pour générer une carte d'activation, mettant en évidence les régions de l'image qui ont contribué le plus à la prédiction de notre modèle.

Nous présentons dans les pages suivantes plusieurs exemples de Grad-CAM appliqué au modèle d'apprentissage par transfert construit à partir de VGG16 et optimisé précédemment, sur des images tirées au hasard dans chacune des classes du jeu de test.

Grad-CAM sur une image de la classe Normale

```
Classe réelle :  NORMAL
Classe prédite par le modèle :  Normal
```

NORMAL-1917.png



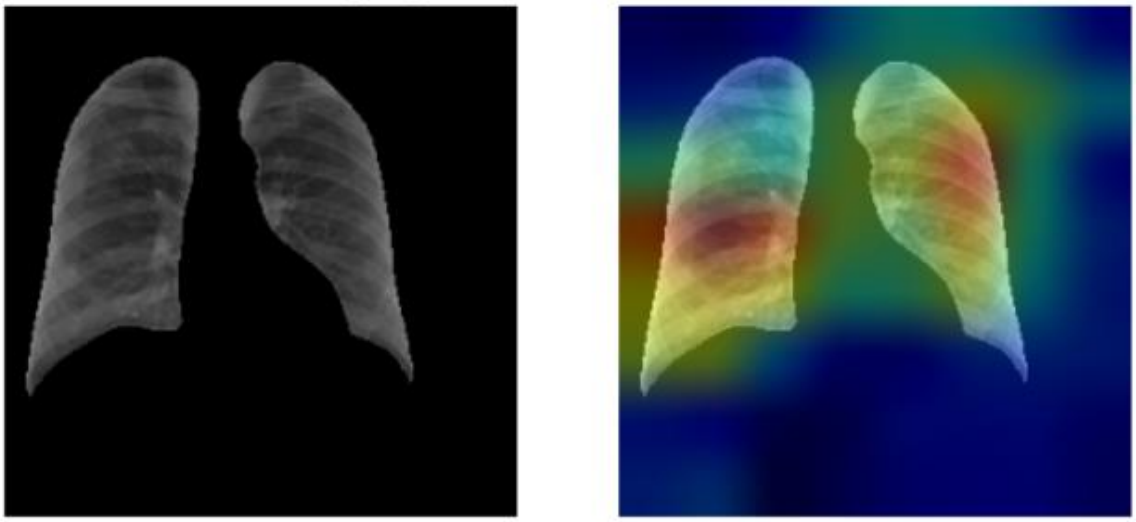
On constate que la zone activée par le modèle est centrée entre les poumons et sur les bords intérieurs dans la partie supérieure des poumons.

[5] Grad-CAM class activation visualization, F. Chollet, https://keras.io/examples/vision/grad_cam/

Grad-CAM sur une image de la classe COVID

```
Classe réelle :  COVID
Classe prédite par le modèle :  COVID
```

COVID-1214.png

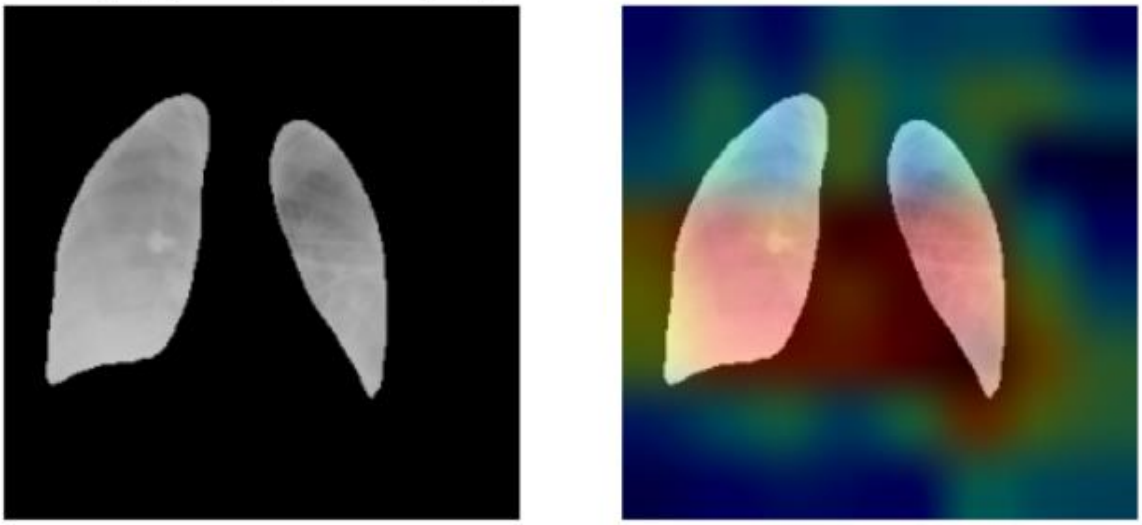


On constate que la zone activée par le modèle est plus localisée à l'intérieur des poumons.

Grad-CAM sur une image de la classe « LUNG OPACITY »

```
Classe réelle :  Lung_Opacity
Classe prédite par le modèle :  Lung_Opacity
```

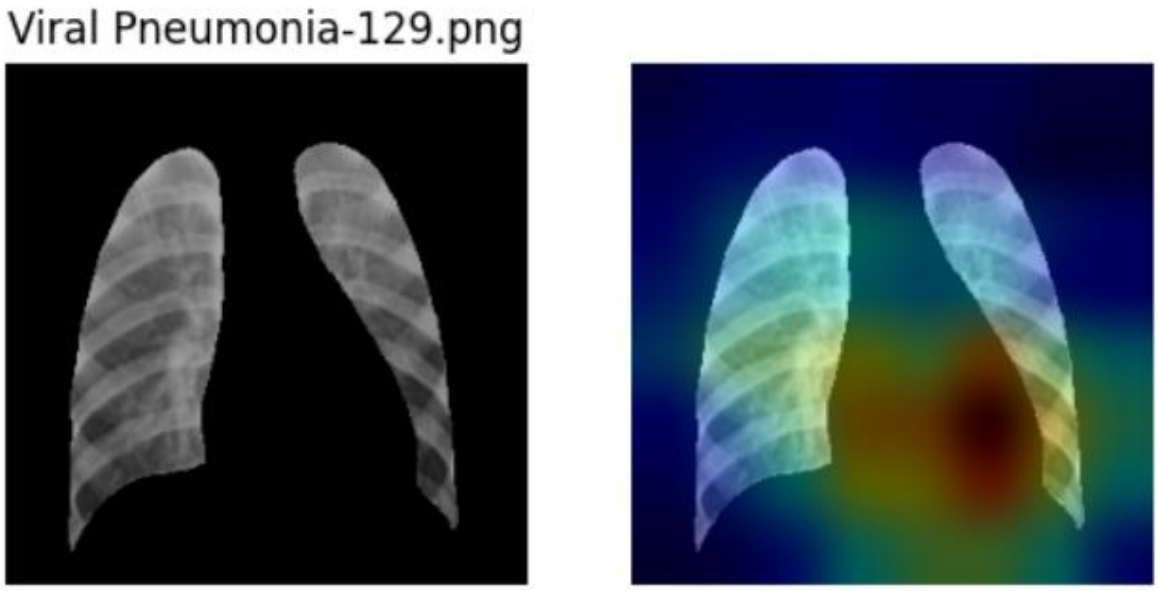
Lung_Opacity-1055.png



On constate que la zone activée par le modèle est plus étendue que pour le Covid. Toutefois, elle est aussi assez différente d'une image à l'autre, et nous n'avons pas réussi à déterminer les caractéristiques particulières à l'œil nu.

Grad-CAM sur une image de la classe « VIRAL PNEUMONIA »

```
Classe réelle :  Viral Pneumonia
Classe prédite par le modèle :  Viral_Pneumonia
```



On constate que la zone activée par le modèle semble localisée entre les poumons, dans la zone « masquée ».

Sans connaissance en analyse d'imagerie médicale, notre capacité à interpréter les sorties de nos modèles est limitée. Pour tenter d'aller plus loin, nous avons sollicité une radiologue professionnelle et lui avons soumis notre approche et nos résultats Grad-CAM sur deux modèles (VGG19 / ResNet152V2).

Il en ressort que les zones activées par les modèles (et mises en évidence par le Grad-CAM) correspondent peu aux signes pathologiques utilisés par les professionnels.

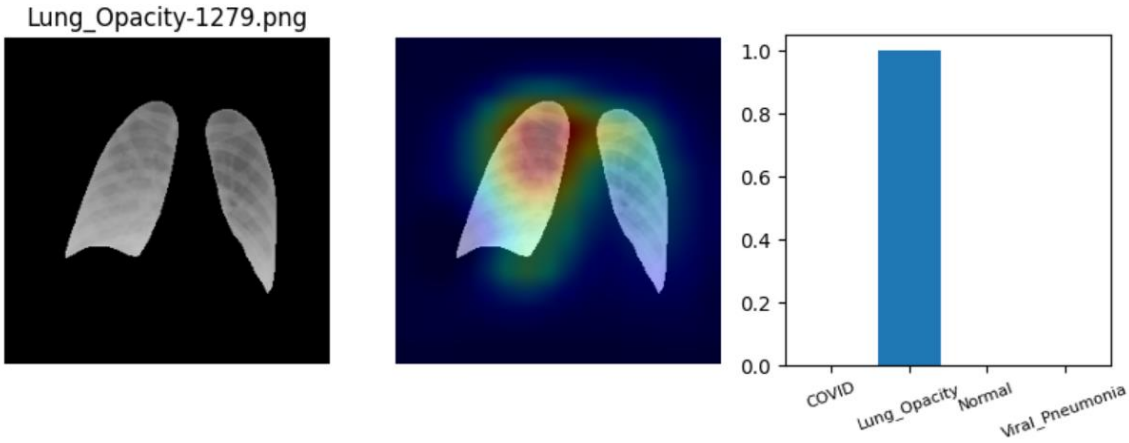
En effet, les radiologues se basent sur des anomalies visibles par rapport à l'image radio d'un patient sain. Les variations d'intensité d'une image radio sont dues aux obstacles rencontrés par les rayons-X sur leur trajectoire. Les poumons d'un patient sain contiennent uniquement de l'air, leur texture est donc homogène, et on visualise bien les frontières des côtes et des organes et vaisseaux sanguins. Sur la radio d'un patient atteint d'une pathologie pulmonaire, un radiologue cherche des signes d'anomalie comme par exemple :

- Des « tâches » visibles à l'œil nu
- Des zones opaques qui traduisent la présence de fluide dans les poumons
- Des frontières floues entre les poumons et les organes (autour du cœur, du diaphragme par exemple)

Les informations en dehors de la zone pulmonaire fournissent aussi des indices sur la gravité de la maladie : intubation, présence de cathéters, …

Interprétation croisée sur une image de la classe Lung Opacity (modèle VGG19)

```
modèle :  vgg19
True class :  Lung_Opacity
Predicted class :  ['Lung_Opacity']
```



Ici, la radiologue a repéré la base des deux poumons comme étant en anomalie (opacité et côtes moins visibles). Or le Grad-CAM montre que le modèle a activé la zone qui paraît « normale » à l'œil du radiologue. L'attribut extrait par le modèle pour la classification est difficilement interprétable d'un point de vue médical.

De plus, il manque sur la radio masquée une partie du diaphragme (à droite sur l'image). Le masque fourni exclut donc une zone utile à l'analyse. Dans les approfondissements (chapitre suivant), nous étudierons l'impact du masquage sur la performance du modèle.

4. Approfondissements

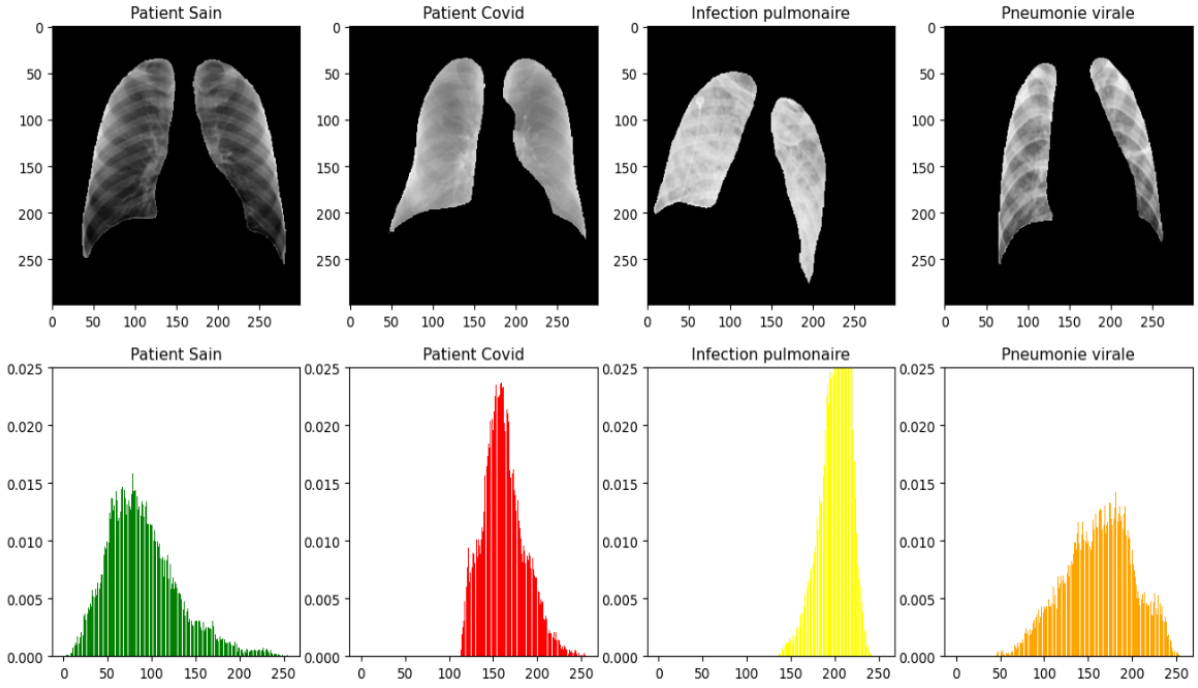
a. Expérimentation 1 : Approche réduite sur la base des distributions d'intensité

De même que la classification directe des images, la classification de la réduction d'image réduites à leur distribution d'intensité est complexe et a été expérimentée comme une branche parallèle de nos modélisations.
Nous avons tout d'abord expérimenté un modèle de classification simple sur la base des vecteurs d'intensité des images.
Ce problème réduit à un choix binaire Malade/Pas Malades est typiquement un problème de diagnostic.
Cependant selon l'usage médical, il existe deux situations de diagnostic qui nécessite de privilégier la performance selon certaines métriques :

- Le diagnostic d'inclusion : privilégie la justesse (*« accuracy »)* et la spécificité
- Le diagnostic d'exclusion privilégie la justesse (*« accuracy »)* et la sensibilité

La métrique de performance principale de cette classification binaire sera la justesse, mais nous considérerons également la sensibilité et spécificité dans un deuxième temps.

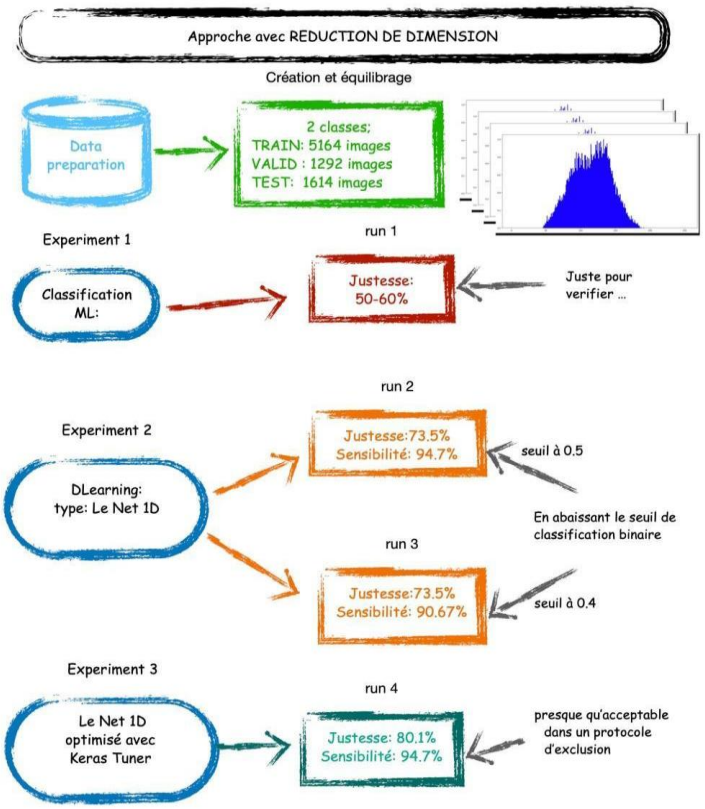
Cette approche malade/normal, nous permettra de disposer d'un jeu de données plus larges. Nous disposons ainsi d'un jeu de 5164 images d'entraînements, 1292 pour la validation et enfin 1614 pour le jeu de test



Choix du modèle et optimisation

1. Expérimentation #1 : KNN, SMV, RandomForest. Justesse < 60%
2. Expérimentation #2 : Apprentissage profond. Réseau type Le net en dimension 1.
3. Expérimentation #3 : Affinage d'une Le Net avec Keras Tuner
4. Optimisation de l'architecture et des paramètres avec Keras Tuner

Par manque de temps nous avons stoppé cette branche d'exploration à ce niveau.



L'amélioration du modèle au sein de chaque expérimentation est souvent venue de l'utilisation des bons paramètres dans Keras, et surtout de l'utilisation de Keras Tuner qui nous a permis de sélectionner la bonne architecture présenter à gauche

Les résultats de l'expérimentation 3 montre un entraînement optimisé du modèle, mais également une excellente performance en sensibilité de 94,5% tout en conservant une justesse acceptable de 80%.

	PREDICTION MALADE	PREDICTION NORMAL
MALADE REEL	VP :764	FN :43
NORMAL REEL	FP :279	VN :528

Avec un faible taux de faux négatif, soit une bonne sensibilité, le modèle devient acceptable pour l'intégrer dans un protocole d'exclusion.

b. Expérimentation 2 : Augmentation de données et pré-traitement

Pour tenter de réduire la sur-adaptation des modèles au jeu d'entraînement, nous avons réalisé plusieurs tests en ajoutant une phase **d'augmentation de données** en amont du réseau de neurones convolutif. Dans la littérature consultée pour le projet, cette approche a permis d'améliorer les performances des modèles entraînés[6].

Nous avons testé trois méthodes d'augmentation de données en amont du réseau de neurones :

1. **Diversification du jeu d'entraînement à taille égale** (transformations aléatoires appliquées aux images)
2. **Augmentation du nombre d'images du jeu d'entraînement** (12 000 images au lieu de 4000)
3. **Pré-traitement visant la réduction des biais** (égalisation d'histogramme et flou gaussien)

Diversification du jeu d'entraînement à taille égale

Nous avons reproduit le test d'apprentissage par transfert mené avec le modèle ResNet152V2, cette fois en ajoutant en amont du modèle une couche d'augmentation de données avec les opérations suivantes :

- Translation aléatoire +/-10%
- Rotation aléatoire +/-10°
- Retournement horizontal aléatoire
- Zoom aléatoire +/-15%
- Variation d'intensité aléatoire +/-10%

Ce test n'a pas obtenu un meilleur résultat que celui sans augmentation de données. La performance était même inférieure, toutes choses égales par ailleurs (voir tableau ci-dessous).

Nous pouvons émettre plusieurs hypothèses :

- Le jeu de données initial présente déjà une **forte variabilité** et l'augmentation de données n'apporte rien au processus d'apprentissage
- Les transformations appliquées n'ont **pas de sens d'un point de vue métier** (par exemple, la position des organes n'est pas symétrique, et moins d'une personne sur 10 000 a les organes inversés ; il n'est peut-être pas pertinent d'appliquer un retournement horizontal)

Résultats obtenus :

Modèle d'apprentissage par transfert	Caractéristiques de l'expérimentation	Métrique de justesse sur le jeu de validation
ResNet152V2	Apprentissage par transfert sans dégel de couches	**82%**
	Entraînement avec transformations en amont	**77%**

[6] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7658227/

Augmentation du nombre d'images du jeu d'entraînement

Pour ce test, nous avons testé l'hypothèse suivante : fournir plus de données lors de l'entraînement du modèle permet d'améliorer sa performance.

Au lieu du jeu d'entraînement à 4304 images que nous avions utilisé pour comparer les modèles, nous avons extrait un jeu d'entraînement de 12 193 images. Le nombre d'images par classe a été fixé par rapport à l'échantillon « COVID » dans le jeu initial, qui comportait 3616 images.
La classe « Viral Pneumonia » ne comportait que 1345 images dans le jeu initial, toutefois nous avons accepté un léger déséquilibre dans le jeu de données en conservant cette valeur, car cette classe avait obtenu les meilleurs scores F1 dans les tests précédents.

Catégorie	Nombre d'images dans le jeu de données initial	Nombre d'images après rééchantillonnage
Normal	10192	**3616**
Opacités pulmonaires (« Lung Opacity »)	6012	**3616**
Covid	**3616**	**3616**
Pneumonie Virale	1345	**1345**

Résultat : nous n'avons pas constaté d'amélioration par rapport au test initial sur le jeu de donnée à 4304 images.

Modèle d'apprentissage par transfert	Caractéristiques de l'expérimentation	Métrique de justesse sur le jeu de validation
VGG19	Apprentissage par transfert avec dégel progressif de couches sur jeu de données à 4304 images	**91%**
	Apprentissage par transfert avec dégel progressif de couches sur jeu de données à 12 193 images	**90%**

Pré-traitement visant la réduction des biais (égalisation d'histogramme et flou gaussien)

Lors de notre phase de pré-traitement, nous avions choisi d'opérer une conversion en niveaux de gris et l'application des masques fournis. Or d'après la littérature scientifique parcourue pour le projet[7], l'application de méthodes de pré-traitement des images peut améliorer la performance des modèles d'apprentissage par transfert classiques. Nous nous sommes donc inspirés d'une de ces publications pour comparer la performance d'un modèle **sans** pré-traitement supplémentaire et **avec pré-traitement : égalisation d'histogramme et flou gaussien.**

L'égalisation d'histogramme a été choisie suite à une discussion avec une radiologue, car le niveau de contraste est un facteur déterminant pour l'analyse des images radio, et pourrait donc constituer un biais pour le modèle (par exemple si toutes les images COVID ont nécessité un réglage particulier lors de l'examen radio).

Le flou gaussien a été choisi car il permet de réduire le « bruit » de l'image, qui peut aussi constituer un biais pour l'apprentissage.

Toutefois, le pré-traitement supplémentaire n'a pas permis d'obtenir une meilleure performance.

Résultats obtenus :

Modèle d'apprentissage par transfert	Caractéristiques de l'expérimentation	Métrique de justesse sur le jeu de validation
VGG19	Apprentissage par transfert sans pré-traitement (jeu de 12 193 images avec dégel de couches)	**90%**
	Apprentissage par transfert avec égalisation d'histogramme et flou gaussien en amont du modèle (jeu de 12 193 images avec dégel de couches)	**88%**

[7] "Improving the performance of CNN to predict the likelihood of COVID-19 using chest X-ray images with preprocessing algorithms" https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7510591/

c. Expérimentation 3 : Classification 3 classes

Une exploration des sources des données utilisées nous a permis d'identifier que la classe « Lung Opacity » était en fait composée d'images de pathologies diverses non-Covid. Il se peut que la classe soit très hétérogène par rapport aux trois autres dont les caractéristiques sont plus ciblées (Covid, pneumonie virale, patient sain).

Nous avons donc conduit une expérimentation pour tester la performance de classification sur 3 classes, en retirant les images « Lung Opacity » de notre jeu de données.

L'impact constaté sur le modèle testé, est très important, avec une amélioration de la justesse de plus de 10 points selon les tests réalisés. Cela conforterait donc l'idée que la qualité et l'homogénéité du jeu de donnée à un impact important sur les performances. Cependant pour faire une telle conclusion, il aurait fallu approfondir et étendre cette comparaison sur d'autres modèles pour au moins confirmer la répétabilité de ce résultat.
(Cette expérience a été menée sur la base du Notebook 1.0 Directe Exp 4-16 ResNet152v2.ipynb)

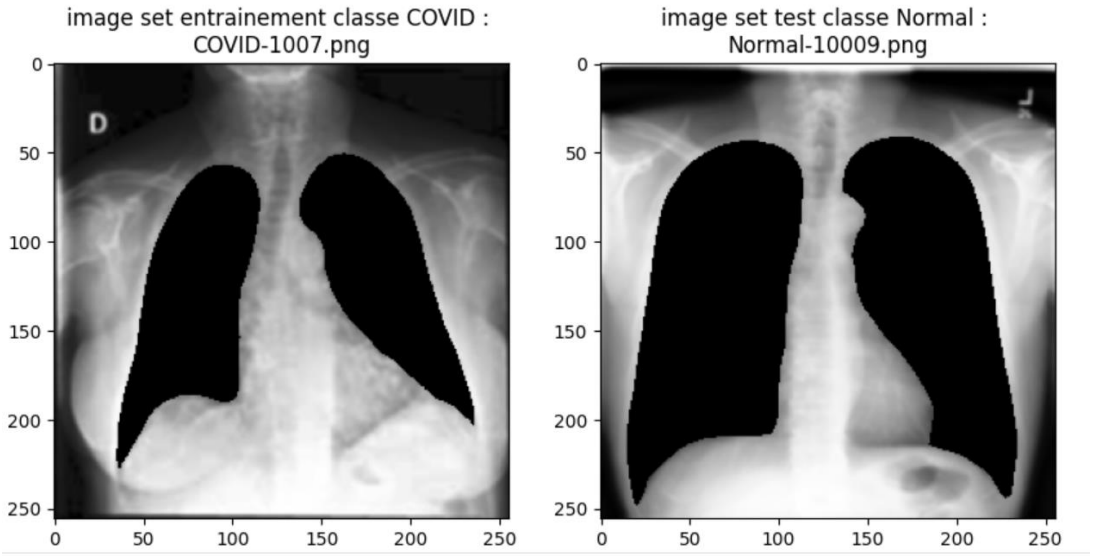
Résultats obtenus :

Modèle d'apprentissage par transfert	**Caractéristiques de l'expérimentation**	**4 classes**	**3 classes**
		Justesse sur le jeu de validation	**Justesse sur le jeu de validation**
Resnet152v2	Apprentissage sans dégel de couches	**68%**	**80%**
	Réapprentissage avec dégel de 21 couches	**69%**	**81%**
	Réapprentissage avec dégel de 61 couches	**74%**	**87%**

d. Expérimentation 4 : Performance avec ou sans poumons masqués

Nous savons que les professionnels ne se basent pas uniquement sur l'intérieur des poumons mais sur un ensemble d'indices visibles sur les radiographies du thorax, comme les frontières des organes. Une expérience réalisée par une équipe de chercheurs[8] montre que les modèles d'apprentissage performent de manière similaire même si l'on masque la zone des poumons. Nous avons tenté de reproduire ce résultat.

Puisque nous disposons des masques fournis avec les données source, nous avons entraîné un modèle sur un jeu d'images sur lesquelles la **zone pulmonaire était masquée.** Voici deux exemples d'images utilisées :



A l'issue de l'expérimentation, **le modèle entraîné sur les images avec poumons masqués a obtenu un meilleur résultat** que le même modèle entraîné uniquement sur la zone des poumons (notre pré-traitement initial utilisé dans le chapitre 3).

Ce résultat nous a fortement étonné et nous a amené aux hypothèses suivantes :

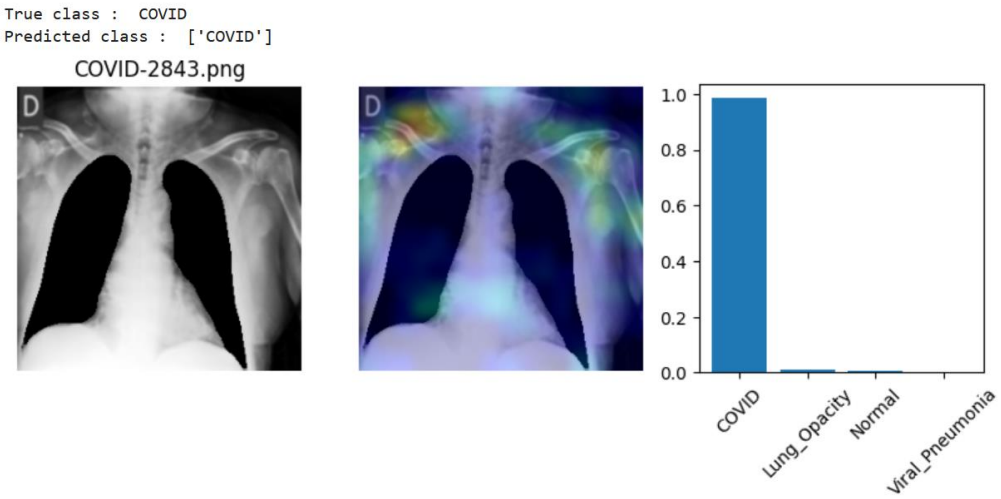
- Le masque fourni avec les images exclut une partie des informations utiles pour le diagnostic médical
- Le jeu de données initial comporte des biais que nous n'avons pas identifiés (par exemple, il se pourrait que toutes les radios d'une classe aient été prises avec un réglage de la machine différent des autres classes).

Résultats :

Modèle testé	Caractéristiques de l'expérimentation	Métrique de justesse sur le jeu de validation
VGG19	Apprentissage par transfert, sans dégel de couches, sur jeu de données à 12193 images : **zone pulmonaire uniquement**	**82%**
	Apprentissage par transfert, sans dégel de couches, sur jeu de données à 12193 images : **poumons masqués**	**89%**

[8] Sadre, R., Sundaram, B., Majumdar, S. et al. Validating deep learning inference during chest X-ray classification for COVID-19 screening. Sci Rep 11, 16075 (2021). https://doi.org/10.1038/s41598-021-95561-y

L'analyse du Grad-CAM du modèle entraîné sur les images « poumons masqués » montre que les zones activées se situent pour partie aux frontières des poumons mais également sur d'autres zones peu pertinentes d'un point de vue médical (exemple : les os pour la classe « Covid » dans l'exemple suivant).



Pour approfondir ce résultat, nous avons comparé la performance d'un modèle donné sur trois jeux de données contenant les mêmes images mais avec des zones masquées différentes :

1. « Poumons seuls » : les masques sont appliqués pour garder uniquement la zone des poumons (notre jeu de donnée prétraité utilisé pour les tests du chapitre 3)
2. « Poumons masqués » : les masques sont appliqués pour enlever l'intérieur des poumons de l'image
3. « Image complète » : les images non masquées fournies dans le jeu Kaggle (converties en niveaux de gris et redimensionnées en 224x224 par cohérence avec les autres tests)

Le modèle utilisé pour les 3 tests utilise l'apprentissage par transfert avec le modèle de base ResNet152V2 pré-entraîné sur ImageNet. Pour chaque test, le modèle a été entraîné une première fois en gelant les couches du modèle de base, puis réentraîné en dégelant les blocs supérieurs du modèle ResNet152V2 (400 couches).

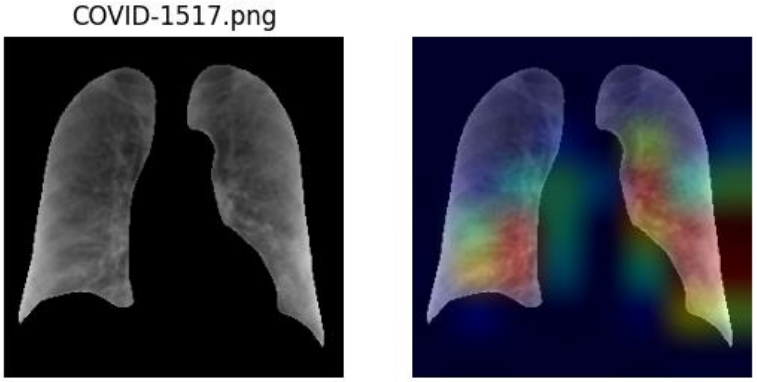
Résultat : **le modèle performe mieux sur les images complètes et les images « poumons masqués » que les images « poumons seuls ».** Cela confirme l'hypothèse selon laquelle les masques excluent une partie de l'information utile. Toutefois, le score sur les images « poumons masqués » n'est pas expliqué, et pourrait être dû à des biais non identifiés dans les données sources.

Modèle testé	Caractéristiques de l'expérimentation	Métrique de justesse sur le jeu de validation
ResNet152V2	Jeu de 4304 images avec « poumons seuls »	**84 %**
	Jeu de 4304 images avec « poumons masqués »	**89 %**
	Jeu de 4304 images complètes	**87 %**

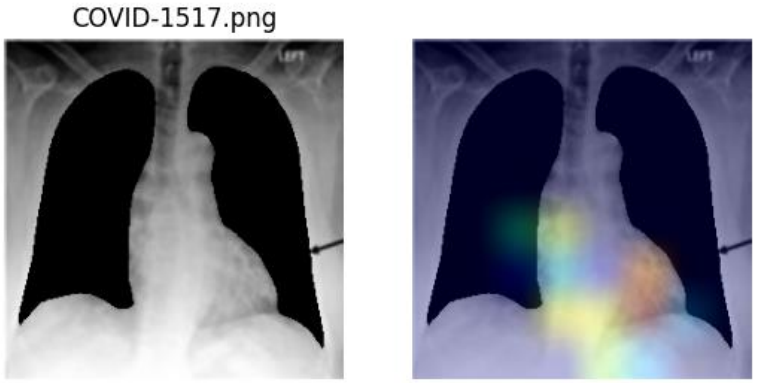
Pour comprendre ces écarts, nous avons comparé la carte d'activation Grad-CAM des 3 tests sur une même image.

Cartes d'activations Grad-CAM du modèle entraîné sur les images avec poumons, sans poumons, et complètes (IMAGE COVID – 1517)

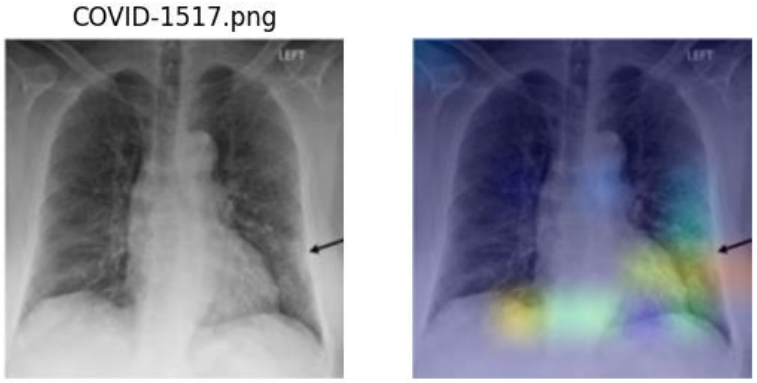
Dans les 3 cas, le modèle a prédit la classe « COVID » correctement. Les 3 Grad-CAM montrent une zone activée dans la partie inférieure droite du poumon et sur la zone du cœur, ce qui laisse penser que le masque **filtre des informations qui sont utiles à la classification** (par exemple, une démarcation moins claire avec le cœur et le diaphragme par rapport aux autres cas. En revanche, sur d'autres images (voir en annexe) on voit que le modèle active des zones qui ne devraient pas l'être (la clavicule ou le cou), ce qui pourrait être expliqué par des biais liés aux sources et à la prise d'image.



Grad-CAM « poumons seuls »



Grad-CAM « poumons masqués »



Grad-CAM « images complètes »

5. Conclusion et perspectives

a. Pertinence de la solution

En quelques semaines d'expérimentation, nous avons démontré qu'il était possible d'obtenir une classification juste à plus de 90% sur 4 classes (3 pathologies et le cas normal), grâce à un modèle de deep learning entraîné en apprentissage par transfert sur un jeu d'image radio en open-source.

Les publications scientifiques consultées pour le projet (voir bibliographie) montrent qu'il est possible d'obtenir une justesse de 95 à 99% sur des problématiques similaires à 2, 3 ou 4 classes.

Au cours de l'exploration de ce problème, nous avons été amenés à modéliser plusieurs approches et variantes :

- ⇨ Une classification dite « **approche directe »** consistant à fournir les radiographies avec leur masque selon 4 classes, tel que proposé par le jeu de donnée initial.
  - Nous avons comparé un réseau de neurones convolutifs de base (LeNet) avec plusieurs modèles d'apprentissage par transfert ;
  - Nous avons appliqué les techniques d'optimisation des hyperparamètres ;
  - Nous avons comparé les performances des modèles après dégel et réentraînement des couches de base ;
  - Nous avons obtenu une justesse de classification de 91% avec des modèles d'apprentissage par transfert utilisant EfficientNet et VGG19.
- ⇨ Une approche parallèle dite « **réduite »** consistant à trier les patients malades de patients sains sur la base exclusive des **profils d'intensités des radiographies**.
  - L'objectif de cette approche était d'obtenir des performances acceptables pour le tri des patients avec une économie de ressources de calcul et de stockage
  - Les différents modèles comparés variaient sur l'architecture des RNN et leurs hyperparamètres
  - Le meilleur modèle présente une justesse modeste de 80.1% mais une sensibilité de 94.7%.

A partir de l'analyse des résultats et de l'interprétation Grad-CAM, nous avons également réalisé plusieurs tests pour tenter d'améliorer la performance de classification :

- ⇨ **L'augmentation de donnée** en entrée des modèles, qui n'a pas permis d'améliorer les performances ;
- ⇨ L'ajout de **prétraitements supplémentaires** en entrée, qui n'a pas permis d'améliorer les performances ;
- ⇨ **L'exclusion d'une classe**, en raison de l'hétérogénéité des ces pathologies, qui a permis d'augmenter très légèrement les performances de justesse
- ⇨ **L'entraînement sur des images « poumons masqués »** et la comparaison des performances « poumons seuls » / « poumons masqués » / « image complète », qui a montré que les modèles parvenaient à obtenir une meilleure performance avec les images complètes, et encore meilleure avec les poumons masqués sur l'image…

Nous avons toutefois identifié plusieurs limites à la généralisation d'un tel modèle dans le monde médical.

b. Limites identifiées

La principale limite à la généralisation d'une telle approche est son **interprétabilité.** L'analyse des cartes d'activation « Grad-CAM » avec une radiologue n'a pas permis d'établir un lien clair entre les signes pathologiques recherchés par les professionnels et les attributs identifiés par les modèles dans la zone des poumons. Il faut pouvoir traduire les critères de classification utilisés par le modèle pour travailler l'acceptabilité d'une telle approche dans le monde professionnel.

De plus, le test sur les images « poumons masqués » a montré que les performances de classification étaient **meilleures si on ne fournit pas la zone des poumons aux modèles.** Nous pouvons donc nous interroger sur la pertinence des attributs extraits par les modèles vis-à-vis de la connaissance médicale sur les pathologies concernées.

La qualité du jeu de données est également une limite forte identifiée. Nos expérimentations nous ont conduit à émettre plusieurs hypothèses (qui n'ont pas été testés dans le temps alloué au projet) :

- Le jeu de données possède des **biais inhérents,** liés par exemple aux sources, aux réglages machines, à la pratique de la prise d'image, qui donnent aux modèles des « indices » non médicaux pour différencier les images. Une analyse plus poussée des sources montre que les classes sont issues de sources différentes avec peu de recouvrement (par exemple une seule source pour Viral Pneumonia)

	Classes			
Sources des données	**COVID**	**LUNG_ OPACITY**	**NORMAL**	**VIRAL_ PNEUMONIA**
BIMCV	2474			
EURORAD	258			
COVID-CXNet	400			
Covid-chestxray-dataset	182			
Covid-19-image-repository	183			
SIRM	119			
RNSA		6012	8851	
Chest-xray-pneumonia			1341	1345
Total général	**3616**	**6012**	**10192**	**1345**

- Une partie importante des images présente des artefacts (par exemple : électrodes visibles sur les poumons) n'ont pas pu être retirés des images ;
- Les masques fournis ne sont pas assez précis et excluent une partie des informations pertinentes d'un point de vue médical
- Les vraies classes comportent un taux d'erreur non négligeable. En effet, nous n'avons pas d'information sur la qualité du diagnostic initial. Le test PCR qui permet d'établir un diagnostic COVID n'a pas 100% de justesse !

Une troisième limite vient de **l'imprécision de l'objectif du projet**. En effet, la pratique médicale diffère selon le cas d'usage : diagnostic initial, examen complémentaire visant à exclure une pathologie possible, tri des patients dans le cas d'une pandémie, etc. Pour réaliser un diagnostic, un professionnel de santé se base sur un faisceau d'indices liés aux profils du patient (âge, sexe, etc.), à ses antécédents médicaux, au délai écoulé depuis les premiers symptômes, aux informations données fournies par les examens médicaux, à la consultation d'équipes pluridisciplinaires… Or notre jeu de données ne contient aucune métadonnée concernant les patients, l'apparition des symptômes, les traitements déjà réalisés ou en cours.

c. Potentiel de généralisation pour un usage réel

Les modèles construits et testés peuvent fournir une aide pour le triage rapide des patients en cas de pandémie par exemple. Mais le manque d'interprétabilité est un frein majeur. Les praticiens médicaux ont besoin d'outils qui leur permettent de confirmer un pré-diagnostic ou de compléter un ensemble d'examens. Ainsi, certains modèles trouvés dans la littérature sont entraînés pour **annoter les images radio avec les anomalies identifiées** (opacités, nodules, …) qui permettent au radiologue de confirmer son diagnostic ou d'effectuer des analyses supplémentaires.

La **disponibilité des données labellisées** est un frein pour l'entraînement et la généralisation de tels modèles à d'autres pathologies. En effet, malgré les jeux partagés en open-source par des équipes de chercheurs, il existe peu de données publiques (comparées aux bases « ImageNet » par exemple) et le diagnostic et l'annotation des images par des radiologues professionnels est coûteuse et chronophage.

Au-delà de la justesse du modèle, c'est tout le processus qui est à construire :

- A quel moment du diagnostic le modèle a-t-il sa place ?
- De quelles informations le ou la professionnelle a-t-elle besoin pour comprendre et valider le diagnostic fait par le modèle ?
- Comment concevoir un système qui permet un gain de temps par rapport au traitement humain ? (Puissance de calcul, intégration dans les outils existants, …)
- Comment traiter les cas où la prédiction du modèle est incertaine ou fausse ?

d. Bénéfice du projet pour l'équipe et difficultés rencontrées

L'équipe a grandement apprécié de pouvoir travailler sur ce projet avec les apports de la formation Datascientest. Le projet nous a permis de maîtriser très rapidement les techniques d'apprentissage profond, en avance par rapport au programme de formation.

Nous avons appris les bases théoriques des réseaux de neurones denses, convolutifs, du traitement d'images automatisé. Nous avons mis en application les bibliothèques Python disponibles en open-source (Tensorflow/Keras, OpencV, Scikit-learn…) et avons appris à utiliser Google Colab pour disposer de capacités de calcul sur GPU à coût raisonnable.

Le projet nous a permis de mettre en œuvre une démarche expérimentale, par essai-erreur, en forte autonomie dans la recherche des solutions et l'analyse des résultats. Nous avons dû travailler en équipe, construire les bases communes et organiser le travail collaboratif sur les expérimentations à effectuer. Nous avons consulté de nombreux articles de publications scientifiques ou sur le site Medium notamment, pour nous inspirer de projets réalisés et de techniques déjà testées.

L'apport du mentor projet a été très utile pour rythmer le projet, nous guider sur les pistes à envisager et challenger les résultats obtenus.

Le projet nous a demandé un investissement fort en temps et nous avons parfois dû prioriser entre le projet et les modules de formation à valider, notamment dans la phase de modélisation.

Il a parfois été difficile de choisir entre toutes les expérimentations possibles en tant que débutants ; et la littérature n'a pas toujours permis de nous guider (peu d'indications notamment sur le choix des modèles et des hyperparamètres pour un problème donné).

Nous avons rapidement été limités par la puissance de calcul de nos ordinateurs personnels et par les capacités gratuites offertes par Google ou Kaggle, et avons dû autofinancer des unités de calcul sur Colab (> 100€ cumulés).

e. Remerciements

Nous remercions l'équipe DataScientest pour l'opportunité de travailler sur ce sujet, Gaël Penessot qui a été notre mentor pour l'accompagnement tout au long du projet, et Martine Mattei (radiologue) pour ses éclairages médicaux et son regard critique sur les cartes d'activation Grad-CAM issues des modèles testés.

6. Annexes

a. Bibliographie

- Yadav, Ruchi & Sahoo, Debasis & Graham, Ruffin. (2020). **Thoracic imaging in COVID–19**. Cleveland Clinic Journal of Medicine. 87. 10.3949/ccjm.87a.ccc032.

- Wong HYF, Lam HYS, Fong AH, et al. **Frequency and distribution of chest radiographic findings in COVID-19 positive patients** [published online ahead of print, 2019 Mar 27]. Radiology 2019;201160.doi:10.1148/radiol.2020201160

- Narin A., Kaya C., Pamuk Z. **Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks**. arXiv preprint arXiv:2003.10849. 2020 [Google Scholar] [Ref list]

- Appasami G, Nickolas S. **A deep learning-based COVID-19 classification from chest X-ray image: case study**. Eur Phys J Spec Top. 2022;231(18-20):3767-3777. doi: 10.1140/epjs/s11734-022-00647-x. Epub 2022 Aug 18. PMID: 35996535; PMCID: PMC9386662.

- Sadre, R., Sundaram, B., Majumdar, S. *et al.* **Validating deep learning inference during chest X-ray classification for COVID-19 screening**. *Sci Rep* **11**, 16075 (2021). https://doi.org/10.1038/s41598-021-95561-y

- Talaat, M.; Si, X.; Xi, J. **Multi-Level Training and Testing of CNN Models in Diagnosing Multi-Center COVID-19 and Pneumonia X-ray Images**. *Appl. Sci.* **2023**, *13*, 10270. https://doi.org/10.3390/app131810270

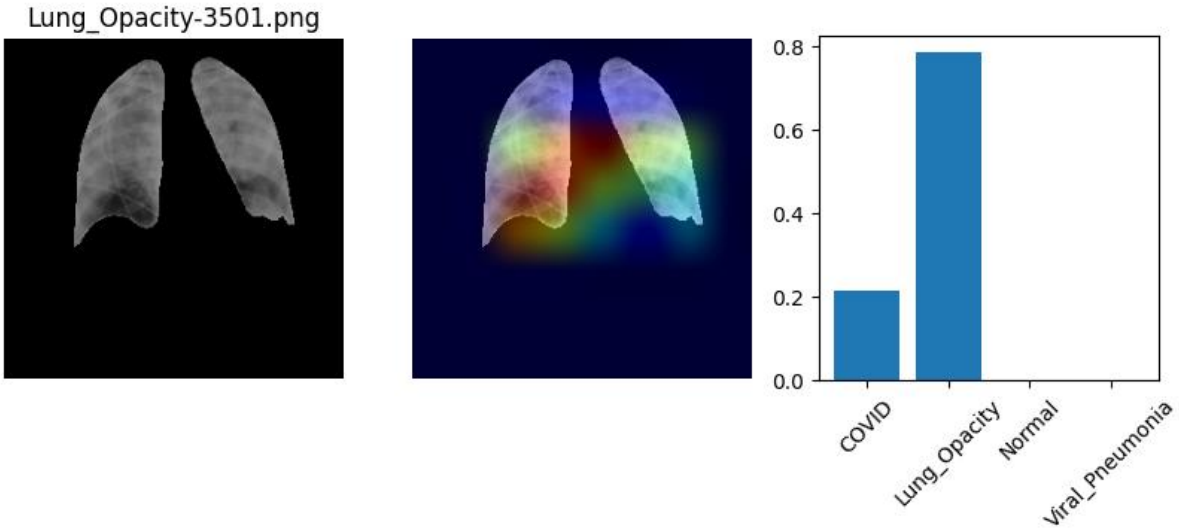
- Heidari M, Mirniaharikandehei S, Khuzani AZ, Danala G, Qiu Y, Zheng B. **Improving the performance of CNN to predict the likelihood of COVID-19 using chest X-ray images with preprocessing algorithms**. Int J Med Inform. 2020 Dec;144:104284. doi: 10.1016/j.ijmedinf.2020.104284. Epub 2020 Sep 23. PMID: 32992136; PMCID: PMC7510591.

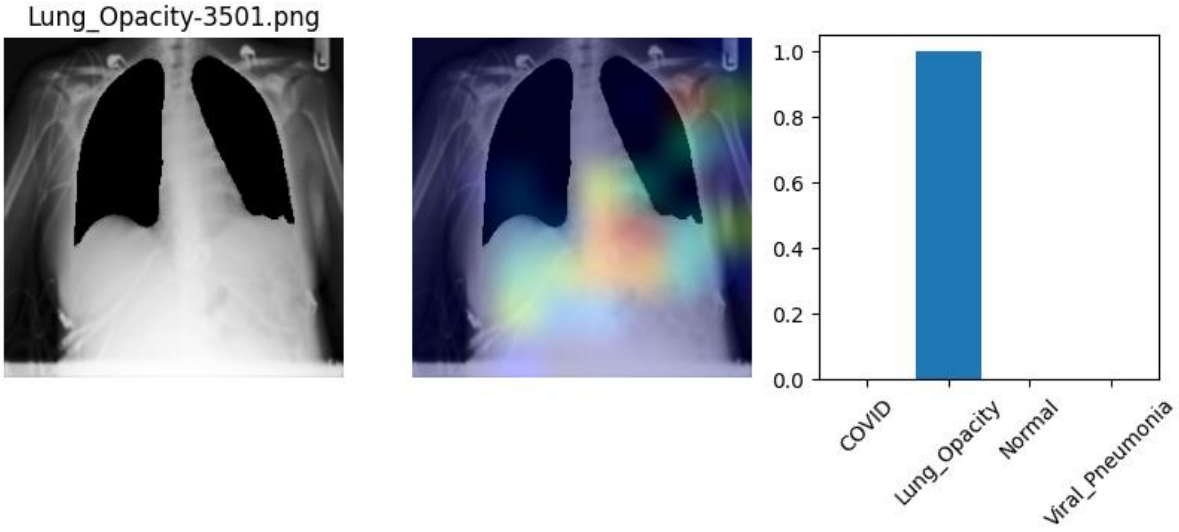
- Wang L, Lin ZQ, Wong A. **COVID-Net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images**. Sci Rep. 2020 Nov 11;10(1):19549. doi: 10.1038/s41598-020-76550-z. PMID: 33177550; PMCID: PMC7658227.

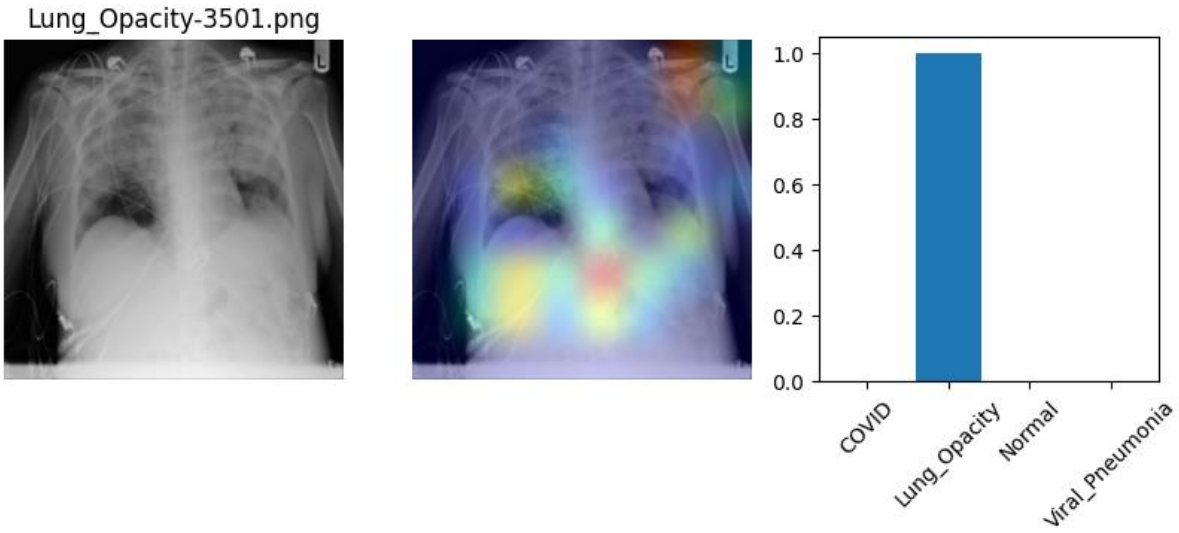
b. Grad-CAM comparées sur plusieurs images de chaque classe (poumons/sans poumons/images complètes)

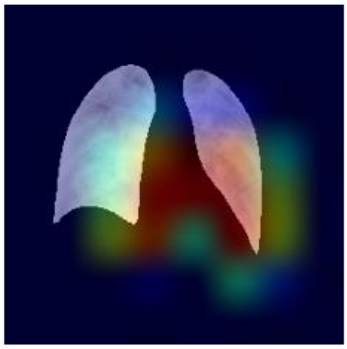
(Le graphique représente la probabilité prédite en sortie du modèle)

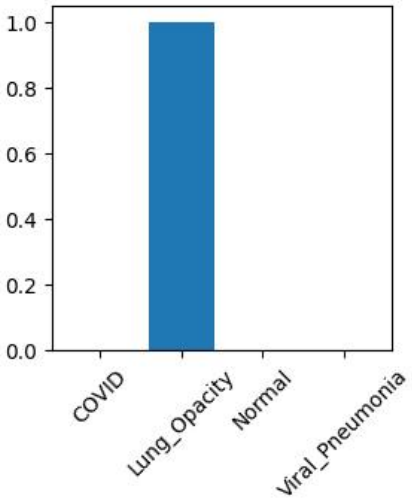
LUNG OPACITY - IMAGE 3501

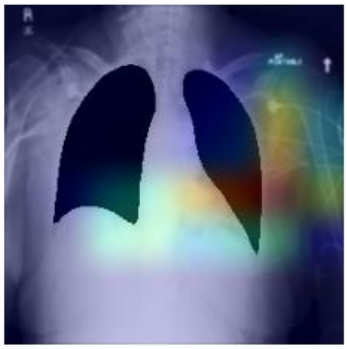


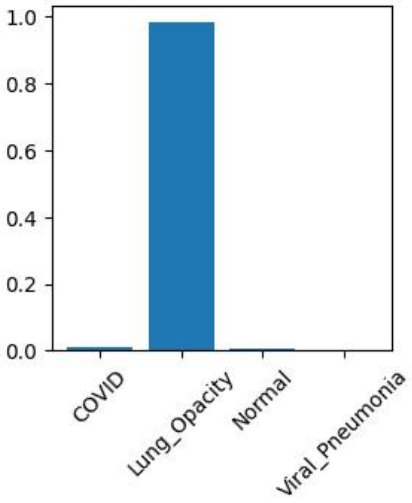


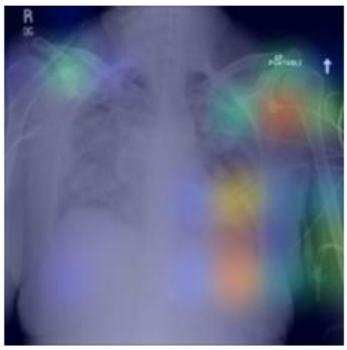


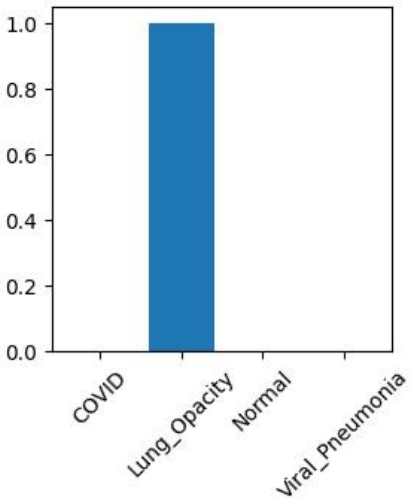








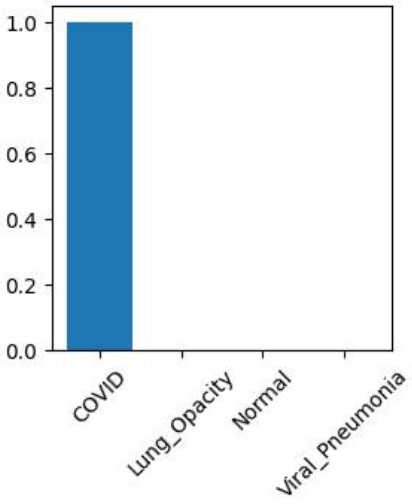




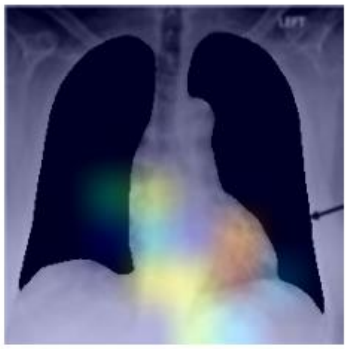
<u>**LUNG OPACITY - IMAGE 4183**</u>

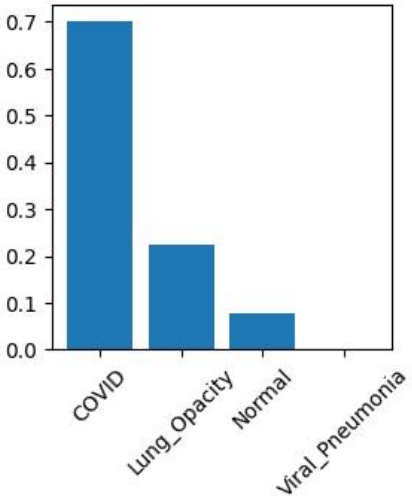
COVID - IMAGE 1517

COVID-1517.png

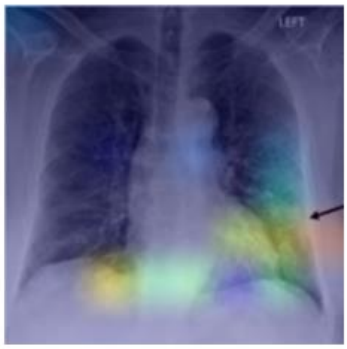


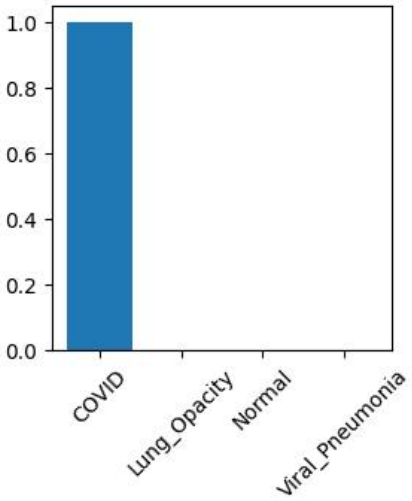
COVID-1517.png





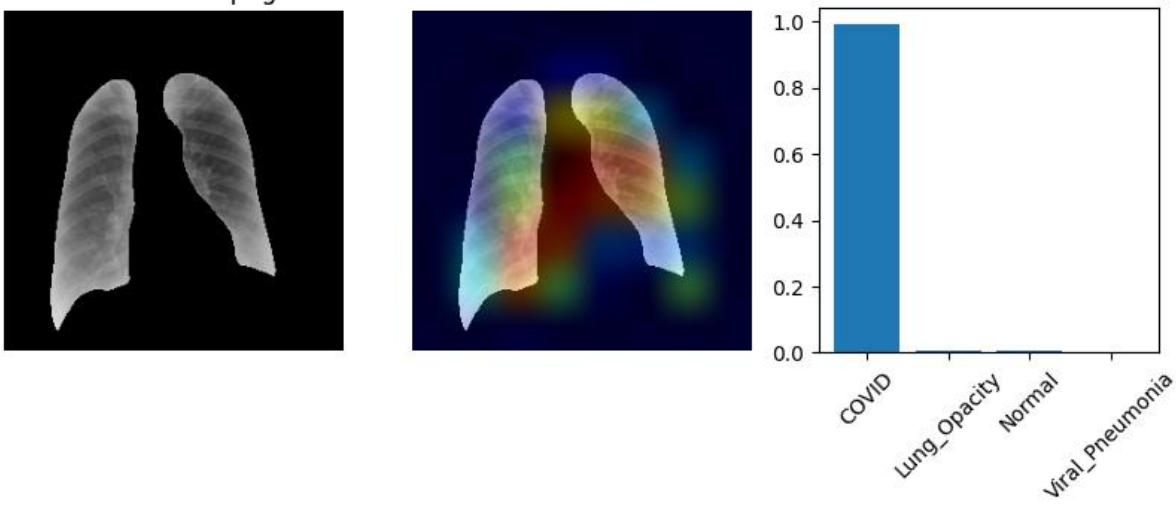
COVID-1517.png





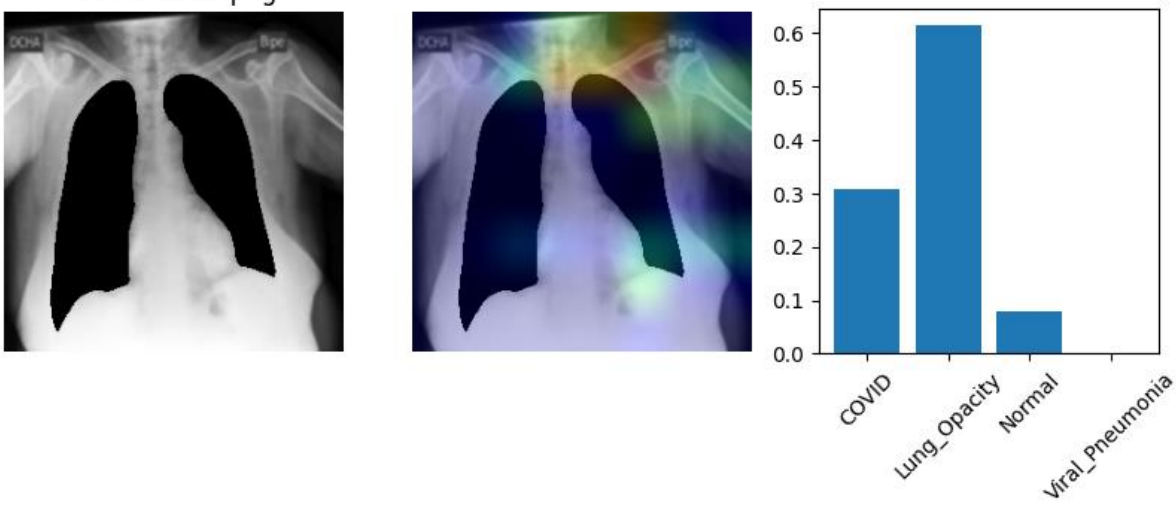
COVID - IMAGE 2910

COVID-2910.png



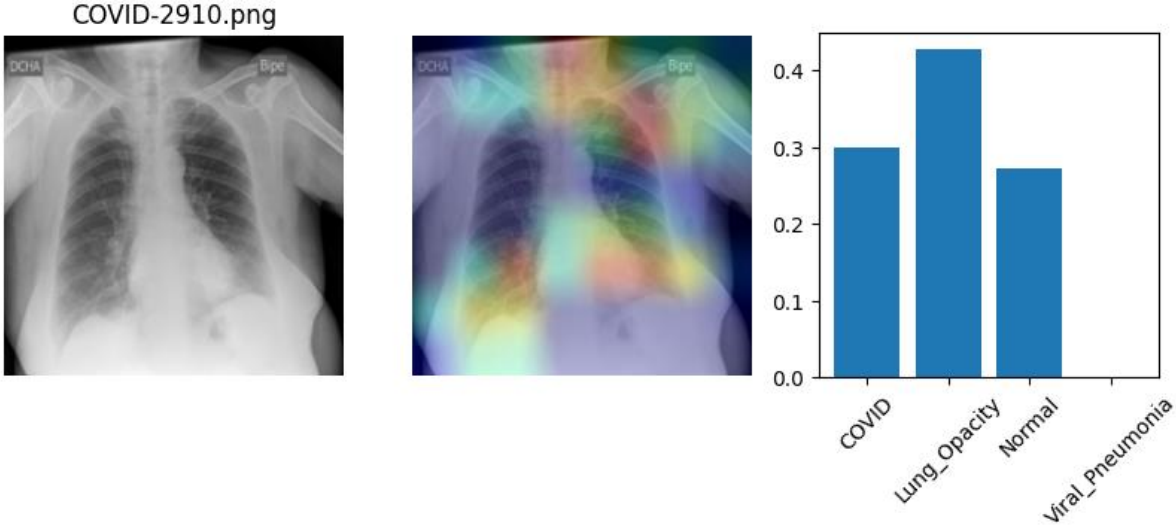
Predicted class : ['Lung_Opacity']

COVID-2910.png

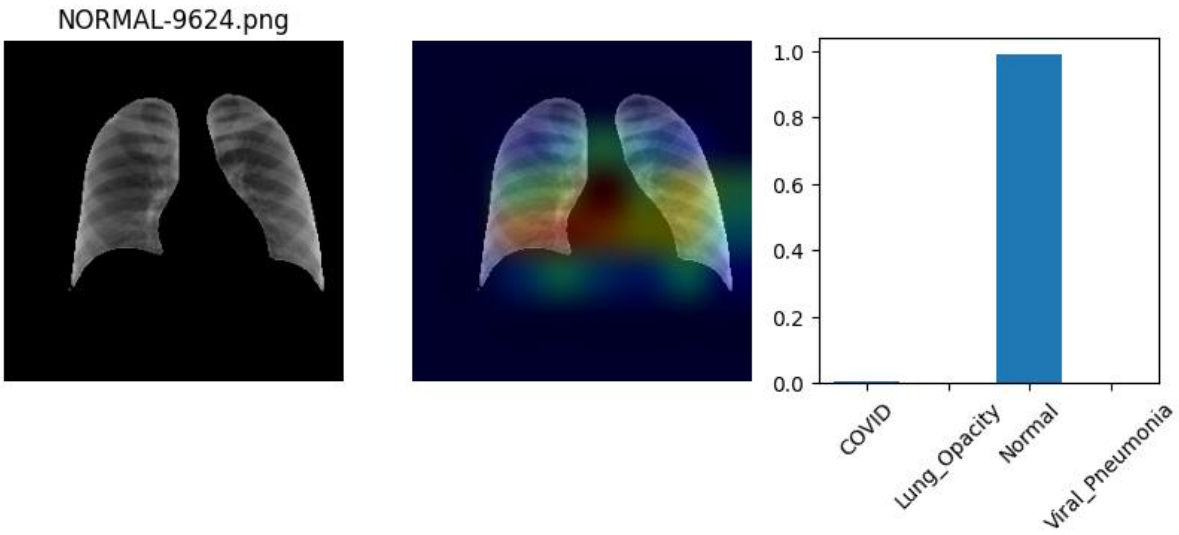


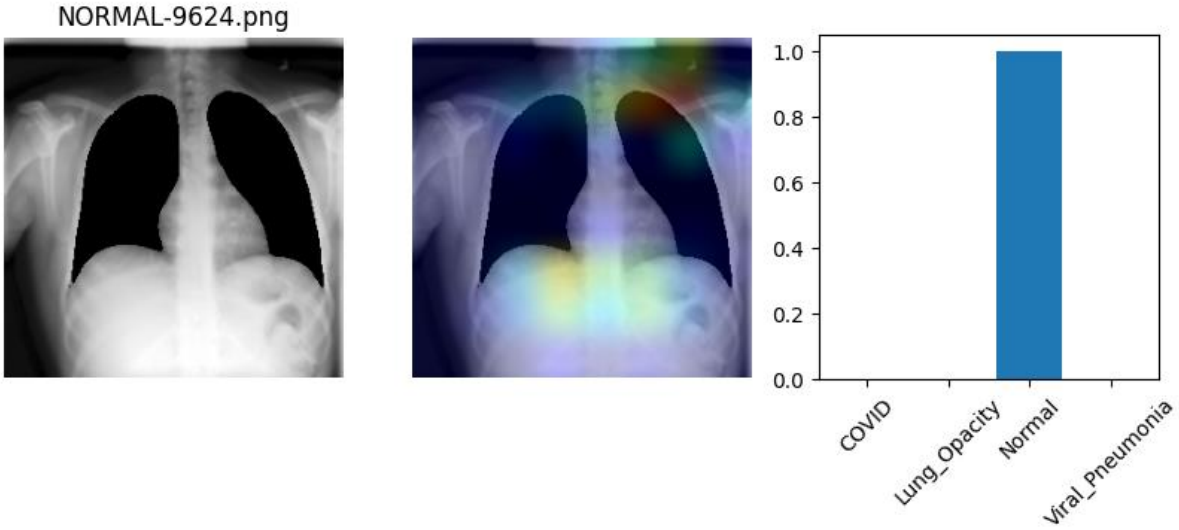
Predicted class : ['Lung_Opacity']

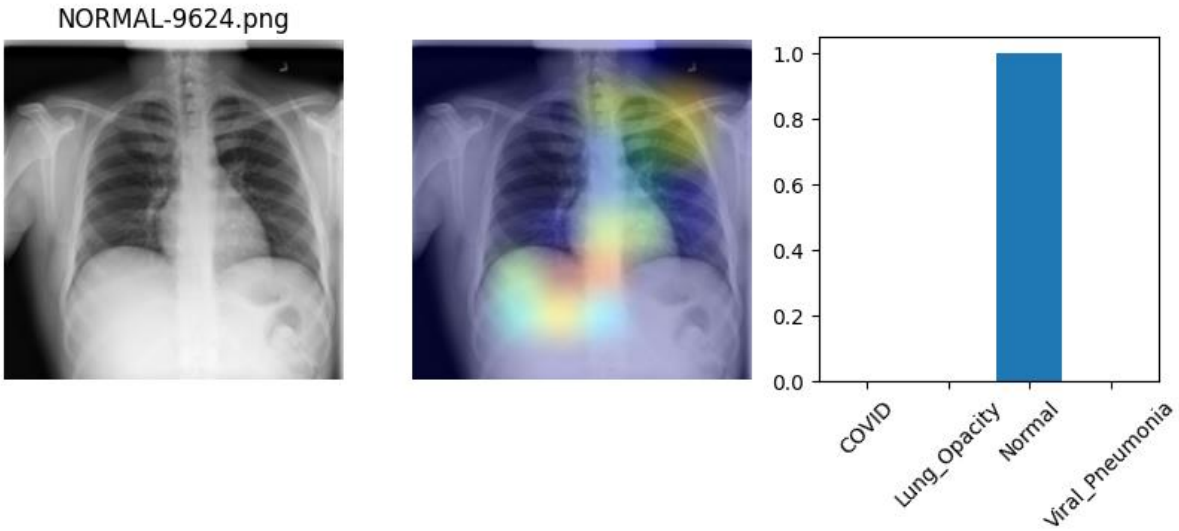
COVID-2910.png



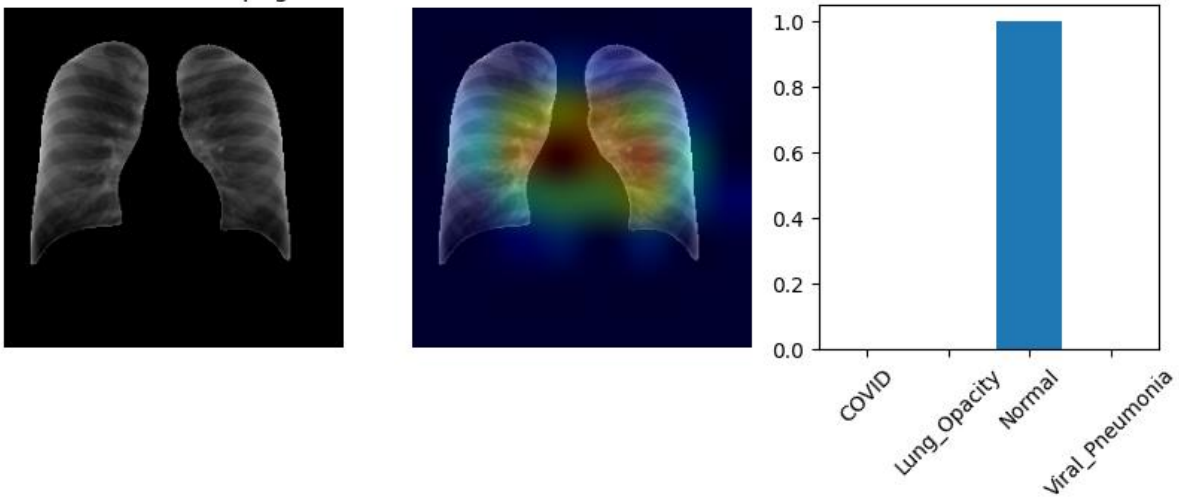
NORMAL - Image 9624

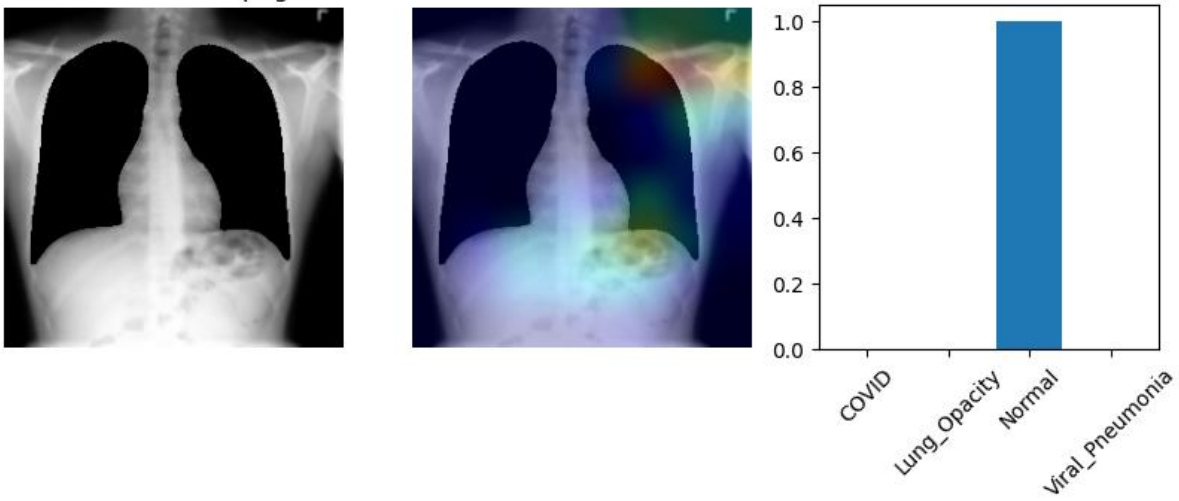


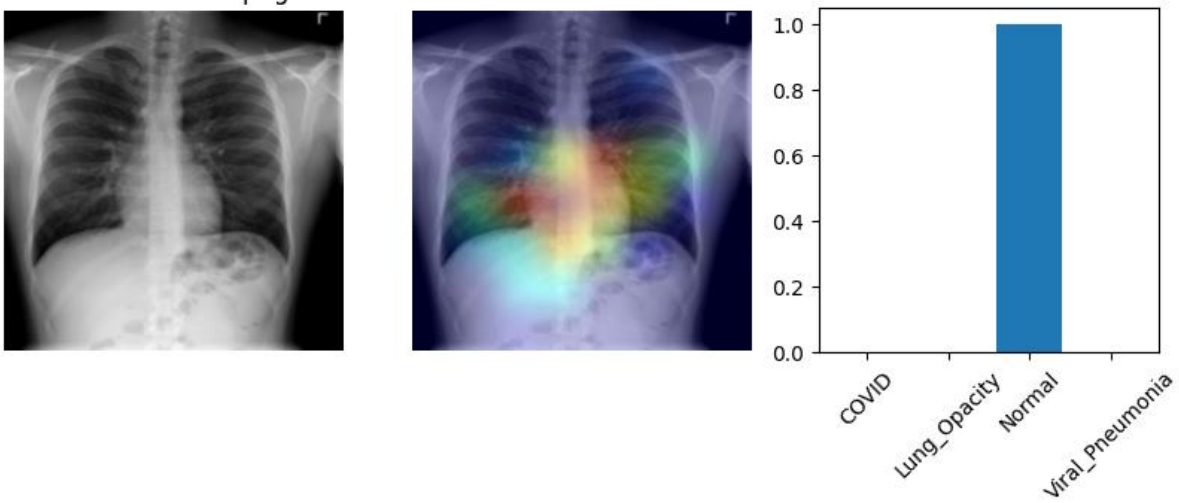


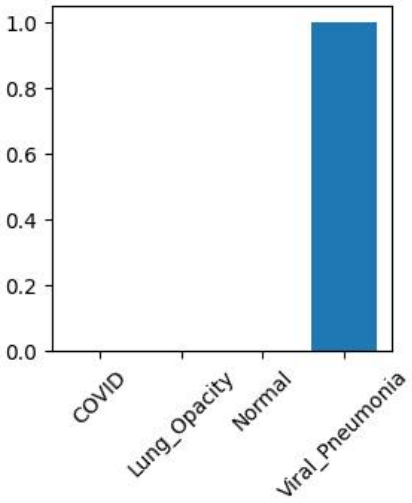


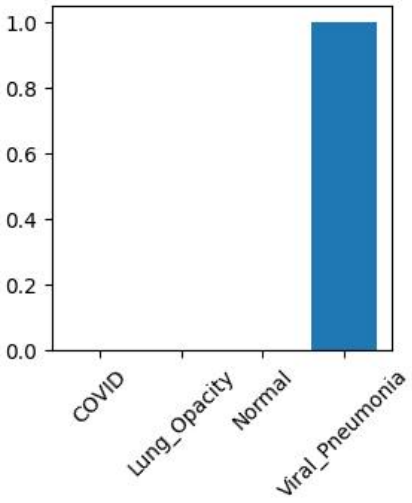
NORMAL - Image 9869

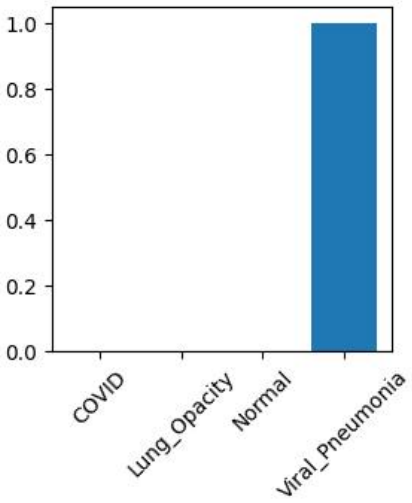












Viral PNEUMONIA - Image 674

Viral PNEUMONIA - Image 1190