

Note méthodologique

Preuve de concept pour la classification automatique d'images

Sommaire

1. Dataset retenu	2
1.1. Fichier de données.....	2
1.2. Dossier "Images".....	2
1.3. Catégorisation des articles.....	2
2. Concepts de l'algorithme EfficientNet-B7	3
2.1. Introduction.....	3
2.2. Principes clés d'EfficientNet.....	3
2.3. Spécificités du modèle EfficientNet-B7.....	4
3. Modélisation.....	5
3.1. Méthodologie de modélisation	5
3.2. Métrique d'évaluation	5
3.3. Démarche d'optimisation.....	5
4. Synthèse des résultats	6
4.1. Résultats précédents	6
4.2. Résultats EfficientNet-B7	7
4.3. Conclusion.....	7
5. Analyse de la feature importance du nouveau modèle	8
6. Limites et les améliorations possibles.....	9
6.1. Limites du modèle	9
6.2. Perspectives d'amélioration	9

1. Dataset retenu

Au sein de l'entreprise "Place de Marché", l'automatisation de l'attribution des catégories aux articles est devenue un enjeu majeur pour améliorer l'expérience des utilisateurs. Actuellement réalisée manuellement par les vendeurs, cette tâche souffre de fiabilité et d'évolutivité limitées. Pour cette preuve de concept, nous nous concentrerons sur la classification automatique d'images pour attribuer une catégorie aux objets en fonction de l'image associée au produit.

1.1. Fichier de données

Le jeu de données est contenu dans le fichier **flipkart_com-ecommerce_sample_1050.csv**. Il comprend les informations suivantes :

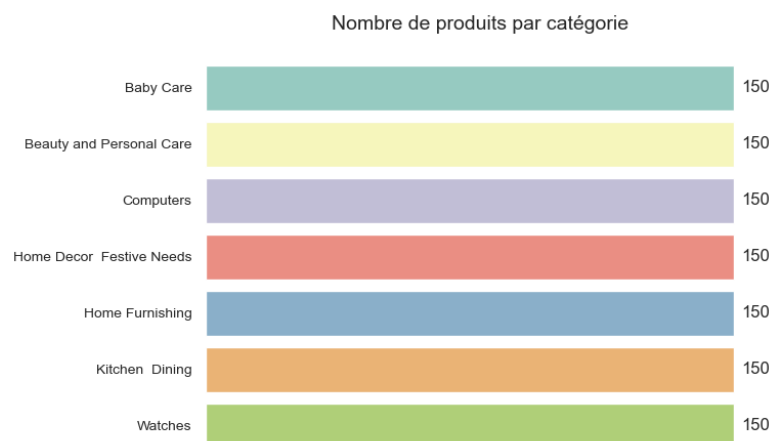
- **Nombre de colonnes** : 15
- **Principales colonnes** : Nom des produits, Catégorie, Description, Nom de l'image associée, Prix, etc.
- **Nombre de lignes** : 1050

1.2. Dossier "Images"

En complément du fichier CSV, un dossier nommé **"Images"** est fourni. Il contient 1050 images correspondant aux articles répertoriés dans le fichier de données. Chaque image est associée à un article via le nom du fichier, qui est référencé dans la colonne "Nom de l'Image Associée" du fichier CSV.

1.3. Catégorisation des articles

Les articles sont classés en sept catégories principales, avec des sous-catégories pour chaque catégorie principale. La catégorisation actuelle est effectuée manuellement par les vendeurs et est sujette à des erreurs et à une faible fiabilité, ce qui justifie le besoin d'automatiser ce processus de classification.



2. Concepts de l'algorithme EfficientNet-B7

2.1. Introduction

Les réseaux de neurones convolutifs (ConvNets ou CNN) sont largement utilisés pour la reconnaissance et la classification d'images. Traditionnellement, leur taille a été augmentée pour une meilleure précision, dépassant souvent les limitations des ressources informatiques. EfficientNet propose une nouvelle méthode de mise à l'échelle "compound scaling" qui équilibre la profondeur, la largeur et la résolution du réseau pour une performance et une efficacité optimales.

- Article de recherche : <https://arxiv.org/pdf/1905.11946.pdf>
- Référentiel GitHub : <https://github.com/tensorflow/tpu/tree/master/models/official/efficientnet>

2.2. Principes clés d'EfficientNet

- **Mise à l'échelle équilibrée** : Contrairement aux approches conventionnelles qui ne mettent à l'échelle qu'une seule dimension (profondeur, largeur ou résolution), EfficientNet met à l'échelle uniformément les trois dimensions en utilisant des ratios constants.

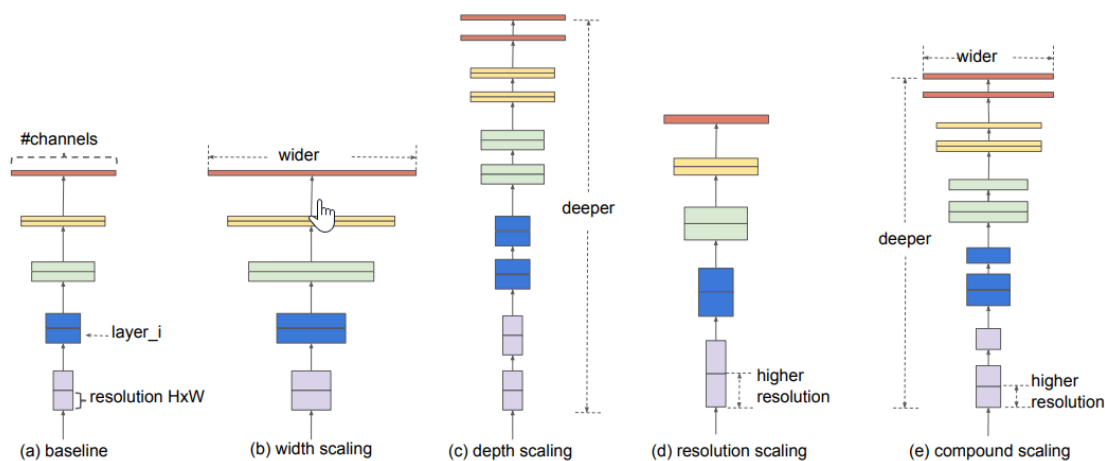


Figure 2. **Model Scaling.** (a) is a baseline network example; (b)-(d) are conventional scaling that only increases one dimension of network width, depth, or resolution. (e) is our proposed compound scaling method that uniformly scales all three dimensions with a fixed ratio.

Figure 1 : Source <https://arxiv.org/pdf/1905.11946.pdf>

- **Formule de mise à l'échelle composée** : L'idée principale est incarnée dans la formule suivante :
 - α affecte la profondeur du réseau (nombre de couches)
 - β affecte la largeur du réseau (nombre de canaux)
 - γ affecte la résolution de l'image

Ces ratios sont déterminés par une recherche sur une grille sur un modèle plus petit pour une mise à l'échelle optimale.

- **Justification** : Cette approche équilibrée est logique car une image d'entrée plus grande nécessite plus de couches pour traiter le champ réceptif élargi et plus de canaux pour capturer des détails plus fins.

- **Soutien théorique et empirique** : Des études antérieures ont établi l'importance de la profondeur et de la largeur pour l'expressivité du réseau. EfficientNet quantifie la relation entre les trois dimensions par des études empiriques.

2.3. Spécificités du modèle EfficientNet-B7

EfficientNet-B7 est un modèle de la famille EfficientNet qui utilise la méthode de mise à l'échelle "compound scaling". Il a été sélectionné pour ses excellentes performances et son efficacité parmi les modèles EfficientNets.

Avantages d'EfficientNet-B7 :

- **Précision de pointe** : Atteint une précision de **84,3 % sur ImageNet**, surpassant les modèles ConvNets existants avec un **nombre de paramètres 8,4 fois plus petit** et une **vitesse d'inférence 6,1 fois plus rapide**.

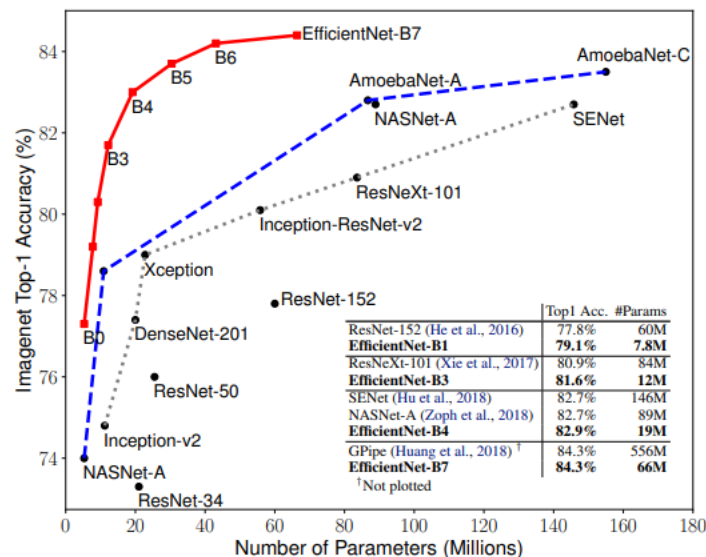


Figure 2 : Source <https://arxiv.org/pdf/1905.11946.pdf>

- **Efficacité optimale** : Offre un équilibre optimal entre la précision et l'efficacité, le rendant idéal pour les applications où les ressources sont limitées.
- **Flexibilité** : Peut être utilisé pour une variété de tâches de vision par ordinateur, telles que la classification d'images, la détection d'objets et la segmentation sémantique.

Utilisation d'EfficientNet-B7 :

Le modèle peut être utilisé directement via des frameworks d'apprentissage automatique tels que TensorFlow et Keras. Il est également possible de le personnaliser en modifiant ses paramètres ou en l'utilisant comme base pour construire un nouveau modèle.

En résumé, EfficientNet-B7 propose une approche révolutionnaire pour la mise à l'échelle des ConvNets en équilibrant soigneusement la profondeur, la largeur et la résolution en utilisant une méthode de mise à l'échelle composée. Cela se traduit par des améliorations significatives des performances tout en maintenant l'efficacité, ce qui en fait un outil précieux pour diverses applications de reconnaissance d'images et d'apprentissage profond.

3. Modélisation

3.1. Méthodologie de modélisation

La classification a été réalisée à l'aide du modèle pré-entraîné EfficientNetB7 et de l'apprentissage par transfert.

La démarche a été la suivante :

- **Création des datasets** : La fonction `dataset_fct` génère les ensembles de données d'entraînement, de validation et de test à partir des images et labels dans les répertoires `Datacat/train` et `Datacat/test`.
- **Création du modèle** : La fonction `create_EfficientNet_model` charge le modèle EfficientNetB7 pré-entraîné sur ImageNet, congèle ses couches, ajoute des couches personnalisées (couches denses et Dropout) et compile le modèle avec l'optimiseur RMSprop et la perte catégorique croisée.
- **Entraînement du modèle** : Le modèle est entraîné sur l'ensemble de données d'entraînement avec validation croisée (25% pour la validation) et arrêt précoce (EarlyStopping) pour éviter le surajustement.

3.2. Métrique d'évaluation

Pour évaluer notre modèle, nous avons utilisé l'accuracy.

Matrice de confusion			
		Réalité	
		0	1
Prédiction	0	TN : True Negatives	FN : False Negatives
	1	FP : False Positives	TP : True Positives

- **Accuracy** : Mesure le pourcentage de prédictions correctes par rapport au nombre total d'échantillons. Elle est simple à interpréter mais peut être trompeuse en cas de déséquilibre de classes (ce qui n'était pas le cas ici).

- **Formule** :
$$\frac{TP+TN}{Total}$$

3.3. Démarche d'optimisation

Enfin, pour optimiser notre modèle, nous avons mis en place les éléments suivants :

- **Fonction de perte** : « Categorical cross-entropy », qui mesure l'écart entre la distribution de probabilité prédite et la distribution de probabilité réelle.
- **Optimiseur** : « RMSprop », un algorithme de descente de gradient efficace pour les problèmes d'apprentissage profond.

- **Early stopping** : Arrêt automatique de l'entraînement lorsque la perte sur le jeu de validation ne s'améliore plus pendant un certain nombre d'époques consécutives.
- **Checkpoint** : Sauvegarde du modèle avec la meilleure performance sur le jeu de validation afin d'éviter le surapprentissage.

4. Synthèse des résultats

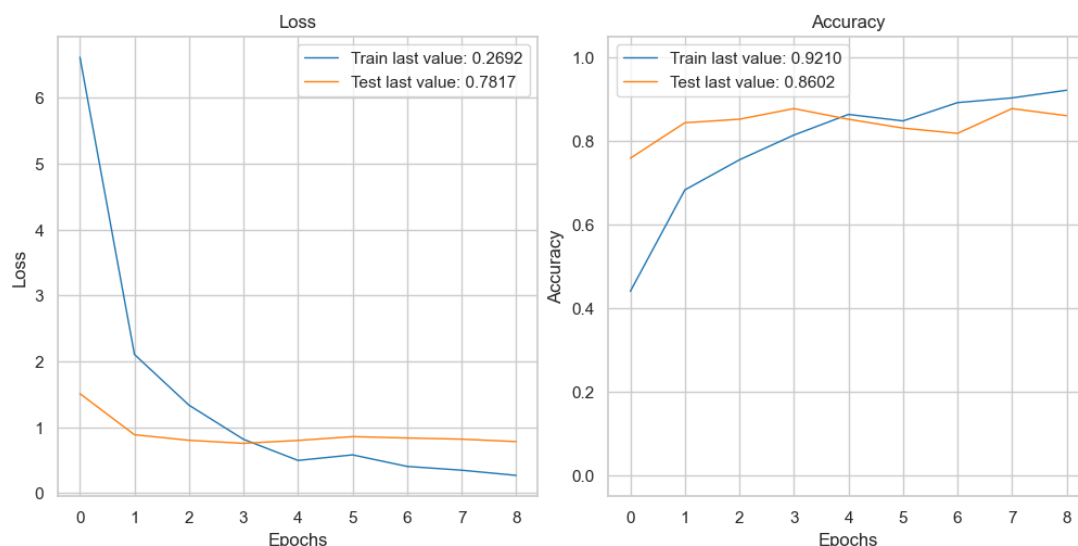
Nous allons à présent comparer les résultats obtenus avec les techniques classiques des réseaux de neurones convolutifs avec les résultats obtenus avec EfficientNet-B7.

4.1. Résultats précédents

Lors du projet 6 OpenClassrooms – « Classifiez automatiquement des biens de consommation », nous avons testé différents modèles pour lesquels nous avons obtenu les résultats suivants :

	Classification supervisée simplifiée (VGG16)	ImageDatagenerator avec augmentation des données (VGG16)	Nouvelle approche avec augmentation intégrée des données (VGG16)	Nouvelle approche avec augmentation intégrée des données (VGG19)	Nouvelle approche avec augmentation intégrée des données (ResNet50)
Validation Accuracy (epoch optimal)	0.8354	0.8517	0.8644	0.8771	0.8559
Test Accuracy (epoch optimal)	0.6000	0.5333	0.5714	0.6381	0.5524
Durée Training (dernier epoch)	336s	348s	312s	311s	96s
Durée Validation (dernier epoch)	97s	116s	105s	103s	35s

Nous avons donc déterminé que le modèle VGG19 avec la nouvelle approche « Augmentation intégrée des données » avait fourni les meilleurs résultats. Pour ce qui est de la fonction de perte et l'accuracy, voici les résultats obtenus :



4.2. Résultats EfficientNet-B7

Après l'entraînement du modèle EfficientNet, nous avons évalué ses performances sur les ensembles de données d'entraînement et de validation. Voici les résultats obtenus :

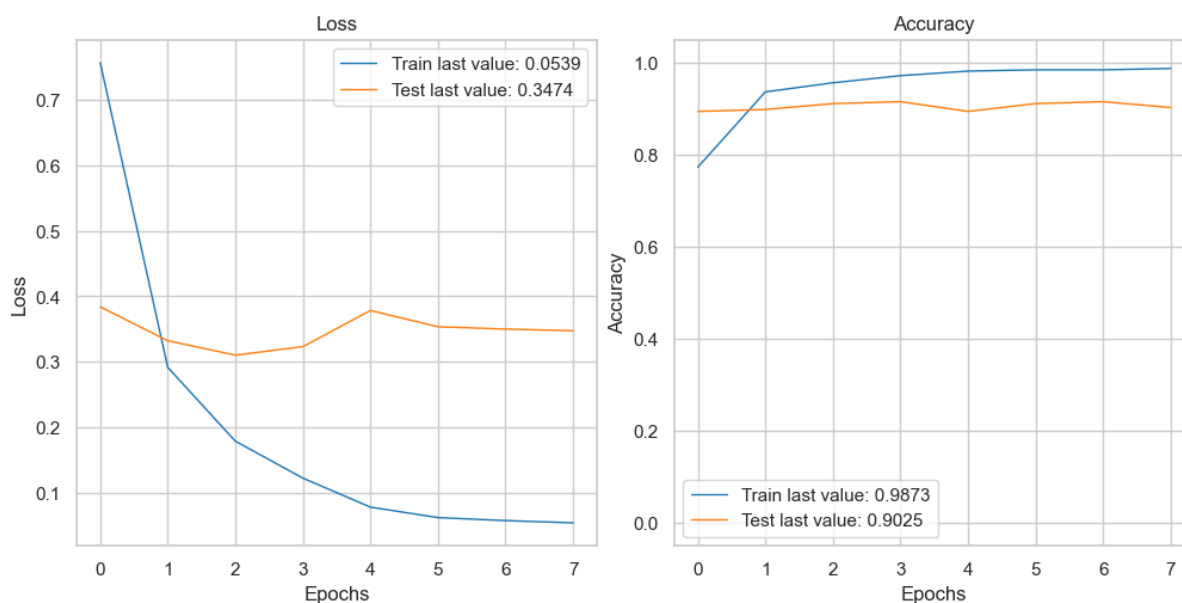
- Training Accuracy : **98.87%**
- Validation Accuracy : **90.25%**

Ces scores témoignent d'une très bonne performance du modèle sur les données d'entraînement et de validation.

Après avoir chargé les poids du modèle correspondant à l'epoch optimal, nous avons évalué ses performances sur l'ensemble de données de validation et de test. Les résultats sont les suivants :

- Validation Accuracy : **91.10%**
- Test Accuracy : **72.38%**

Ces résultats confirment l'efficacité du modèle, notamment sur l'ensemble de validation où il atteint une précision de 91.10%. Pour ce qui est de la fonction de perte et l'accuracy, voici les résultats obtenus :



4.3. Conclusion

	Résultats précédents	EfficientNet-B7
Validation Accuracy (Epoch optimal)	87,71 %	91,10 %
Test Accuracy (Epoch optimal)	63,81 %	72,38 %

En conclusion, les résultats démontrent clairement que l'utilisation d'EfficientNet-B7 a conduit à une amélioration significative de l'accuracy, tant au niveau de la validation que du test, par rapport aux résultats obtenus avec les méthodes précédentes.

5. Analyse de la feature importance du nouveau modèle

Pour évaluer l'importance des caractéristiques (features) dans la décision du modèle EfficientNet-B7, nous avons utilisé la méthode SHAP (SHapley Additive exPlanations). Cette technique nous permet d'interpréter les prédictions de notre modèle en attribuant des valeurs Shapley à chaque caractéristique d'une observation.

Nous avons suivi les étapes suivantes pour cette analyse :

- **Étape 1 : Préparation des données pour SHAP**

Nous avons sélectionné une image représentative de la catégorie "Beauty and Personal Care" pour notre analyse. L'image a été redimensionnée à une taille de 224x224 pixels, puis prétraitée pour être conforme aux exigences de l'exploration SHAP.

- **Étape 2 : Définition de la fonction pour SHAP**

Nous avons défini une fonction, appelée $f(X)$, qui prend en entrée les données et renvoie les prédictions du modèle EfficientNet-B7.

- **Étape 3 : Configuration du masker pour les images**

Nous avons utilisé le masker "inpaint_telea" de SHAP pour masquer les parties de l'image et évaluer l'impact de différentes régions sur la prédiction du modèle.

- **Étape 4 : Création de l'explainer SHAP et calcul des valeurs Shapley**

En utilisant la fonction `shap.Explainer()`, nous avons créé un explainer SHAP pour notre modèle. Cet explainer nous permet d'explorer l'importance des caractéristiques dans les prédictions du modèle.

- **Étape 5 : Visualisation des valeurs Shapley**

Les valeurs Shapley calculées ont été visualisées à l'aide de la fonction `shap.image_plot()`. Cette visualisation nous permet de comprendre quelles parties de l'image contribuent le plus à la prédiction du modèle. En exemple, voici l'image analysée :



Cette analyse de la feature importance à l'aide de SHAP nous fournit des informations précieuses sur les décisions prises par le modèle EfficientNet-B7, ce qui nous aide à mieux comprendre son fonctionnement et à interpréter ses prédictions.

6. Limites et les améliorations possibles

Malgré les avancées significatives apportées par l'algorithme EfficientNet-B7 et les résultats encourageants obtenus dans le cadre de cette étude de preuve de concept, plusieurs limites et opportunités d'amélioration nécessitent une attention particulière.

6.1. Limites du modèle

- **Dépendance aux données d'entraînement** : La performance du modèle dépend de la qualité et de la représentativité des données d'entraînement. Des biais dans les données peuvent influencer négativement la capacité du modèle à généraliser.
- **Interprétabilité des modèles** : Comprendre les décisions des modèles de deep learning est un défi. La complexité des réseaux neuronaux rend difficile la compréhension des facteurs derrière une prédiction.
- **Évolution des données et des exigences** : Les données, les tendances du marché et les préférences des consommateurs évoluent constamment. Les modèles doivent s'adapter rapidement pour rester pertinents.

6.2. Perspectives d'amélioration

- **Diversification des données d'entraînement** : En élargissant la diversité des sources de données et en incorporant des variations plus vastes d'images, il est possible d'améliorer la capacité du modèle à généraliser et à traiter un éventail plus large de scénarios d'utilisation.
- **Optimisation des hyperparamètres** : Explorer les paramètres du modèle et utiliser des techniques d'optimisation avancées peut améliorer les performances du modèle.
- **Évaluation de diverses architectures de modèles** : Outre EfficientNet-B7, l'évaluation d'autres architectures de réseaux neuronaux et de méthodes d'apprentissage automatique peut offrir des perspectives nouvelles et complémentaires pour la résolution des défis spécifiques à la classification d'images.

En conclusion, bien que les progrès réalisés jusqu'à présent soient prometteurs, l'amélioration continue et la recherche de solutions innovantes demeurent essentielles pour consolider l'efficacité et la fiabilité des systèmes de classification d'images dans le contexte dynamique du commerce en ligne.