

الجمهورية الجزائرية الديمقراطية الشعبية
République algérienne démocratique et populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'enseignement supérieur et de la recherche scientifique
جامعة عين تموشنت بلحاج بوشعيب
Université –Ain Témouchent- Belhadj Bouchaib
Faculté des Sciences et de Technologie
Département d'Électronique et des Télécommunications



Projet de Fin d'Etudes
Pour l'obtention du diplôme de Master en :
Domaine : Sciences et Technologie
Filière : TELECOMMUNICATIONS
Spécialité : Réseaux et Télécommunications
Thème

Arabic Image Captioning Using GoogLeNet

Présenté Par :

- 1) Melle BENSALAH Dounia Abir
- 2) Mr BOULEFDAOUI Abdelaziz

Devant le jury composé de :

Mme BADIR. B Lahouaria	MCB UAT.B. B (Ain Témouchent)	Président
Mr HACHEMI Belkacem	MCB UAT.B. B (Ain Témouchent)	Examineur
Mme MEKAMI Hayat	MCB UAT.B. B (Ain Témouchent)	Encadrant
Mme Boutkhil Malika	MAA UAT.B. B (Ain Témouchent)	Co-Encadrante

Année Universitaire 2025/2026

Dédicaces

C'est avec tout mon amour et ma reconnaissance que je dédie ce travail...

À moi-même,

En reconnaissance de ma persévérance et de mon courage pour mener à bien ce projet académique ; j'espère que mes efforts seront couronnés de succès.

À mes chers parents, mon père Jamal et ma mère bien-aimée Houria,

Les mots ne sauraient exprimer l'amour et la gratitude que je vous porte. Merci pour votre soutien inconditionnel, vos sacrifices incessants, ainsi que pour vos prières et vos encouragements constants. Vous êtes la source de ma force et la lumière de mes jours ; que Dieu vous protège et veille sur vous.

À mes chers frères et sœurs : Boucif, Tahar, Chamse Eddine et Hiba,

Pour leur amour, leur soutien et leur dévouement, ainsi que pour tous les beaux moments que nous avons partagés. Merci d'être toujours à mes côtés.

À toute ma famille bien-aimée et tout particulièrement à ma grand-mère, que Dieu vous prête longue vie.

Petits et grands, merci pour votre affection et vos vœux de réussite.

À Ayoub,

En reconnaissance de ton soutien indéfectible, de ta patience et de tes encouragements constants tout au long de ce parcours. Merci d'avoir été un véritable soutien pour moi.

À mes chères amies : Sarah, Rahmouna, Nour et Manal,

Merci pour votre présence chaleureuse, votre gentillesse et vos encouragements précieux ; vous êtes de véritables perles dans ma vie.

À mon cher cousin, Mohamed,

Que ton âme repose en paix. Tu nous as quittés bien trop tôt, mais tu resteras à jamais gravé dans nos cœurs. Je prie Dieu de t'accorder une place dans Son vaste Paradis.

Dounia

Dédicaces

Je dédie ce modeste travail à :

- ❖ *Mes très chers parents (**Zakia et Saïd**), pour votre éducation, vos encouragements, votre disponibilité et affection, durant tous mes études ; Qu'ALLAH vous protège.*
- ❖ *A mes chères sœurs **Asmaa et Amina**, pour leur soutien moral, disponibilité et leur encouragement. Elles ont toujours été là pour moi, Qu'ALLAH les gardes.*
- ❖ *A toute la famille **Boulefdaoui et Belloula**.*
- ❖ *A notre promotrice **Mme. Mekami Hayat**, qui a été toujours avec nous.*
- ❖ *A tous ceux qui ont été à mes côtés de près ou de loin*

Abdelaziz

Remerciements

Je tiens tout d'abord à remercier Dieu, le Tout-Puissant, de m'avoir donné la force, la patience et le courage pour mener à bien ce travail

Je tiens à exprimer ma profonde gratitude à notre encadrante, **Mme MEKAMI H.**, ainsi qu'à ma Co-encadrante, **Mme BOUTKHIL M.**, pour leur accompagnement précieux, leurs conseils avisés et leur soutien constant tout au long de la réalisation de ce mémoire.

Je remercie chaleureusement les membres du jury pour l'honneur qu'ils me font en acceptant d'évaluer ce travail.

Mes remerciements les plus sincères s'adressent à **Algérie Télécom**, qui m'a offert l'opportunité d'effectuer mon stage au sein de son établissement. Je tiens particulièrement à remercier **M. Habi Sidi Mohammed**, Chef du département des ressources humaines, pour son accueil chaleureux, sa disponibilité, son professionnalisme ainsi que pour la confiance qu'il m'a accordée. Je lui suis profondément reconnaissante pour ses précieux conseils, son accompagnement, son soutien constant et les excellentes conditions de travail mises à ma disposition tout au long de cette expérience, qui a grandement contribué à enrichir mes connaissances et à renforcer mes compétences professionnelles.

J'adresse également mes sincères remerciements à **la société Sonelgaz** pour son accueil et pour m'avoir permis de découvrir son environnement professionnel. Je remercie l'ensemble de son personnel pour leur disponibilité, leur bienveillance et leur collaboration, qui ont contribué au bon déroulement de cette expérience.

Je remercie chaleureusement les membres du jury pour l'honneur qu'ils me font en acceptant d'évaluer ce travail.

Un grand merci à tous mes enseignants pour la qualité de leur formation, leur disponibilité et leur dévouement tout au long de mon parcours universitaire.

Je tiens aussi à remercier du fond du cœur ma famille et mes proches pour leur soutien moral, leur patience et leurs encouragements tout au long de mon parcours.

Enfin, je remercie toute personne ayant contribué, de près ou de loin, à l'élaboration de ce travail.

Résumé

Ce mémoire s'inscrit dans le domaine de la génération automatique de descriptions d'images en langue arabe (Arabic Image Captioning), une problématique qui associe la vision par ordinateur (Computer Vision) au traitement automatique du langage naturel (Natural Language Processing – NLP). L'objectif est de concevoir et de mettre en œuvre un système capable d'interpréter le contenu d'une image et de générer automatiquement une description textuelle pertinente en langue arabe.

La première partie de ce travail présente les concepts fondamentaux de l'Image Captioning, son évolution, son principe de fonctionnement ainsi que ses principaux domaines d'application, notamment l'assistance aux personnes malvoyantes, le secteur médical, le commerce électronique, les réseaux sociaux et les plateformes d'apprentissage en ligne. Les architectures les plus utilisées sont également étudiées, en particulier le modèle Encodeur-Décodeur, dans lequel un réseau de neurones convolutif (CNN) assure l'extraction des caractéristiques visuelles, tandis qu'un décodeur basé sur des réseaux LSTM ou sur les Transformers est chargé de générer progressivement la description textuelle. Une attention particulière est portée aux mécanismes d'attention (Attention Mechanism), qui permettent d'améliorer la qualité des légendes en orientant le modèle vers les régions les plus pertinentes de l'image.

La seconde partie est consacrée à la mise en œuvre du système proposé. Elle décrit l'environnement de développement, le jeu de données Flickr8k, ainsi que les différentes étapes de prétraitement des images et des descriptions textuelles. Les caractéristiques visuelles sont extraites à l'aide de la technique du Transfer Learning en exploitant deux modèles pré-entraînés, VGG16 et GoogLeNet (InceptionV3), dont les performances sont comparées. Les descriptions en arabe font également l'objet d'un prétraitement comprenant le nettoyage, la normalisation et la tokenisation, en utilisant aussi bien une approche classique qu'une approche basée sur AraBERT.

Enfin, ce mémoire présente l'architecture complète du modèle, son processus d'entraînement, les principaux hyperparamètres retenus ainsi que les méthodes utilisées pour évaluer ses performances. Les résultats expérimentaux montrent que l'approche proposée permet de générer des descriptions cohérentes et pertinentes en langue arabe. Ils mettent également en évidence l'intérêt du Transfer Learning ainsi que l'apport de la tokenisation basée sur

AraBERT pour améliorer la qualité des descriptions générées et mieux prendre en compte les particularités morphologiques de la langue arabe.

ملخص

تندرج هذه المذكرة في إطار الوصف التلقائي للصور باللغة العربية (Arabic Image Captioning)، وهو مجال بحثي يجمع بين الرؤية الحاسوبية (Computer Vision) والمعالجة الآلية للغات الطبيعية (NLP). يكمن الهدف الرئيس لهذا العمل في تصميم وتطوير نظام ذكي قادر على استيعاب وتفسير محتوى الصورة، ومن ثم توليد وصف نصي دقيق وملئم لها باللغة العربية.

يستعرض الجزء الأول من هذا العمل المفاهيم الأساسية لتقنية وصف الصور، وتطورها التاريخي، ومبدأ عملها، إلى جانب أبرز مجالات تطبيقها؛ لا سيما في دعم ذوي الإعاقة البصرية، والمجال الطبي، والتجارة الإلكترونية، ومنصات التواصل الاجتماعي، وبيئات التعلم الإلكتروني. كما يتناول هذا الجزء بالدراسة البنى الهيكلية (Architectures) الأكثر شيوعاً، وتحديدًا نموذج المشفر والمفكّر (Encoder-Decoder)؛ حيث تُستخدم الشبكات العصبية الالتفافية (CNN) لاستخلاص الميزات المرئية، بينما يتولى فك التشفير شبكات الذاكرة الطويلة قصيرة المدى (LSTM) أو نماذج المحولات (Transformers) لتوليد النص الوصفي تدريجيًا. وقد أُعطيت أهمية خاصة لآليات الانتباه (Attention Mechanisms)، التي تسهم في رفع جودة النصوص المولدة عبر توجيه النموذج للتركيز على المناطق الأكثر أهمية في الصورة.

بينما خُصص الجزء الثاني للجانب التطبيقي وتنفيذ النظام المقترح؛ حيث يسلط الضوء على بيئة التطوير المستخدمة، ومجموعة بيانات Flickr8k، بالإضافة إلى المراحل المختلفة لمعالجة الصور والنصوص مسبقًا. تم استخلاص الميزات المرئية بالاعتماد على تقنية التعلم بنقل المعرفة (Transfer Learning) باستغلال نموذجين مدربين مسبقًا هما VGG16 و GoogLeNet (InceptionV3)، مع إجراء دراسة مقارنة لأدائهما. وفي المقابل، خضعت الأوصاف العربية لمعالجة أولية شملت التنظيف، والتقييس (Normalization)، والتقطيع (Tokenization)، وذلك باستعمال منهجية كلاسيكية وأخرى متقدمة تعتمد على نموذج AraBERT.

أخيرًا، يعرض هذا البحث الهيكلية الكاملة للنموذج المطور، ومراحل تدريبه، وأبرز المعاملات الفائقة (Hyperparameters) التي تم ضبطها، فضلًا عن المقاييس المعتمدة لتقييم الأداء. وتُظهر النتائج التجريبية قدرة المقاربة المقترحة على توليد أوصاف متماسكة وذات دلالة واضحة باللغة العربية. كما تبرز النتائج بوضوح الأثر الإيجابي للاعتماد على التعلم بنقل المعرفة، والقيمة المضافة للتقطيع النصي المستند إلى AraBERT في تحسين جودة الأوصاف المولدة ومراعاة الخصائص الصرفية الفريدة للغة العربية.

Abstract

This thesis falls within the scope of Arabic image captioning, a research field that combines computer vision and natural language processing (NLP). The primary objective of this work is to design and develop an intelligent system capable of comprehending and interpreting image content, and subsequently generating an accurate and appropriate textual description in Arabic.

The first part of this work reviews the fundamental concepts of image captioning technology, its historical evolution, and its operating principles, alongside key application areas—particularly in assisting the visually impaired, as well as in medicine, e-commerce, social media platforms, and e-learning environments. This section also examines the most common architectures, specifically the encoder-decoder model; here, Convolutional Neural Networks (CNNs) are employed to extract visual features, while Long Short-Term Memory (LSTM) networks or Transformer models handle the decoding process to generate descriptive text incrementally. Special emphasis is placed on attention mechanisms, which enhance the quality of generated text by guiding the model to focus on the most significant regions of the image.

The second part is dedicated to the practical implementation of the proposed system, highlighting the development environment, the Flickr8k dataset, and the various stages of image and text preprocessing. Visual features were extracted using transfer learning techniques—leveraging two pre-trained models, VGG16 and GoogLeNet (InceptionV3)—and a comparative analysis of their performance was conducted. Meanwhile, the Arabic descriptions underwent preprocessing that included cleaning, normalization, and tokenization, utilizing both a classical approach and an advanced method based on the AraBERT model.

Finally, this research presents the complete architecture of the developed model, its training stages, the key hyperparameters tuned, and the metrics used for performance evaluation. Experimental results demonstrate the proposed approach's ability to generate coherent and semantically meaningful Arabic descriptions. Furthermore, the results clearly highlight the positive impact of leveraging transfer learning, as well as the added value of AraBERT-based text segmentation in enhancing the quality of generated descriptions and accounting for the unique morphological characteristics of the Arabic language.

Table des matières

Résumé	5
ملخص.....	6
Abstract	7
Abréviation	11
Liste des figures.....	12
Liste des Tableaux.....	13
INTRODUCTION GÉNÉRALE	15
Chapitre I : Fondements Théoriques de l'Intelligence Artificielle et du Traitement des Images ..	17
I.1.Introduction	18
I.2.Intelligence Artificielle	19
I.3. Machine Learning	21
I.3.1. Définition du Machine Learning	21
I.3.2. Types d'apprentissage	21
I.4. Deep Learning	24
I.4. 1.Historique du Deep Learning	24
I.4.2. Réseaux de neurones artificiels (ANN)	25
I.4.3. Les architectures du Deep Learning	26
I.5. Introduction au Traitement des Image	33
I.5.1. Définition d'une image numérique	34
I.5.2. Les principales opérations de traitement d'images.....	35
I.5.2.1. L'amélioration d'images	35
I.5.2.2. L'analyse d'images (la compréhension)	35
I.5.2.3. Le recalage d'images (la superposition parfaite)	36
I.5.3. Applications du traitement d'images	36
I.6. Introduction au Traitement Automatique du Langage Naturel (NLP)	37
I.7.conclusion	37
Chapitre II: Images Captioning	38
II.1 Introduction	39
II.2 Présentation de l'Image Captioning	40
II.2.1 Définition	40
II.2.2 Historique.....	41
II.2.3 Domaines d'application.....	43
II.3. Architectures et fonctionnement de l'Image Captioning.....	45
II.3.1. L'architecture classique : CNN + RNN / LSTM	46

II.3.2. L'architecture avec mécanisme d'Attention	48
II.3.3 L'architecture moderne : Les Transformers (Vision + Texte)	49
II.3.4 Dense Image Captioning	50
II.5. ImageCaptioning en langue Arabe	50
II.5.1 Particularités de la langue arabe	51
II.5.2 Les défis de l'Arabic Image Captioning	51
II.5.3 Jeux de données arabes	53
II.6 Méthodes d'évaluation	54
II.6.1 BLEU	54
II.6.2 METEOR	55
II.6.3 ROUGE	55
II.6.4 CIDEr	56
II.6.5 SPICE	56
II.7. Conclusion	57
III.1 Introduction	59
III.2 Environnement de développement	59
III.2.1 Environnement matériel	59
III.2.2 Environnement logiciel	59
III.2.3 Bibliothèques et outils utilisés	60
III.3 Présentation et préparation des données images	61
III.3.1 Présentation du dataset Flickr8k	61
III.3.2 Prétraitement des images	62
III.4 Extraction des caractéristiques par Transfer Learning	62
III.4.1 Principe du Transfer Learning	63
III.4.2 Avantages du Transfer Learning	64
III.4.3 Extraction des caractéristiques avec VGG16	64
III.4.4 Extraction des caractéristiques avec GoogLeNet	65
III.4.5 Comparaison entre VGG16 et GoogLeNet	67
III.5 Prétraitement du texte	68
III.5.1 Tokenisation classique (KerasTokenisation)	70
III.5.2 Tokensization avec AraBERT	70
III.6. Implémentation du modèle de génération des descriptions	71
III.7 Résultats et évaluation	73
III.7.1 Métriques d'évaluation	78
III.7.2 Exemples de descriptions générées	80
III.8 Conclusion	84

Conclusion Générale	86
Références	89

Abréviation

ANN :Réseau de neurones artificiels

AraBERT :Représentations d'encodeurs bidirectionnels issus de Transformers pour l'arabe

BLEU :Étudiant en évaluation bilingue (BLEU)

CNN : Réseau de neurones à convolution

Flickr8k Flickr 8k Dataset

IA: Intelligence artificielle

LSTM = Long Short-Term Memory

MSCOCO : MicrosoftcommonObjectsdans le context

NLP : Traitement du langage naturel

RNN : Réseau de neurones récurrent

VGG16: Visual Geometry Group 16

Liste des figures

Figure I. 1:Schéma d'apprentissage pour l'intelligence artificielle (1)	19
Figure I. 2:Domaines d'application d'IA (3).....	21
Figure I. 3:Historique de l'évolution dans le domaine de l'apprentissage profond.....	24
Figure I. 4:Réseaux de neurones artificiels (ANN)	26
Figure I. 5:Convolution naturel network (CNN)	27
Figure I. 6:Pooling Layers.....	28
Figure I. 7:Exemple de couche entièrement connectée	29
Figure I. 8:réseau neuronal récurrent(RNN)	32
Figure I. 9:traitement d'images.....	34
Figure I. 10:Image numérique	35
Figure II. 1: Illustration du cadre de génération de légendes d'images basé sur un réseau CNN-RNN	40
Figure II. 2:Structure de la méthodologie du projet « L'encodeur-décodeur ».....	43
Figure II. 3:Exemples de quelques images avec des légendes d'exemple	45
Figure II. 4:Présentation d'un modèle de sous-titrage basé sur l'architecture encodeur-décodeur.	46
Figure II. 5:Schéma de l'architecture de Transfer Learning.....	47
Figure II. 6:Réseau LSTM avec l'état initial et la séquence pour ce qui est de l'alimentation en entrée et en sortie	48
Figure II. 7:Le schéma d'architecture d'images captioning basé sur le mécanisme attention.	49
Figure II. 8:Exemples de sous-titrage dense.....	50
Figure III. 1 : Étapes du prétraitement des images.....	62
Figure III. 2:Le principe du processus d'apprentissage par transfert (Transfer Learning)	63
Figure III. 3:Processus d'extraction avec VGG16	65
Figure III. 4:Processus d'extraction avec GoogLeNet	66
Figure III. 5:Étapes du prétraitement des descriptions arabes.....	69
Figure III. 6:Courbe d'apprentissage avec GoogLeNET en Tokenisation classique.....	74
Figure III. 7:Courbe d'apprentissage avec googleNET en Tokenisation AraBERT	75
Figure III. 8:Courbe d'apprentissage avec VGG16 en Tokenisation classique.....	75
Figure III. 9:Courbe d'apprentissage avec VGG16 en Tokenisation AraBERT	76
Figure III. 10:Quelques résultats avec AraBERT	81
Figure III. 11:Résultats erroné par AraBERT	82
Figure III. 12: Résultats erroné par AraBERT	82
Figure III. 13:Résultats avec tokenisation classique	83
Figure III. 14:Comparisons de quelque résultat obtenu par Kerastokanisaton (en haut) et AraBERT (en bas)	84

Liste des Tableaux

Tableau I. 1:comparaison entre deep learning et machine learning.....	25
Tableau I. 2:Comparaison des modèles d'apprentissage profond populaire	31
Tableau III. 1: Environnement logiciel utilisé pour le développement du système.....	60
Tableau III. 2:Bibliothèques et outils utilisés dans le développement du système.....	60
Tableau III. 3:Répartition du jeu de données Flickr8k	61
Tableau III. 4:présente une comparaison des principales caractéristiques des modèles VGG16 et GoogLeNet utilisés pour l'extraction des caractéristiques visuelles.....	68
Tableau III. 5: Synthèse des hyperparamètres et configuration d'entraînement du modèle.	74
Tableau III. 6:Évaluation comparative des performances et des diagnostics de convergence	76
Tableau III. 7:Analyse comparative de l'évolution de la fonction de perte (Loss) et du phénomène de surapprentissage	77
Tableau III. 8:RÉSULTATS DE L'ÉVALUATION	78
Tableau III. 9:INTERPRETATION DU RÉSULTATS DE L'ÉVALUATION.....	79

INTRODUCTION GÉNÉRALE

INTRODUCTION GÉNÉRALE

Le domaine des technologies de l'information et de la communication connaît une évolution très rapide grâce aux progrès réalisés dans le domaine du Deep Learning. Ces avancées permettent aux systèmes informatiques d'accomplir des tâches complexes qui nécessitaient auparavant l'intelligence humaine, notamment l'analyse et l'interprétation de données multidimensionnelles. Parmi les défis importants de ce domaine figure la génération automatique de descriptions d'images, connue sous le nom d'Image Captioning. Cette technique vise à établir un lien entre la vision par ordinateur et le traitement automatique du langage naturel en transformant le contenu visuel d'une image en une phrase cohérente, correcte sur le plan grammatical et facilement compréhensible.

L'intérêt des systèmes de description d'images ne se limite pas au domaine de la recherche. Ils possèdent également de nombreuses applications pratiques. Dans le domaine de l'accessibilité, ils permettent aux personnes malvoyantes d'accéder plus facilement aux informations visuelles grâce à la conversion des descriptions textuelles en parole. Ces systèmes jouent également un rôle important dans l'indexation des bases de données multimédias, l'amélioration des moteurs de recherche visuelle ainsi que dans les applications de surveillance intelligente.

Même si les modèles de description d'images ont atteint de bonnes performances en langue anglaise, leur adaptation à la langue arabe, appelée Arabic Image Captioning, reste un défi important. En effet, la langue arabe se caractérise par une morphologie riche, une structure syntaxique flexible et des particularités orthographiques complexes. À cela s'ajoute un manque de ressources linguistiques et de bases de données annotées de grande taille par rapport aux langues occidentales. Malgré ces difficultés, le développement de modèles capables de comprendre à la fois le contenu visuel et les spécificités linguistiques de l'arabe constitue un domaine de recherche actif et prometteur.

La plupart des travaux récents utilisent une architecture de type « Encodeur-Décodeur ». Dans cette approche, un réseau neuronal convolutif (CNN) est utilisé comme encodeur afin d'extraire les caractéristiques visuelles importantes de l'image. Ces caractéristiques sont ensuite transmises à un réseau neuronal récurrent (RNN) ou à un modèle LSTM qui génère la

description textuelle mot par mot. Afin de réduire les besoins en données et les coûts de calcul, l'apprentissage par transfert (Transfer Learning) est largement utilisé. Cette technique consiste à exploiter les connaissances acquises par des modèles pré-entraînés sur de grandes bases de données d'images telles qu'ImageNet.

Dans ce contexte, l'objectif de ce projet de fin d'études est d'étudier et de développer un système capable de générer automatiquement des descriptions d'images en langue arabe. Ce travail vise également à analyser l'impact de différentes architectures d'extraction de caractéristiques visuelles sur la qualité des légendes produites. Pour atteindre cet objectif, ce mémoire est organisé en trois chapitres principaux :

Chapitre 1 : État de l'art sur l'apprentissage profond et les réseaux neuronaux.

Ce chapitre présente les notions théoriques nécessaires à la compréhension du travail réalisé. Il décrit les réseaux neuronaux CNN et RNN (LSTM), les fonctions d'activation, les méthodes d'optimisation ainsi que les mécanismes d'apprentissage.

Chapitre 2 : Problématique et spécificités de la description d'images

Ce chapitre introduit les concepts fondamentaux liés à la génération automatique de légendes d'images. Il présente et compare différentes architectures utilisées dans ce domaine, notamment les mécanismes d'attention et les Transformers. Il aborde également les particularités et les défis de la langue arabe, puis décrit les principales métriques utilisées pour l'évaluation des modèles.

Chapitre 3 : Implémentation, réalisation et évaluation expérimentale

Ce chapitre présente la partie pratique du projet. Il décrit l'environnement matériel et logiciel utilisé ainsi que les différentes étapes de prétraitement des images et des textes du jeu de données Flickr8k en arabe. Il explique également l'utilisation de l'apprentissage par transfert et propose une comparaison détaillée entre deux architectures CNN largement utilisées, à savoir VGG16 et GoogLeNet. Enfin, il présente et analyse les résultats obtenus à l'aide d'indicateurs d'évaluation quantitatifs et qualitatifs.

Chapitre I : Fondements Théoriques de l'Intelligence Artificielle et du Traitement des Images

I.1.Introduction

L'intelligence artificielle (IA) est devenue l'un des domaines les plus dynamiques de l'informatique. Elle vise à concevoir des systèmes capables de reproduire certaines capacités cognitives humaines, telles que l'apprentissage, le raisonnement, la résolution de problèmes et la prise de décision. Grâce aux progrès réalisés dans les technologies numériques et à la disponibilité de grandes quantités de données, l'IA connaît un essor considérable et trouve aujourd'hui des applications dans de nombreux secteurs, notamment la santé, l'industrie, les transports, la sécurité et les systèmes intelligents.

Parmi les disciplines qui composent l'intelligence artificielle, le Machine Learning occupe une place centrale. Cette approche permet aux systèmes informatiques d'apprendre à partir des données et d'améliorer progressivement leurs performances sans être explicitement programmés pour chaque tâche. L'évolution du Machine Learning a conduit à l'émergence du Deep Learning, qui s'appuie sur des réseaux de neurones profonds capables d'extraire automatiquement des représentations complexes des données et de résoudre des problèmes de plus en plus sophistiqués.

En parallèle, le Traitement Automatique du Langage Naturel (Natural Language Processing – NLP) constitue une branche essentielle de l'intelligence artificielle dédiée à la compréhension, à l'analyse et à la génération du langage humain. Il est à l'origine de nombreuses applications, telles que la traduction automatique, les assistants virtuels, les systèmes de dialogue et la génération automatique de texte. Toutefois, le traitement de la langue arabe demeure un défi en raison de sa richesse lexicale, morphologique et syntaxique.

L'association de ces différentes disciplines a favorisé le développement de systèmes intelligents capables de générer automatiquement une description textuelle à partir d'une image, un domaine connu sous le nom d'Image Captioning. Cette tâche repose sur la combinaison de la vision par ordinateur, chargée d'extraire les caractéristiques visuelles de l'image, et du traitement automatique du langage naturel, qui permet de produire une description cohérente et pertinente.

Ce chapitre a pour objectif de présenter les concepts fondamentaux nécessaires à la compréhension de cette problématique. Il introduit successivement les notions d'intelligence artificielle, de Machine Learning, de Deep Learning et de Traitement Automatique du Langage Naturel, qui constituent les bases théoriques indispensables à l'étude et à la conception du système d'Image Captioning développé dans ce mémoire.

I.2.Intelligence Artificielle

L'intelligence artificielle (IA), c'est l'art d'apprendre aux ordinateurs à imiter notre propre intelligence. Le but est de créer des programmes capables de réfléchir, de décider et d'agir un peu comme nous le ferions. Pour faire tourner une IA, il faut réunir trois ingrédients indispensables :

- Des ordinateurs puissants (le matériel).
- Une grande quantité de données structurées.
- Des lignes de code et des formules mathématiques (les algorithmes).

Les performances d'un système d'IA s'améliorent généralement avec l'augmentation de la quantité de données disponibles et de la puissance de calcul. La phase d'entraînement joue un rôle essentiel dans l'amélioration des performances du modèle (Figure I.1).

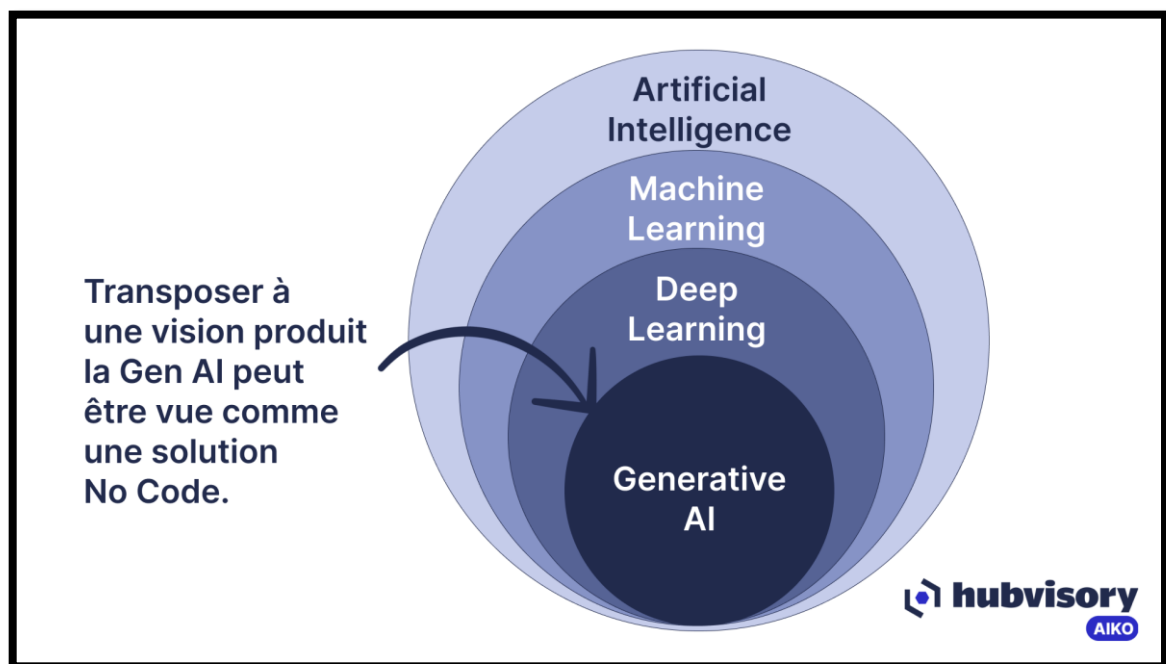


Figure I. 1Schéma d'apprentissage pour l'intelligence artificielle (1)

I.2.1. Domaines d'application de l'IA

Aujourd'hui, l'intelligence artificielle est largement intégrée dans de nombreux services et applications du quotidien. Que ce soit pour sécuriser nos transactions bancaires, faciliter nos achats en ligne ou améliorer les services numériques, elle intervient discrètement afin de rendre de nombreuses tâches plus simples, plus rapides et plus efficaces (2)

Parmi les domaines où l'IA joue un rôle essentiel, on peut citer :

I.2.1.1. La sécurité des services bancaires

Dans le secteur financier, l'intelligence artificielle contribue à renforcer la sécurité des opérations et à améliorer la prise de décision. Elle est notamment utilisée pour évaluer rapidement la solvabilité d'un client lors d'une demande de crédit en analysant un grand nombre d'informations. Elle permet également de détecter les fraudes en surveillant les transactions bancaires en temps réel. Lorsqu'une opération inhabituelle est repérée, le système peut immédiatement déclencher une alerte afin de limiter les risques (2)

I.2.1.2. Les services clients et le chatbots

L'intelligence artificielle est également très présente dans le domaine de la relation client grâce au chat bot, ces assistants virtuels capables de répondre automatiquement aux questions des utilisateurs. Ils prennent en charge les demandes les plus courantes à tout moment de la journée. Lorsque la demande est trop complexe, le chatbots transmet la conversation à un conseiller humain. Les interactions sont ensuite analysées afin d'améliorer les performances du système et de rendre les réponses futures plus pertinentes (2)

I.2.1.3. Le traitement du langage naturel et la vision par ordinateur

Deux technologies majeures expliquent une grande partie des capacités actuelles de l'intelligence artificielle.

Le **traitement du langage naturel (NLP)** permet aux machines de comprendre, d'interpréter et de produire le langage humain. Cette technologie est utilisée dans les assistants vocaux, les traducteurs automatiques ou encore les systèmes de conversation intelligents **Source spécifiée non valide..**

La **vision par ordinateur**, quant à elle, permet aux machines d'analyser et d'interpréter des images ou des vidéos. Elle trouve de nombreuses applications, comme la reconnaissance faciale, le diagnostic médical à partir d'images, les filtres des réseaux sociaux ou encore les systèmes de conduite autonome.

L'adoption de ces technologies connaît une croissance rapide dans de nombreux secteurs, notamment la santé, la finance et l'industrie automobile. Elles permettent aux entreprises d'améliorer leurs services, d'optimiser leurs coûts et de développer des solutions toujours plus innovantes (2)

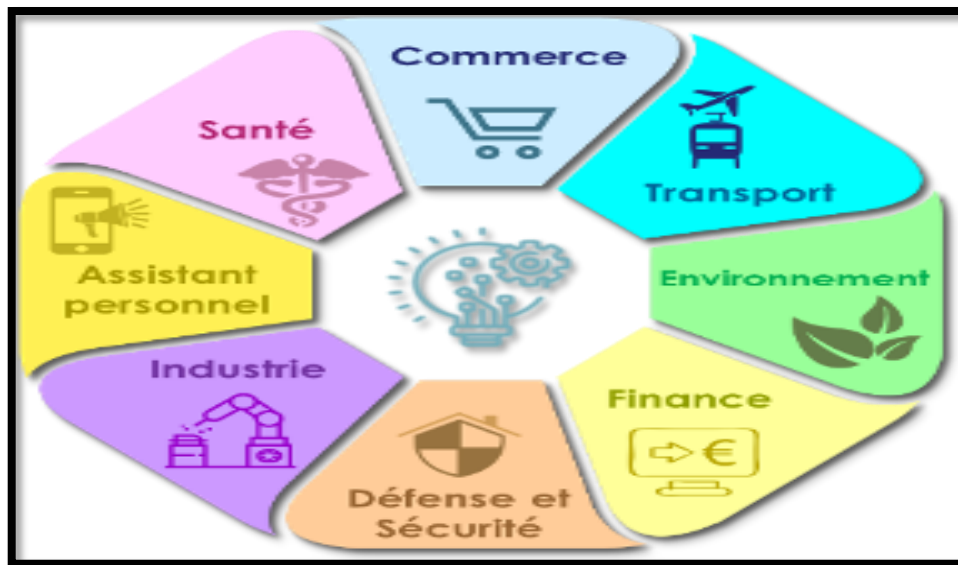


Figure I. 2: Domaines d'application d'IA (3)

L'intelligence artificielle (IA) est un domaine qui vise à créer des machines capables de reproduire certaines capacités humaines (Figure I.2). Le Machine Learning est l'une des principales méthodes utilisées pour atteindre cet objectif. Au lieu de suivre des règles programmées à l'avance, les machines apprennent à partir de grandes quantités de données afin d'améliorer progressivement leurs performances.

I.3. Machine Learning

I.3.1. Définition du Machine Learning

L'apprentissage automatique (Machine Learning) est la branche de l'intelligence artificielle (IA) spécialisée dans les algorithmes capables de « comprendre » les modèles des données d'entraînement et, par la suite, de tirer des déductions précises sur de nouvelles données. Cette capacité de reconnaissance de motifs permet aux modèles d'apprentissage automatique de prendre des décisions ou de faire des prédictions sans instructions explicites et codées à la main (4)

I.3.2. Types d'apprentissage

I.3.2. 1. Apprentissage supervisé

Le Machine Learning supervisé est une méthode d'apprentissage dans laquelle le modèle est entraîné à partir de données déjà étiquetées. Cela signifie que, pour chaque exemple utilisé pendant l'entraînement, la réponse correcte est connue à l'avance. Le modèle apprend ainsi à établir une relation entre les données d'entrée et le résultat attendu afin de pouvoir effectuer des prédictions sur de nouvelles données.

Par exemple, pour prédire la survenue d'une tornade, le modèle reçoit des informations telles que la date, la localisation, la température ou encore la vitesse du vent, ainsi que l'information indiquant si une tornade s'est effectivement produite. En analysant un grand nombre de ces exemples, il apprend à reconnaître les conditions favorables à l'apparition d'une tornade.

L'apprentissage supervisé est largement utilisé dans plusieurs domaines, notamment la reconnaissance d'images, la détection de fraudes bancaires, l'analyse prédictive et l'évaluation des risques (5)

Parmi les principaux algorithmes de l'apprentissage supervisé, on distingue :

- **Les algorithmes de régression**, qui permettent de prédire des valeurs numériques continues, comme le prix d'une maison, la température ou le salaire. La régression linéaire est l'un des exemples les plus connus.
- **Les algorithmes de classification**, qui servent à attribuer une catégorie à une donnée, par exemple déterminer si un e-mail est un spam ou non, ou si une image représente un chat ou un chien. Parmi les algorithmes les plus utilisés figurent la régression logistique, les k-plus proches voisins (KNN) et les machines à vecteurs de support (SVM).
- **Les classificateurs bayésiens naïfs**, qui reposent sur les probabilités pour classer les données. Ils sont particulièrement efficaces pour traiter de grands volumes de données, notamment dans les applications de filtrage des e-mails et de classification de textes.
- **Les réseaux de neurones**, inspirés du fonctionnement du cerveau humain, sont capables d'apprendre des relations complexes entre les données. Ils sont largement utilisés dans la reconnaissance d'images, la reconnaissance vocale, la traduction automatique et la génération d'images.
- **Les forêts d'arbres décisionnels (Random Forest)**, qui combinent plusieurs arbres de décision afin d'améliorer la précision des prédictions et de réduire les erreurs. Elles peuvent être utilisées aussi bien pour des problèmes de classification que de régression (5)

I.3.2. 2. Apprentissage non supervisé

Le Machine Learning non supervisé est une méthode d'apprentissage dans laquelle le modèle travaille sur des données qui ne sont pas étiquetées. Contrairement à l'apprentissage supervisé, il ne connaît pas les réponses à l'avance. Son objectif est d'analyser les données afin d'identifier automatiquement des structures, des relations ou des groupes présentant des caractéristiques similaires.

Cette approche est principalement utilisée pour l'analyse exploratoire des données, la segmentation de clients, la détection d'anomalies ainsi que les systèmes de recommandation.

L'une des techniques les plus utilisées est le « clustering », qui consiste à regrouper les données selon leur degré de similarité. Il permet, par exemple, de segmenter une clientèle en différents profils ou de détecter des comportements inhabituels.

Parmi les principaux algorithmes de l'apprentissage non supervisé, on trouve :

- **L'algorithme des k-moyennes (K-Means)**, qui répartit les données en plusieurs groupes (clusters) en fonction de leur proximité. Il est largement utilisé pour la segmentation de marché, le regroupement de documents et la compression d'images.
- **Le clustering hiérarchique**, qui construit progressivement une hiérarchie de groupes. Les données peuvent être fusionnées progressivement ou, au contraire, un grand groupe peut être divisé en plusieurs sous-groupes.
- **Le regroupement probabiliste**, qui classe les données selon la probabilité qu'elles appartiennent à un même groupe. Cette méthode est particulièrement utile lorsque les frontières entre les groupes ne sont pas clairement définies.
- **L'analyse en composantes principales (PCA)**, qui réduit le nombre de variables d'un jeu de données tout en conservant les informations les plus importantes. Cette technique facilite la visualisation et accélère le traitement des données.

Par ailleurs, les modèles d'apprentissage non supervisé sont largement utilisés dans les systèmes de recommandation. Par exemple, les plateformes de commerce en ligne peuvent suggérer des produits en se basant sur les comportements d'achat de clients ayant des préférences similaires (5)

I.3.2.3. Apprentissage par renforcement

L'apprentissage par renforcement est un type de programmation dynamique qui entraîne des algorithmes à l'aide d'un système de récompense et de punition. Pour déployer l'apprentissage par renforcement, un agent effectue des actions dans un environnement spécifique pour atteindre un objectif prédéterminé. L'agent est récompensé ou pénalisé pour ses actions en fonction d'un indicateur établi (généralement des points), ce qui l'encourage à poursuivre les bonnes pratiques et à éliminer les mauvaises. À force de répétition, l'agent apprend les meilleures stratégies.

Les algorithmes d'apprentissage par renforcement sont courants dans le développement de jeux vidéo et sont fréquemment utilisés pour apprendre aux robots à reproduire des tâches humaines (5)

I.4. Deep Learning

L'apprentissage profond (deep Learning) est une branche de l'apprentissage automatique qui repose sur des réseaux neuronaux multicouches, inspirés du fonctionnement du cerveau humain. Ces modèles sont actuellement utilisés dans la plupart des applications avancées de l'intelligence artificielle, de la vision par ordinateur à l'intelligence artificielle, en passant par les véhicules autonomes et les robots (**Tableau I.1**). (6)

I.4. 1.Historique du deep Learning

En 1952, Arthur Samuel, de la société IBM, a développé un programme capable d'apprendre à jouer aux dames. Ce programme pouvait construire de nouveaux modèles en observant les mouvements des pions et en les utilisant pour améliorer ses compétences de jeu. En 1959, l'apprentissage automatique a été présenté comme un domaine de recherche capable d'enseigner à une machine une compétence spécifique sans nécessiter de programmation prédéfinie. Au fur et à mesure de l'évolution de l'apprentissage automatique, divers modèles ont été proposés, notamment l'apprentissage profond. En raison de sa structure complexe et de la nécessité de calculs importants, le coût de calcul est très élevé, ce qui explique pourquoi il n'a pas suscité beaucoup d'intérêt au départ. Cependant, grâce à l'amélioration considérable de la puissance de calcul, les excellentes performances de l'apprentissage profond l'ont propulsé au rang des domaines de recherche les plus intéressants. Les principaux modèles d'apprentissage profond seront brièvement décrits, et les perspectives de développement de l'apprentissage profond seront examinées (FigureI.3).

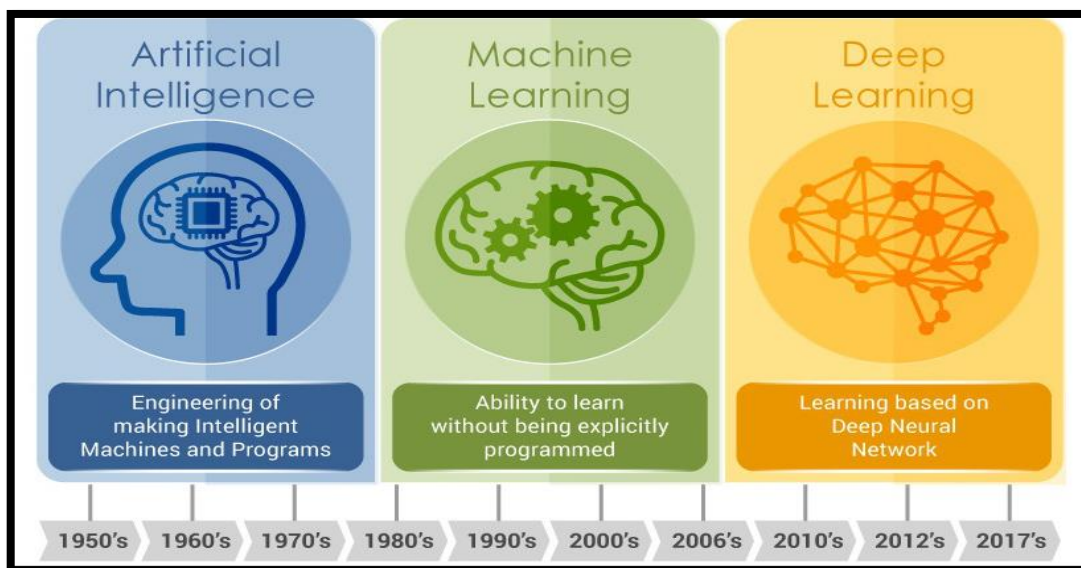


Figure I. 3: Historique de l'évolution dans le domaine de l'apprentissage profond

Critère	Machine Learning	Deep Learning
Interopérabilité	Plus interprétable	Moins interprétable (boîte noire)
Extraction des caractéristiques	Manuelle (feature engineering)	Automatique
Quantité des données	Fonctionne avec des petites ou moyennes données	Nécessite de grandes quantités de données
Performance	Bonne pour les problèmes simples	Très élevée pour les tâches complexes
Complexité des modèles	Modèles simples (SVM, arbres, régression...)	Réseaux neuronaux multicouches
Temps de calcul	Moins élevé	Élevé (besoin de GPU souvent)
Application	Prédiction, classification simple	Vision par ordinateur, NLP, image captioning

Tableau I. 1: comparaison entre deep learning et machine learning

I.4.2. Réseaux de neurones artificiels (ANN)

Un réseau neuronal artificiel (ANN) est un modèle mathématique inspiré de la structure et du fonctionnement du cerveau humain (Figure I.4). Il est constitué d'un réseau de nœuds interconnectés, appelés neurones artificiels, qui traitent les informations d'entrée et génèrent des sorties.

Dans un réseau neuronal artificiel, chaque neurone artificiel reçoit une ou plusieurs entrées et la traite à l'aide d'une fonction d'activation qui détermine la sortie du neurone. Les sorties de chaque neurone sont transmises aux autres neurones via des connexions pondérées, qui servent à ajuster la contribution de chaque neurone aux sorties finales.

Pendant l'entraînement du réseau neuronal artificiel, le modèle ajuste les poids des connexions entre les neurones afin de réduire l'écart entre les sorties souhaitées et les sorties réelles. Ce processus est mis en œuvre à l'aide d'algorithmes d'optimisation, tels que l'algorithme du gradient descendant.

Les réseaux neuronaux artificiels sont utilisés dans un large éventail d'applications d'apprentissage automatique, telles que la classification d'images, le traitement du langage naturel et la prévision de séries chronologiques. Ils sont également utilisés pour résoudre des problèmes complexes, tels que l'apprentissage profond et la création de réseaux neuronaux récurrents et convolutifs (7)

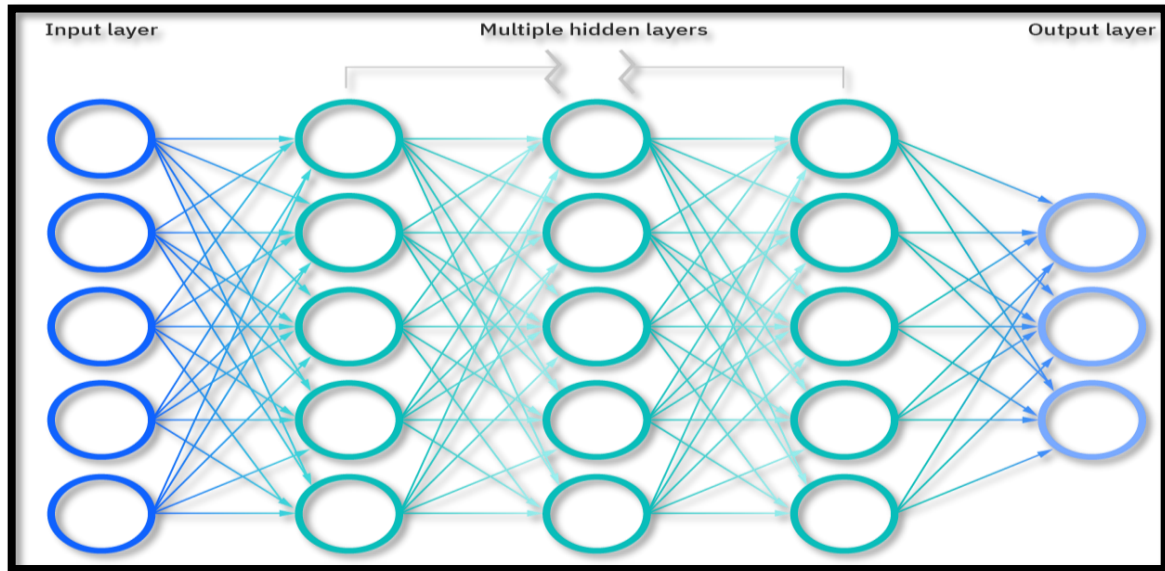


Figure I. 4: Réseaux de neurones artificiels (ANN)

I.4.3. Les architectures du Deep Learning

I.4.3.1. CNN

Les réseaux neuronaux convolutifs (CNN), qui se trouvent à la croisée de la biologie et de l'informatique, et qui sont inspirés par le cortex visuel humain, représentent une innovation significative introduite particulièrement par Yann LeCun dans les années 1990. Ces outils sont spécifiquement développés pour l'analyse des données visuelles, permettant d'extraire automatiquement des caractéristiques clés telles que les formes, textures et objets. Cela les rend très performants pour des opérations comme la classification, la détection et la segmentation d'images. Les CNN, par leur invariance spatiale, leur aptitude à extraire des caractéristiques de manière hiérarchique et grâce à la mutualisation des paramètres qui diminue la complexité computationnelle, présentent d'excellentes performances tout en demeurant efficaces.

Ils peuvent aussi être ajustés pour divers problèmes en utilisant le transfert d'apprentissage sur de vastes ensembles de données. Actuellement, ils sont couramment employés en vision par ordinateur, particulièrement dans les systèmes de reconnaissance d'images et de véhicules

autonomes. Ils occupent une place centrale dans les dispositifs de création automatique de descriptions d'images, où ils fonctionnent comme des encodeurs pour capter les données visuelles avant leur conversion en texte par des modèles séquentiels (8)

Architectures d'un réseau neuronal convolutifs (CNN)

Dans cette section, nous passerons en revue les principales rubriques du réseau CNN, puis nous aborderons les différentes évolutions et modifications de sa structure.

La structure du réseau CNN peut varier d'une version à l'autre, mais certaines caractéristiques restent toujours identifiables.

- **La couche de convolution** est l'élément central des réseaux neuronaux convolutifs, elle figure systématiquement au moins parmi les premières couches. Ce que l'on nomme des filtres compose les couches de convolution. Les filtres se composent de tableaux de valeurs connus sous le nom de cartes de caractéristiques. Chaque couche convolutive reçoit une image en input et génère une carte de caractéristiques (Figure I.5). Chaque carte de caractéristiques est générée en appliquant le filtre à l'image. Par souci d'illustration, si l'image mesure 5x5 et que le filtre a une dimension de 3x3, la carte des caractéristiques se présentera sous la forme d'une taille 3x3. Ainsi, la couche de convolution prend plusieurs images en entrée et effectue la convolution de chacune d'elles avec chaque filtre. Les filtres sont précisément alignés avec les caractéristiques que nous désirons identifier dans les images (8)

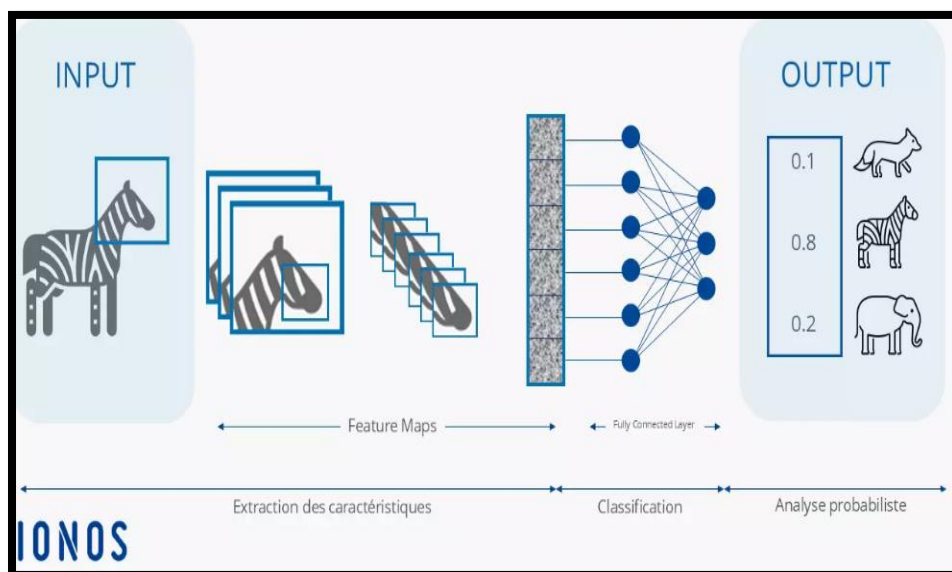


Figure I. 5: Convolution naturel network (CNN)

- **Couche de mise en commun (pooling)** On a souvent recours à ce genre de couche entre deux couches de convolution : elle prend en entrée divers featuremaps et applique une opération de pooling à chacun d'eux (Figure I.6). Une couche de pooling fonctionne en tant que couche de diminution. Elle segmente l'image en sections et conserve uniquement la valeur maximale de chaque section. Cette méthode offre la possibilité de diminuer la taille de l'image tout en préservant les traits les plus significatifs. Le nombre de featuremaps en sortie correspond au nombre de featuremaps en entrée, cependant leur taille est significativement réduite. (8)

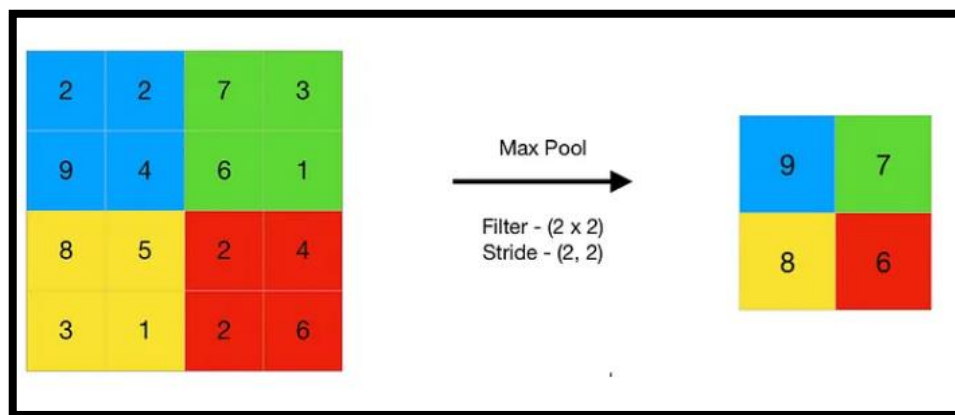


Figure I. 6: Pooling Layers

- **Couche d'entièrement connectée (fully-connected)** Généralement, les réseaux de neurones convolutifs sont constitués de multiples couches de convolution et de pooling, suivies d'une couche entièrement connectée qui fusionne les caractéristiques dérivées des couches antérieures afin de classifier l'image. Ce processus produit un vecteur de dimension N , où N représente le nombre de catégories dans notre tâche de classification d'images. Chaque composant du vecteur représente la chance que l'image d'entrée appartienne à une certaine classe (Figure I.7).

La couche fully-connected constitue toujours la dernière couche d'un réseau de neurones, convolutif ou non, elle n'est donc pas caractéristique d'un CNN. Cette sorte de couche prend un vecteur comme entrée et génère un nouveau vecteur en sortie. Ainsi, elle utilise une combinaison linéaire suivie éventuellement d'une fonction d'activation sur les valeurs qu'elle reçoit en entrée (8)

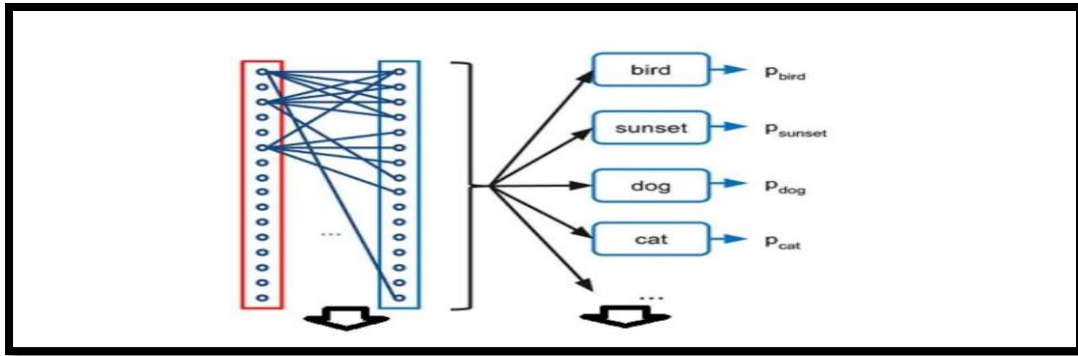


Figure I. 7: Exemple de couche entièrement connectée

Les architectures de réseaux neuronaux convolutifs célèbres (Tableau I.2)

- **LeNet-5 (CNN Classique)**

Développé par **Yann LeCun**, c'est l'un des premiers réseaux convolutifs appliqués avec succès. Il a été conçu pour la reconnaissance de caractères manuscrits (comme les codes postaux sur les chèques) (9)

- **AlexNet**

Ce modèle créé par Alex Krizhevsky, Geoffrey Hinton et Ilya Sutskever, ce modèle a révolutionné le domaine en remportant la compétition *ImageNet* avec une marge impressionnante. Il a popularisé l'utilisation des cartes graphiques (GPU) pour entraîner des réseaux beaucoup plus profonds (9)

- **VGGNet(VGG16/VGG19)**

VGG est un modèle de réseau de neurones développé par K. Simonyan et A. Zisserman à l'Université d'Oxford. Il a été présenté au concours ILSVRC (ImageNet Large Scale Visual Recognition Challenge) en 2014, où il a obtenu une précision impressionnante de 92,7 % dans la classification d'images, dépassant de nombreux modèles antérieurs.

Ce modèle a été créé pour utiliser l'architecture profonde tout en restant simple dans sa structure, ce qui lui a permis de devenir une référence dans le domaine de la vision par ordinateur.

Le but principal du modèle VGG était d'améliorer le rendement des réseaux de neurones pour la reconnaissance d'images en utilisant une structure plus profonde et uniforme. En utilisant des filtres de convolution de petite taille (3x3), VGG a permis d'apprendre des caractéristiques plus compliquées tout en gardant un nombre limité de paramètres. Ce modèle a aussi été créé pour être facilement ajustable à d'autres tâches de classification d'images, ce qui en fait une option précisée pour le transfert Learning (9)

- **GoogleNet(Inception)**

C'est un réseau neuronal convolutif profond créé par Google avec l'architecture inception. Au lieu d'un processus séquentiel, le travail parallèle est favorisé dans ce réseau. Il a 22 couches et 9 inception modules, ce qui conduit à une meilleure précision et performance. Il a un taux d'erreur de 6,7 %. Il est fondé sur l'idée de connexions éparses, donc chaque canal de sortie n'est pas lié à un canal d'entrée.

Il se distingue des autres modèles car il peut soit convolutionner, soit regrouper directement l'entrée. Le principal bénéfice est que les exigences de calcul diminuent grâce à la réduction de la dimensionnalité des données.

Google Net diverge de l'approche qui consiste simplement à empiler des couches convolutionnelles et à les superposer, tout en ajoutant beaucoup de filtres. Il peut extraire des volumes de fines informations détaillées en raison de la disponibilité du réseau dans la couche réseau. Une autre caractéristique notable de Google Net est l'ajout d'une couche de goulot d'étranglement, entraînant des réductions significatives des exigences informatiques (9)

- **Réseaux Résiduels (ResNet)**

ResNet, aussi connu sous le nom de réseaux résiduels, est une architecture de réseau composée de 152 couches. Sa conception a permis d'améliorer considérablement les techniques de classification, de détection et de localisation. L'architecture ResNet est constituée d'une série de modules résiduels, qui représentent les blocs de base. Le réseau est élaboré intégralement en superposant les réseaux résiduels les uns aux autres. Dans ResNet, l'approche de prétraitement des données est modifiée. Les données sont initialement segmentées en correctifs avant d'être intégrées au réseau. Il vient à bout de deux obstacles principaux rencontrés par les modèles précédents : l'extinction du gradient et la dégradation. On considère que ResNet est efficace car lors du processus de rétro propagation, le gradient reste fluide. Un ResNet présente deux options : il peut soit effectuer un ensemble de fonctions sur l'entrée, soit ignorer totalement cette étape. Il présente un taux d'erreur de 3,6 %. Il est principalement basé sur trois filtres, tels que VGG, et utilise le pooling moyen global, suivi de la classification. L'avantage majeur de ResNet réside dans sa capacité à construire et à former un réseau même en utilisant des milliers de couches résiduelles (9)

- **Mobile Net**

Conçu pour les appareils mobiles et les systèmes à faible consommation d'énergie ; Léger et rapide ; Idéal pour les applications en temps réel sur téléphones ou systèmes embarqués (9)

Modèle	Caractéristiques principales	Cas d'utilisation
LeNet-5	Modèle simple et précoce	Reconnaissance de l'écriture manuscrite
AlexNet	Modèle profond et de grande taille	Classification générale d'images
VGGNet	Grand nombre de couches, petits filtres	Apprentissage par transfert
Google Net	Blocs Inception	Reconnaissance efficace
ResNet	Connexions résiduelles (skip connections)	Modèles très profonds
Mobile Net	Léger et rapide	Applications mobiles et appareils embarqués

Tableau I. 2: Comparaison des modèles d'apprentissage profond populaire

- **RNN**

Le réseau neuronal récurrent (RNN) est un modèle d'apprentissage profond formé pour transformer une entrée de données séquentielle en une sortie de données séquentielle spécifique. Les données séquentielles se réfèrent à des informations, comme des mots, des phrases ou des données de séries temporelles, où les éléments successifs sont reliés selon des règles complexes de sémantique et de syntaxe. Un réseau neuronal récurrent (RNN) est un architecteur composé de multiples éléments liés entre eux, simulant la manière dont les humains réalisent des transformations de données séquentielles comme la traduction de textes d'une langue à une autre. L'intelligence artificielle basée sur des transformateurs et les grands modèles de langage tend à supplanter le réseau neuronal récurrent (RNN), offrant une efficacité supérieure dans le traitement séquentiel des données (10)

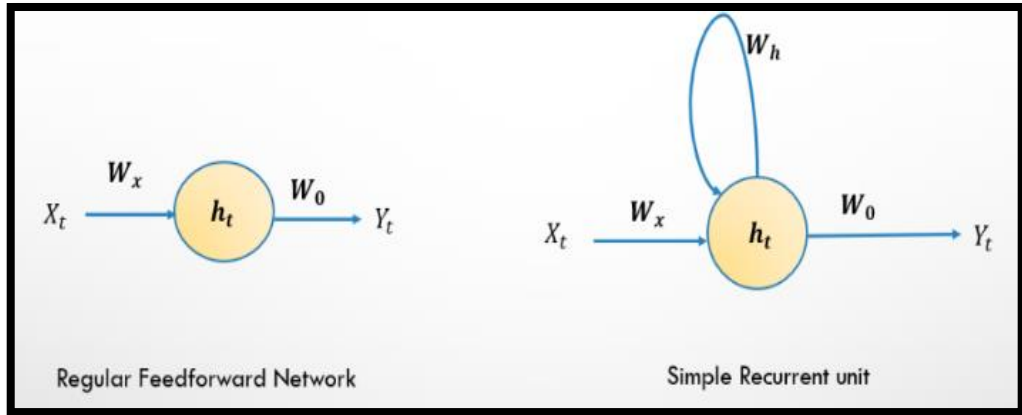


Figure I. 8:réseau neuronal récurrent (RNN)

D'après l'image(Figure I.8) ci-dessus, on peut formuler les équations suivantes :

$$H_t = \phi(X_t W_x + H_{t-1} W_h + b_h) \quad (10-1)$$

$$O_t = H_t W_o + b_t \quad (1-2)$$

La valeur Y_t représente la sortie de la couche récurrente au moment t . Ainsi, c'est un vecteur de longueur h .

L'état H_t , qui n'est pas employé en sortie, mais qui participera au calcul pour la prochaine date. C'est d'ailleurs pour cette raison que l'on parle d'état caché, car H_t n'est utilisé que pour le calcul et ne sort jamais des neurones de la couche.

W_x : Les poids qui sont introduits dans la couche récurrente.

W_h : Les poids au sein des neurones pour l'état caché.

W_o : Les poids sortants de la couche récurrente.

• Fonctionnement de RNN

Les réseaux neuronaux récurrents (RNN) sont constitués de neurones : des points d'analyse de données qui travaillent ensemble pour réaliser des tâches complexes. Les neurones sont organisés en couches d'entrée, de sortie et masquées. La couche d'entrée se charge de recevoir les données à examiner, alors que la couche de sortie fournit le produit fini. La couche cachée est chargée de gérer, d'analyser et de prédire les données.

Les réseaux neuronaux récurrents (RNN) fonctionnent en acheminant progressivement les données séquentielles qu'ils reçoivent à travers les différentes couches cachées. Toutefois, ils disposent également d'un système de fonctionnement en cycle continu ou répétitif : la couche cachée possède

l'aptitude de conserver et d'utiliser les données antérieures pour des prévisions futures, agissant en tant que mémoire à court terme. Elle utilise l'entrée actuelle et la mémoire stockée pour prévoir la séquence suivante.

Par exemple, considérons la séquence suivante : La pomme a une couleur rouge. Vous voulez que les réseaux neuronaux récurrent (RNN) prédisent la couleur rouge lorsqu'il reçoit la séquence d'entrée « pomme est ». Lorsque le niveau caché traite l'élément « pomme », il en garde une réplique dans sa mémoire. Dans ce qui suit, lorsqu'elle tombe sur le mot « est », elle se remémore « pomme » par l'intermédiaire de sa mémoire et saisit la totalité de la séquence : « pomme » renvoie au contexte. Par la suite, il peut prévoir le rouge avec une précision accrue. C'est ce qui confère aux réseaux neuronaux récurrent (RNN) leur pertinence pour des applications comme la reconnaissance vocale, la traduction automatique et d'autres tâches de modélisation du langage (10)

- **Apprentissage**

Les ingénieurs spécialisés en apprentissage automatique (ML) entraînent des réseaux de neurones profonds, comme les réseaux neuronaux récurrents (RNN), en nourrissant le modèle avec des données d'apprentissage et en perfectionnant ses performances. En apprentissage automatique, les poids neuronaux sont des indicateurs qui déterminent l'impact des données assimilées lors de l'entraînement sur la prévision du résultat. Dans un réseau neuronal récurrent (RNN), chaque couche partage le même poids.

Les ingénieurs en apprentissage automatique modifient les poids afin d'accroître la précision des prédictions. Ils emploient une méthode nommée rétro propagation temporelle (Back propagation Through Time, BPTT) pour évaluer l'erreur du modèle et modifier son poids en fonction de cela. La BPTT renvoie le résultat à la période de temps antérieure et refait le calcul du taux d'erreur. Ainsi, elle est capable de déterminer quel état latent de la séquence est responsable d'une erreur notable et d'ajuster le poids afin de minimiser l'écart d'erreur (10)

I.5. Introduction au Traitement des Image

Le traitement d'images est une branche du traitement du signal qui s'intéresse à l'analyse et à la manipulation des images et des vidéos. Il regroupe un ensemble de techniques visant à améliorer la qualité des images afin de les rendre plus claires et plus faciles à interpréter (Figure I.9). Parmi les opérations les plus courantes figurent l'amélioration du contraste, la réduction du bruit et la correction du flou. Le traitement d'images permet également d'extraire des informations utiles,

notamment grâce à des techniques telles que la segmentation des images et la détection des contours, qui facilitent l'analyse et la compréhension du contenu visuel.



Figure I. 9:traitement d'images

I.5.1. Définition d'une image numérique

Une image numérique (Figure I.10) est une image représentée sous forme de données numériques, ce qui permet de la créer, de la traiter, de la stocker et de l'afficher à l'aide d'un ordinateur. Elle est obtenue grâce à un processus appelé « numérisation », qui consiste à transformer une image réelle, dont les informations sont continuées, en une représentation discrète composée d'un ensemble de pixels. Chaque pixel est associé à une valeur numérique correspondant à son niveau de luminosité ou à sa couleur. Ces informations sont ensuite codées en langage binaire (0 et 1), ce qui permet aux logiciels informatiques de manipuler, d'analyser et de modifier facilement l'image selon les besoins (11)

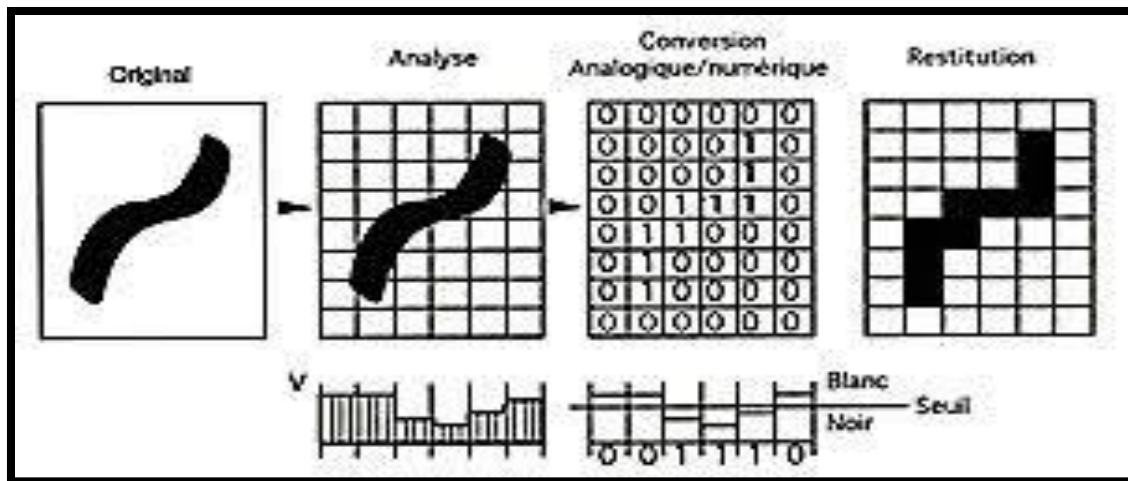


Figure I. 10:Image numérique

I.5.2. Les principales opérations de traitement d'images

Le traitement d'images, c'est un ensemble de techniques qui permettent de faire deux choses essentielles : rendre une image plus belle ou plus nette, et y dénicher des informations cachées pour pouvoir l'analyser.

Pour faire simple, tout cela s'articule autour de trois grandes étapes :

I.5.2.1. L'amélioration d'images

Le but ici est purement visuel : on veut rendre l'image plus claire et plus facile à exploiter. Cette opération consiste à améliorer la qualité visuelle de l'image afin de faciliter son exploitation. Pour cela, on va :

- Ajuster le contraste pour faire ressortir les détails.
- Réduire le bruit présent dans l'image (le grain ou le flou qui gâche l'image).
- Utiliser des filtres et équilibrer la luminosité (l'égalisation d'histogramme). Résultat : une image nette et prête à être utilisée.

I.5.2.2. L'analyse d'images (la compréhension)

L'analyse d'images vise à extraire et à interpréter les informations pertinentes contenues dans une image numérique. Son objectif est d'identifier les objets, les structures ou les caractéristiques significatives afin de faciliter leur exploitation par les systèmes informatiques. Pour ce faire,

diverses techniques sont utilisées, notamment la détection de contours, l'extraction d'objets et l'analyse des caractéristiques visuelles.

- La détection des contours (pour dessiner les formes).
- L'extraction d'objets (pour isoler un élément de l'arrière-plan).
- La mesure des caractéristiques des zones qui nous intéressent.

I.5.2.3. Le recalage d'images (la superposition parfaite)

Imaginez que vous ayez plusieurs photos de la même scène, mais prises sous des angles différents ou à des moments différents. Le recalage sert à les aligner parfaitement pour les superposer. Des algorithmes permettent de corriger automatiquement la rotation, la taille ou la position des images pour les fusionner proprement. Cette technique est particulièrement utile, par exemple, pour combiner un scanner et une IRM en médecine (12)

I.5.3. Applications du traitement d'images

Le traitement d'images regroupe l'ensemble des méthodes et des algorithmes permettant d'analyser, d'améliorer, de transformer ou d'extraire des informations à partir d'images numériques. Grâce à son évolution, cette discipline est aujourd'hui utilisée dans de nombreux domaines où l'analyse des images joue un rôle essentiel.

Dans le domaine de la santé, le traitement d'images est utilisé pour analyser les images médicales telles que les IRM, les scanners ou les radiographies, afin de faciliter le diagnostic précoce de certaines pathologies, la détection de tumeurs et la planification des interventions chirurgicales (13)

Dans le secteur de la sécurité, il est au cœur des systèmes de vidéosurveillance intelligents, de la reconnaissance faciale ainsi que de la lecture automatique des plaques d'immatriculation.

En industrie et en robotique, ces techniques permettent d'automatiser le contrôle de la qualité des produits, de détecter les défauts de fabrication et d'assurer le guidage des robots et des véhicules autonomes.

Le traitement d'images est également largement exploité dans le domaine de l'imagerie satellitaire et de la géomatique, où il contribue à l'analyse des données de télédétection, à la cartographie, au suivi de l'environnement, à l'agriculture de précision et aux prévisions météorologiques (14)

Par ailleurs, dans le secteur du multimédia, il est utilisé pour la retouche et l'amélioration des photographies, la création d'effets spéciaux dans les productions audiovisuelles ainsi que la

génération d'images de synthèse. Ces nombreuses applications montrent que le traitement d'images constitue aujourd'hui une technologie indispensable dans plusieurs domaines scientifiques, industriels et technologiques. (12)

I.6. Introduction au Traitement Automatique du Langage Naturel (NLP)

Le traitement du langage naturel (NLP) est une branche de l'intelligence artificielle qui offre la capacité aux ordinateurs de saisir, déchiffrer et produire le langage humain, qu'il soit sous forme écrite ou orale. On le retrouve dans de nombreuses applications telles que les assistants numériques (Siri, Alexa, Cortana), la recherche en ligne, la traduction automatique, l'analyse des émotions, le tri des spams et le condensé de textes.

Le traitement du langage naturel (NLP) se divise en deux volets essentiels : la compréhension du langage naturel (NLU), qui autorise les machines à interpréter le langage humain, et la génération de langage naturel (NLG), qui offre la possibilité de créer des textes en langue naturelle.

L'histoire du NLP remonte aux années 1950, mais la majorité des avancées récentes s'explique par l'évolution du machine learning et du deep learning. Ces techniques permettent d'acquérir des modèles sophistiqués à partir de vastes ensembles de données. (15)

I.7.conclusion

En conclusion, ce chapitre a présenté les notions fondamentales de l'intelligence artificielle, du Machine Learning, du Deep Learning, du traitement automatique du langage naturel et du traitement d'images. Ces concepts constituent la base scientifique de l'Image Captioning, qui combine la vision par ordinateur et le traitement du langage pour générer des descriptions d'images. Ils servent de fondement au chapitre suivant, consacré aux architectures, au fonctionnement et à l'évaluation des systèmes d'Image Captioning, en particulier pour la langue arabe.

Chapitre II : Images Captioning

II.1 Introduction

La génération automatique de descriptions d'images (Image Captioning) est aujourd'hui l'un des domaines les plus actifs de l'intelligence artificielle. Elle combine les avancées de la vision par ordinateur (Computer Vision) et du traitement automatique du langage naturel (Natural Language Processing – NLP) afin de permettre à une machine d'analyser une image et d'en produire une description en langage naturel. Cette tâche nécessite non seulement l'identification des objets présents dans une scène, mais également la compréhension de leurs interactions, de leur contexte et de leurs relations spatiales, dans le but de générer une phrase cohérente et fidèle au contenu de l'image.

L'essor de l'apprentissage profond (Deep Learning) a profondément transformé ce domaine en remplaçant les méthodes traditionnelles par des modèles capables d'être entraînés de bout en bout (end-to-end). La plupart des systèmes actuels reposent sur une architecture de type Encodeur–Décodeur. L'encodeur, généralement basé sur un réseau de neurones convolutif (CNN), extrait les informations visuelles de l'image, tandis que le décodeur, construit à partir de réseaux LSTM ou plus récemment de Transformers, génère progressivement la description textuelle. L'ajout des mécanismes d'attention a permis d'améliorer davantage les performances de ces modèles en leur donnant la capacité de se concentrer sur les régions les plus pertinentes de l'image lors de la génération de chaque mot.

Grâce à ces avancées, l'Image Captioning trouve aujourd'hui de nombreuses applications, notamment dans l'assistance aux personnes malvoyantes, l'indexation automatique d'images, la recherche multimédia, les réseaux sociaux ou encore les systèmes intelligents d'aide à la décision.

Dans ce chapitre, nous présentons les notions essentielles relatives à la génération automatique de descriptions d'images. Nous introduisons d'abord les concepts fondamentaux de l'Image Captioning, puis les principales architectures utilisées dans ce domaine, avant d'aborder les spécificités et les défis liés à la génération de descriptions en langue arabe (Arabic Image Captioning).

II.2 Présentation de l'Image Captioning

II.2.1 Définition

L'Image Captioning est une tâche qui consiste à générer automatiquement une description textuelle à partir du contenu d'une image. Contrairement aux problèmes classiques de vision par ordinateur, qui se limitent à reconnaître ou localiser des objets, cette approche cherche à produire une phrase complète décrivant la scène observée de manière claire et naturelle(Figure II.1).

Pour y parvenir, le système doit identifier les différents objets présents dans l'image, reconnaître les actions réalisées, comprendre les relations entre les éléments de la scène et prendre en compte leur contexte. Les informations visuelles extraites sont ensuite converties en une description rédigée en langage naturel, capable de refléter le plus fidèlement possible le contenu de l'image.

Le sous-titrage automatique d'images est ainsi considéré comme une tâche data-to-text, dans laquelle l'entrée correspond aux données visuelles de l'image et la sortie à une phrase descriptive. Cette problématique est plus complexe que la simple classification ou la détection d'objets, car elle nécessite une collaboration étroite entre les techniques de vision par ordinateur et les modèles de traitement automatique du langage afin de produire des descriptions proches de celles rédigées par un être humain. (9)

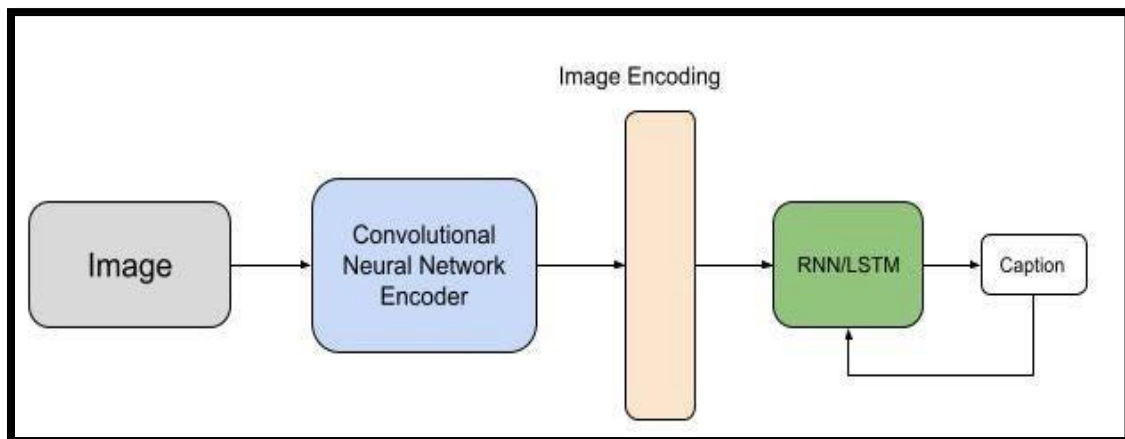


Figure II. 1: Illustration du cadre de génération de légendes d'images basé sur un réseau CNN-RNN

II.2.2 Historique

II.2.2.1 Approches traditionnelles et frameworks classiques

Les premières avancées dans le domaine du sous-titrage automatique d'images (image captioning) reposaient fortement sur l'extraction de caractéristiques artisanales (hand crafted features) et sur des méthodes de synthèse déterministes. Ces configurations pionnières s'appuyaient généralement sur un processus découplé où les descripteurs visuels étaient initialement extraits, puis classés à l'aide de classiques tels que les machines à vecteurs de support (SVM). Cependant, l'ingénierie manuelle des caractéristiques s'est avérée inefficace pour gérer des ensembles de données de grande dimension et hétérogènes, notamment en raison de la complexité sémantique et de la nature multi-interprétable des médias visuels réels. Historiquement, ces techniques classiques se subdivisent en trois méthodologies principales :

- **Approches basées sur des modèles rigides (Template-Based) :** Ces systèmes utilisaient des structures grammaticales prédéfinies. Des classificateurs spécialisés étaient déployés pour détecter des objets, des attributs et des actions discrets, qui étaient ensuite insérés dans des gabarits textuels fixes (par exemple, "Un [sujet] [action] un [objet]"). Bien que syntaxiquement correctes, ces méthodes souffraient d'une grande rigidité linguistique.
- **Méthodes basées sur la recherche (Retrieval-Based) :** Au lieu de générer de nouvelles séquences textuelles, ces approches formulaient le problème comme une tâche de recherche et d'appariement. En calculant les similitudes visuelles à l'aide de descripteurs manuels tels que SIFT (Scale-Invariant Feature Transform) ou HOG (Histogram of Oriented Gradients), le système extrayait la description la plus pertinente à partir d'une base de données existante de paires images-textes. Par conséquent, ces modèles manquaient de capacité de généralisation et étaient incapables de synthétiser des descriptions inédites pour des scénarios inconnus.
- **Systèmes en cascade (Pipeline Systems) :** Ces méthodes adoptaient une configuration modulaire liant séquentiellement la détection d'objets, la compréhension de la scène et la génération du langage naturel. La vulnérabilité majeure de cette architecture résidait dans la propagation des erreurs : toute imprécision introduite lors des phases initiales de détection visuelle dégradait inévitablement la fidélité et la précision de la production textuelle finale.

II.2.2.2 L'ère moderne : apprentissage profond et paradigmes multimodaux

Le changement de paradigme vers l'apprentissage profond (deep learning) a fondamentalement transformé le domaine en introduisant des architectures entraînables de bout en bout (end-to-end), capables d'optimiser conjointement les représentations visuelles et les structures linguistiques. Contrairement à leurs prédécesseurs, les réseaux de neurones profonds apprennent automatiquement des représentations de caractéristiques hiérarchiques directement à partir de volumes massifs et diversifiés de données.

La pierre angulaire de cette ère moderne est le framework unifié Encodeur-Décodeur(Figure II.2) :

- **L'Encodeur** : En exploitant des réseaux de neurones convolutifs profonds (CNN) tels que ResNet ou Inception, l'encodeur extrait des cartes de caractéristiques spatiales de haut niveau à partir de l'image d'entrée, projetant ainsi les données visuelles dans un espace latent dense.
- **Le Décodeur** : En s'appuyant sur des réseaux de neurones récurrents (RNN), plus précisément les réseaux LSTM (Long Short-Term Memory) ou les unités GRU (Gated Recurrent Units), et plus récemment sur les Transformers basés sur l'auto-attention, le décodeur traite séquentiellement les vecteurs visuels latents pour synthétiser des phrases fluides et proches du langage humain.

Une étape cruciale de cette évolution a été l'intégration des mécanismes d'attention (introduits de manière notable par des travaux pionniers comme "Show, Attend and Tell" en 2015). En pondérant dynamiquement les différentes régions spatiales des caractéristiques visuelles à chaque étape de la génération des mots, les mécanismes d'attention ont résolu le problème de col de bouteille (bottleneck) lié à la compression d'une scène entière en un seul vecteur statique, améliorant ainsi considérablement l'alignement contextuel et la précision des modèles contemporains. (9)

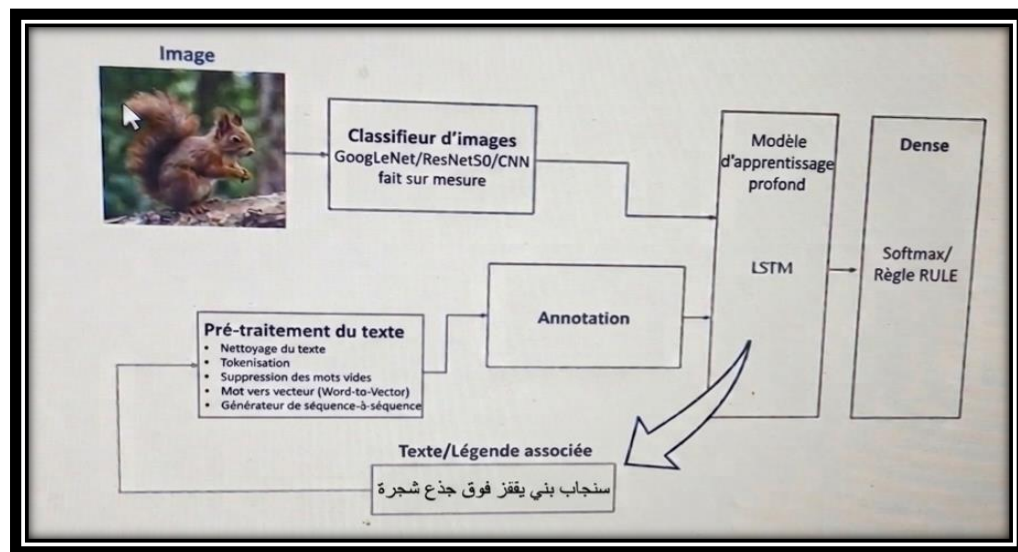


Figure II. 2: Structure de la méthodologie du projet « L'encodeur-décodeur »

II.2.3 Domaines d'application

Le sous-titrage automatique d'images (image captioning) constitue une technologie transversale aux retombées significatives dans divers secteurs industriels et académiques. Son intégration permet d'automatiser l'interprétation visuelle et d'optimiser les flux de travail complexes.

II.2.3.1 Le secteur de la santé et de l'imagerie médicale

Dans le domaine biomédical, cette technologie est exploitée pour rationaliser l'analyse de l'imagerie diagnostique. Les algorithmes d'intelligence artificielle permettent de générer des descriptions textuelles rigoureuses à partir de clichés radiologiques, constituant ainsi un outil d'aide à la décision pour les praticiens.

Application clinique : Les systèmes automatisés analysent les examens d'imagerie par résonance magnétique (IRM) ou les radiographies standards pour produire des comptes rendus préliminaires mettant en évidence les anomalies structurelles ou les signes pathologiques majeurs. Ce processus permet d'accélérer le tri clinique et de sécuriser le diagnostic radiologique. (16)

II.2.3.2 Le commerce électronique (E-commerce)

Les plateformes de commerce électronique s'appuient sur l'indexation visuelle pour enrichir sémantiquement les catalogues de produits. La génération automatisée de métadonnées descriptives

détaillant les caractéristiques techniques, esthétiques et fonctionnelles des articles permet d'optimiser l'expérience utilisateur et les taux de conversion.

Application opérationnelle : Dans le secteur de la distribution de mode en ligne, les modèles légendent automatiquement les attributs des vêtements (coupe, texture textile, style), structurant ainsi les données pour faciliter les requêtes de recherche des utilisateurs et affiner les systèmes de recommandation. (16)

II.2.3.3 Les plateformes de médias sociaux

Au sein des réseaux socio-numériques, le sous-titrage automatique répond à un double enjeu : l'amélioration de l'interaction homme-machine (IHM) et la réduction de la charge cognitive des utilisateurs lors de la publication de contenus visuels.

Application technique : Des plateformes à forte composante visuelle, telles qu'Instagram, intègrent des modèles pré-entraînés pour suggérer des descriptions contextuelles basées sur l'analyse sémantique de l'image, fluidifiant ainsi le partage de contenus à grande échelle. (16)

II.2.3.4 Le domaine de l'éducation et du e-learning

Dans les environnements pédagogiques, la description automatique d'images sert de levier pour soutenir l'apprentissage multimodal. En associant des descriptions textuelles explicatives à des schémas techniques ou des diagrammes complexes, elle facilite l'assimilation conceptuelle.

Application pédagogique : Les architectures d'apprentissage en ligne (*e-learning*) déploient ces outils pour générer des annotations textuelles dynamiques sous les illustrations didactiques, permettant aux apprenants de décoder plus efficacement les concepts abstraits ou les processus d'ingénierie. (16)

II.2.3.5 L'industrie du voyage et du tourisme

Le secteur touristique valorise cette technologie pour l'indexation et la contextualisation de larges bases de données iconographiques. L'attribution automatique de légendes géographiques, historiques ou culturelles enrichit l'expérience immersive des utilisateurs.

Application commerciale : Les portails d'agences de voyages exploitent ces descriptions pour indexer sémantiquement les photographies de destinations, optimisant ainsi le référencement naturel (SEO) et renforçant l'attractivité visuelle des interfaces numériques de réservation. (16)

II.3. Architectures et fonctionnement de l'Image Captioning

La génération automatique de légendes d'images consiste à produire une description textuelle d'une image de manière automatique. Cette tâche repose sur la compréhension de la sémantique visuelle, ce qui implique l'identification des objets présents, de leurs attributs, de leurs positions ainsi que des interactions entre eux dans la scène. Elle nécessite également l'inférence des informations sémantiques implicites afin de générer des descriptions cohérentes et pertinentes.

La Figure II.3 illustre quelques exemples d'images accompagnées de leurs légendes, telles que : « Un couple d'enfants se promenant avec des parapluies colorés », « Un oiseau vert posé sur des bananes à l'arrière-plan » et « Un homme en tenue de football jouant sur un terrain », correspondant respectivement aux Figure II.3(a), (b) et (c).

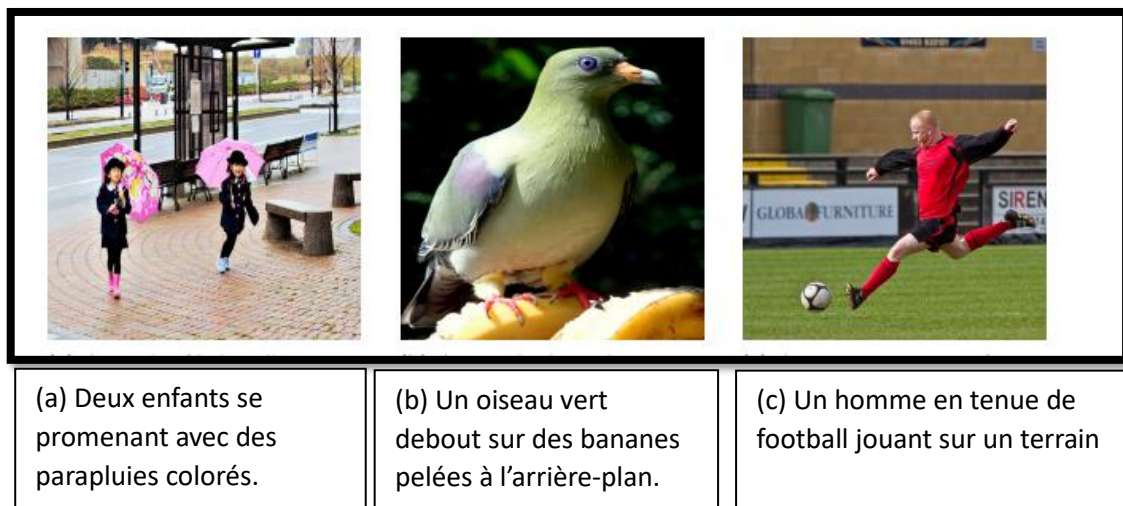


Figure II. 3:Exemples de quelques images avec des légendes d'exemple

L'Image Captioning repose principalement sur une structure Encoder-Decoder (Encodeur-Décodeur) qui fait le pont entre la vision par ordinateur et le traitement du langage naturel (NLP). Le rôle de l'encodeur est de compresser l'image en un vecteur de caractéristiques, tandis que le décodeur traduit ce vecteur en une phrase textuelle cohérente. La Figure II.4 présente une vue d'ensemble du modèle de sous-titrage.

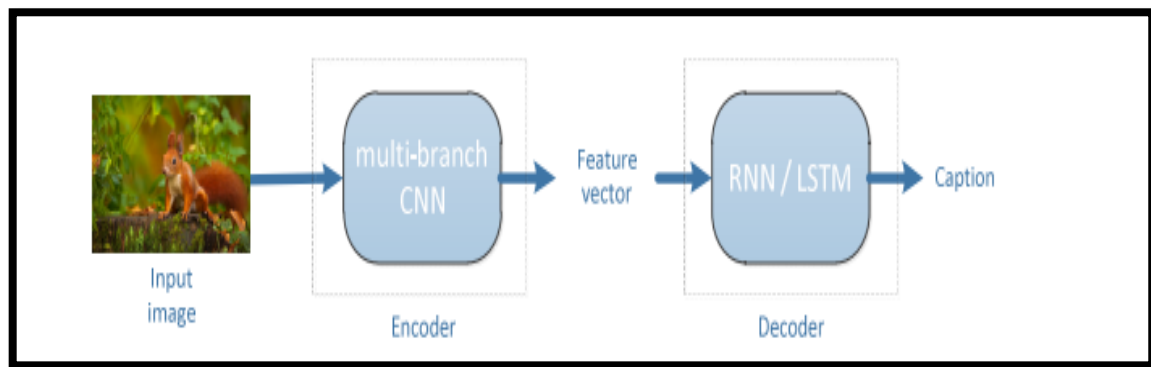


Figure II. 4:Présentation d'un modèle de sous-titrage basé sur l'architecture encodeur–décodeur.

II.3.1. L'architecture classique : CNN + RNN / LSTM

L'Encodeur (CNN) : L'extraction des caractéristiques visuelles

L'encodeur est un réseau de neurones convolutif (CNN). Son rôle est d'analyser l'image pour comprendre les objets, les formes et les textures ce l'étape d'extraction des caractéristiques visuelles.

L'extraction de caractéristiques consiste à transformer des données brutes (images) en représentations numériques significatives. Dont, l'extraction de caractéristiques est la première étape d'un système d'image Captioning, agissant comme l'encodeur visuel qui transforme les pixels bruts en vecteurs numériques exploitables par un modèle de langage (décodeur).

Au lieu d'entraîner le modèle CNN à partir de zéro on utilise le Transfer Learning ; ou on choisit un CNN déjà expert en reconnaissance d'images (comme ResNet, VGG ou Inception). Ce modèle a déjà appris à observer des millions d'images sur la base de données ImageNet. Ensuite, on modifie ce réseau en pratiquant la suppression de la tête de classification. Un CNN classique se termine par une couche (Softmax) qui donne le nom de l'objet (par exemple : "chat" ou "chien"). Pour générer du texte, cette dernière couche est inutile, donc on la supprime. Enfin, on extrait le vecteur d'encapsulation (embedding). On récupère les données juste avant la couche supprimée, au niveau du pooling global. On obtient alors un vecteur de caractéristiques fixes, par exemple d'une taille de 2048 pour le GoogLeNET (Inception3) et 4096 pour le modèle VGG 16, comme il est illustré sur la Figure II.5. Ce vecteur est un résumé mathématique abstrait de tout le contenu visuel de l'image.

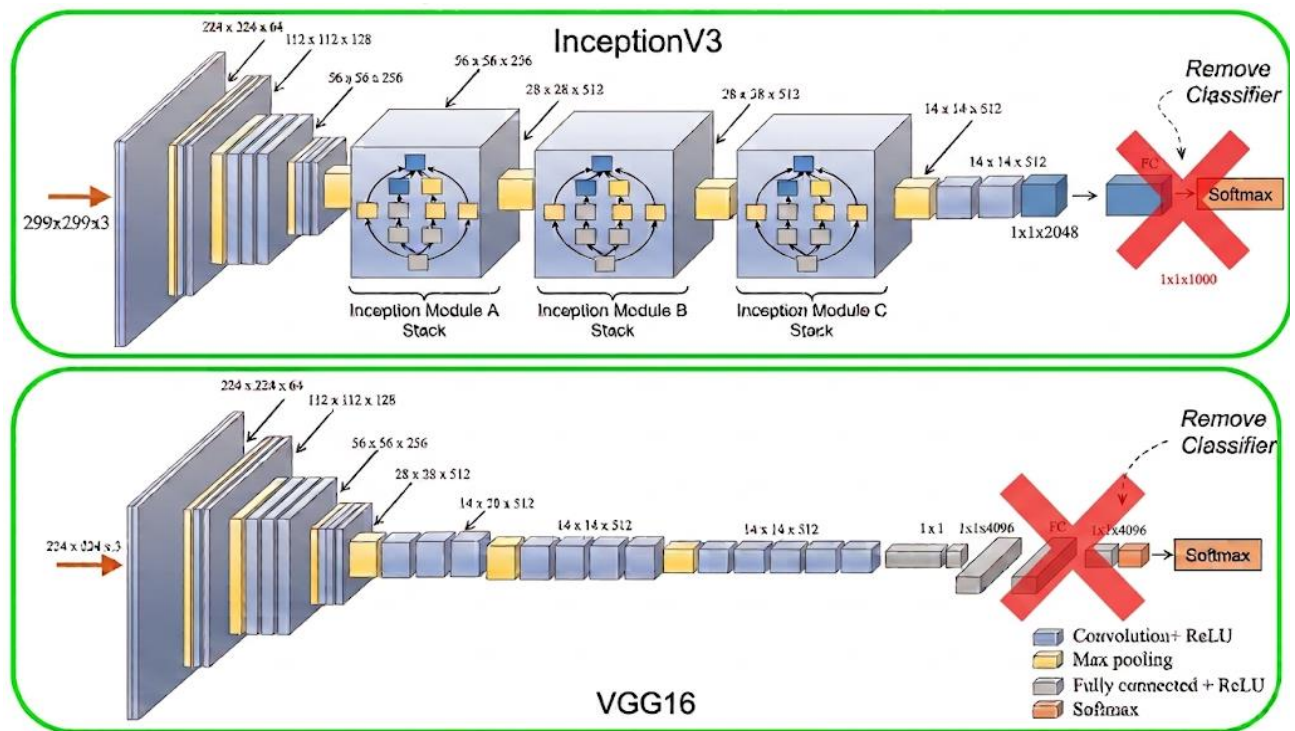


Figure II. 5: Schéma de l'architecture de Transfer Learning

Le Décodeur (RNN / LSTM) : La génération de la phrase

Le décodeur de texte est le composant chargé de générer une description en langage naturel (caption) à partir des caractéristiques visuelles extraites par l'encodeur (Figure II. 6). Son rôle consiste à transformer les informations contenues dans la représentation de l'image en une séquence de mots cohérente et descriptive. Dans les architectures classiques de génération de descriptions d'images, le décodeur repose généralement sur une couche d'incorporation des mots (embedding layer) suivie d'un ensemble de couches LSTM empilées. Cette structure permet au modèle de capturer les dépendances temporelles entre les mots et de produire progressivement une description adaptée au contenu de l'image.

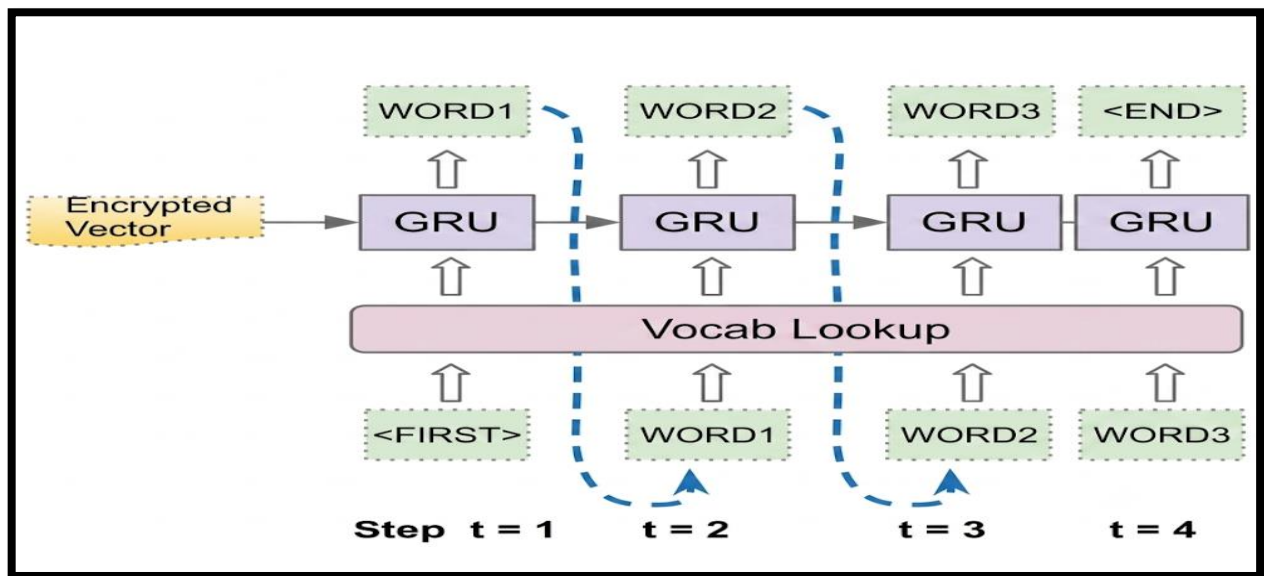


Figure II. 6: Réseau LSTM avec l'état initial et la séquence pour ce qui est de l'alimentation en entrée et en sortie

Le processus de génération débute à partir du vecteur représentant l'image, associé à une séquence d'entrée initiale contenant uniquement le jeton « Start ». À partir de cette représentation visuelle, le décodeur produit progressivement une séquence de mots correspondant à la description de l'image.

La génération s'effectue de manière itérative : à chaque étape, un nouveau mot est prédit puis réinjecté dans le modèle comme entrée pour l'étape suivante. Ainsi, chaque prédiction repose à la fois sur les informations visuelles de l'image et sur les mots déjà générés.

Ce mécanisme se poursuit jusqu'à la production du jeton « End », qui indique la fin de la description et clôture la séquence générée (9)

II.3.2. L'architecture avec mécanisme d'Attention

Dans les modèles Encoder-Decoder classiques (Figure II. 7), l'encodeur CNN compresse l'intégralité de l'image en un unique vecteur de taille fixe. Cette compression globale entraîne une perte massive d'informations géométriques et spatiales. Le décodeur LSTM doit générer toute la phrase à partir de ce seul résumé initial. Face à des descriptions longues ou à des scènes visuelles complexes,

Pour résoudre cette perte de données, le mécanisme d'attention spatiale a été mis en place pour résoudre cette perte de données. Où le CNN extrait une carte de caractéristiques sous forme de grille, divisant l'image en régions locales distinctes. À chaque étape temporelle, le décodeur LSTM

interagit dynamiquement avec cette grille : le module d'attention calcule un score d'importance, normalisé par une fonction Softmax, en comparant l'état caché du décodeur avec chaque zone visuelle. Finalement, une moyenne pondérée de ces régions crée un vecteur de contexte dynamique, permettant au modèle de focaliser mathématiquement son regard sur un objet précis au moment exact de sa transcription

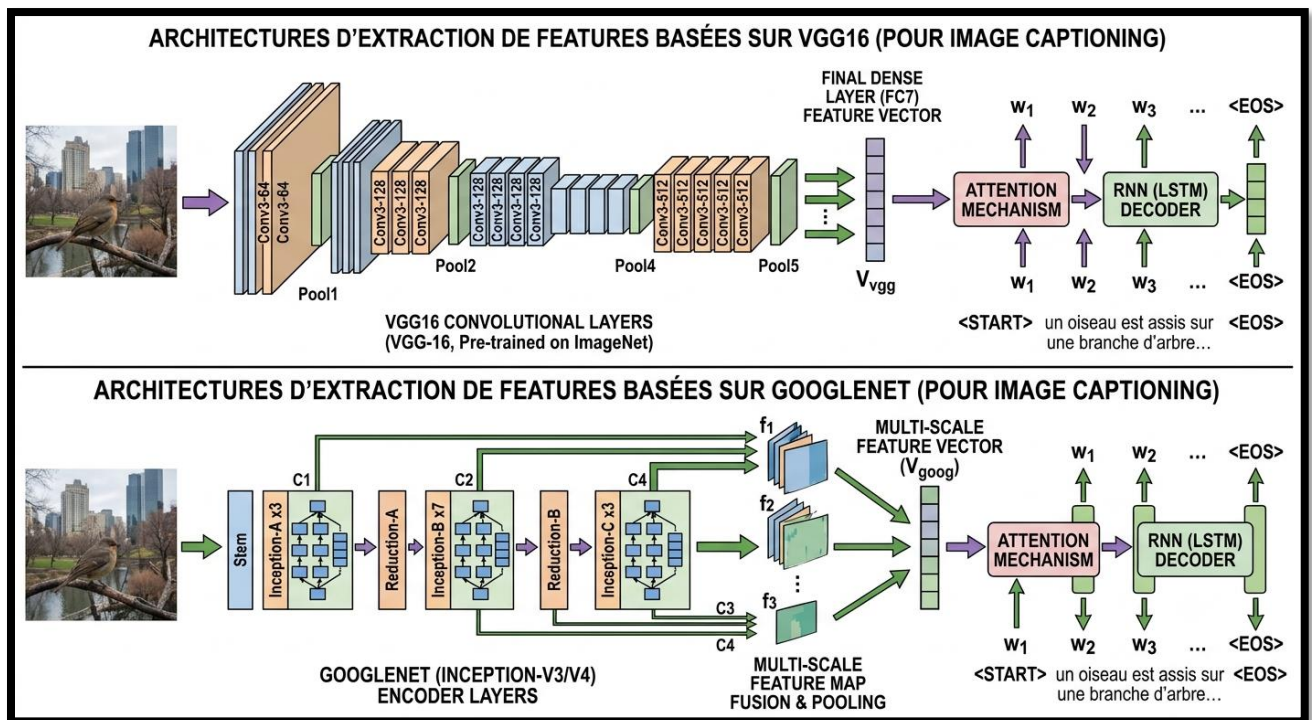


Figure II. 7: Le schéma d'architecture d'images captioning basé sur le mécanisme attention.

II.3.3 L'architecture moderne : Les Transformers (Vision + Texte)

Aujourd'hui, les architectures basées sur les Transformers dominent le domaine en éliminant la récurrence séquentielle des RNN au profit du calcul parallèle. (17).

Encodeur (Vision Transformer - ViT) : L'image est découpée en petits patches, projetés comme des tokens visuels. Un encodeur Transformer traite ces patches pour capturer les relations contextuelles globales au sein de l'image.

Décodeur (Text Transformer) : Un décodeur Transformer classique génère le texte en utilisant l'attention croisée (*cross-attention*) pour aligner les tokens textuels avec les caractéristiques visuelles issues du ViT.

Modèles Fondations (VLM) : Des modèles massifs d'aujourd'hui, tels que CLIP d'OpenAI, BLIP ou les MLLM (MiniGPT-4, LLaVA), exploitent le pré-entraînement contrastif ou l'alignement direct avec des grands modèles de langage (LLM) pour générer des descriptions extrêmement riches et contextuelles.

II.3.4 Dense Image Captioning

Le dense captioning constitue une autre variante des approches basées sur la détection d'objets. L'idée principale repose sur le fait qu'une image contient souvent plusieurs objets ainsi que différentes actions réparties dans diverses régions (Figure II.8).

Ainsi, au lieu de produire une seule description globale pour toute l'image, cette méthode génère plusieurs légendes, chacune associée à une zone spécifique de l'image. Grâce à cette approche, le modèle est capable de capturer et de décrire de manière plus fine et détaillée l'ensemble des éléments présents dans la scène (9)

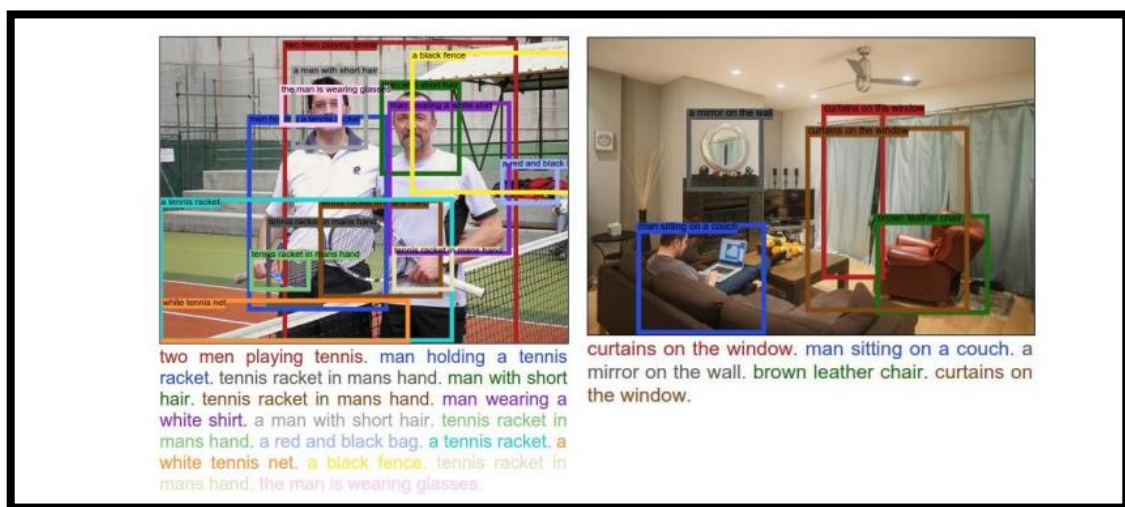


Figure II. 8:Exemples de sous-titrage dense

II.5.ImageCaptioning en langue Arabe

Le développement du sous-titrage automatique d'images en langue arabe (AIC) fait face à deux défis majeurs : la pénurie de jeux de données publics et la complexité morphologique et dialectale de la langue (18). Pour surmonter ces obstacles, la recherche a évolué à travers plusieurs approches clés :

Modèles basés sur les racines(19): Exploitation des mots-racines via des réseaux RNN pour générer directement du texte arabe, introduisant les premiers scores BLEU dans ce domaine.

Architectures CNN-LSTM (20)(21): Combinaison de modèles profonds pour la fusion visuelle-textuelle, entraînés sur des données traduites par crowdsourcing.

Création de Benchmarks (21) : Introduction de la version arabe de Flickr8k (référence publique pour éviter l'apprentissage excessif) et comparaison entre les modèles de traduction automatique et les modèles de bout en bout (end-to-end).

Mécanismes d'Attention(22) : Intégration de l'attention visuelle basée sur les objets et évaluation de la précision sémantique et lexicale des légendes.

II.5.1 Particularités de la langue arabe

Le traitement automatique de la langue arabe (TAL) se heurte à des défis linguistiques et computationnels majeurs, intrinsèquement liés à ses spécificités structurelles. L'étape de prétraitement s'avère particulièrement cruciale pour normaliser le texte et optimiser les performances des modèles d'apprentissage automatique. Ce processus standard englobe plusieurs phases critiques, telles que la tokenisation(segmentation textuelle), l'extraction des racines morphologiques (stemming/lemmatization) visant à réduire la forte variabilité flexionnelle des termes, ainsi que le filtrage des mots vides (stop words) sémantiquement faibles (tels que « » ,« من على » ,« في »).

Bien que ces étapes restreintes contribuent à la simplification dimensionnelle des données, la modélisation de l'arabe demeure complexe. Cette situation est accentuée d'une part par la richesse morpho-syntaxique de la langue, et d'autre part par la pénurie relative de corpus annotés et de benchmarks publics comparativement aux ressources massives disponibles pour la langue anglaise. (23)

II.5.2 Les défis de l'Arabic Image Captioning

La génération automatique de descriptions d'images en langue arabe (Arabic Image Captioning) se heurte à des verrous scientifiques et techniques majeurs. Ces obstacles découlent directement des propriétés intrinsèques de la langue arabe, qui compliquent l'apprentissage multimodal (vision-langage). Ces défis peuvent être structurés autour de quatre axes principaux :

II.5.2.1 La complexité morphologique

Sur le plan linguistique, l'arabe se distingue par une double complexité : énumérative (un nombre élevé de déclinaisons morphosyntaxiques) et intégrative (les règles de dérivation à partir des mots-racines). Pour les modèles de Deep Learning, cette richesse engendre une immense variété lexicale. L'analyse statistique des corpus (via des indicateurs comme l'entropie textuelle) montre que cette forte variabilité augmente la dimensionnalité du vocabulaire, rendant la prédiction du mot suivant par le décodeur textuel beaucoup plus difficile que pour l'anglais. (18)

II.5.2.2 L'ambiguïté liée à l'absence de diacritiques

Le système scriptural arabe repose majoritairement sur une orthographe dite "profonde", où les signes diacritiques (les voyelles brèves ou Harakat) sont omis dans les textes standards. Cette absence de repères phonétiques crée une forte ambiguïté homographique (des mots ayant la même écriture mais des sens totalement différents). Dans un système d'Image Captioning, le modèle ne peut pas s'appuyer uniquement sur le texte déjà généré ; il doit constamment fusionner les indices sémantiques de l'image pour lever l'ambiguïté et choisir le terme exact adapté au contexte visuel. (25)

II.5.2.3 Le phénomène de diglossie et le continuum linguistique

L'arabe est caractérisé par une diglossie marquée : l'arabe standard moderne (الفصحى), codifié et utilisé dans les milieux académiques et officiels, coexiste avec une multitude de dialectes régionaux (العامية) propres aux interactions quotidiennes. De plus, l'émergence de formes hybrides comme l'Arabe Parlé Éduqué (ESA) transforme cette dualité en un véritable continuum linguistique. Pour l'annotation d'images, cette situation pose deux problèmes majeurs : le manque d'un système d'écriture stable pour les dialectes et la difficulté de standardiser les annotations au sein des jeux de données, souvent partagés entre expressions formelles et dialectales. (26)

II.5.2.4 Les contraintes de directionnalité et le formatage cursif

Le traitement informatique du script arabe impose des contraintes techniques strictes au niveau des pipelines de traitement de texte (Natural Language Processing) :

La nature cursive : Les lettres changent de forme (glyphe) selon leur position dans le mot (initiale, médiane, finale ou isolée), ce qui complique l'étape de segmentation en sous-mots (tokenisation).

La bidirectionnalité (BiDi) : Bien que l'arabe s'écrive de droite à gauche, l'insertion de données numériques (chiffres) ou de caractères latins nécessite un affichage de gauche à droite. Cette gestion

bidirectionnelle complique l'alignement séquentiel des tokens est dans les architectures neuronales (comme les Transformers où LSTMs) lors de la génération de la légende. (27)

II.5.3 Jeux de données arabes

Les jeux de données constituent un élément fondamental dans le développement et l'évaluation des modèles d'Image Captioning en langue arabe. Ils fournissent les images ainsi que les descriptions textuelles nécessaires à l'entraînement, à la validation et au test des modèles. Parmi les jeux de données les plus utilisés dans ce domaine figurent Flickr8k Arabic et MS-COCO Arabic, qui représentent deux références importantes pour la génération automatique de descriptions d'images en arabe. (28)

II.5.3.1 Flickr8k Arabic

Le jeu de données Flickr8k, proposé par Hodosh(29) contient 8 092 images, chacune accompagnée de cinq descriptions en anglais, soit un total de 40 460 légendes. Ces descriptions ont été collectées via la plateforme Amazon Mechanical Turk afin d'assurer une diversité et une qualité des annotations.

Afin de disposer d'un corpus adapté à la langue arabe, plusieurs travaux ont été réalisés pour traduire ces descriptions. Parmi eux, ElJundi (21) ont proposé Arabic Flickr8k, une version largement utilisée par la communauté scientifique. Le corpus est réparti en 6 000 images pour l'entraînement, 1 000 images pour la validation et 1 000 images pour le test. Chaque image est associée à trois descriptions en arabe, soit un total de 24 000 légendes.

La traduction a été réalisée en deux étapes. Dans un premier temps, les descriptions anglaises ont été traduites automatiquement à l'aide de Google Translate API. Dans un second temps, elles ont été relues et corrigées par des traducteurs professionnels arabophones. Les auteurs ont finalement retenu les trois meilleures descriptions pour chaque image. Malgré cette validation humaine, certaines légendes peuvent encore présenter quelques imperfections linguistiques. (28)

II.5.3.2 MS-COCO Arabic

La base de données Microsoft Common Object in Contexte (MS-COCO), introduit par Lin et al. (30), est l'un des plus importants corpus utilisés dans les domaines de la vision par ordinateur et de la génération automatique de descriptions d'images.

Il comprend 123 287 images, chacune accompagnée de cinq descriptions, soit un total de 616 435 légendes. Les images représentent des scènes complexes de la vie quotidienne contenant plusieurs objets dans leur environnement naturel, ce qui en fait une référence pour l'entraînement et l'évaluation des modèles d'Image Captioning.

Dans la plupart des travaux de recherche, la répartition proposée par Karpathy et Fei-Fei (31) est adoptée. Elle comprend 113 287 images d'entraînement, 5 000 images de validation et 5 000 images de test.

Pour obtenir une version adaptée à la langue arabe, 414 113 descriptions correspondant à 82 783 images d'entraînement ont été traduites automatiquement à l'aide de Google Cloud Translation API, donnant naissance au corpus MS-COCO Arabic, aujourd'hui largement utilisé dans les recherches consacrées à la génération de descriptions d'images en arabe. (28)

II.6 Méthodes d'évaluation

L'évaluation de la qualité des descriptions d'images générées automatiquement (Image Captioning) repose sur des métriques linguistiques et sémantiques. Ces outils mesurent la proximité entre la phrase candidate produite par le modèle et un ensemble de phrases de référence rédigées par des experts humains.

II.6.1 BLEU

La métrique BLEU (Bilingual Evaluation Understudy), introduite par Papinen et al (32), est historiquement issue du domaine de la traduction automatique. Elle évalue la similarité entre deux énoncés en calculant la moyenne géométrique de la précision des n-grammes (généralement de 1 à 4). Afin d'éviter que le modèle ne privilégie des prédictions excessivement courtes pour maximiser sa précision, BLEU intègre un facteur de pénalité de brièveté (33)

BLEU-1 (Unigrammes) : La précision du vocabulaire

Le score BLEU-1 regarde les mots un par un. Il calcule simplement le pourcentage de mots générés par votre modèle qui apparaissent dans les légendes de référence.

BLEU-2 (Bigrammes) : L'association des mots

BLEU-2 analyse les correspondances par paires de deux mots consécutifs (exemple : "chien noir" [كلب أسود], "dans l'eau" [في الماء]).

BLEU-3 (Trigrammes) : La structure des propositions

BLEU-3 évalue des groupes de trois mots consécutifs (exemple : "un chien noir" [كلب أسود يسبح], "dans l'eau chaude" [في الماء الدافئ]).

BLEU-4 (4-grammes) : La fluidité et la syntaxe globale

C'est le standard international en recherche. Il évalue des séquences de quatre mots consécutifs. Il mesure la fluidité textuelle globale et la syntaxe à long terme. C'est l'examen final de la langue.

II.6.2 METEOR

Développée par Banerjee et Lavie (2005) (34), METEOR (Metric for Evaluation of Translation with Explicit Ordering) cherche à pallier certaines lacunes de BLEU, notamment l'absence de prise en compte du rappel et de l'alignement explicite des mots. Elle est définie comme la moyenne harmonique de la précision et du rappel des correspondances d'unigrammes. De plus, METEOR intègre une dimension sémantique en gérant les synonymes et les variations morphologiques via la base de données WordNet. Bien que cette approche textuelle capture mieux la flexibilité linguistique que le simple chevauchement de n-grammes, la stricte dépendance aux synonymes reste parfois insuffisante pour saisir une similarité sémantique globale. (33)

II.6.3 ROUGE

Proposée par Lin(35) pour l'évaluation des systèmes de résumé automatique et de génération de texte, la métrique ROUGE (Recall-Oriented Understudy for Gisting Evaluation) se base sur le chevauchement de n-grammes, de séquences de mots et de paires de termes. Dans le cadre de l'image captioning, la variante ROUGE-L est la plus couramment adoptée. Elle mesure la plus longue sous-séquence commune (Longest Common Subsequence - LCS) entre deux phrases. Étant fortement orientée vers le rappel, ROUGE tend à surestimer et à privilégier les phrases longues. (33)

II.6.4 CIDEr

CIDEr (Consensus-based Image Description Evaluation), introduite par Vedantam et al. (2015) (36), est une métrique spécifiquement conçue pour l'évaluation des descriptions d'images. Elle mesure le degré de consensus entre une phrase candidate et un ensemble de propositions de référence fournies par des annotateurs humains. Après une étape de racinisation (stemming), chaque phrase est décomposée en n-grammes (de 1 à 4). À l'instar du principe TF-IDF employé en recherche d'information, CIDEr attribue un poids plus faible aux n-grammes fréquemment présents dans l'ensemble du corpus afin de valoriser les termes porteurs d'information. Enfin, le score final est obtenu par le calcul de la similarité cosinus entre les vecteurs de n-grammes. Néanmoins, fonctionnant de manière purement statistique et lexicale, CIDEr peut parfois surévaluer des détails textuels secondaires au détriment de la structure globale de la description. (33)

II.6.5 SPICE

Pour s'affranchir des limitations purement lexicales, Anderson et al. (37) ont proposé SPICE (Semantic Propositional Image Caption Evaluation). Cette métrique repose sur l'alignement des graphes de scènes sémantiques (Scene Graphs). Le graphe de scène décompose une phrase en unités sémantiques distinctes : les classes d'objets (C), les types de relations (R) et les attributs (A). Formellement, une légende candidate est modélisée par un graphe

$$G(c) = \langle K(c), E(c), O(c) \rangle \quad (II.3)$$

où:

$O(c) \subseteq C$ Représente l'ensemble des objets identifiés.

$E(c) \subseteq O(c) * R * O(c)$ Désigne l'ensemble des relations unissant ces objets.

$K(c) \subseteq O(c) * A$ Est l'ensemble des attributs associés aux objets.

À partir de ce graphe, un ensemble de tuples est généré, et le score SPICE est calculé comme le score F1 mesurant l'accord entre les tuples de la phrase candidate et ceux des références. À l'instar de METEOR, SPICE utilise WordNet pour la correspondance des synonymes. Cependant, cette métrique est fortement tributaire de la qualité et de la précision du parseur syntaxique. Une erreur d'analyse syntaxique (par exemple, interpréter une action comme un objet ou un sujet comme un simple attribut) peut induire un calcul erroné de la métrique et fausser l'évaluation globale du modèle. (33)

II.7. Conclusion

En conclusion, ce chapitre a permis de présenter les concepts fondamentaux de la génération automatique de descriptions d'images (Image Captioning). Nous avons retracé l'évolution des principales approches, depuis les architectures basées sur les réseaux de neurones convolutifs (CNN) et les réseaux LSTM avec mécanisme d'attention jusqu'aux architectures modernes de type Vision Transformer (ViT). Nous avons également mis en évidence les défis spécifiques de l'Arabic Image Captioning, notamment la complexité linguistique de la langue arabe, l'absence de diacritiques, la diglossie ainsi que le manque de ressources annotées. Enfin, les principales métriques d'évaluation, telles que BLEU, METEOR, ROUGE, CIDEr et SPICE, ont été présentées afin d'évaluer la qualité des descriptions générées. L'ensemble de ces éléments constitue une base théorique essentielle pour la suite de ce travail, consacrée à la mise en œuvre de notre méthodologie et à l'analyse des résultats expérimentaux.

Chapitre III : Réalisation et Implémentation du système

III.1 Introduction

Dans ce chapitre nous présentons les étapes fondamentales de la conception de notre système de génération automatique de descriptions d'images (Image Captioning). Nous avons reposé sur le paradigme du Transfer Learning (apprentissage par transfert), qui consiste à réutiliser les connaissances acquises par un modèle préalablement entraîné sur une base de données de grande envergure, telle qu'ImageNet, afin de faciliter l'apprentissage d'une nouvelle tâche.

Dans le cadre de cette étude, deux architectures de référence, VGG16 et GoogLeNet (Inception V3), ont été retenues pour l'extraction des caractéristiques visuelles. Les sections suivantes présentent les principes du Transfer Learning, le processus d'extraction des caractéristiques propre à chacun de ces modèles, ainsi qu'une comparaison de leurs principales caractéristiques architecturales.

III.2 Environnement de développement

L'environnement de développement qui a été utilisé pour la conception et la réalisation de notre système de génération automatique de légendes d'images en langue arabe. Nous y détaillons les spécifications matérielles (hardware), l'écosystème logiciel, ainsi que les principales bibliothèques et Framework exploités pour l'implémentation, l'entraînement et l'évaluation des performances du modèle

III.2.1 Environnement matériel

Le modèle a été entièrement développé en Python sur la plateforme Google Colab. Ce choix s'explique par les ressources offertes par la plateforme (CPU, RAM et accélérateurs GPU), qui ont permis de gérer efficacement les charges de calcul élevées des algorithmes d'apprentissage profond et de dépasser les limites de l'infrastructure locale.

III.2.2 Environnement logiciel

Le tableau (**Tableau III. 3**) suivant présente l'environnement logiciel utilisé pour le développement et l'exploitation du système.

Logiciel	Version	Rôle
Python	3.12.11	Langage de programmation exécuté sur Google Colab.
Google Colab	En ligne	Plateforme cloud utilisée pour le développement, l'entraînement et l'évaluation du modèle.

Tableau III. 1: Environnement logiciel utilisé pour le développement du système

III.2.3 Bibliothèques et outils utilisés

Le tableau suivant présente les principales bibliothèques et outils utilisés pour le développement de notre système.

Bibliothèque	Rôle
TensorFlow	Développement et entraînement du modèle de Deep Learning.
NumPy	Manipulation des tableaux numériques et traitement des données.
Matplotlib	Visualisation des courbes d'entraînement et des résultats.
Pickle	Sauvegarde et chargement des fichiers de caractéristiques, des modèles et du tokenisation.
OS	Gestion des fichiers et des répertoires du projet.
arabic_resaper	Restructuration du texte arabe afin d'assurer un affichage correct.
bidirectional	Gestion du sens d'écriture de droite à gauche (RTL) pour les textes arabes.
VGG16	Extraction des caractéristiques visuelles des images.
GoogLeNet	Modèle utilisé pour comparer les performances de l'extraction des caractéristiques avec VGG16.
AraBERT	Tokenisation et représentation contextuelle des descriptions en langue arabe.
KerasTokenisation	Tokenisation classique utilisée comme méthode de référence pour la comparaison avec AraBERT.

Tableau III. 2: Bibliothèques et outils utilisés dans le développement du système

Le tableau (**Tableau III. 4**) récapitule l'ensemble des bibliothèques et des outils logiciels utilisés lors de la conception de notre système de génération automatique de descriptions d'images. Ces ressources ont contribué à la mise en œuvre de l'ensemble du processus de développement : depuis le prétraitement des données et l'extraction des descriptions visuelles jusqu'à la modélisation textuelle du contenu en arabe, en passant par les étapes de conception de l'architecture du modèle, de son apprentissage et de son évaluation expérimentale.

III.3 Présentation et préparation des données images

Cette section est consacrée à la présentation de l'ensemble de données sélectionné, ainsi qu'aux différentes opérations de prétraitement appliquées aux éléments visuels (images). Ces étapes préparatoires sont indispensables pour harmoniser et normaliser les données avant de lancer la phase d'entraînement du modèle.

III.3.1 Présentation du dataset Flickr8k

Le système proposé s'appuie sur l'ensemble de données Flickr8k (Tableau III. 5), largement reconnu et considéré comme une référence (benchmark) dans la littérature scientifique consacrée à la génération automatique de légendes d'images (Image Captioning). Cet ensemble offre une grande diversité de scènes visuelles, chacune étant associée à plusieurs légendes textuelles, ce qui constitue un support idéal pour l'apprentissage et la validation de nos modèles.

Dans le cadre de notre protocole expérimental, le jeu de données a été structuré puis divisé en trois sous-ensembles distincts : l'ensemble d'entraînement (Training Set), l'ensemble de validation (Validation Set) et l'ensemble de test (Test Set). Cette segmentation précise des données permet d'optimiser les poids du réseau, d'ajuster objectivement les hyperparamètres et d'évaluer avec précision les capacités de généralisation du système face à de nouveaux cas.

Ensemble	Rôle
Training Set (6000 images)	Entraînement du modèle
Validation Set (1000 images)	Ajustement des hyperparamètres et suivi de l'apprentissage
Test Set (1000 images)	Évaluation finale des performances

Tableau III. 3: Répartition du jeu de données Flickr8k

III.3.2 Prétraitement des images

Le prétraitement des images représente une étape fondamentale dans le développement d'un système d'Image Captioning. Il consiste à préparer les images avant leur exploitation par les modèles d'apprentissage profond, en assurant une représentation homogène des données d'entrée et une compatibilité avec les exigences des architectures utilisées.

Dans le cadre de ce travail, les images du jeu de données Flickr8k sont d'abord chargées, puis redimensionnées en fonction du modèle retenu. Les images destinées au modèle VGG16 sont redimensionnées à 224×224 pixels, tandis que celles utilisées avec InceptionV3 sont redimensionnées à 299×299 pixels (Figure III.1).

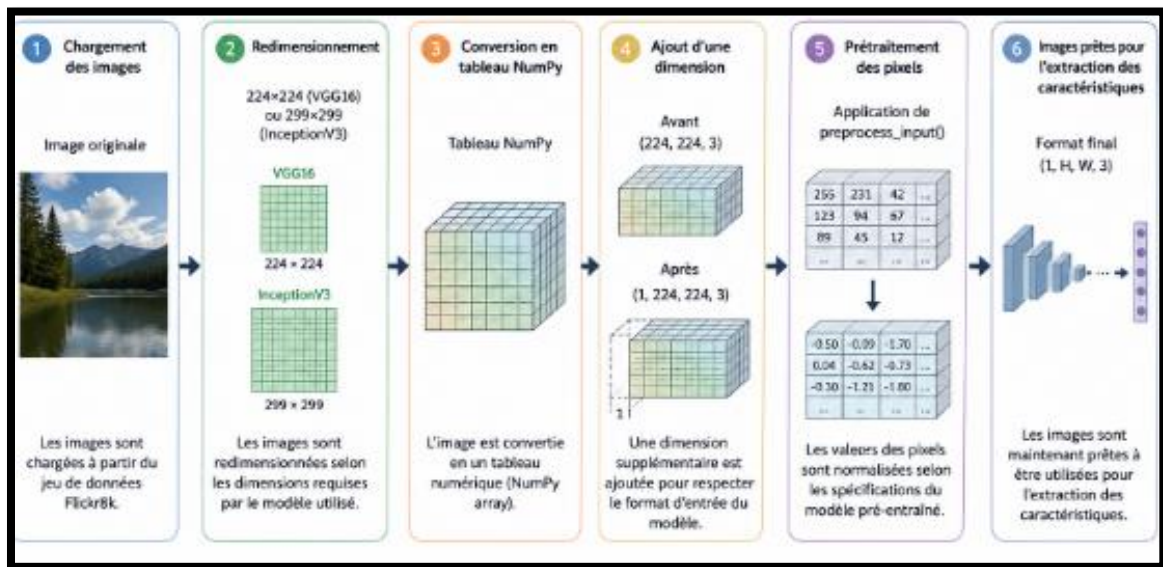


Figure III. 1 : Étapes du prétraitement des images

III.4 Extraction des caractéristiques par Transfer Learning

L'extraction de caractéristiques est une étape essentielle dans les systèmes de génération automatique de légendes d'images (Image Captioning). Les images brutes ne pouvant pas être directement fournies au modèle de génération, elles sont d'abord soumises à un traitement préalable par un réseau neuronal convolutif (CNN) pré-entraîné, dans le but de générer une représentation numérique dense appelée vecteur de caractéristiques.

Cette approche s'appuie sur le modèle d'apprentissage par transfert (Transfer Learning), qui consiste à exploiter les connaissances acquises par un modèle préalablement entraîné sur une base de

données à grande échelle telle qu'ImageNet. Cette stratégie permet d'extraire des descripteurs visuels riches et distinctifs, tout en évitant l'étape complexe consistant à entraîner un réseau convolutif à partir de zéro (from scratch).

Dans le cadre de ce travail, deux architectures pré-entraînées ont été utilisées pour accomplir cette tâche : VGG16 et GoogLeNet. Leurs caractéristiques structurelles seront détaillées dans les sections suivantes.

III.4.1 Principe du Transfer Learning

L'apprentissage par transfert (Transfer Learning) est un modèle majeur dans le domaine de l'apprentissage automatique. Il repose sur la réutilisation des connaissances et des compétences acquises par un modèle préalablement entraîné sur une tâche source afin de résoudre une tâche cible connexe. Au lieu de procéder à un apprentissage à partir de zéro, cette approche tire parti des représentations hiérarchiques et des caractéristiques globales apprises à partir de bases de données à grande échelle, telles qu'ImageNet. L'importance de cette stratégie est particulièrement évidente dans les architectures d'apprentissage profond (Deep Learning), où l'ensemble du processus d'optimisation est libéré de contraintes lourdes, telles que le besoin considérable en ressources informatiques et la nécessité de disposer d'énormes quantités de données étiquetées.

Dans le cadre de ces recherches, l'apprentissage par transfert est appliqué pour extraire des caractéristiques visuelles à l'aide de réseaux neuronaux convolutifs (CNN) pré-entraînés (Figure III.2). Ces descriptions de haut niveau sont ensuite utilisées comme entrées pour un modèle de génération de légendes d'images (Image Captioning).

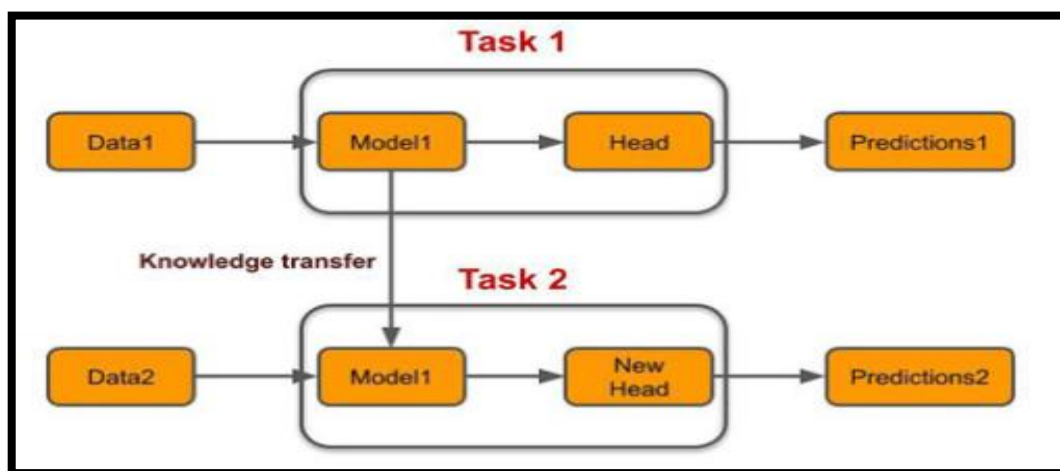


Figure III. 2: Le principe du processus d'apprentissage par transfert (Transfer Learning)

III.4.2 Avantages du Transfer Learning

L'intégration du modèle d'apprentissage par transfert (Transfer Learning) dans les architectures d'apprentissage profond offre des avantages fondamentaux, ce qui justifie son adoption à grande échelle pour les tâches complexes dans le domaine de la vision par ordinateur :

Amélioration de la convergence et réduction du temps d'apprentissage : accélération notable de la phase de convergence grâce à la pré-configuration du modèle avec des poids pré-entraînés, ce qui réduit au minimum le temps nécessaire à son adaptation à la tâche visée.

Allègement des contraintes liées au volume de données : cette approche permet d'atteindre des niveaux de performance élevés et une robustesse algorithmique remarquable, même en cas de rareté ou de dimensions limitées des données d'entraînement (petites données).

Optimisation des ressources informatiques : en évitant un entraînement complet à partir de zéro, la charge de calcul et la consommation de ressources matérielles (GPU/TPU) sont considérablement réduites.

Amélioration de la précision grâce à l'extraction de caractéristiques générales : l'apprentissage préalable sur des ensembles de données volumineux et variés (tels qu'ImageNet) dote le modèle de caractéristiques visuelles de haut niveau, ce qui renforce la précision lors de la phase d'inférence.

Robustesse et transférabilité : la réutilisation de structures de caractéristiques ayant fait leurs preuves confère au modèle une meilleure capacité de généralisation, ce qui réduit le risque de surapprentissage (overfitting).

III.4.3 Extraction des caractéristiques avec VGG16

Dans le cadre de ce travail, l'architecture VGG16 a été utilisée comme extracteur de caractéristiques visuelles, en s'appuyant sur le modèle d'apprentissage par transfert (Transfer Learning) (Figure III.3). Afin d'adapter ce modèle à notre problématique, le module de classification final a été supprimé pour ne conserver que les couches convolutives du réseau. Cette configuration permet d'isoler les informations visuelles et de les représenter sous la forme d'un vecteur de caractéristiques de haut niveau (Feature Vector). Ce vecteur est ensuite utilisé comme entrée (ou contexte initial) pour le modèle de génération de descriptions, chargé de traduire le contenu sémantique des images en texte en langue arabe.

Le processus d'extraction des caractéristiques visuelles via l'architecture VGG16 se compose de plusieurs étapes successives. Dans un premier temps, les images du jeu de données sont soumises à un redimensionnement géométrique uniforme à une résolution de $(224 * 224)$ pixels, conformément aux exigences de la couche d'entrée du réseau. Ces images sont ensuite soumises à la fonction de normalisation [preprocess_input], ce qui garantit l'alignement de la distribution des valeurs de pixels avec les statistiques de centralisation et de réduction appliquées lors de l'entraînement initial du modèle sur ImageNet. Chaque échantillon est ensuite transmis au réseau VGG16, préalablement configuré en mode « cut-off » (include_top=False) afin d'exclure les couches denses de la classification finale. Les activations extraites à la sortie du dernier bloc convolutif sont soumises à un aplatissement (flattening) ou à une agrégation globale afin de générer un vecteur de caractéristiques (Feature Vector), qui regroupe les éléments sémantiques essentiels de l'image. Au cours de la phase finale, ces vecteurs sont classés et enregistrés dans un fichier dédié, ce qui facilite leur réutilisation ultérieure lors de la phase d'entraînement du modèle de génération automatique de légendes (Image Captioning).

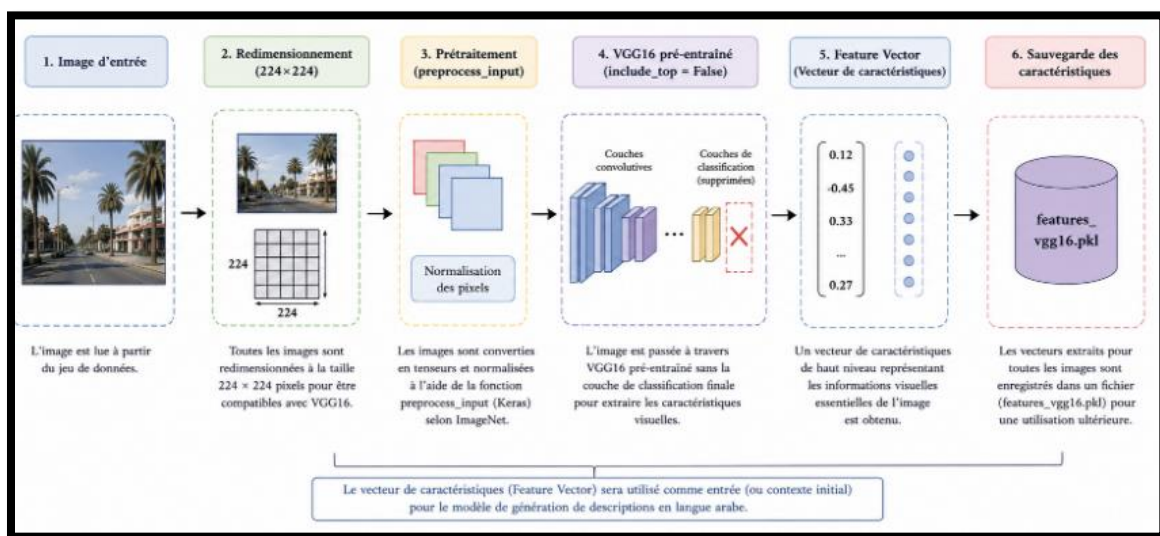


Figure III. 3:Processus d'extraction avec VGG16

III.4.4 Extraction des caractéristiques avec GoogLeNet

Outre l'architecture VGG16, le modèle GoogLeNet (appartenant à la famille des architectures Inception) a été implémenté comme extracteur de descripteurs visuels (Figure III.4), s'appuyant également sur le paradigme de l'apprentissage par transfert (Transfer Learning). Suivant une approche analogue, la couche de classification finale (top layer) a été tronquée afin d'isoler

uniquement le sous-réseau dédié à la capture des primitives visuelles. Cette configuration permet de projeter chaque image dans un espace latent sous la forme d'un vecteur de caractéristiques (Feature Vector), destiné à alimenter le modèle de génération automatique de descriptions textuelles en langue arabe.

Le pipeline d'extraction s'articule de la manière suivante : les images du jeu de données subissent d'abord un redimensionnement géométrique standardisé à une résolution de (299 * 299) pixels correspondant aux contraintes spécifiques de la couche d'entrée du réseau, suivi d'une phase de normalisation alignée sur les statistiques de la base de données ImageNet. Après propagation des données à travers le réseau configuré sans son bloc classificatoire, les activations issues de la dernière couche de mise en commun globale (Global Average Pooling) sont capturées pour former les représentations denses finales. Ces vecteurs de caractéristiques sont enfin sérialisés et sauvegardés, optimisant ainsi leur réutilisation lors de la phase d'entraînement ultérieur du modèle de génération.

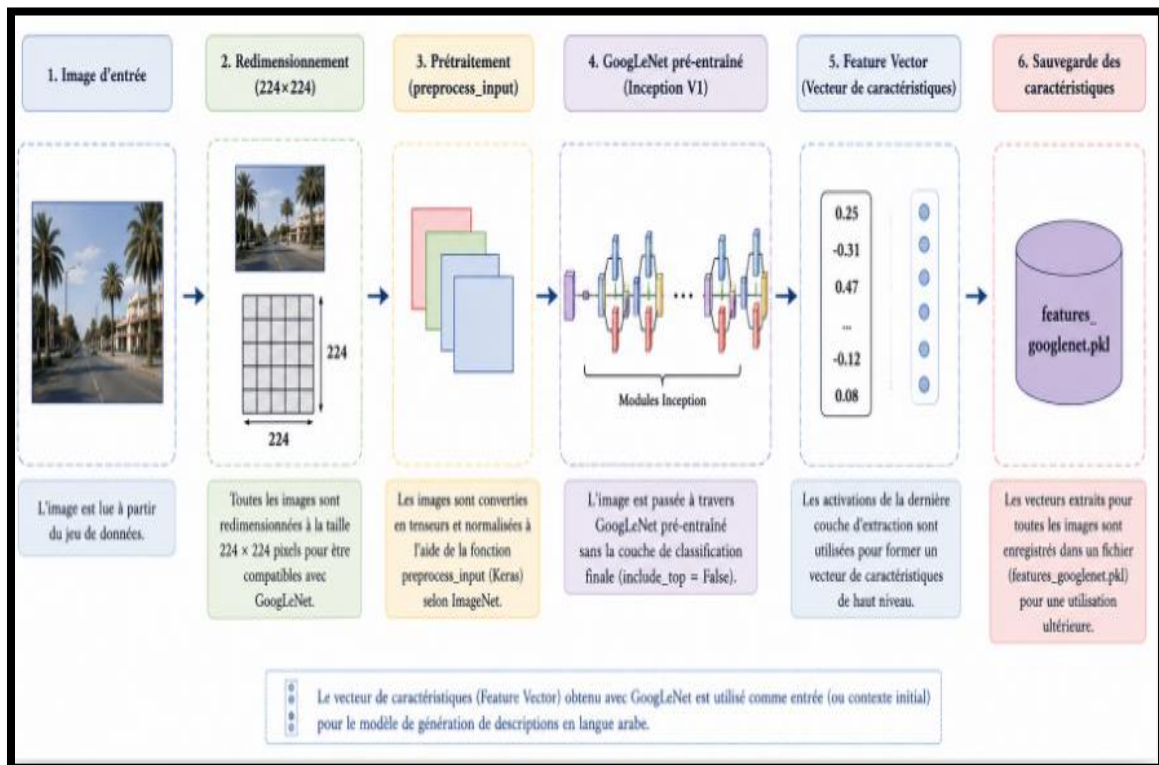


Figure III. 4:Processus d'extraction avec GoogLeNet

III.4.5 Comparaison entre VGG16 et GoogLeNet

Les architectures VGG16 et GoogLeNet représentent deux jalons majeurs dans l'évolution des réseaux de neurones convolutifs (CNN) appliqués à l'extraction de descripteurs visuels. Bien que partageant le même paradigme d'apprentissage par transfert (Transfer Learning) via leur pré-entraînement initial sur le corpus standardisé ImageNet, leurs philosophies de conception divergent structurellement en termes de complexité topologique, de densité paramétrique et d'efficacité computationnel (Tableau III. 6) .

D'une part, l'architecture VGG16 se distingue par une topologie séquentielle et homogène, caractérisée par l'empilement systématique de couches de convolution à filtres de petite taille (3×3) intercalées de couches de sous-échantillonnage (Max-Pooling). Si cette régularité structurelle facilite grandement son implémentation et l'analyse de sa dynamique interne, elle induit en contrepartie une explosion du volume des paramètres. Cette redondance paramétrique engendre une empreinte mémoire substantielle et des coûts de calcul intensifs lors des phases d'inférence.

À l'inverse, GoogLeNet (Inception-v1) introduit une rupture conceptuelle par le biais des modules Inception. Ces blocs fonctionnels s'affranchissent de la linéarité classique en exécutant des convolutions multi-échelles en parallèle (1×1 , 3×3 , 5×5) combinées à des mécanismes de réduction de dimensionnalité. Cette optimisation architecturale permet à GoogLeNet de surpasser la capacité de représentation de VGG16 tout en réduisant drastiquement le nombre de paramètres.

En définitive, le choix algorithmique entre ces deux architectures relève d'un arbitrage technofonctionnel : VGG16 demeure privilégié pour sa robustesse intrinsèque et sa simplicité conceptuelle, tandis que GoogLeNet s'impose comme une alternative optimale face à des contraintes de ressources computationnelles strictes.

Critère	VGG16	GoogLeNet (Inception V3)
Année	2014	2015
Développeur	Visual Geometry Group (Oxford)	Google
Profondeur	16 couches	48 couches
Architecture	Convolutions successives	Modules Inception
Taille des filtres	3x3	Modules Inception (filtres parallèles de différentes tailles)
Taille d'image d'entrée (Input)	224x224	299x299
Nombre de paramètres	≈ 138 millions	≈ 24 millions (léger et efficace)
Taille du modèle	Grande	Plus compacte
Temps de calcul	Plus élevé	Plus faible
Consommation mémoire	Très gourmand	Optimisé (idéal pour les plateformes limitées)
Efficacité	Faible : Consomme beaucoup de ressources pour une précision aujourd'hui dépassée.	Élevée : Offre une excellente précision avec 5 fois moins de paramètres.

Tableau III. 4: Présente une comparaison des principales caractéristiques des modèles VGG16 et GoogLeNet utilisés pour l'extraction des caractéristiques visuelles.

III.5 Prétraitement du texte

Le prétraitement des descriptions textuels constitue une étape importante de notre système, visant à améliorer la qualité du corpus linguistique avant son intégration dans l'architecture d'apprentissage profond. Ce processus a pour objectif d'uniformiser les descriptions et de réduire la variabilité lexicale inhérente à la langue arabe (Figure III.5).

Le protocole commence par l'extraction des données textuelles à partir d'une source structurée, suivie d'une phase d'uniformisation visant à harmoniser la représentation des caractères. Un processus de nettoyage minutieux est ensuite appliqué pour éliminer le bruit informationnel : suppression des signes de ponctuation, des signes de diacritique (mouvements), des segments écrits

en caractères latins, ainsi que des lettres allongées (tamad). De plus, les entités numériques sont converties au format textuel, et les unités lexicales composées d'un seul caractère sont exclues.

Les différentes méthodes de normalisation sont :

- Suivre les règles grammaticales ou de structure tout au long du texte, en respectant exemple en supprimant les espaces ou signes de ponctuation Ou bien, préserver les lettres minuscules dans l'intégralité du texte.
- Enlever les préfixes et suffixes des mots et les réduire à leur base.
- Élimination des mots inutiles comme « et », « est », « le » et d'autres qui n'apportent pas de prix au contenu.
- Élimination des mots vides (Suppression des Stop words) : Retrait des mots sans signification intense (ex. "هذا هو".)
- Stemming et Lemmatisation : Retrait des mots à leur base ou forme standard ex ("يكتب" → "كتب")

Une fois ce processus de nettoyage terminé, chaque séquence est marquée par les balises syntaxiques « startseq » et « endseq », qui définissent respectivement les limites initiales et finale de la phrase. Les descriptions sont ensuite converties en vecteurs sous forme de séquences d'identifiants numériques à l'aide de l'outil Tokenisation. Enfin, les dimensions du vocabulaire et la longueur maximale des vecteurs sont calculées afin de déterminer les contraintes dimensionnelles de la couche d'entrée du modèle.

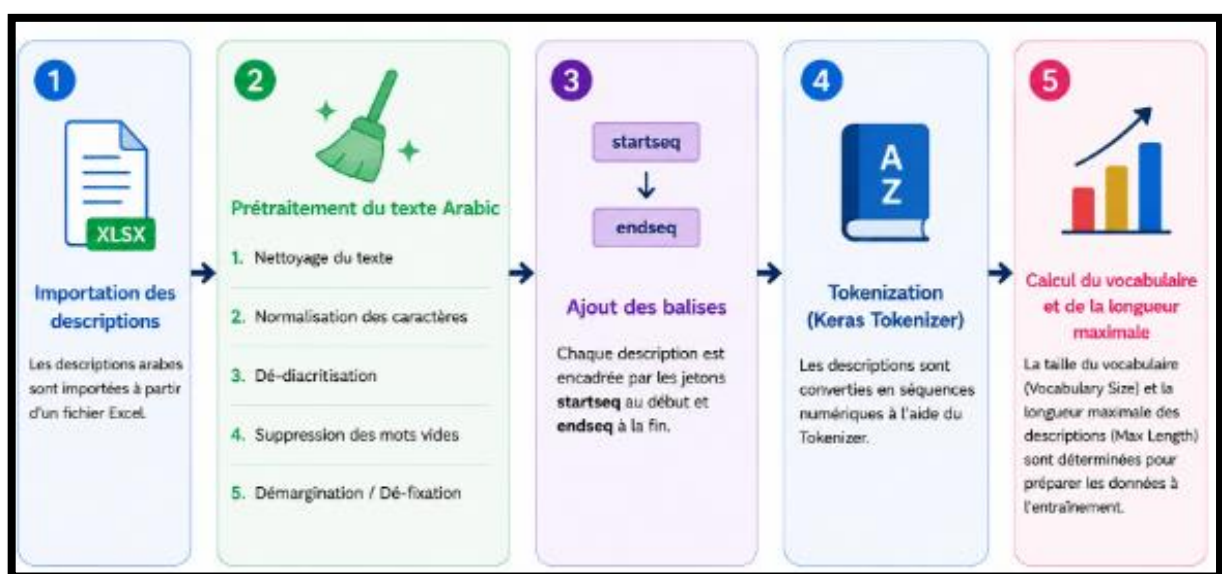


Figure III. 5:Étapes du prétraitement des descriptions arabes

III.5.1 Tokenisation classique (KerasTokenisation)

La tokenisation classique (Word-level Tokenisation) représente l'une des approches les plus utilisées en traitement automatique du langage naturel. Elle repose sur la segmentation d'un texte brut en mots distincts, en utilisant principalement les espaces et les signes de ponctuation comme séparateurs. Dans ce cadre, chaque mot différent présent dans le corpus d'apprentissage est associé à un identifiant numérique unique au sein d'un vocabulaire fixe.

Cette méthode présente plusieurs limites importantes lorsqu'elle est appliquée à la langue arabe.

Tout d'abord, l'arabe se caractérise par une richesse morphologique élevée et un phénomène d'agglutination. Un même mot peut combiner un radical avec plusieurs préfixes et suffixes. Ainsi, une forme comme « وبالمكتبة » est considérée par un tokenisation classique comme un seul tokens, sans prise en compte de sa structure interne.

Ensuite, cette variabilité morphologique entraîne une explosion du vocabulaire. Un même concept peut apparaître sous un grand nombre de formes différentes, ce qui augmente considérablement la taille du dictionnaire nécessaire à la représentation des données textuelles.

Enfin, la tokenisation par mots entiers souffre du problème des mots hors vocabulaire (Out Of Vocabulary – OOV). Lorsqu'un mot rencontré pendant la phase de test n'existe pas exactement sous la même forme dans les données d'apprentissage, il est remplacé par le symbole générique [UNK] (Unknown). Cette substitution provoque une perte d'information sémantique, particulièrement pénalisante dans une tâche de génération automatique de descriptions d'images, où la précision des mots utilisés joue un rôle essentiel dans la qualité de la légende produite.

III.5.2 Tokensization avec AraBERT

Afin de surmonter les limites de la tokenisation classique, les architectures basées sur les Transformers, telles qu'AraBERT (31) s'appuient sur une approche de tokenisation par sous-mots (*Subword Tokensization*). Cette technique utilise des algorithmes statistiques tels que Word Piece ou Sentence Piece, souvent associés, dans le cas de la langue arabe, à un prétraitement morphologique réalisé par des outils spécialisés.

Contrairement à la tokenisation par mots entiers, cette méthode ne se limite pas à la segmentation selon les espaces. Elle consiste à décomposer les mots complexes, rares ou fortement agglutinés en unités plus petites appelées *sous-mots*, appartenant à un vocabulaire prédéfini et optimisé. Dans

AraBERT, ce vocabulaire est généralement limité à plusieurs dizaines de milliers de tokens, ce qui permet de maintenir un bon équilibre entre couverture lexicale et efficacité du modèle.

Le principe de fonctionnement est relativement simple : lorsqu'un mot apparaît fréquemment dans le corpus d'apprentissage, il est conservé sous sa forme complète. En revanche, les mots rares ou morphologiquement complexes sont fragmentés en plusieurs sous-unités. Dans certaines implémentations basées sur BERT, les sous-mots qui suivent le premier fragment sont précédés d'un préfixe spécifique, tel que « ## », afin d'indiquer qu'ils appartiennent au même mot.

Par exemple, le mot arabe « وبالسّيارات » peut-être segmenté en plusieurs unités distinctes correspondant à ses différentes composantes morphologiques : و + ب + ال + سّيارات. Cette décomposition permet au modèle de mieux exploiter la structure interne des mots et de capturer les relations linguistiques propres à la langue arabe.

Une autre caractéristique importante de la tokenisation utilisée par AraBERT est l'introduction de tokens spéciaux destinés à structurer les séquences textuelles et à faciliter le fonctionnement du mécanisme d'attention :

[بداية] : placé au début de la séquence, il sert à représenter l'information globale du texte.

[نهاية] : ajouté à la fin de la phrase afin de marquer explicitement la limite de la séquence.

[PAD] : utilisé pour compléter les séquences plus courtes et obtenir des longueurs uniformes, nécessaires au traitement par lots (*mini-batch*).

Grâce à cette approche, AraBERT réduit considérablement le problème des mots hors vocabulaire (*Out Of Vocabulary*). Même lorsqu'un mot n'a jamais été rencontré durant l'entraînement, il peut être décomposé en sous-unités connues par le modèle. Cette capacité améliore la robustesse du système et favorise une meilleure compréhension du texte, ce qui constitue un avantage majeur pour les tâches de génération automatique de descriptions d'images en langue arabe.

III.6. Implémentation du modèle de génération des descriptions

Cette section présente la mise en œuvre logicielle du système de génération de descriptions développé dans le cadre de ce travail. Notre approche repose sur une architecture de type Merge (fusion tardive), dans laquelle les informations visuelles et textuelles sont traitées séparément avant d'être combinées. Afin d'évaluer l'impact de l'extraction des caractéristiques visuelles sur les performances du modèle, deux encodeurs d'images différents, VGG16 et InceptionV3, ont été

utilisés et associés à un décodeur séquentiel basé sur les réseaux LSTM. Cette configuration permet de comparer l'efficacité des deux extracteurs visuels dans la tâche de génération automatique de descriptions.

Architecture du modèle

L'architecture proposée repose sur deux branches distinctes dont les représentations sont combinées avant la phase de génération du texte.

1. Encodeur la partie visuelle

L'extraction des caractéristiques visuelles est réalisée à l'aide de deux réseaux profonds pré-entraînés sur la base de données ImageNet. Les couches finales de classification sont supprimées afin de conserver uniquement les représentations visuelles utiles à la description de l'image.

Configuration VGG16 : les caractéristiques sont récupérées à la sortie de la couche entièrement produisant un vecteur de représentation de dimension 4096.

Configuration InceptionV3 : les caractéristiques sont extraites à partir de la dernière couche précédant la classification. Cette architecture, héritée de GoogLeNet, exploite les modules Inception et les différentes couches de concaténation, permettant d'obtenir une représentation visuelle compacte et riche en informations sémantiques de dimension 2048.

Couche de projection : indépendamment de l'encodeur utilisé, les caractéristiques extraites sont d'abord régularisées à l'aide d'une couche Dropout avec un taux de 0,5 afin de limiter le surapprentissage. Elles sont ensuite projetées dans un espace latent commun grâce à une couche Dense de 256 neurones utilisant la fonction d'activation ReLU, facilitant ainsi leur fusion avec les représentations textuelles.

2. Décodeur (partie textuelle)

La partie textuelle est chargée de traiter les descriptions en langue arabe afin de générer les séquences de mots correspondantes. Dans un premier temps, les légendes sont transformées en représentations vectorielles à l'aide d'une couche Embedding de dimension 256. L'option `mask_zero=True` est utilisée pour ignorer les valeurs de remplissage (padding), ce qui permet au modèle de se concentrer uniquement sur les mots significatifs de la séquence.

Afin de limiter le surapprentissage, une couche Dropout avec un taux de 0,5 est appliquée avant le traitement séquentiel. Les séquences sont ensuite analysées par une couche LSTM composée de 256

unités, capable de capturer les dépendances temporelles et les relations syntaxiques propres à la langue arabe.

3. Bloc de fusion et couche de sortie

Les représentations issues des branches visuelle et textuelle, toutes deux de dimension 256, sont ensuite combinées au sein d'un module de fusion. Cette fusion est réalisée par une addition élément par élément (Add), permettant d'intégrer simultanément les informations visuelles et linguistiques dans une représentation commune.

Le vecteur résultant est ensuite transmis à une couche dense intermédiaire de 256 neurones utilisant la fonction d'activation ReLU, afin d'enrichir les interactions entre les deux modalités. Enfin, une couche Dense finale associée à une fonction Softmax génère une distribution de probabilité sur l'ensemble du vocabulaire (vocab_size), permettant ainsi de prédire le mot suivant de la légende à chaque étape de la génération.

III.7 Résultats et évaluation

Cette section présente l'évaluation quantitative et qualitative des modèles entraînés après 30 époques(Tableau III. 7).

Les hyperparamètres réels extraits du code sont synthétisés dans le tableau suivant :

Hyperparamètre	Valeur Réelle dans le Programme	Rôle Technique
Optimiseur	Adam	Algorithme de descente de gradient stochastique adaptatif.
Fonction de perte	categorical_crossentropy	Mesure de l'erreur
Nombre d'époques	30	Nombre d'itérations complètes sur le jeu de données.
Dimension de l'Embedding	256	Taille de la représentation vectorielle des mots.

Cellules de mémoire LSTM	256	Capacité de rétention du contexte textuel arabe.
Taux de Dropout	0.5	Prévention du surapprentissage sur les deux branches.

Tableau III. 5: Synthèse des hyperparamètres et configuration d'entraînement du modèle.

L’expérimentation menée dans ce travail a permis d’évaluer l’influence de l’encodeur visuel sur les performances du système de génération de descriptions en arabe, comme le montrent les figures (Figure III.6, Figure III.7, Figure III.8, Figure III. 9).

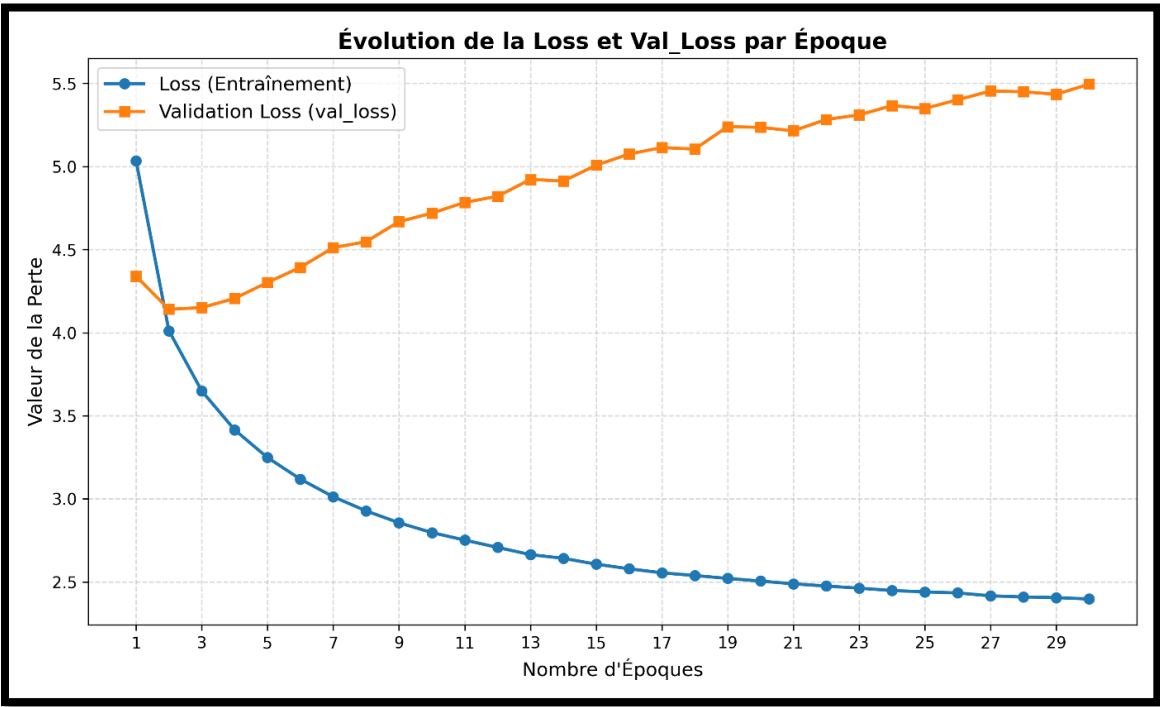


Figure III. 6:Courbe d’apprentissage avec GoogLeNET en Tokenisation classique

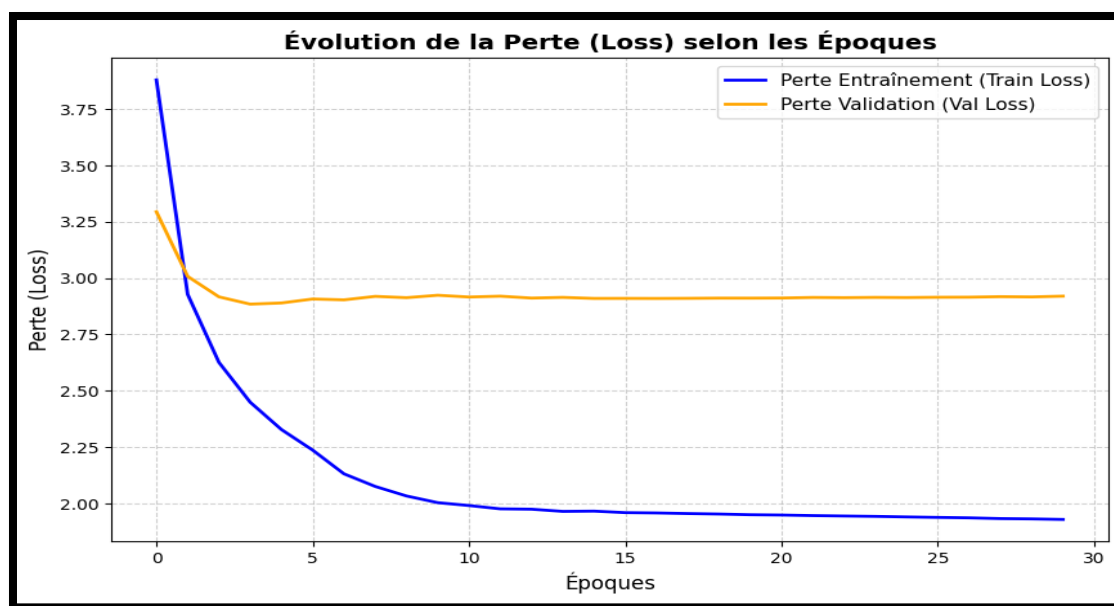


Figure III. 7: Courbe d'apprentissage avec googleNET en Tokenisation AraBERT

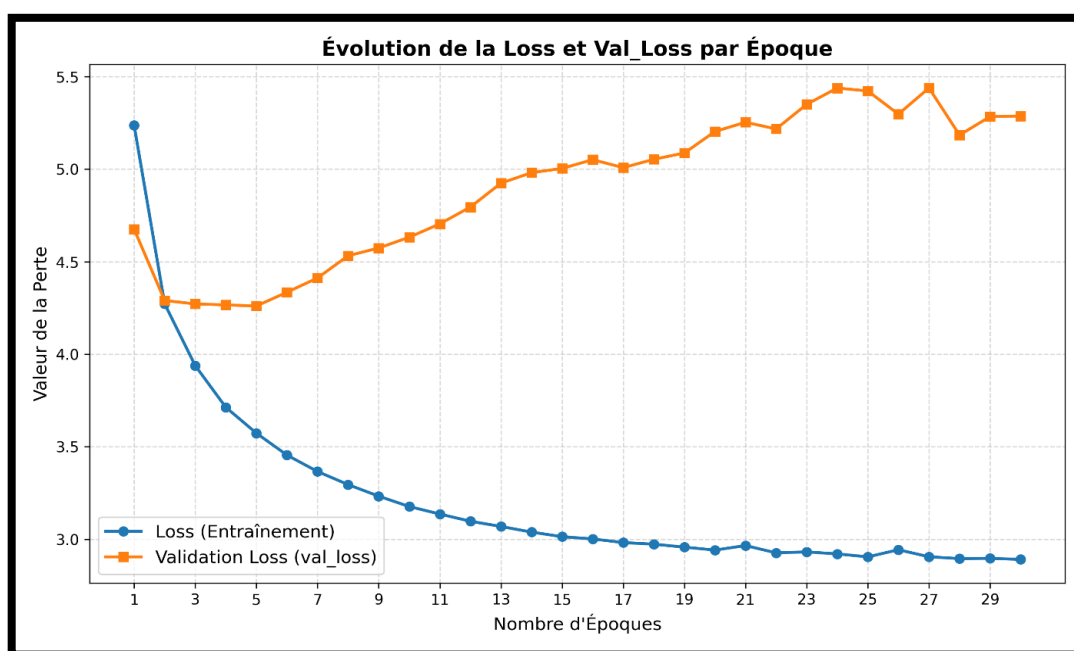


Figure III. 8: Courbe d'apprentissage avec VGG16 en Tokenisation classique

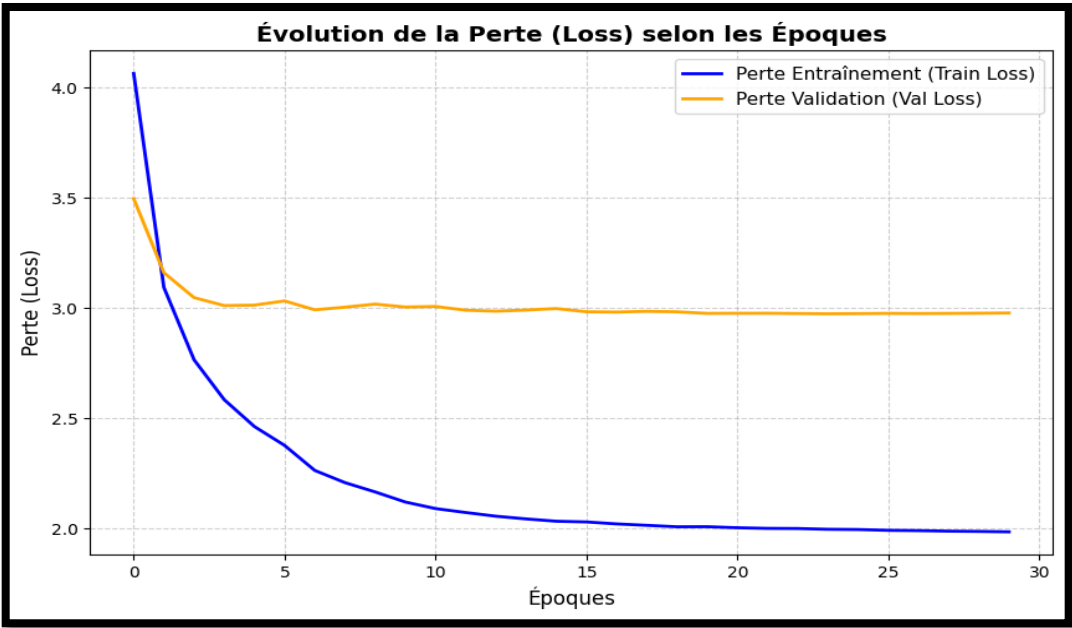


Figure III. 9: Courbe d'apprentissage avec VGG16 en Tokenisation AraBERT

Critère d'Analyse	GoogleNET en Tokenisation classique	GoogleNET en Tokenisation AraBERT
Allure de la <i>Train Loss</i>	Diminution régulière et progressive (de ≈ 5.0 à ≈ 2.4)	Diminution fluide et asymptotique plus rapide (de ≈ 3.8 à ≈ 1.9).
Comportement de la <i>Val Loss</i>	Décroissance initiale minimale (époches 1-2) suivie d'une remontée stricte et continue vers 5.5	Diminution initiale rapide, puis stabilisation stricte en plateau (≈ 2.9) dès la 5ème époque.
Écart (<i>Gap</i>) Train/Val	Divergence orthogonale majeure et croissante au fil des époques.	Écart constant, parallèle et maîtrisé sans aucune divergence
Diagnostic Global	Surapprentissage (Overfitting) précoce et critique.	Convergence stable, avec un léger risque de Sous-apprentissage (Underfitting) ou saturation précoce.
Capacité de Généralisation	Très faible (le modèle mémorise le bruit du jeu d'entraînement).	Satisfaisante et stable, bien que les performances globales restent perfectibles.

Tableau III. 6: Évaluation comparative des performances et des diagnostics de convergence

L'analyse comparative des courbes d'apprentissage (Tableau III. 6) met clairement en évidence l'influence de la méthode de tokenisation sur les performances du modèle GoogleNet. Les résultats montrent qu'une tokenisation classique entraîne un phénomène marqué de surapprentissage (overfitting), se traduisant par une augmentation progressive de la perte de validation (Validation Loss), qui atteint une valeur d'environ 5,5 en fin d'entraînement. En revanche, l'utilisation de la tokenisation basée sur AraBERT permet de stabiliser significativement le processus d'apprentissage. La perte de validation converge rapidement vers un plateau stable, situé autour de 2,9 dès la cinquième époque, traduisant une meilleure capacité de généralisation du modèle. Cette amélioration peut être attribuée à l'approche de segmentation en sous-mots adoptée par AraBERT, qui permet de mieux représenter les particularités morphologiques de la langue arabe tout en réduisant la variabilité lexicale. Par conséquent, le modèle apprend des représentations plus pertinentes et limite le risque de mémorisation des données d'entraînement. Toutefois, malgré cette amélioration notable, le niveau atteint par la perte de validation indique que des optimisations supplémentaires de l'architecture ou du processus d'entraînement pourraient encore être envisagées afin d'améliorer davantage les performances globales du système

Critère d'Analyse	VGG16 en Tokenisation classique	VGG16 en Tokenisation AraBERT
Évolution de la <i>Train Loss</i>	Décroissance régulière (de ≈ 5.23 à ≈ 2.90).	Décroissance fluide et asymptotique (de ≈ 4.07 à ≈ 1.98).
Comportement de la <i>Val Loss</i>	Baisse initiale éphémère (époques 1-5), suivie d'une remontée continue et oscillatoire jusqu'à ≈ 5.28	Baisse rapide (époques 1-3), suivie d'une stabilisation parfaite en plateau horizontal (≈ 2.98).
Comportement des Courbes	Divergence orthogonale majeure (les courbes s'écartent fortement après l'époque 5).	Convergence stable (les deux courbes restent parallèles sans aucune divergence).
Diagnostic Global	Surapprentissage (Overfitting) sévère et instabilité en fin de cycle.	Convergence robuste et stable (léger risque de saturation précoce).
Capacité de Généralisation	Très faible (le modèle n'est pas capable de traiter de nouvelles données).	Satisfaisante et maîtrisée (le modèle conserve sa robustesse face aux données de test).

Tableau III. 7: Analyse comparative de l'évolution de la fonction de perte (Loss) et du phénomène de surapprentissage

L'analyse de la courbe d'apprentissage indique clairement comment le choix de l'architecture et du prétraitement textuel influence le processus d'entraînement. Les résultats démontrent que l'utilisation du modèle VGG16 associé à une tokenisation classique souffre de manière critique d'un phénomène de surapprentissage (overfitting) précoce et oscillatoire. Comme en témoigne le comportement de la perte de validation (Validation Loss), celle-ci entame une augmentation consécutive après la cinquième époque, pour atteindre un pic d'environ 5,28 à la fin de l'entraînement (Tableau III.7).

Ce comportement divergent est révélateur d'une capacité de généralisation extrêmement limitée du modèle vis-à-vis de données invisibles. Cette défaillance est principalement imputable à la grande complexité structurelle de VGG16 (nombre élevé de paramètres) qui, combinée à une tokenisation classique rudimentaire, pousse le réseau à mémoriser le bruit et les spécificités du jeu d'entraînement au lieu d'assimiler des caractéristiques sémantiques réutilisables.

III.7.1 Métriques d'évaluation

L'évaluation est réalisée à l'aide des métriques BLEU (BLEU-1, BLEU-2, BLEU-3 et BLEU-4), METEOR, et CIDEr. Le Tableau III. 8 présente les résultats obtenus

Métrique	VGG 16		GoogleNet	
	Tokenisation de Keras	Tokenisation AraBERT	Tokenisation de Keras	Tokenisation AraBERT
BLEU-1	0.324261	0.205175	0.275218	0.193536
BLEU-2	0.174151	0.103117	0.148187	0.094666
BLEU-3	0.089769	0.047464	0.080747	0.039806
BLEU-4	0.039181	0.019936	0.038640	0.016207
METEOR	0.184816	0.177569	0.170155	0.159489
CIDEr	0.175907	0.143900	0.165274	0.118739

Tableau III. 8:RÉSULTATS DE L'ÉVALUATION

Les scores BLEU (1 à 4)

Les scores BLEU mesurent la précision des n-grammes (groupes de n mots consécutifs).

- **Dégradation rapide** : Pour toutes les configurations, le score s'effondre à mesure que n augmente (par exemple, pour VGG16/Keras : BLEU-1 = 0.324 $\rightarrow$ BLEU-4 = 0.039).

- **Interprétation** : Le modèle réussit à générer les bons mots isolés (BLEU-1 correct), mais il a beaucoup de mal à générer de longues séquences de mots fluides et syntaxiquement correctes (BLEU-4 très bas). Le décodeur LSTM manque probablement de données ou de régularisation pour apprendre la structure grammaticale des phrases de référence.

METEOR

- Les scores METEOR sont relativement stables et serrés (entre **0.159** et **0.184**).
- Comme METEOR prend en compte les synonymes, la lemmatisation et les correspondances exactes, cela indique que même lorsque les mots exacts ne sont pas trouvés (ce qui pénalise le BLEU), le modèle génère des mots qui restent globalement dans le bon champ lexical.

CIDEr (Consensus-based Image Description Evaluation)

Contrairement à BLEU (issu de la traduction automatique) ou METEOR (issu de la génération de texte générale), CIDEr a été spécifiquement créé pour l'évaluation de la description d'images. Il mesure à quel point les descriptions de votre modèle correspondent au consensus humain.

Nous pouvons résumer le comportement global de notre système sur le tableau (Tableau III.7) récupératif suivant :

Métrique	Score obtenu	Diagnostic du modèle
BLEU-1	0.324	Bonne mémoire lexicale : Le modèle sait reconnaître les objets individuellement et choisir le bon vocabulaire.
METEOR	0.184	Bonne flexibilité sémantique : Même s'il se trompe de mot exact, le modèle reste proche du sens et utilise des synonymes valides.
CIDEr	0.175	Alignement sur le consensus humain : Le modèle met correctement l'accent sur les éléments importants de l'image (objets et actions clés) plutôt que sur les mots de liaison.

Tableau III. 9:INTERPRETATION DU RÉSULTATS DE L'ÉVALUATION

III.7.2 Exemples de descriptions générées

Nous avons présente quelques exemples de descriptions générées par différents architecteurs

- **Quelques résultats avec AraBERT**

La Figure III.10 montre bien que le modèle GoogleNet se caractérise par des descriptions académiques concises et précises, centrées sur l'action principale, sans introduire de détails superflus susceptibles d'entraîner des erreurs

GNet: بداي صبي صغير يسبح في ماء
VGG: بداي صبي صغير يسبح في حمام سباح



GNet: بداي رجل يركب دراج على طريق تراب
VGG: بداي رجل يركب دراج على طريق تراب



GNet: بداي كلب يلعب في عشب
VGG: بداي كلب بني يركض على شاطئ



GNet: بداي طفل صغير يلعب على أرجوح
VGG: بداي صبي صغير يرتدي قميص أحمر يلعب على أرجوح



GNet: بداي مجموع من ناس يركب طواف على طوف
VGG: بداي مجموع من أشخاص يرتد ستر نجا يركض في نهر



GNet: بداي كلب أسود أبيض يركض على مضمار
VGG: بداي كلب سلوقي يرتدي كممام يركض على مضمار



Figure III. 10: Quelques résultats avec AraBERT

- Quelques résultats erronés par AraBERT



Figure III. 11: Résultats erroné par AraBERT

Ces images (Figure III.11) apportent une preuve supplémentaire et solide de la supériorité du modèle GNet sur le modèle VGG16 en matière d'analyse visuelle et de correspondance des couleurs.

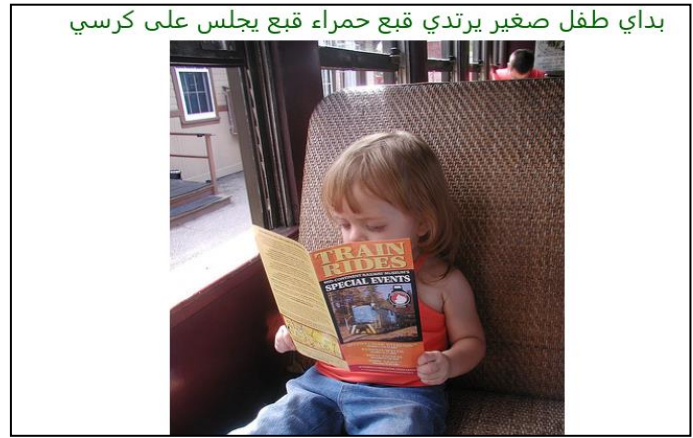


Figure III. 12: Résultats erroné par AraBERT

Les résultats (Figure III.12) démontrent même si le modèle s'est trompé mais est parfaitement compris le contenu visuel

• Quelques résultats avec tokenisation classique



Figure III. 13: Résultats avec tokenisation classique

Les deux modèles (GoogleNet et VGG16) présentent un effondrement total des performances ainsi qu'une grave défaillance sémantique (Figure III.13) ; les deux architectures se sont révélées totalement incapables de générer des phrases cohérentes, produisant à la place des mots aléatoires sans lien avec le contenu de l'image.



Figure III. 14: Comparisons de quelque résultat obtenu par Kerastokanisaton (en haut) et AraBERT (en bas)

L'ambiguïté visuelle dans ces échantillons entraîne une dégradation sévère de la structure linguistique, provoquant un effondrement par répétition excessive chez VGG16 et une génération aléatoire hybride chez GoogleNet (Figure III.14).

III.8 Conclusion

En conclusion, ce chapitre a présenté la réalisation pratique de notre système d'Arabic Image Captioning. Après le prétraitement des données, nous avons utilisé VGG16 et GoogLeNet pour l'extraction des caractéristiques visuelles, puis comparé la tokenisation classique à celle d'AraBERT pour le traitement du texte. Enfin, le modèle a été entraîné et évalué afin d'analyser ses performances dans la génération automatique de descriptions en langue arabe.

Conclusion Générale

Conclusion Générale

Dans ce travail, nous avons étudié et appliqué un système automatique de génération de descriptions d'images en arabe (Arabic Image Captioning), qui est un domaine multidisciplinaire combinant la vision par ordinateur et le traitement du langage naturel. L'objectif principal de ce travail était de concevoir un modèle capable de fournir une description sémantique en arabe à partir du seul contenu d'une image ; et, à travers nos expériences, nous avons analysé comment différents mécanismes d'extraction des caractéristiques visuelles, ainsi que la représentation textuelle, pouvaient influencer sur les performances du système.

Dans un premier temps, nous nous sommes appuyés sur la théorie de l'intelligence artificielle, du machine learning, du deep learning, puis sur le traitement d'images et le traitement du langage naturel. Ces concepts constituent la base des architectures modernes utilisées dans les systèmes de génération de descriptions d'images.

Le deuxième chapitre s'est intéressé plus précisément à une analyse détaillée de la génération de descriptions d'images. Nous avons présenté son développement, son fonctionnement général et ses principaux types : en particulier les modèles Encoder-Decoder, les mécanismes d'attention et les Transformers ; puis nous avons discuté des particularités et des problèmes liés à la production de descriptions en arabe. Vous avez également analysé les métriques clés utilisées pour évaluer la génération : BLEU, METEOR, ROUGE, CIDEr et SPICE.

Ici, nous avons effectué la partie pratique du troisième chapitre du projet en utilisant le principe de transfert d'apprentissage. Après le prétraitement des images et des descriptions arabes du jeu de données Flickr8k, nous avons extrait des caractéristiques visuelles avec les architectures VGG16 et GoogLeNet. Nous avons également comparé les résultats de deux approches de tokenisation : l'approche traditionnelle basée sur le Tokenisation de Keras et la représentation contextuelle d'AraBERT. L'ensemble du système a ensuite été entraîné et évalué à l'aide de plusieurs métriques quantitatives, ainsi qu'en réalisant une évaluation qualitative globale des descriptions générées.

Les résultats expérimentaux démontrent que les choix concernant l'extraction des caractéristiques visuelles et la représentation du texte influencent significativement les descriptions générées. L'étude comparative conclut que GoogLeNet + AraBERT produisent globalement les meilleurs résultats. Cette hybridation offre davantage de capacités de généralisation, atténue le phénomène de

surapprentissage, et présente de manière efficace et une meilleure compréhension à la fois des caractéristiques morphologiques et contextuelles de la langue arabe, ce qui aboutit à des descriptions mathématiques exactes ainsi qu'à des descriptions naturelles.

Bien que certains résultats prometteurs aient été obtenus, de nombreuses possibilités d'amélioration restent envisageables. En particulier, il serait intéressant d'intégrer des mécanismes d'attention (Attention Mechanism), d'entraîner des architectures basées sur les transformeurs ou de nouvelles classes de modèles multimodaux dans des jeux de données arabes plus, ainsi que d'analyser le potentiel complet des grands modèles de langage pour produire des artefacts plus diversifiés et contextualisés.

Pour conclure, ce travail confirme que mettre fin aux techniques modernes de Deep Learning et de Transfer Learning offre une méthode efficace pour développer des systèmes de génération de légendes d'images en langue arabe. Par conséquent, il s'agit d'une contribution à un domaine encore peu exploré et d'une plateforme pour des recherches futures afin d'obtenir une meilleure qualité des descriptions automatiques et des champs d'application plus larges (par exemple : assistance aux personnes malvoyantes, indexation ou recherche d'applications et d'images multimédias, y compris dans des systèmes intelligents).

Références

Références

1. *Comprendre l'IA, le Machine Learning, le Deep Learning et la Gen AI*. **Rosier, Benoit**. mars 2026.
2. *QU'EST-CE QUE L'INTELLIGENCE ARTIFICIELLE ?* **NetApp**. mars 2026.
3. *Vers un déploiement industriel de l'intelligence artificielle*. **Mattioli, Juliette**. avril 2026.
4. *Machine learning*. **Bergmann, Dave**. avril 2026.
5. *Les cinq types de machine learning à connaître*. **China, Chrystal R**. mars 2026.
6. *deep learning*. **Begmman, dave**. mars 2026.
7. *Réseaux de neurones artificiels*. **Quantinuum**. mars 2026.
- 8 Réseaux neuronaux convolutifs(CNN)
9. *Arabic Image Captioning Using Neural Networks*. **Zakaria, EL HABIB DAHO et Wissal, HADDOUCHE**. mars 2026.
10. *What is a Recurrent Neural Network (RNN)?* **Amazon Web Services**. avril 2026.
11. *L'image numérique*. **École nationale supérieure d'architecture de Toulouse**. avril 2026.
12. *Traitment d'image*. **MathWorks**. avril 2026.
13. *What is Medical Image Processing?* **Synopsys**. avril 2026.
- 14 TRAITEMENT NUMÉRIQUE D'IMAGES ET APPLICATIONS
- 15 Traitement automatique du langage naturel
16. **Akridata), Alexander Berkovich (Principal AI/ML Engineer chez**. *Applications Of Image Captioning In AI: Enhancing User Experience*. avril 2026.
17. *Image Captioning Using Transformer Architecture*. **Nanal, W. A.** 2022.
18. *A survey on Arabic Image Captioning Systems Using Deep Learning Models*. **Elnagar, A. Attai and A.** Al Ain, United Arab Emirates : International Conference on Innovations in Information Technology (IIT), 2020.
19. *Generating image captions in Arabic using root-word based recurrent neural networks and deep neural networks*. **JINDAL, Vasu**. s.l. : Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop, 2018.

20. *Automatic Arabic image captioning using RNN-LSTM-based language model and CNN*. **AL-MUZAINI, Huda A., AL-YAHYA, Tasniem N., et BENHIDOUR, Hafida**. 6, s.l. : International Journal of Advanced Computer Science and Applications, 2018, Vol. 09.
21. *Resources and End-to-End Neural Network Models for Arabic Image Captioning*. **ElJundi, O., Dhaybi, M., Mokadam, K., Hajj, H. and Asmar, D.** s.l. : n Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2020).
22. *AraCap: A hybrid deep learning architecture for Arabic Image Captioning*. *Procedia Computer Science*, 2021, vol. 189,. **Afyouni, Imad, Imtinan Azhar, and Ashraf Elnagar**. s.l. : Procedia Computer Science, 2021, vol. 189, p. 382-389.
23. *Traitement automatique du dialecte algérien : construction de ressources et d'outils linguistiques*. **Abdelhakim Benali, Lamia Hadrich Belguith, Mohamed Hedi Maaloul**. 11 avril 2026.
24. **Barbaresi, Adrien**. La complexite linguistique Methode d'analyse. 27 avril 2026.
25. *Le traitement orthographique du mot écrit en arabe*. **Zeghli .H, Bouanani.M**. 23 avril 2026.
26. *A Sociolinguistic Analysis Of Arabic Diglossia Inside The Classroom: Tlemcen Speech Community As A Case In Point*. **Chahrazed, Hamzaoui**. 01 avril 2026.
27. *Mistara*. **Tabet, Maxime**. 26 avril 2026.
28. *Proceedings of the 15th International Conference on Natural Language Generation: Generation Challenges*. **Samira Shaikh, Thiago Ferreira, Amanda Stent**. 2026.
29. *Framing image description as a ranking task: Data, models and evaluation metrics*. **Hodosh, Micah, Peter Young, and Julia Hockenmaier**. s.l. : Journal of Artificial Intelligence Research,, 2013, Vol. 47. p. 853-899.
30. *Microsoft COCO: Common Objects in Context*. **Lin, Tsung-Yi, et al**. Cham : s.n., 2014. pp 740–755.
31. *Deep Visual-Semantic Alignments for Generating Image Descriptions*. **Karpathy, Andrej et Fei-Fei, Li**. 2026. pp. 3128–3137.
32. *BLEU: A Method for Automatic Evaluation of Machine Translation*. **Papineni, Kishore, et al**. 2002. pp. 311–318.
33. *Re-evaluating Automatic Metrics for Image Captioning*. *Ankara, TURKEY : s.n*. **Kilickaya, Mert, et al**. avril 2026.
34. *METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments*. **Banerjee, Satanjeev et Lavie, Alon**. 2005. pp. 65–72.

35. *ROUGE: A Package for Automatic Evaluation of Summaries*. **Lin, Chin-Yew**. s.l. : Association for Computational Linguistics, 2004. pp. 74–81.
36. *CIDeR: Consensus-Based Image Description Evaluation*. **Vedantam, Ramakrishna, Zitnick, C. Lawrence et Parikh, Devi**. 2015. pp. 4566–4575.
37. *SPICE: Semantic Propositional Image Caption Evaluation*. **Anderson, Peter, et al.** s.l. : Springer International Publishing, 2016, Vol. 9909. pp. 382–398.
38. *Arabert: Transformer-based model for arabic language understanding*. **Antoun, W., Baly, F., & Hajj, H.** s.l. : In Proceedings of the 4th workshop on open-source arabic corpora and processing tools, with a shared task on offensive language detection , 2020, May. pp.9-15.