République Algérienne Démocratique et Populaire

Ministère de l'enseignement Supérieur et de la Recherche Scientifique

Université Kasdi Merbah – Ouargla

Faculté des Nouvelles Technologies de l'information et la Communication

Département de l'informatique et Technologies de l'information

Filière : Mathématiques et Informatique

Secteur : Informatique

Spécialité : Administration et Sécurité des Réseaux

Thème

Evaluation de qualité du vidéo sous Spark

Réalisé par : Meriouma Safa et Tati Firdaws

Discuté publiquement : 15 /06/2025

Membres du jury

Dr. Merzougui Naima	Superviseur	UNIV Ouargla
Dr. Ameur Khadidja	Président	UNIV Ouargla
Dr. Khelili Farida Khadidja	Examinateur	UNIV Ouargla

Année Universitaire : 2024/2025

Dédicace

Pour qui j'ai toujours été un soutien, une prière et un refuge, À celui qui est resté éveillé tard pour me rendre heureux et qui s'est sacrifié pour que j'arrive à ce jour

À ma mère bien-aimée," Khadouj", le sommet de ma tête et la source de ma générosité,

Je vous dédie ce travail comme le fruit de vos efforts, de votre patience et de votre soutien inestimable. Ce succès est votre droit, pas le mien.

À ceux qui ont quitté ce monde mais dont l'impact reste dans mon cœur et ma vie,

À celui qui m'a appris la force, l'honnêteté et l'autonomie,

À l'âme pure de mon père"Ali"...

Je prie Allah de le récompenser pour cet exploit et d'avoir pitié de lui.

À mes chers frères,

Mohamed Lamine Saad Eddin, Abd el Moneim, Munira, Rawya, Khawla et Hana, vous êtes mon soutien et mes encouragements dans toutes les étapes de ma vie...

Tout mon amour et ma reconnaissance pour toi.

À ma chère famille,

À vous tous, merci pour votre amour et vos prières, et pour chaque moment de soutien et d'encouragement.

À mes fidèles amis,

Compagnons de chemin, propriétaires de souvenirs inoubliables...

Les amis des dortoirs universitaires sont votre deuxième famille et la compagnie qui allège le fardeau de l'aliénation.

À mon collègue et colocataire"Firdaws",

À vous, qui alliez un grand professionnalisme et une âme bienveillante,

Ô toi qui fus une sœur en exil, et un soutien dans un voyage qui ne fut pas facile,

Merci pour vos rires qui ont illuminé les journées, votre patience et votre soutien inoubliable.

Meriouma Safa

Dédicace

Louange à Allah, Seigneur des mondes, pour Sa lumière et la force qu'Il m'a accordée.
Alhamdulillah pour chaque étape.

À mes parents,

Merci pour votre amour, vos sacrifices et vos prières. Votre soutien m'a portée jusqu'ici.

À mes grands-parents,

Une pensée pleine de gratitude à ma grand-mère paternelle pour sa douceur et sa présence rassurante.
Je remercie également sincèrement mon grand-père maternel, ma grand-mère et mon grand-père paternel – que Dieu leur accorde sa miséricorde.
Ils ont été un pilier dans ma vie et mes études, et leur souvenir reste vivant.

À mes frères et sœurs,

Merci pour votre affection et votre présence rassurante.

À ma famille,

À mes oncles, tantes, cousines et cousins, merci pour votre amour, vos prières et vos encouragements.

À mes amis et camarades de résidence et de classe,

Merci pour votre soutien et votre esprit de partage. Merci pour votre entraide et votre esprit de partage durant cette aventure.

À mon encadrante, Dr. Merzougui Naima,

Merci pour votre patience et vos conseils précieux.

Et à moi-même,
À celle qui a douté, lutté et persévéré. Merci pour ta force et ton courage. Ce travail est le fruit de ta résilience.

À Bien-aimée Palestine,

Puisse Allah accorder Sa protection, Sa justice et Sa paix au peuple palestinien. Que leurs souffrances ne soient jamais oubliées, et que la lumière triomphe de l'injustice.

Ce travail, fruit de savoir et de persévérance, leur est aussi dédié, en hommage à leur résistance et à leur dignité.

Qu'Allah bénisse chacun de vous, vous récompense pour votre soutien, et remplisse vos vies de paix et de lumière.

Tati Firdaws

Remerciement

Avant tout, nous remercions Dieu pour sa grâce et sa lumière qui nous ont guidées à travers les défis et les incertitudes de ce parcours.

Nous exprimons notre profonde reconnaissance à notre encadrante, **Dr. Merzougui Naima**, pour sa patience, sa bienveillance et ses conseils toujours justes. Elle a cru en nous et nous a poussés à donner le meilleur de nous-mêmes.

Nous adressons également nos sincères remerciements à **l'ensemble des enseignants du Département d'Informatique de la Faculté des Nouvelles Technologies de l'Information de l'Université Kasdi Merbah – Ouargla**, pour la qualité de la formation qu'ils nous ont dispensée, leur accompagnement et leur engagement tout au long de notre cursus. Leur savoir et leur disponibilité ont été une source précieuse d'apprentissage et d'inspiration.

Nous remercions également tous ceux qui, de près ou de loin, ont contribué à la réalisation de ce travail. Que ce soit par un mot d'encouragement, un regard bienveillant, ou un simple geste, vous avez tous eu un impact.

Nos familles, nos amis, nos camarades… merci pour vos encouragements, votre écoute, et votre présence. Vous avez été notre moteur quand la fatigue prenait le dessus.

Enfin, nos pensées les plus sincères vont au peuple palestinien, symbole de résilience et de dignité. Que leur courage inspire les cœurs, et que la justice et la paix triomphent. Ce travail leur est dédié avec respect et solidarité.

Ce mémoire est bien plus qu'un travail académique. C'est l'aboutissement d'un chemin semé d'apprentissages, de remises en question et de belles rencontres.

Tati Firdaws~Meriouma Safa

Résumé

Avec l'essor exponentiel du contenu vidéo et la demande croissante pour des médias de haute qualité, l'évaluation fiable et évolutive de la qualité vidéo est devenue un enjeu majeur. Les vidéos numériques, soumises à de multiples étapes de traitement — telles que la compression ou la transmission sur des canaux sujets aux erreurs — subissent souvent des dégradations avant d'atteindre l'utilisateur final. Dans ce contexte, il est crucial de pouvoir estimer de manière précise la qualité perçue par les utilisateurs.

L'évaluation de la qualité vidéo (VQA) permet de répondre à cette problématique, mais les méthodes subjectives traditionnelles, bien qu'efficaces, sont coûteuses, longues à mettre en œuvre et peu adaptées aux grandes échelles. Ce travail propose une méthode d'évaluation objective sans référence, reposant sur l'apprentissage automatique appliqué à des caractéristiques extraites directement des vidéos.

En tirant parti de la puissance du traitement distribué avec Apache Spark et de l'efficacité de l'algorithme **GBTRegressor**, notre système permet de prédire la qualité perçue de manière fiable, rapide et scalable, à partir de bases de données de référence telles que **LIVE-VQC**, **KoNViD-1k** et **YouTube-UGC**.

Les résultats obtenus valident la pertinence de cette approche, démontrant que la combinaison d'outils Big Data et de modèles de régression avancés constitue une solution performante pour relever les défis contemporains de l'évaluation automatique de la qualité vidéo.

Mots-clés : Évaluation de la qualité vidéo, Apache Spark, GBTRegressor, PySpark, Apprentissage automatique, No-Reference VQA.

Abstract

With the exponential growth of video content and the increasing demand for high-quality media, ensuring reliable and scalable video quality assessment has become crucial. Digital videos typically undergo several processing stages — such as compression and transmission over error-prone channels — which may lead to noticeable degradation before reaching the end user.

Video Quality Assessment (VQA) aims to estimate how a user would perceive the quality of a given video. While traditional subjective evaluation methods are accurate, they are time-consuming, costly, and impractical for large-scale applications.

This study investigates an objective, no-reference approach for predicting video quality using machine learning techniques applied to features extracted directly from videos. By leveraging the distributed computing capabilities of Apache Spark and the performance of the **GBTRegressor** algorithm, our system enables efficient prediction of perceived video quality from large-scale datasets, including **LIVE-VQC**, **KoNViD-1k**, and **YouTube-UGC**.

The results demonstrate the effectiveness of combining Spark with ensemble regression models to build a fast, scalable, and reliable framework for modern video quality assessment.

Keywords: Video Quality Assessment, Apache Spark, Machine Learning, PySpark, GBTRegressor, No-Reference, Big Data.

ملخص

مع النمو الهائل لمحتوى الفيديو والطلب المتزايد على الوسائط عالية الجودة، أصبح ضمان تقييم جودة الفيديو بشكل موثوق وقابل للتطوير أمرًا بالغ الأهمية. تخضع مقاطع الفيديو الرقمية عادةً لعدة مراحل معالجة — مثل الضغط والنقل عبر القنوات المعرضة للخطأ — مما قد يؤدي إلى تدهور ملحوظ قبل الوصول إلى المستخدم النهائي.

يهدف تقييم جودة الفيديو (VQA) إلى تقدير كيفية إدراك المستخدم لجودة فيديو معين. على الرغم من أن طرق التقييم الذاتي التقليدية دقيقة، إلا أنها تستغرق وقتًا طويلاً ومكلفة وغير عملية للتطبيقات واسعة النطاق.

تبحث هذه الدراسة في نهج موضوعي بدون مرجع للتنبؤ بجودة الفيديو باستخدام تقنيات التعلم الآلي المطبقة على الميزات المستخرجة مباشرة من مقاطع الفيديو. من خلال الاستفادة من قدرات الحوسبة الموزعة لـ **Apache Spark** وأداء خوارزمية **GBTRegressor**، يتيح نظامنا التنبؤ الفعال بجودة الفيديو المتصورة من مجموعات البيانات واسعة النطاق، بما في ذلك **LIVE-VQC** و**KoNViD-1k** و**YouTube-UGC**.

توضح النتائج فعالية الجمع بين Spark ونماذج الانحدار الجماعي لبناء إطار عمل سريع وقابل للتطوير وموثوق به لتقييم جودة الفيديو الحديث.

الكلمات المفتاحية: تقييم جودة الفيديو، Apache Spark، التعلم الآلي، PySpark، GBTRegressor، عدم وجود مرجع، البيانات الضخمة.

Table des matières

Table des matières 4
Liste des figures 8
Liste des tableaux 9
Introduction générale 10
Chapitre 1 : Qualité de vidéo 12
1. Introduction 12
2. Définition de la vidéo 12
2.1. C'est quoi vidéo ? 12
2.2. Concepts fondamentaux de la vidéo 12
3. Qualité de vidéo 13
3.1. Qu'est-ce que la qualité vidéo ? 13
3.2. Evaluation de la qualité de vidéo 13
3.3. Méthodes d'évaluation de la qualité vidéo 14
3.3.1 Evaluation subjective 14
3.3.2 Evaluation objective 15
a. Évaluation de la qualité des vidéos de référence complètes (FR-VQA) 15
b. Évaluation de la qualité des vidéos de référence réduite (RR-VQA) 16
c. Évaluation de la qualité des vidéos sans référence (NR-VQA) 16
3.3.3 Subjective vs objective 17
4. Méthodes récentes et performantes pour NR-VQA 17
4.1 . V-BLIINDS 18
4.2 . Netflix VMAF (Video Multimethod Assessment Fusion) 18
4.3 . VIDEVAL (VIDeo EVALuation) 18
4.4 . RAPIQUE (ReAL-time Perceptual Image QUality Evaluator) 19
4.5 . CNN-TLVQM (A Hybrid Video Quality Assessment Model Using Deep and Handcrafted Features) 19
5. Les conséquences et Problématiques 20
6. Conclusion 21
Chapitre 2 : Apache Hadoop et Apache Spark 22
1. Introduction 22
2. Big data 22
2.1. Définition de Big Data 22

2.2. Type des données Big Data 22
2.2.1 Données structurées 22
2.2.2 Données non-structurées 23
2.2.3 Données semi-structurées 23
2.3. Caractéristiques de Big Data (les 5 V) 23
2.3.1 Volume 23
2.3.2 Vitesse 23
2.3.3 Variété 24
2.3.4 Véracité 24
2.3.5 Valeur 24
2.4. Pile de technologies Big Data 24
3. Apache Hadoop 25
3.1 . Historique 25
3.2 . Présentation de Hadoop 26
3.3 . Caractéristique d'Hadoop 26
3.4 . Architecture 27
3.4.1 MapReduce 27
3.4.2 HDFS 28
3.4.3 YARN 29
3.4.4 Hadoop commun ou utilitaires communs 29
4. Apache Spark 29
4.1 Historique 29
4.2 Définition d'apache Spark 30
4.3. Composants de l'Apache Spark 30
4.4. Architecture d'Apache Spark 32
4.5. Exemples d'utilisation d'Apache Spark 33
4.6 Caractéristiques de Spark 34
4.7. Avantages de Spark par rapport à Hadoop MapReduce 35
5. Conclusion 36
Chapitre 3 : Modélisation et expérimentation de la qualité vidéo ..... **Error! Bookmark not defined.**
1. Introduction ..... **Error! Bookmark not defined.**
2. Présentation des outils et des données ..... **Error! Bookmark not defined.**
2.1. Environnement de travail ..... **Error! Bookmark not defined.**

2.1.1 Matériel utilisé ..... **Error! Bookmark not defined.**
2.1.2 Outils et technologies utilisés ..... **Error! Bookmark not defined.**
a. Langage de programmation utilisé : Python ..... **Error! Bookmark not defined.**
b. Technologies principales ..... **Error! Bookmark not defined.**
2.1.3 Frameworks et bibliothèques ..... **Error! Bookmark not defined.**
2.1.4 Configuration de l'environnement Spark ..... **Error! Bookmark not defined.**
2.1.5 Justification du choix des outils ..... **Error! Bookmark not defined.**
2.2. Présentation des bases des données vidéo utilisées ..... **Error! Bookmark not defined.**
2.2.1 Présentation de la base de données LIVE ..... **Error! Bookmark not defined.**
2.2.2 Présentation de la base de données KonVid-1k ..... **Error! Bookmark not defined.**
2.2.3 Présentation de la base de données YouTube ..... **Error! Bookmark not defined.**
3. Conception et mise en œuvre du modèle ..... **Error! Bookmark not defined.**
3.1 Choix du modèle : GBTRegresso ..... **Error! Bookmark not defined.**
3.2 Traitement sous Spark ..... **Error! Bookmark not defined.**
3.2.1 Chargement des fichiers Excel (features et MOS) ..... **Error! Bookmark not defined.**
3.2.2 Ajout d'un identifiant unique et fusion des données ..... **Error! Bookmark not defined.**
3.2.3 Conversion des colonnes au type Double ..... **Error! Bookmark not defined.**
3.2.4 Traitement des valeurs manquantes ..... **Error! Bookmark not defined.**
3.2.5 Division du jeu de données ..... **Error! Bookmark not defined.**
3.2.6 Assemblage et normalisation des caractéristiques ..... **Error! Bookmark not defined.**
3.3 Entraînement du modèle ..... **Error! Bookmark not defined.**
3.4 Evaluation du modèle ..... **Error! Bookmark not defined.**
3.4.1 Prédiction sur les ensembles de données ..... **Error! Bookmark not defined.**
3.4.2 Métriques d'évaluation utilisées ..... **Error! Bookmark not defined.**
3.4.3 Métriques statistiques de corrélation ..... **Error! Bookmark not defined.**
3.4.4 Résultats obtenus ..... **Error! Bookmark not defined.**
3.5 Évaluation des performances ..... **Error! Bookmark not defined.**
3.5.1 Coefficient RMSE ..... **Error! Bookmark not defined.**
3.5.2 Coefficient de détermination R^2 ..... **Error! Bookmark not defined.**
3.5.3 Corrélations de Pearson, Spearman et Kendall ..... **Error! Bookmark not defined.**
3.5.4 Visualisation de la performance du modèle ..... **Error! Bookmark not defined.**
3.6 Comparaison avec les études antérieures ..... **Error! Bookmark not defined.**
3.6.1 La base KoNViD-1K ..... **Error! Bookmark not defined.**

3.6.2 La base LIVE ..... **Error! Bookmark not defined.**
3.6.3 La base YouTube-UGC ..... **Error! Bookmark not defined.**
.4 Organigramme général du projet ..... **Error! Bookmark not defined.**
5. Conclusion ..... **Error! Bookmark not defined.**
Conclusion générale 38
Perspectives 38
Références 40

Liste des figures

Figure 2. 1: éléments clés du modèle 5v 24
Figure 2. 2 : Mécanisme d'action du MapReduce [25] 27
Figure 2. 3 : Architecture du HDFS [25] 28
Figure 2. 4 : Composants d'Apache Spark 31
Figure 2. 5 : Architecture d'Apache Spark 32

Figure 3. 1 : installation jupyter notebook ..... **Error! Bookmark not defined.**
Figure 3. 2 : interface jupyter notebook ..... **Error! Bookmark not defined.**
Figure 3. 3 : Connexion à Jupyter et installation pyspark ..... **Error! Bookmark not defined.**
Figure 3. 4 : création d'un Sparksession ..... **Error! Bookmark not defined.**
Figure 3. 5 : Captures d'écran de cadres de certains de ceux présentés lors de l'étude de la base de données live. [45] ..... **Error! Bookmark not defined.**
Figure 3. 6 : Capture d'écran d'une collection de vidéos de base de données KoNViD-1k [47] ..... **Error! Bookmark not defined.**
Figure 3. 7 : Toutes les catégories de l'ensemble de données UGC de YouTube. [48] **Error! Bookmark not defined.**
Figure 3. 8 : Architecture de la régression GBT. [49] ..... **Error! Bookmark not defined.**
Figure 3. 9 : Nuage de points la base LIVE ..... **Error! Bookmark not defined.**
Figure 3. 10 : Nuage de points la base KonVid-1k ..... **Error! Bookmark not defined.**
Figure 3. 11 : Nuage de points la base YouTube-UGC ..... **Error! Bookmark not defined.**

Liste des tableaux

Tableau 1.1 : comparaison entre l'évaluation subjective et objective..... 17

Tableau 3. 1 : Résumé des ensembles de données UGC-VQA d'évaluation comparative. **Error! Bookmark not defined.**
Tableau 3. 2 : Les résultats d'évaluation des performances..... **Error! Bookmark not defined.**
Tableau 3. 3 : Résultats des études précédentes et résultats de notre étude la base KoNViD-1K ..... **Error! Bookmark not defined.**
Tableau 3. 4 : Résultats des études précédentes et résultats de notre étude la base LIVE..**Error! Bookmark not defined.**
Tableau 3. 5 : Résultats des études précédentes et résultats de notre étude la base YouTube-UGC.... **Error! Bookmark not defined.**

Introduction générale

Dans un monde de plus en plus connecté, les contenus vidéo occupent une place centrale dans la communication, le divertissement, l'enseignement et les médias sociaux. L'explosion du volume de vidéos diffusées en ligne, conjuguée aux attentes croissantes des utilisateurs en matière de qualité, pose de nouveaux défis aux acteurs du numérique. Avant d'être visionnées, ces vidéos subissent souvent divers traitements (compression, transmission sur des réseaux sujets aux erreurs), ce qui peut altérer leur qualité perçue.

L'évaluation de la qualité vidéo (VQA) est un domaine clé du traitement multimédia qui se concentre sur l'évaluation de la façon dont les spectateurs perçoivent le contenu vidéo. Il joue un rôle essentiel dans des domaines tels que le streaming, la diffusion et les communications à distance, où la qualité vidéo affecte directement la satisfaction de l'utilisateur. Les approches subjectives, fondées sur l'opinion d'observateurs humains, bien que fiables, sont souvent coûteuses, longues et peu pratiques pour les applications à grande échelle. Cela a conduit au développement d'approches objectives de VQA, visant à estimer la qualité vidéo automatiquement en utilisant des modèles informatiques.

Ces dernières années, l'apprentissage automatique est devenu un outil puissant dans ce domaine. En apprenant des ensembles de données annotées, les modèles peuvent prédire les scores de qualité à partir des caractéristiques extraites. Parmi ces modèles, la régression des arbres à gradient amplifié (GBT) qui se distingue par sa précision et sa capacité à capturer des motifs complexes.

Compte tenu de la grande taille des jeux de données vidéo modernes, les outils de traitement traditionnels sont insuffisants. Pour cette raison, nous utilisons Apache Spark, un framework de calcul distribué conçu pour traiter efficacement des données massives et entraîner des modèles de machine learning à grande échelle grâce à sa bibliothèque MLlib.

Dans ce mémoire, nous construisons une méthode d'évaluation de la qualité de vidéos sans référence en utilisant la régression GBT et implémenté sous Spark. Nous utilisons des ensembles de données de référence comme KoNViD-1k, LIVE et YouTube-UGC, et nous suivons un pipeline structuré : fonctions de chargement et de prétraitement, alignement avec les scores MOS, application de la normalisation, entraînement du modèle et évaluation de sa performance à l'aide de RMSE et des coefficients de corrélation (Pearson, Spearman, Kendall).

Toutes les étapes, du traitement des données à la formation et l'évaluation des modèles, sont exécutées dans Spark pour assurer l'évolutivité et la performance.

La problématique à laquelle nous tentons de répondre est la suivante : **Comment développer une méthode d'évaluation automatique, évolutif et fiable de la qualité vidéo, capable de s'adapter à la diversité des contenus et à la complexité des distorsions sans avoir besoin de référence ?**

Pour répondre à cette question, ce mémoire s'organise en trois chapitres :

- Le chapitre 1 présente les concepts fondamentaux de la qualité vidéo, y compris les types d'évaluations, les mesures et l'importance des approches subjectives et objectives.
- Le chapitre 2 porte sur les outils de mégadonnées, en particulier Apache Spark, et met en évidence son architecture et son rôle dans le traitement des données vidéo à grande échelle pour l'apprentissage machine.
- Le chapitre3 présente la mise en œuvre pratique, détaillant la méthodologie, l'intégration des ensembles de données, les étapes de prétraitement, la formation au modèle GBT et l'analyse du rendement.

Chapitre 1 : Qualité de vidéo

1. Introduction

La vidéo est un média clé à l'ère numérique, et l'évaluation de sa qualité est cruciale pour assurer la satisfaction des utilisateurs. Avec la croissance du Big Data, les jeux de données vidéo à grande échelle sont essentiels pour la formation et le test des modèles de prédiction. Ce chapitre présente les bases de l'évaluation de la qualité des vidéos, et discute brièvement des exemples d'études d'évaluation antérieures dans ce domaine.

2. Définition de la vidéo

2.1. C'est quoi vidéo ?

Selon Le Nouveau Petit Robert de la Langue Française 2008 « est une technologie qui permet d'enregistrer des images et des sons sur des supports magnétiques ou numériques et de les retransmettre sur un écran ». [1]

Plus techniquement, La vidéo est une séquence d'images fixes montrées rapidement pour créer du mouvement. Il s'appuie sur des éléments clés tels que les cadres, la résolution et les méthodes de codage couleur telles que les signaux composants et composites.

2.2. Concepts fondamentaux de la vidéo

• **Le trame :** est une seule image dans une séquence de photos. En général, une seconde d'une vidéo est composée de 24 ou 30 images par seconde également connu sous le nom de FPS (Frame Per Second). Le cadre est une combinaison de l'image et du temps de l'image lorsqu'elle est exposée à la vue. Un extrait des cadres d'une rangée fait l'animation. Les cadres sont mis en vue une fois qu'ils sont reconsidérés par les prochaines images. [2]

• **les pixels :** un pixel est le plus petit élément d'une image, et une vidéo se compose de centaines de milliers à des millions de pixels, qui se combinent pour former sa résolution. Les pixels peuvent être composés de 3 composants (RVB : Rouge, vert, bleu) ou 4 composants (CMJN : cyan, magenta, jaune, noir).

• **Résolution vidéo :** Désigne le nombre de pixels horizontaux et verticaux dans une vidéo, par exemple 1080p (1920 1080 pixels).

• **Aspect ratio :** Définit le rapport largeur-hauteur d'une vidéo, tel que 4 :3 (TV standard) et 16 :9 (écran large). Elle ne détermine pas la résolution mais affecte la forme globale de l'image. En conclusion, les pixels jouent un rôle fondamental dans la définition de la résolution, de la qualité et du ratio d'aspect d'une vidéo. [3]

• **Débit binaire :** En termes simples, Débit binaire fait référence à la quantité de données vidéo transférées dans un laps de temps particulier. Cela affecte principalement la taille et la qualité de votre vidéo. L'unité de mesure est bit/s, ou bps, ce qui signifie bits par seconde. Pour les vidéos en streaming, le débit est généralement décrit en mégabits par seconde (Mbps), et pour l'audio en kilobits par seconde (kbps).[4]

3. Qualité de vidéo

3.1. Qu'est-ce que la qualité vidéo ?

En termes simples, la qualité vidéo est la caractéristique d'une vidéo qui détermine à quel point la vidéo transmise au spectateur est claire, nette et détaillée. Une vidéo de meilleure qualité fournit plus d'informations, une meilleure clarté et une expérience de visionnement plus fluide sans distorsion ni artefacts.

3.2. Evaluation de la qualité de vidéo

L'évaluation de la qualité vidéo est essentielle pour s'assurer que le contenu visuel répond aux attentes des utilisateurs.

Cette étape permet de déterminer dans quelle mesure une vidéo conserve sa clarté, ses couleurs et son expérience globale après le traitement ou la transmission.

Cependant, au-delà des mesures techniques, ce qui importe vraiment est la façon dont les utilisateurs perçoivent la qualité, c'est ce qu'on appelle l'évaluation de la qualité vidéo perceptuelle.

3.3. Méthodes d'évaluation de la qualité vidéo

Avec l'essor des plateformes de streaming, de la vidéoconférence, et des contenus numériques en général, évaluer la qualité d'une vidéo est devenu un enjeu crucial. La qualité perçue d'une vidéo ne dépend pas uniquement de sa résolution, mais aussi de nombreux facteurs comme la compression, le débit binaire, les artefacts visuels, et le rendu des mouvements.

Pour cela, différentes méthodes d'évaluation de la qualité vidéo ont été développées, que l'on peut regrouper en deux grandes catégories :

3.3.1 Evaluation subjective

L'évaluation subjective de la qualité vidéo implique des observateurs humains qui regardent les séquences vidéo dans un environnement contrôlé et évaluent leur qualité perçue. Ces évaluations reflètent la perception humaine réelle et sont donc considérées comme très fiables. Les cotes individuelles sont agrégées pour produire des scores d'opinion moyens (MOS) ou des scores d'opinion moyens différentiels (DMOS), qui servent de base à l'élaboration et à la validation de modèles automatisés. Bien qu'elles soient précises et perspicaces, les évaluations subjectives sont coûteuses, longues et difficiles à mettre à l'échelle en raison de la nécessité d'une participation humaine et de paramètres contrôlés. [5]

L'évaluation subjective implique généralement des méthodes structurées pour recueillir les opinions des téléspectateurs. Une approche courante est l'utilisation de :

Questionnaires :

Les téléspectateurs répondent à des questions structurées portant sur des aspects tels que la netteté, la précision des couleurs, le flou et la fréquence d'image. Leurs réponses permettent d'évaluer la satisfaction globale quant à la qualité de la vidéo.

Séances de visionnage contrôlé :

Les participants regardent différents échantillons vidéo dans un environnement surveillé, et leurs impressions sont enregistrées. Ces séances peuvent être individuelles ou en groupe, et parfois guider les spectateurs à se concentrer sur des facteurs de qualité spécifiques.

S'appuyant sur les points de vue des téléspectateurs, il capture l'expérience réelle de l'utilisateur et met en évidence les principaux facteurs de satisfaction, bien qu'il soit limité par les coûts élevés, le temps requis et les préjugés personnels potentiels.

3.3.2 Evaluation objective

L'évaluation de la qualité vidéo (VQA) est un domaine crucial qui vise à mesurer la qualité de l'expérience de visionnement, jouant un rôle clé dans l'amélioration des services vidéo et assurant la satisfaction des utilisateurs. Les méthodes VQA sont généralement classées en trois types principaux, selon la disponibilité de la vidéo originale (de référence) :

a. Évaluation de la qualité des vidéos de référence complètes (FR-VQA)

Cette méthode est basée sur une comparaison complète de la vidéo traitée (ou déformée) avec la vidéo d'origine (de référence). Cet accès complet à la vidéo originale permet une analyse détaillée et complète, qui donne des résultats très précis.

Mécanisme : Ces méthodes utilisent des informations structurelles (comme le SSIM), des informations sur les mouvements et des modèles du système visuel humain (HVS) pour simuler la façon dont l'œil humain perçoit la qualité vidéo. Les mesures de qualité sont extraites en comparant les caractéristiques des deux vidéos.

Méthodes notables :

Indice de similarité structurelle (SSIM) [6] : mesure la similarité entre les structures visuelles des deux vidéos.

Indice de similarité structurelle multi-échelle (MC-SSIM) [7] : Une extension du SSIM qui prend en compte les effets des différentes fréquences.

MOVIE Index : Basé sur un modèle HVS et simule la perception des distorsions.

Vidéo Multimethod Assessment Fusion (VMAF) [8] : Repose sur l'apprentissage automatique et combine plusieurs métriques.

Spatio-Temporal VMAF (ST-VMAF) : Une version améliorée de VMAF qui tient compte des effets temporels.

b. Évaluation de la qualité des vidéos de référence réduite (RR-VQA)

Cette méthode est un compromis entre FR-VQA et NR-VQA. RR-VQA nécessite une quantité limitée de données provenant de la vidéo de référence, ce qui le rend plus efficace que FR-VQA et plus précis que NR-VQA.

Mécanisme : Ces méthodes utilisent des fonctionnalités à faible bande passante de la vidéo de référence, telles que VQM, ST-RRED et SpEED-QA. RR-VQA s'appuie sur des modèles de transformation par ondelettes et de statistiques de scène naturelle (NSS).

Méthodes classiques notables :

Vidéo QualityMetric (VQM) : Une mesure de qualité vidéo basée sur un modèle HVS.

Détection d'erreurs spatio-temporelles de référence réduite (ST-RRED) [9]: Une métrique de référence réduite qui tient compte des effets temporels.

Évaluation de la qualité spatio-temporelle efficace et efficiente orientée vers la distorsion (SpEED-QA) : Une mesure rapide et efficace pour évaluer la qualité vidéo. [10]

c. Évaluation de la qualité des vidéos sans référence (NR-VQA)

Cette méthode permet d'évaluer la qualité vidéo sans avoir besoin de la vidéo originale, ce qui en fait un outil idéal pour les applications réelles où la vidéo originale n'est généralement pas disponible (comme le streaming en direct ou la vidéo stockée sur des appareils).

Mécanisme :

Il y a deux approches principales dans NR-VQA :

- Distorsion-spécifique NR-VQA : Se concentre sur des types spécifiques de distorsions (tels que la compression ou les erreurs de transmission).
- Distortion-Agnostic NR-VQA : utilise les propriétés statistiques et d'apprentissage profond de la vidéo pour évaluer la qualité en général.

Méthodes classiques notables :

Visual-CORTEX Inspired No-Reference Image Assessment (V-CORNIA) : Une métrique basée sur un modèle perceptuel.

Évaluation de la qualité des images vidéo sans regard/sans référence (VBLIINDS) : Une mesure basée sur les modèles NSS.

Vidéo Intégrité intrinsèque et évaluation de la distorsion Oracle (VIIDEO) : Une métrique basée sur l'apprentissage automatique.

Évaluation vidéo sans référence basée sur le codage spatio-angulaire (SACONVA) : Une métrique basée sur les réseaux neuronaux.

Métrique spatio-temporelle guidée par l'information (ST-IGM) [11] : Une métrique basée sur l'information de mouvement et l'intégration temporelle. [12]

3.3.3 Subjective vs objective

Tableau 1.1 : comparaison entre l'évaluation subjective et objective.

Caractéristique	**Evaluation subjective**	**Evaluation objective**
Définition	Les spectateurs évaluent la qualité vidéo	Les algorithmes estiment automatiquement la qualité
Exemple	Les téléspectateurs attribuent des notes (p. ex., MOS) après avoir regardé	Des mesures comme le PSNR et le SSIM calculent la qualité
Précision	Très précise (reflète la perception réelle)	Varie selon le modèle (nécessite un étalonnage)
Temps et coûts	Élevé (configuration du test nécessaire et participants)	Faible (automatisé, évolutif)
Application	Benchmarking datasets, validating models	Real-time monitoring, streaming optimization
Factures d'influence	Humeur du spectateur, arrière-plan, dispositif d'affichage	Type de métrique, robustesse de l'algorithme

4. Méthodes récentes et performantes pour NR-VQA

Plusieurs chercheurs se sont intéressés à l'évaluation de la qualité vidéo sans référence (NR-VAQ), parmi les méthodes les plus performantes on peut citer les études suivantes :

4.1. V-BLIINDS

V-BLIINDS est un modèle d'évaluation objective sans référence (no-référence) qui se base sur les statistiques du mouvement et de l'apparence extraites des vidéos. Contrairement aux méthodes qui nécessitent une version originale de la vidéo pour comparaison, V-BLIINDS peut estimer la qualité à partir de la vidéo seule. Il a été entraîné et validé sur des bases de données comme LIVE, et a démontré une performance remarquable en se rapprochant des scores obtenus par les évaluateurs humains. Ce modèle repose sur des techniques d'apprentissage automatique et sur l'analyse des caractéristiques temporelles et spatiales des vidéos. [13]

4.2. Netflix VMAF (Video Multimethod Assessment Fusion)

VMAF est un modèle développé par **Netflix** pour évaluer la qualité perceptuelle de la vidéo telle qu'elle est vue par l'utilisateur final. Il combine plusieurs métriques objectives (comme VIF, DLM, etc.) à l'aide de techniques de machine learning pour produire un score global qui correspond mieux aux impressions humaines.

Netflix utilise VMAF pour ajuster ses algorithmes de compression et de streaming tout en maintenant une bonne expérience utilisateur. [9]

4.3. VIDEVAL (VIDeo EVALuation)

Cette méthode repose sur une combinaison judicieuse de caractéristiques visuelles extraites à la fois d'images individuelles et d'informations temporelles issues de la séquence vidéo. VIDEVAL s'appuie sur un ensemble de descripteurs préexistants issus de l'évaluation de la qualité d'image (NR-IQA), tels que BRISQUE, NIQE, V-BLIINDS, ainsi que sur des mesures statistiques complémentaires liées au bruit, à la texture, au mouvement ou encore à la dynamique temporelle. L'originalité de l'approche réside dans une étape de sélection automatique des caractéristiques les plus discriminantes, grâce à un algorithme de type Recursive Feature Elimination.

Une fois les caractéristiques pertinentes sélectionnées, VIDEVAL utilise un modèle de régression supervisée (Support Vector Regressor (SVR) pour prédire les scores de qualité subjective (MOS) à partir des vecteurs de caractéristiques extraites. [14]

Grâce à sa rapidité d'exécution, sa simplicité d'implémentation et sa robustesse face à la diversité des contenus, VIDEVAL constitue une méthode de choix pour des applications en contexte réel.

4.4. RAPIQUE (ReAL-time Perceptual Image QUality Evaluator)

Est une méthode NR-VQA récente et performante, développée pour prédire la qualité vidéo de manière rapide et efficace. Elle se distingue par sa structure hybride, combinant des caractéristiques perceptuelles traditionnelles et des représentations profondes extraites via des réseaux pré-entraînés. [15]

L'idée centrale de RAPIQUE est de tirer parti à la fois :

- De caractéristiques perceptuelles classiques, telles que les statistiques de type natural scene statistics (NSS) — utilisées dans BRISQUE, NIQE, etc. ;
- Et de représentations issues de réseaux de type Deep CNN, plus précisément à partir des couches d'un réseau ResNet-50 pré-entraîné.

Ces deux types de descripteurs sont concaténés pour former un vecteur de caractéristiques global, qui est ensuite utilisé comme entrée d'un modèle de régression supervisée (SVR : Support Vector Regressor) pour prédire les scores de qualité perceptuelle.

Conçue pour une exécution quasi en temps réel, RAPIQUE est bien adaptée aux systèmes nécessitant un compromis entre précision et vitesse, comme le streaming, les systèmes embarqués ou les plateformes de vidéo UGC. Elle affiche d'excellentes performances sur les bases LIVE-VQC, KoNViD-1k et YouTube-UGC, avec des résultats comparables voire supérieurs à certaines approches plus lourdes de type deep learning pur.

4.5. CNN-TLVQM (A Hybrid Video Quality Assessment Model Using Deep and Handcrafted Features)

La méthode CNN-TLVQM intègre des caractéristiques visuelles classiques et des représentations profondes issues de réseaux de neurones convolutifs, elle vise à améliorer la performance de prédiction de la qualité vidéo tout en conservant une certaine légèreté computationnelle.

Le modèle combine :

- Des caractéristiques handcrafted extraites à deux niveaux, incluant des informations sur la luminance, le contraste, la texture, le mouvement et la dynamique temporelle.
- Et des features profondes extraites automatiquement via un réseau ResNet-50 pré-entraîné, appliqué sur des images clés de la vidéo.

Les descripteurs extraits sont ensuite fusionnés en un vecteur de caractéristiques global, utilisé comme entrée d'un modèle de régression supervisée pour prédire le score de qualité perceptuelle.

CNN-TLVQM a été évalué sur plusieurs bases de données de référence et a démontré de meilleures performances grâce à l'enrichissement apporté par les descripteurs issus du deep learning.

Cette approche représente un compromis intéressant entre précision et coût de calcul, et s'avère bien adaptée aux systèmes d'évaluation de qualité vidéo qui nécessitent à la fois robustesse et efficacité. [16]

5. Les conséquences et Problématiques

Principaux défis en matière d'évaluation de la qualité vidéo :

➢ **Impact du contenu vidéo :**

La diversité des contenus vidéo a une incidence importante sur l'évaluation de la qualité. Les différents genres, la complexité des mouvements et les variations de scène peuvent influencer les évaluations subjectives et objectives, ce qui rend difficile le développement de modèles d'évaluation universels.

➢ **Effets de la mémoire :**

La perception humaine de la qualité vidéo est influencée par des facteurs temporels, comme l'ordre des scènes et l'exposition antérieure au contenu. La prise en compte de ces effets sur la mémoire est essentielle pour des évaluations subjectives précises, mais reste complexe à modéliser dans des mesures objectives.

➢ **Efficacité de calcul :**

L'équilibre entre la précision des méthodes d'AQV et les exigences du calcul est un défi persistant. Les modèles de haute précision nécessitent souvent des ressources informatiques considérables, ce qui peut ne pas être réalisable pour les applications en temps réel.

- **Prédiction de la qualité vidéo personnalisée :**

Les différences individuelles dans les préférences des spectateurs et les sensibilités perceptuelles nécessitent des prédictions de qualité personnalisées. La mise au point de modèles pouvant s'adapter à des expériences personnelles de visionnement est un domaine de recherche en cours.

- **Évaluation des formats vidéo émergents :**

Les nouveaux formats et technologies vidéo, comme la gamme dynamique élevée (RDD), les vidéos à 360 degrés et la réalité virtuelle (RV), présentent des défis uniques pour l'évaluation de la qualité. Les méthodes VQA existantes ne sont peut-être pas directement applicables, ce qui nécessite la mise au point de nouvelles techniques d'évaluation adaptées à ces formats.

Il est essentiel de relever ces défis pour faire progresser les méthodologies VQA et garantir des évaluations précises et fiables dans divers contenus vidéo et technologies émergentes. [10]

6. Conclusion

L'évaluation de la qualité vidéo est essentielle pour assurer la satisfaction des utilisateurs et maintenir les performances des systèmes multimédias modernes. À mesure que le volume, la variété et la vitesse des données vidéo augmentent, les outils traditionnels ne parviennent pas à gérer une telle complexité. Cela a ouvert la voie aux technologies de mégadonnées pour prendre les devants en proposant des frameworks avancés comme Apache Spark qui sont conçus pour traiter et analyser des ensembles de données vidéo à grande échelle de manière efficace et intelligente. Dans le chapitre suivant on va discuter sur l'Apache Spark.

Chapitre 2 : Apache Hadoop et Apache Spark

1. Introduction

Avec la croissance massive des volumes de données générés quotidiennement, notamment dans les domaines du multimédia, du streaming et des réseaux sociaux, les architectures traditionnelles de traitement se révèlent rapidement insuffisantes. Le besoin de traiter, analyser et extraire des connaissances à partir de données volumineuses et hétérogènes a donné naissance à des frameworks distribués capables de gérer efficacement ce que l'on appelle le Big Data.

Dans ce contexte, Apache Hadoop et Apache Spark sont rapidement imposé comme des outils incontournables dans les applications modernes de traitement massif de données, notamment dans l'apprentissage automatique distribué.

Ce chapitre vise à présenter ces deux frameworks majeurs en exposing leurs principes fondamentaux, leurs architectures respectives, ainsi que leurs forces et limites comparées dans le contexte d'applications à grande échelle.

2. Big data

2.1. Définition de Big Data

Le terme « big data » désigne les différentes techniques utilisées pour analyser ou extraire de la valeur et des informations utiles à partir de données trop volumineuses ou trop complexes. Ces dernières elles ont été collectées à partir de sources diverses (capteurs, appareils, etc.) sont très difficiles à traiter avec les technologies de traitement classiques. [17]

2.2. Type des données Big Data

2.2.1 Données structurées

Les données structurées font référence aux données avec un format et une longueur définie, faciles à stocker et à analyser, et hautement organisées. Cela signifie que les données sont organisées dans une structure reconnaissable afin de pouvoir répondre aux requêtes pour récupérer des informations à des fins d'organisation.

Une base de données relationnelle comme le langage de requête structuré (SQL) représente un bon exemple pour les données structurés, il contient des nombres organisés, des dates, des groupes de mots et des nombres appelés chaînes / texte.

En raison de la structure transparente de la base de données, elle peut être recherchée avec des algorithmes de recherche simples et directs qui peuvent être par type de données dans le contenu réel. [18]

2.2.2 Données non-structurées

Les données non structurées sont des informations, sous de nombreuses formes différentes, qui ne correspondent pas aux modèles de données conventionnels et qui ne conviennent donc généralement pas à une base de données relationnelle traditionnelle. Cela rend le traitement et l'analyse des données non structurées très difficiles et longues.

D'après Feldman et Sanger, Les données non structurées n'ont pas de structure spécifique. Les données non structurées comprennent généralement des images / objets bitmap, du texte, des e-mails et d'autres types de données qui ne font pas partie de la base de données. [19]

2.2.3 Données semi-structurées

Les données semi-structurées sont des données irrégulières qui peuvent être incomplètes et avoir une structure qui change rapidement ou de manière imprévisible mais qui ne se conforme pas à un schéma fixe ou explicite.

Hanig, Schierle et Trabold ont précisé que le modèle de données semi-structurées permet aux informations provenant de plusieurs sources, avec des propriétés liées mais différentes, d'être regroupées en un tout, par exemple, des fichiers de courrier électronique, XML et Doc. [20]

2.3. Caractéristiques de Big Data (les 5 V)

Les caractéristiques sous-jacentes des mégadonnées ou Big Data comprennent :[21]

2.3.1 Volume

Le volume se réfère à la quantité de données générées quotidiennement par des entreprises ou des personnes.

2.3.2 Vitesse

La vitesse est une caractéristique importante du Big Data, elle signifie la vitesse de génération des données. Par exemple une grande vitesse de génération des résultats des données aboutit à une large quantité des données en peu de temps.

2.3.3 Variété

Les données à traiter sont sous forme structurés ou non structurés : data bases, textes, données de capteurs, sons, vidéos, de parcours, fichiers journaux etc.

2.3.4 Véracité

La véracité fait référence à la précision et la fiabilité des données. Pour extraire la valeur des données, les données doivent être nettoyées pour supprimer le bruit, ce nettoyage des données est important pour que les données incorrectes et défectueuses puissent être filtrées. [22]

2.3.5 Valeur

La valeur des données se réfère à l'utilité des données en fonction du but prévu. L'objectif final de tout le système d'analyse des mégadonnées consiste à extraire cette valeur des données. La valeur des données est également liée à la véracité ou à l'exactitude des données. Pour certaines applications, la valeur dépend également de la vitesse à laquelle nous pouvons traiter les données.

Figure 2. 1: éléments clés du modèle 5v

2.4. Pile de technologies Big Data

La pile technologique Big Data fait référence à un ensemble d'outils et de technologies utilisés pour collecter, stocker, traiter et analyser de grandes quantités de données. Ces technologies permettent de gérer différentes étapes du cycle de vie du Big Data, de l'acquisition des données à l'analyse des données. Que nous citons à d'exemple :

• Apache Spark :

C'est un framework open source de traitement en temps réel, offrant des capacités de calcul en mémoire.

Il est particulièrement performant pour les tâches de traitement de données en temps réel, comme dans le cas de l'analyse de flux de données ou des applications de recommandation. Spark est utilisé par des entreprises comme Uber pour l'analyse en temps réel des données.

• Apache Hadoop :

C'est un framework open source permettant le traitement distribué des données massives, idéal pour le traitement par lots. Il permet de stocker et traiter de grandes quantités de données sur plusieurs machines. Par exemple, il est utilisé dans des entreprises comme Yahoo pour le traitement de données en grande échelle.

• Apache Flink :

Un autre framework open source pour le traitement de données en temps réel, très performant pour les calculs d'événements complexes et le traitement de flux en temps réel.

• Apache Kafka :

Une plateforme de streaming open source utilisée pour la gestion des flux de données en temps réel. Elle permet de stocker, de traiter et d'envoyer des messages entre systèmes en temps réel, comme dans le cas du traitement des logs de serveur ou des données IoT.

3. Apache Hadoop

3.1. Historique

Hadoop a été lancé en 2002 par Doug Cutting et Mike Cafarella, initialement comme un projet appelé Apache Nutch pour créer un moteur de recherche capable d'indexer un milliard de pages. Cependant, les défis d'évolutivité et les coûts élevés liés au stockage et au traitement de grandes quantités de données ont conduit à la séparation de l'informatique distribuée de Nutch et à la création de Hadoop en 2006.

Doug Cutting, inspiré par les travaux de Google sur GFS et MapReduce, a rejoint Yahoo et a pris Hadoop avec lui. Hadoop a été nommé d'après l'éléphant jaune en peluche de son fils. Il est rapidement devenu une solution pour gérer efficacement de vastes ensembles de données à moindre coût. En 2009, Hadoop a prouvé son efficacité dans le traitement de données à grande échelle, ce qui a incité Cutting à rejoindre Cloudera pour promouvoir davantage l'adoption de Hadoop. Le projet a été transféré à l'Apache Software Foundation en 2008 et a atteint des jalons significatifs avec les versions 1.0 en décembre 2011, 2.0.6 en août 2013 et 3.0 en décembre 2017, marquant son évolution en tant que technologie fondamentale à grande échelle. [23]

3.2. Présentation de Hadoop

Hadoop est un Framework open source d'Apache écrit en Java et est utilisé pour stocker des processus et analyser des données de très gros volume. Il est largement utilisé dans le domaine du Big Data pour répondre aux défis liés à la gestion de données à grande échelle. Hadoop est basé sur les composants principaux : Hadoop Distributed File System (HDFS), un système de planification des traitements (YARN), un Framework de traitement (MapReduce).[23]

3.3. Caractéristique d'Hadoop

Hadoop se caractérise par les caractéristiques suivantes : [24]

• **Robustesse :** Hadoop est conçu pour gérer efficacement les pannes matérielles. Lorsqu'une machine tombe en panne, ses tâches sont automatiquement réaffectées aux autres nœuds disponibles dans le cluster. De plus, les blocs de données sont répliqués sur plusieurs nœuds pour maintenir la disponibilité des données même en cas de défaillance du nœud.

• **Rentabilité :** En répartissant le traitement des données sur de nombreux nœuds d'un cluster, Hadoop optimise l'utilisation des ressources. Cette approche réduit le besoin de matériel coûteux et performant en permettant l'utilisation d'un grand nombre de serveurs à faible coût.

• **Prise en charge de plusieurs langues :** Hadoop prend en charge différents langages de programmation pour le traitement et le stockage des données, ce qui améliore sa flexibilité et le rend adapté à un large éventail de tâches d'analyse de données.

• **Haute disponibilité :** Grâce à la réplication des données, Hadoop assure une accessibilité continue des données. Même en cas de défaillance d'une machine ou d'un composant matériel, le système peut récupérer les données à partir d'un autre réplica, tout en maintenant un accès ininterrompu.

3.4. Architecture

L'architecture Hadoop se compose principalement de 4 composants :

3.4.1 MapReduce

MapReduce est un modèle puissant de traitement des données conçu pour le traitement d'ensembles de données à grande échelle. Il fonctionne en deux étapes principales : la fonction Map, qui traite les données d'entrée et les transforme en paires clé-valeur, et la fonction Reduce, qui regroupe ces paires par clé et effectue des opérations telles que le tri ou l'agrégation pour générer le résultat final.

La tâche Map commence avec le RecordReader, qui convertit les entrées brutes en paires clé-valeur. Ceux-ci sont ensuite traités par la fonction Map, éventuellement passés à travers un combinateur pour l'optimisation locale et dirigés par un partitionneur vers le réducteur approprié. Dans la tâche Réduire, le système effectue un tri aléatoire pour organiser les données intermédiaires, après quoi la fonction Réduire agrège les résultats. Enfin, l'OutputFormat écrit les données traitées dans le stockage.

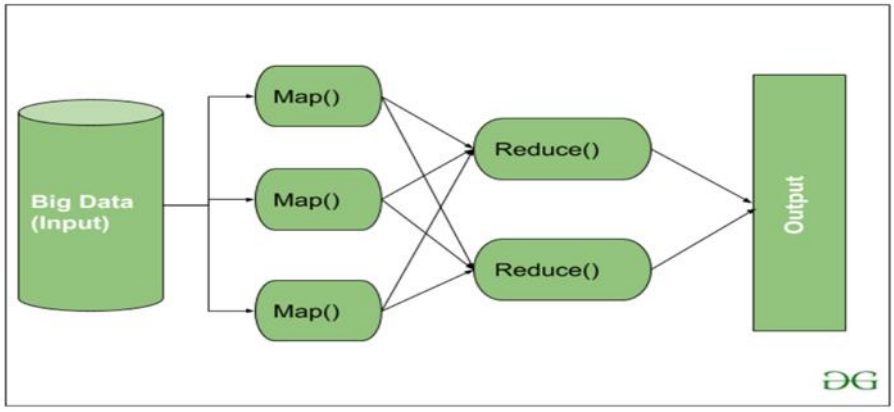

Figure 2. 2 : Mécanisme d'action du MapReduce [25]

3.4.2 HDFS

HDFS est un système distribué conçu pour stocker et traiter de très grands volumes de données de manière fiable sur des clusters Hadoop. Il prend en charge des données structurées, semi-structurées et non structurées, et permet un accès rapide et efficace grâce à une architecture distribuée. Étant un logiciel open source, il est gratuit et n'implique aucun coût de licence. HDFS garantit la disponibilité des données en les répliquant sur plusieurs nœuds connectés via un réseau à haut débit. Chaque nœud du cluster est une machine indépendante dotée de son propre processeur, ce qui permet d'atteindre des vitesses de lecture/écriture très élevées.

Il se distingue par :

- Le stockage et traitement distribués sur un cluster de serveurs, favorisant le parallélisme.
- Une interface en ligne de commande pour interagir avec le système (création, chargement, suppression de fichiers, etc.).
- Une architecture maître-esclave avec un NameNode central et plusieurs DataNodes assurant le stockage réel des données.
- Un accès continu et transparent aux données, indépendamment de leur emplacement physique sur le cluster.

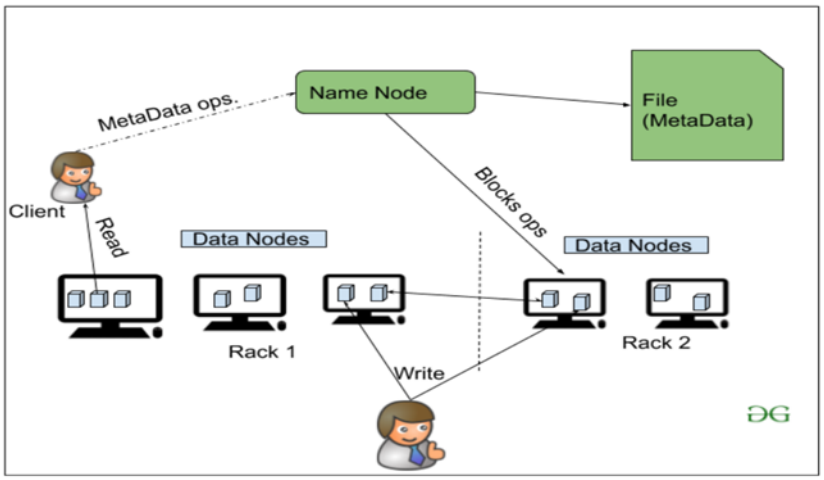

Figure 2. 3 : Architecture du HDFS [25]

3.4.3 YARN

YARN est un framework qui supporte l'exécution de MapReduce en gérant deux fonctions principales : la planification des tâches et la gestion des ressources. Le planificateur de tâches décompose les tâches volumineuses en sous-tâches plus petites et les répartit sur divers nœuds du cluster Hadoop pour améliorer l'efficacité du traitement. Il gère également les priorités des tâches, les dépendances et le calendrier d'exécution. Entre-temps, le gestionnaire des ressources est responsable de l'allocation efficace et de la supervision des ressources disponibles au sein du groupe.

YARN se caractérise par :

- Multilocation
- Grande évolutivité
- Utilisation efficace des clusters
- Compatibilité avec les systèmes et applications existants.

3.4.4 Hadoop commun ou utilitaires communs

Hadoop commun ou utilitaires communs ne sont rien d'autre que notre bibliothèque java et les fichiers java ou nous pouvons dire les scripts java dont nous avons besoin pour tous les autres composants présents dans un cluster Hadoop. Ces utilitaires sont utilisés par HDFS, YARN et MapReduce pour exécuter le cluster. Hadoop Common vérifie que la défaillance matérielle dans un cluster Hadoop est courante, il doit donc être résolu automatiquement dans le logiciel par Hadoop Framework.

4. Apache Spark

4.1 Historique

Apache Spark a été lancé en 2009 à l'Université de Californie à Berkeley sous forme de projet de recherche, et publié en 2010 dans un article intitulé "Spark: Cluster Computing with Working Sets". À l'époque, Hadoop MapReduce était le moteur dominant pour le traitement parallèle des données. Cependant, MapReduce avait des limites pour les applications nécessitant plusieurs passes sur les données, comme les algorithmes d'apprentissage automatique.

Pour résoudre ce problème, l'équipe de Spark a conçu une API basée sur la programmation fonctionnelle et un moteur permettant un partage efficace des données en mémoire. La première version de Spark, lancée en 2010, supportait uniquement les applications par lots. En 2011, Spark a introduit Shark, un moteur SQL intégré permettant des requêtes interactives.

Le projet a évolué en 2013 avec l'intégration de nouvelles bibliothèques comme MLlib, Spark Streaming et GraphX. La même année, Spark a été offert à la fondation Apache pour devenir un projet open source. En 2014, Spark 1.0 a été lancé, suivi de Spark 2.0 en 2016. Ces versions ont introduit des optimisations majeures, des API de traitement de données structurées et des fonctionnalités avancées comme les DataFrames et le traitement en streaming structuré.[26]

4.2 Définition d'apache Spark

Apache Spark est un Framework de traitement distribué open source pour les charges de travail Big Data. Il utilise la mémoire cache et la fonction d'exécution de requête optimisée pour interroger rapidement des données de toute taille. Spark est donc un moteur rapide et polyvalent pour le traitement de données à grande échelle. Moteur Spark Core utilise l'ensemble de Données Distribué Résilient, ou RDD (Resilient Distributed Data), comme type de données de base. [19]

Apache Spark peut effectuer rapidement des tâches de traitement sur de très grands ensembles de données, et peut également distribuer des tâches de traitement de données sur plusieurs ordinateurs, seul ou avec d'autres outils informatiques distribués. Ces deux qualités sont essentielles aux Concepts « Big Data » et l'apprentissage automatique, qui nécessitent la mobilisation d'une puissance de calcul massive pour parcourir les grands magasins de données.

Spark peut être implémenté avec différents langages de programmation fonctionnels Scala (qui s'exécute dans Java VM), java, python et R. [19]

4.3. Composants de l'Apache Spark

Parce que le moteur de base de Spark est à la fois rapide et polyvalent, il alimente de multiples composants de haut niveau spécialisés pour diverses charges de travail, tels que SQL ou l'apprentissage automatique. Ces composants sont conçus pour inter opérer étroitement, vous

permettant de les combiner comme les bibliothèques dans un projet logiciel. Spark Core : contient les fonctionnalités de base de Spark, y compris les composants pour la planification des tâches, gestion de la mémoire, la reprise après incident, interaction avec les systèmes de stockage, et plus. Spark Core est également l'API qui définit les ensembles de données distribués élastiques (RDD), qui sont les principales abstractions de programmation de Spark. RDD représentent une collection d'objets répartis sur plusieurs nœuds de calcul qui peuvent être manipulés en parallèle. Spark Core offre de nombreuses API pour la construction et la manipulation de ces collections. Autre que Spark core API, il ya des bibliothèques supplémentaires qui font partie de l'écosystème de Spark et fournissent des capacités supplémentaires dans l'analyse des Big Data. Ces bibliothèques sont: Spark streaming, Spark SQL, Spark MLlib, Spark GraphX. [27]

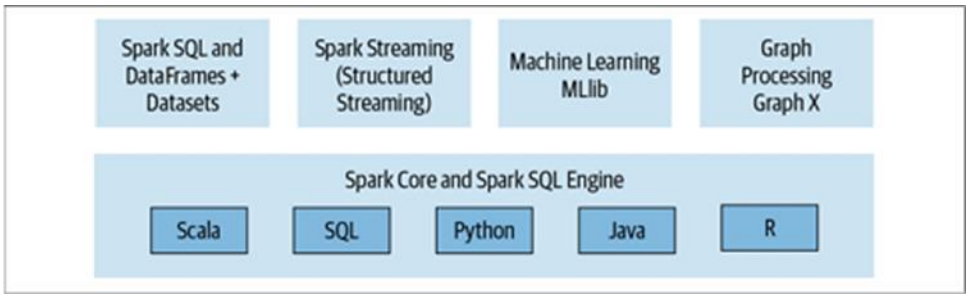

Figure 2. 4 : Composants d'Apache Spark

- **Spark core :**

Les fonctions de base de Spark incluent la planification des tâches, la gestion de la mémoire, la récupération après une panne, l'interaction avec les systèmes de stockage, etc. Démontrez le concept de base de l'ensemble de données distribuées élastiques de Spark, qui peut traiter les données via l'API RDD de Spark.

- **Spark Streaming :**

Spark streaming est utilisé pour le traitement des données en temps réel. Bien que Spark soit conçu comme une infrastructure de traitement par lots, il peut être utilisé pour traiter des données en continu à l'aide du concept de micro-lots fourni par Spark Streaming. Toutes ces bibliothèques fonctionnent sur les RDD en tant qu'abstraction de données, de sorte qu'elles peuvent toutes être combinées ensemble dans une seule application de manière transparente pour offrir des avantages significatifs aux utilisateurs. [28]

- **Spark MLlib :**

MLlib est une bibliothèque d'apprentissage automatique évolutive, incluse dans Apache Spark, qui contient des fonctionnalités d'apprentissage automatique. MLlib contient des algorithmes d'apprentissage automatique courants (classification, régression, clustering et filtrage collaboratif) et offre des fonctionnalités telles que l'évaluation de modèle et l'importation de données. [29]

- **Spark SQL :**

Spark SQL est un module de traitement de données construit via l'interface SQL. Les données sont accessibles à partir de diverses sources de données (telles que JSON, Parquet, tables Hive et JDBC). Spark SQL fournit les fonctions suivantes : manipuler des données via SQL, HQL2 ou une API d'ensemble de données de haut niveau personnalisée, et mélanger SQL avec la manipulation de données fournie par l'API (Application Programming Interface) Spark RDD. [30]

- **Spark GraphX :**

GraphX est un composant pour la manipulation de graphes et les calculs parallèles aux graphes. [31]

4.4. Architecture d'Apache Spark

Il s'agit d'une architecture maître-esclave composée d'un pilote, qui s'exécute en tant que nœud maître, et de nombreux exécuteurs qui s'exécutent en tant que nœuds de travail dans le cluster. [32]

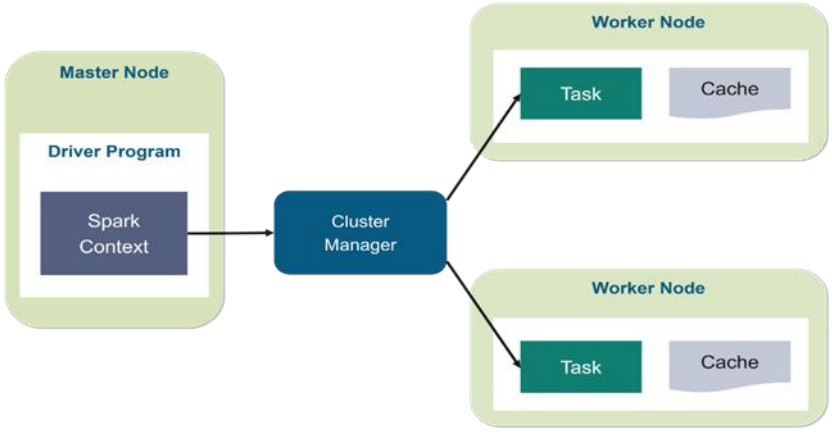

Figure 2. 5 : Architecture d'Apache Spark

- **Le Driver :**

Le driver ou le pilote est le programme ou le processus chargé de coordonner l'exécution de l'application Spark. Il exécute la fonction principale et crée le SparkContext, qui se connecte au gestionnaire de cluster.

- **Les exécuteurs :**

Les exécuteurs sont des processus de travail responsables de l'exécution des tâches dans les applications Spark. Ils sont lancés sur les nœuds Worker et communiquent avec le programme pilote et le gestionnaire de cluster. Les exécuteurs exécutent des tâches simultanément et stockent les données en mémoire ou sur disque pour la mise en cache et le stockage intermédiaire.

- **The cluster manager :**

Le gestionnaire de cluster est responsable de l'allocation des ressources et de la gestion du cluster sur lequel l'application Spark s'exécute. Spark prend en charge divers gestionnaires de clusters comme Apache Mesos, Hadoop YARN et standalone cluster manager.

- **SparkContext :**

SparkContext est le point d'entrée de toute fonctionnalité Spark. Il représente la connexion à un cluster Spark et peut être utilisé pour créer des RDD (Resilient Distributed Datasets), des accumulateurs et des variables de diffusion. SparkContext coordonne également l'exécution des tâches.

- **Task :**

C'est une tâche. Elle est la plus petite unité de travail dans Spark, représentant une unité de calcul qui peut être effectuée sur une seule partition de données. Le programme pilote divise la tâche Spark en tâches et les attribue aux nœuds d'exécution pour exécution.

4.5. Exemples d'utilisation d'Apache Spark

✓ **Traitement du flux en temps réel**

Avec Spark Streaming, il peut traiter des données en direct (par exemple, les journaux, les transactions) en temps réel pour des applications telles que la détection de fraude ou la surveillance des médias sociaux.

✓ Apprentissage automatique (ML)

Spark MLlib prend en charge les tâches de machine learning à grande échelle.Par exemple : Prédire la qualité d'image ou de vidéo à l'aide de modèles de régression formés sur des ensembles de données riches en fonctionnalités.

✓ Traitement des graphiques

Spark GraphX est utilisé pour analyser des graphiques et des réseaux tels que les moteurs de recommandation ou le suivi du comportement des utilisateurs.

4.6 Caractéristiques de Spark

Parmi les fonctionnalités les plus importantes d'Apache Spark, nous pouvons mentionner ce qui suit : [33]

- **Vitesse :** Spark est connu pour sa grande vitesse – jusqu'à 100 fois plus rapide qu'Hadoop lorsqu'il s'exécute en mémoire – grâce à moins d'opérations de lecture/écriture sur disque.
- **Utilisation de RDD (ensembles de données distribués résilients) :** Spark traite les données à l'aide de RDD, qui sont des collections de données en lecture seule et partitionnées. Les résultats intermédiaires peuvent être mis en cache pour éviter l'accès répété au disque.
- **Tolérance aux pannes :** Au lieu de répliquer les données entre les nœuds, Spark suit la lignée des données (les étapes de transformation entre les RDD), ce qui lui permet de recalculer les données perdues si nécessaire.
- **Capacités de traitement polyvalentes :** Spark prend en charge le traitement des données par lots et en continu, les requêtes SQL, l'apprentissage automatique et l'analyse graphique, ce qui en fait un cadre très flexible.
- **Souplesse et soutien linguistique :** Spark prend en charge plusieurs langages de programmation (Java, Scala, Python et R) et peut s'exécuter en tant qu'application autonome, sur Amazon EC2 ou localement.

- **Facilité d'utilisation et de configuration :** La vitesse de Spark, sa configuration simple et les bibliothèques disponibles (par exemple, les API Java) en ont fait un choix approprié pour des expériences dans le prototype KG-OLAP décrit.

4.7. Avantages de Spark par rapport à Hadoop MapReduce

✓ La rapidité Spark est une plateforme de calcul open source plus rapide que Hadoop, grâce à l'utilisation de la mémoire distribuée pour optimiser les traitements itératifs et les requêtes interactives. Il peut être de 10 à 100 fois plus rapide que Hadoop MapReduce en réduisant les lectures et écritures sur le disque.

✓ Traitement itératif Contrairement à Hadoop MapReduce, qui stocke les données sur disque entre chaque opération, Spark utilise les RDD (Resilient Distributed Datasets) pour garder les données en mémoire, évitant ainsi les barrières de synchronisation et réduisant les opérations d'E/S. Cela permet d'accélérer le traitement des données en minimisant les temps d'attente liés au disque.

✓ Requêtes interactives Dans le traitement des algorithmes d'extraction de données interactives, Spark charge les données une seule fois et les stocke en mémoire distribuée, tandis qu'Hadoop charge les mêmes données plusieurs fois à partir du disque pour chaque requête, ce qui entraîne des traitements supplémentaires.

✓ Plus riche Spark propose des API concises pour Scala, Java et Python, avec plusieurs fonctions (actions et transformations), contrairement à Hadoop qui ne dispose que de Map et Reduce en Java.

✓ Facilité d'utilisation Spark permet de développer rapidement des applications en Java, Scala ou Python grâce à des instructions simples et lisibles.

✓ Généralité Spark couvre un large éventail de charges de travail, y compris le traitement en temps réel, les algorithmes itératifs, les requêtes interactives et le streaming, tout en étant facile et peu coûteux à combiner dans des pipelines d'analyse de données.

✓ Streaming en temps réel Contrairement à Hadoop, qui ne traite que des données stockées, Spark permet de traiter des flux de données en temps réel grâce à Spark Streaming.

✓ Traitement graphique Spark peut être utilisé pour le traitement graphique, mappant les relations entre différentes entités dans les données, comme les personnes et les objets.

✓ Algorithmes d'apprentissage Spark inclut une bibliothèque d'apprentissage, MLlib, avec des algorithmes pour la classification, la régression, le regroupement, et le filtrage collaboratif, tandis qu'Hadoop nécessite l'intégration de Mahout.

✓ Gestion rapide des données structurées Spark SQL permet de travailler avec des données structurées et d'effectuer des requêtes sur des ensembles de données distribuées via des API intégrées en Python, Scala et Java.

5. Conclusion

Dans ce chapitre, nous avons présenté les deux principaux frameworks de traitement distribué de l'écosystème Big Data : Apache Hadoop et Apache Spark. Apache Spark s'impose aujourd'hui comme une alternative de référence. Grâce à son moteur de calcul en mémoire, à son modèle de programmation plus expressif et à ses bibliothèques intégrées (telles que MLlib pour l'apprentissage automatique), Spark permet de développer des pipelines de traitement de données puissants, efficaces et adaptables à différents cas d'usage.

Forts de ces atouts, nous avons opté pour Apache Spark afin de construire notre méthode d'évaluation de qualité de vidéo, dont la conception et l'implémentation seront détaillées dans le chapitre suivant.

Conclusion générale

Ce mémoire a exploré le développement d'une méthode d'évaluation de la qualité vidéo sans référence (NR-VQA), basé sur l'algorithme GBTRegressor et exécuté dans un environnement distribué à l'aide d'Apache Spark. L'évaluation de la qualité vidéo reste une tâche difficile qui combine des aspects perceptifs et techniques, nécessitant des outils efficaces capables de traiter des données à grande échelle tout en fournissant des prévisions précises. Dans un premier temps, nous avons introduit les fondements théoriques de la qualité vidéo, y compris les méthodes d'évaluation subjective et objective, et l'importance d'une évaluation fiable dans les systèmes multimédias modernes. Dans le deuxième chapitre, nous avons présenté les outils du Big Data, en mettant l'accent sur Apache Spark, en décrivant son architecture, ses fonctionnalités et sa capacité à traiter des données à grande échelle. Nous avons également discuté de sa bibliothèque d'apprentissage automatique (MLlib), qui a joué un rôle central dans notre implémentation. Dans le troisième chapitre, nous avons détaillé notre méthodologie, y compris la préparation des ensembles de données, les étapes de prétraitement (alignement, normalisation), la formation du modèle à l'aide de GBTRegressor et l'évaluation à l'aide de mesures de rendement telles que RMSE, Coefficients de corrélation de Pearson, Spearman et Kendall. Nous avons utilisé trois larges ensembles de données variés : KoNViD-1k, LIVE VQC et YouTube-UGC. Nos résultats ont démontré l'efficacité de la combinaison des capacités de traitement distribué de Spark avec la régression GBT pour une prédiction de qualité vidéo évolutive et précise. Le modèle a réussi à saisir les relations complexes entre les caractéristiques extraites et la qualité perçue, validant ainsi le potentiel de l'apprentissage automatique pour remplacer les évaluations humaines coûteuses et subjectives.

Perspectives

Pour affiner l'évaluation de la qualité vidéo, plusieurs pistes prometteuses s'offrent à nous. D'abord, il est essentiel d'**optimiser la sélection des caractéristiques pertinentes** et **d'ajuster les paramètres de nos modèles pour des prédictions plus précises**. Ensuite, l'intégration de **techniques d'apprentissage profond**, comme les réseaux de neurones convolutifs (CNN) ou les transformeurs adaptés à la vidéo, représente une voie très intéressante pour saisir les informations visuelles complexes.

De plus, pour que nos résultats se rapprochent davantage des perceptions humaines de la qualité, nous pourrions adopter des **approches hybrides** combinant modèles statistiques et perceptifs. Enfin, une application concrète et très utile serait de déployer ce modèle dans des **contextes en temps réel** (comme le streaming, la surveillance ou la vidéoconférence), en exploitant pleinement les capacités de traitement distribué et en flux d'Apache Spark. Cela répondrait aux exigences croissantes de l'industrie multimédia.

Références

[1] Le Robert, *Le Petit Robert de la langue française*. Paris, France: Dictionnaires Le Robert, 2008.

[2] DARVIDEO, “Frame,” [Online]. Disponible sur :https://darvideo.tv/dictionary/frame/.

[3] VDOCIPHER, “Video Resolution,” [Online]. Disponible sur : https://www.vdocipher.com/blog/video-resolution/.

[4] MOVAVI, “What is Video Bitrate,” [Online]. Disponible sur : https://www.movavi.com/learning-portal/what-is-video-bitrate.html.

[5] Q. Zheng *et al.*, “Video quality assessment: A comprehensive survey,” *arXiv preprint* arXiv:2412.04508, 2024.

[6] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004, doi: 10.1109/TIP.2003.819861.

[7] A. K. Moorthy and A. C. Bovik, “Efficient video quality assessment along temporal trajectories,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 11, pp. 1653–1658, Nov. 2010.

[8] Z. Li, A. Aaron, I. Katsavounidis, *et al.*, “Toward a practical perceptual video quality metric,” *Netflix Tech Blog*, 2016. [Online]. Disponible sur :https://medium.com/netflix-techblog/toward-a-practical-perceptual-video-quality-metric-653f208b9652.

[9] R. Soundararajan and A. C. Bovik, “Video quality assessment by reduced reference spatio-temporal entropic differencing,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 4, pp. 684–694, Apr. 2013.

[10] D. Li, T. Jiang, and M. Jiang, “Recent advances and challenges in video quality assessment,” *ZTE Commun.*, vol. 17, no. 1, Mar. 2019.

[11] Y. Zhu, Y. F. Wang, and Y. Shuai, “Blind video quality assessment based on spatio-temporal internal generative mechanism,” in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, 2017, pp. 305–309.

[12] Y. Derfoufi, *Programmation en langage Python*. 2019.

[13] A. Mittal, M. A. Saad, and A. C. Bovik, “A completely blind video integrity oracle,” *IEEE Trans. Image Process.*, vol. 23, no. 3, pp. 1352–1365, Mar. 2014, doi: 10.1109/TIP.2013.2299802.

[14] V. Hosu, F. Hahn, and D. Saupe, “VIDEVAL: A No-Reference Video Quality Evaluation Metric,” *IEEE Trans. Image Process.*, vol. 30, pp. 5085–5100, 2021.

[15] Z. Tu, C.-J. Chen, Y. Wang, N. Birkbeck, B. Adsumilli, and A. C. Bovik, “RAPIQUE: Rapid and Accurate Video Quality Prediction via Content Understanding,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2021, pp. 14028–14037.

[16] J. Krause *et al.*, “CNN-TLVQM: A Hybrid Video Quality Assessment Model Using Deep and Handcrafted Features,” in *Proc. 13th Int. Conf. Quality of Multimedia Experience (QoMEX)*, 2021.

[17] S. E. Henouda, *Une approche par l’apprentissage approfondi pour l’extraction des connaissances dans les big data*, Ph.D. dissertation, Univ. Kasdi Merbah Ouargla, 2024.

[18] A. C. Eberendu, “Unstructured data: An overview of the data of big data,” *Int. J. Comput. Trends Technol.*, vol. 3, no. 1, pp. 46–50, 2016.

[19] R. Feldman and J. Sanger, *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. Cambridge, UK: Cambridge Univ. Press, 2007.

[20] C. Hänig, M. Schierle, and D. Trabold, “Comparison of structured vs. unstructured data for industrial quality analysis,” in *Proc. World Congr. Eng. Comput. Sci.*, 2010.

[21] S. ROBERT, *Big Data Essentials*. New York, NY, USA: Data Insights Press, 2018.

[22] Y. Demchenko, C. de Laat, and P. Membrey, “Defining architecture components of the Big Data Ecosystem,” in *Proc. 2014 Int. Conf. Collaboration Technol. Syst. (CTS)*, IEEE, 2014.

[23] T. White, *Hadoop: The Definitive Guide*, 4th ed. Sebastopol, CA, USA: O’Reilly Media, 2015.

[24] A. Sultana, “Using Hadoop to Support Big Data Analysis: Design and Performance Characteristics,” *Culminating Projects in Information Assurance*, no. 27, 2015.

[25] GEEKSFORGEEKS, “Hadoop Architecture,” [Online]. Disponible sur :https://www.geeksforgeeks.org/hadoop-architecture/.

[26] M. Chambers and M. Zaharia, *Spark: The Definitive Guide: Big Data Processing Made Simple*. Sebastopol, CA, USA: O’Reilly Media, 2018.

[27] S. Mezzoudj, *Approches de classification des images à grande échelle sur des architectures massivement parallèles*, Ph.D. dissertation, Univ. de Batna 2, 2020. [Online]. Disponible sur :http://eprints.univ-batna2.dz/1898/

[28] M. Zaharia *et al*., “Discretized streams: an efficient and fault-tolerant model for stream processing on large clusters,” in *Proc. USENIX Symp.*, 2012.

[29] I. Mistrík *et al.*, *Software Architecture for Big Data and the Cloud*. San Francisco, CA, USA: Morgan Kaufmann, 2017.

[30] M. Armbrust *et al*., “Spark SQL: Relational data processing in Spark,” in *Proc. 2015 ACM SIGMOD Int. Conf. Management of Data*, 2015.

[31] R. S. Xin *et al*., “GraphX: A resilient distributed graph system on Spark,” in *Proc. 1st Int. Workshop on Graph Data Management Experiences and Systems*, 2013.

[32] H. Karau, A. Konwinski, P. Wendell, and M. Zaharia, *Learning Spark*. Sebastopol, CA, USA: O’Reilly Media, 2015.

[33] J. Klar, *Implementing query operations for knowledge graph OLAP in Apache Spark*, M.S. thesis, Johannes Kepler Univ. Linz, 2023.

[34] Y. Derfoufi, *Programmation en langage Python*. 2019.

[35] W. McKinney, *Python for Data Analysis*. Sebastopol, CA, USA: O'Reilly Media, 2022.

[36] Zaharia, M., Xin, R. S., Wendell, P., Das, T., Armbrust, M., Dave, A., ... & Stoica, I. (2016). *Apache Spark: A unified engine for big data processing*. Communications of the ACM, 59(11), 56–65.

[37] M. Zaharia *et al.*, "Apache Spark: A unified engine for big data processing," *Commun. ACM*, vol. 59, no. 11, pp. 56–65, 2016.

[38] DATABRICKS, "What is PySpark?" [Online]. Disponible sur :https://www.databricks.com/glossary/pyspark.

[39] Stack Overflow, "Using monotonically_increasing_id() for assigning row number to PySpark DataFrame," 2018. [Online]. Disponible sur :https://stackoverflow.com.

[40] Apache Spark, *pyspark.sql.types.DoubleType — PySpark 3.5.0 documentation*, The Apache Software Foundation, 2024. [Online]. Disponible sur :https://spark.apache.org/docs/3.5.0/api/python/reference/pyspark.sql/types.html#pyspark.sql.types.DoubleType.

[41] Apache Spark, *pyspark.ml.feature.VectorAssembler — PySpark 3.5.0 documentation*, The Apache Software Foundation, 2024. [Online]. Disponible sur :https://spark.apache.org/docs/3.5.0/api/python/reference/pyspark.ml.feature.html#pyspark.ml.feature.VectorAssembler.

[42] A. Salihu and S. M. Shamsuddin, "The choice of scaling technique matters for classification performance," *arXiv preprint arXiv:2212.12343*, 2022.

[43] P. Virtanen *et al.*, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[44] D. Varga, "No-reference image quality assessment with global statistical features," *J. Imaging*, vol. 7, no. 2, p. 29, 2021.

[45] Z. Sinno and A. C. Bovik, “Large-scale study of perceptual video quality,” IEEE Trans. Image Process., vol. 28, no. 2, pp. 612–627, 2018.

[46] Hosu, V., Hahn, F., Jenadeleh, M., Lin, H., Men, H., Szirányi, T., Li, S., & Saupe, D. (2017). The Konstanz Natural Video Database (KoNViD-1k). In *2017 Ninth International Conference on Quality of Multimedia Experience (QoMEX)* (pp. 1–6). IEEE.

[47] konvid1k - Dropbox

[48] Y. Wang, S. Inguva, and B. Adsumilli, “YouTube UGC dataset for video compression research,” in Proc. IEEE Int. Workshop on Multimedia Signal Processing (MMSP), Kuala Lumpur, Malaysia, Sept. 2019, pp.1-5.

[49] SpringerOpen, "Environmental Sciences Europe," 2024. [Online]. Available: https://enveurope.springeropen.com/articles/10.1186/s12302-024-00897-7/figures/5.

[50] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," *IEEE Transactions on Image Processing*, vol. 21, no. 12, pp. 4695–4708, Dec. 2012, doi: 10.1109/TIP.2012.2214050.

[51] J. Korhonen, "Two-level approach for no-reference consumer video quality assessment," IEEE Transactions on Image Processing, vol. 28, no. 12, pp. 5923–5938, 2019.

[52] M. A. Saad, A. C. Bovik, and C. Charrier, “Blind prediction of natural video quality,” IEEE Trans. Image Process., vol. 23, no. 3, pp. 1352–1365, 2014.

[53] D. Kundu, D. Ghadiyaram, A. C. Bovik, and B. L. Evans, “Noreference quality assessment of tone-mapped HDR pictures,” IEEE Trans. Image Process., vol. 26, no. 6, pp. 2957–2971, 2017.

[54] D. Ghadiyaram and A. C. Bovik, “Perceptual quality prediction onauthentically distorted images using a bag of features approach,” J.Vision, vol. 17, no. 1, pp. 32–32, 2017