أنظمة تحليل بيانات الزمن الحقيقي الضخمة الناتجة عن انترنت الأشياء

طالبة الماجستير : آلاء محمد كي السباعي كلية الهندسة المعلوماتية – جامعة البعث

إشراف د. محسن حسين و د. وسيم رمضان

الملخص

مع تطور عالم الاتصالات وانتشار الشبكات عالية السرعة والهواتف والأجهزة الذكية، لمع مصطلح إنترنت الأشياء Internet of Things (IoT) في الأفق ، وما ينتجه من بيانات، وازدادت أهيمته خصوصاً بعد أن زادت كمية البيانات بشكل ملحوظ وباتت ترقى لأن نطلق عليها مسمى البيانات الضخمة. انتشرت الدراسات التي تسعى للاستفادة من بيانات انترنت الأشياء لأغراض التطوير في شتى المجالات، وبالتالي ظهرت الحاجة إلى أطر عمل تجمع بين تقنيات البيانات الضخمة وخدمات إنترنت الأشياء.

من تحديات التعامل مع بيانات انترنت الأشياء أنها تتميز بالتدفق السريع والحاجة إلى جمعها ومعالجتها بسرعة في الوقت الحقيقي (real time)، فكان لابد من دراسة أهم أطر عمل البيانات الضخمة المناسبة لمعالجتها.

تمت مقارنة نظامي هادوب وسبارك في هذا البحث لاختيار الأكثر ملائمة لبيانات انترنت الأشياء والذي يوفر متطلباتها من التدفق والتحليل السريع وسهولة الاستخدام. بيّنت النتائج تفوّق اطار سبارك خصوصاً في قدرته على إتاحة إمكانية تحليل بيانات الزمن الحقيقي وسرعة المعالجة والكفاءة في استخدام الذاكرة.

الكلمات المفتاحية: إنترنت الأشياء IoT – البيانات الضخمة – هادوب – سبارك

Analysis Systems of Real-time Big data generated by the Internet of Things

Abstract

With the development of the world of communications and the spread of high-speed networks, phones and smart devices, the term Internet of Things (IoT) appeared on the horizon, and the data it produces, and its importance increased, especially after the amount of data increased significantly and became worthy of being called Big Data. Studies that seek to benefit from IoT data for development purposes have spread in various fields, and thus the need for frameworks that combine big data technologies and IoT services has emerged.

One of the challenges of dealing with Internet of Things data is that it is characterized by rapid flow and the need to collect and process it quickly in real time, so it was necessary to study the most important frameworks for big data appropriate to process it.

Hadoop and Spark systems were compared in this research to choose the most suitable for IoT data, which provides its requirements of flow, rapid analysis and ease of use. The superiority of the Spark framework was found, especially in its ability to provide the possibility of analyzing real-time data

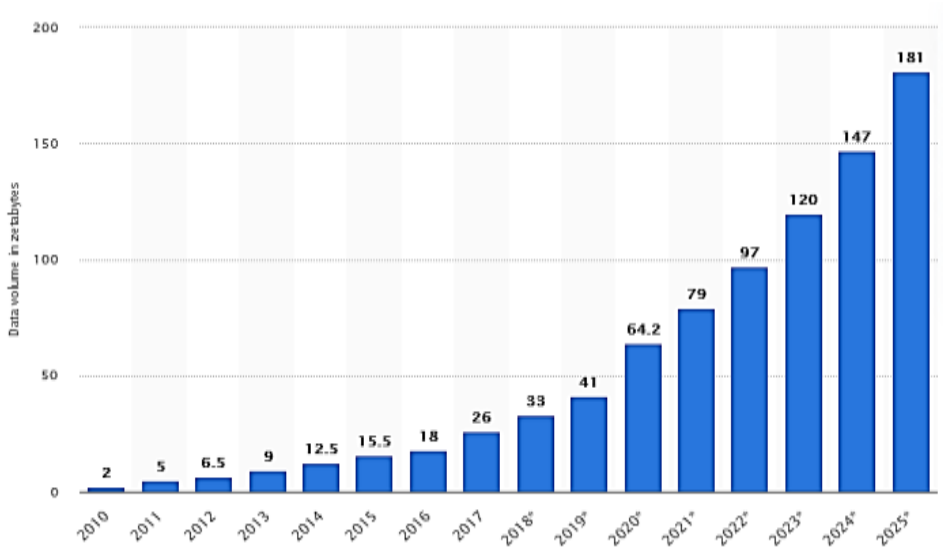
Keywords: Internet of things IOT – big data – Hadoop – spark

1 مقدمة

"البيانات (النفط الجديد)" هي واحدة من أكثر العبارات شيوعًا في وقتنا الحالي[12]، فهي تسلط الضوء على أهمية البيانات، في وقت أصبحت فيه المعلومات الرقمية موجودة وراسخة في جميع جوانب حياتنا ومجتمعنا. كما يبدو أنه لا يمكن إيقاف النمو في إنتاج المعلومات، حيث تقوم الشركات دائمًا بجمع البيانات مثل بيانات المبيعات، وبيانات العمليات، والبيانات المالية، وبيانات الموارد البشرية، وبيانات المستهلك وغيرها. يتم ذلك عادةً بهدف تحليلها واكتساب رؤى محددة والاستفادة منها في اتخاذ القرار.

تعتبر مواقع التواصل الاجتماعي أحد أهم مصادر البيانات، ففي كل يوم يتم إنشاء 500 مليون تغريدة جديدة، و294 مليار رسالة بريد إلكتروني، و4 ملايين غيغابايت من بيانات فيسبوك، و65 مليار رسالة واتس أب، و720000 ساعة من المحتوى الجديد المضاف يوميًا على يوتيوب[13]. كل ذلك ولم يتم ذكر انتشار منتجات إنترنت الأشياء التي أدت إلى إغراق العالم الرقمي بكميات من البيانات لم تكن متاحة من قبل.

تزداد كمية البيانات التي يتم إنشاؤها والتقاطها ونسخها واستهلاكها عالميًا بسرعة، فقد وصلت إلى 64.2 زيتابايت في عام 2020 [3]، و بلغ حجم البيانات التي تم إنشاؤها وتكرارها مستوى جديداً، وكان النمو أعلى من المتوقع سابقًا بسبب زيادة الطلب الذي تسببت فيه جائحة COVID-19، حيث عمل المزيد من الناس وتعلموا من المنزل واستخدموا خيارات الترفيه المنزلي في كثير من الأحيان. ومن المتوقع أن يستمر نمو وازدياد حجم هذه البيانات لتصل إلى أكثر من 180 زيتابايت على مدى السنوات الأربعة المقبلة حتى عام 2025 (انظر الشكل 1).

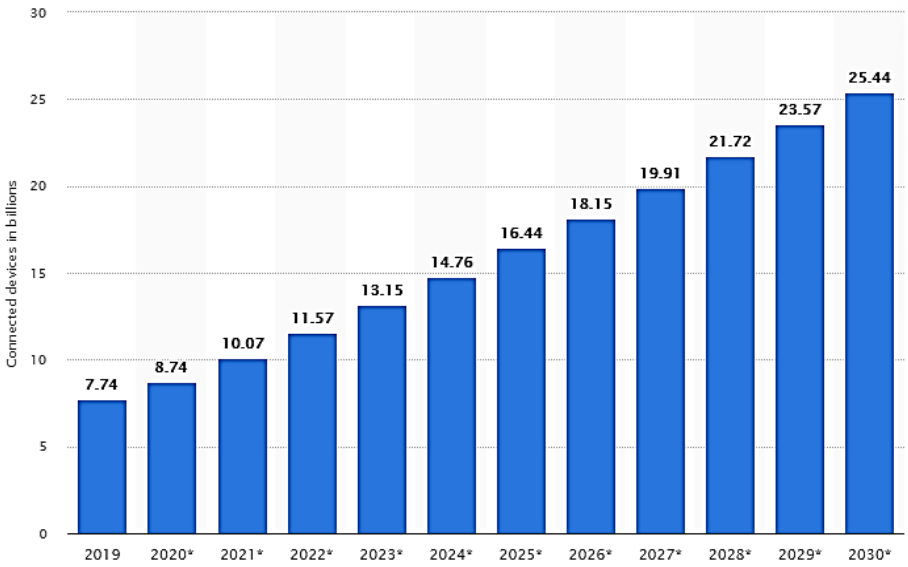


الشكل 1. إنتاج البيانات من عام 2010 و حتى 2025 مقدراً بالزيتابايت حسب موقع www.statista.com

وبما أن البيانات هي معلومات في صورة خام كان لابد من استخدام الأساليب العلمية، والمعالجات والخوارزميات والنظم لاستخراج المعرفة والأفكار اللازمة للتطوير في كافة المجالات. تم ذلك في السابق باستخدام وسائل تقليدية في التخزين والعرض يتعذر استخدامها في الوقت الحالي، وذلك نتيجة التضخم الهائل الحاصل في كمية البيانات المنتجة وتنوع مصادرها، حيث أصبحت ترقى لمصطلح "البيانات الضخمة" وتحتاج لمنصات خاصة للتعامل معها (تخزين وتحليل وتمثيل) للحصول على نتائج مفيدة.

انفجر في السنوات الأخيرة نظام الكائنات الشبكية المعروف بإنترنت الأشياء Internet Of Things (IOT) والذي ساهم في زيادة تضخم كمية البيانات وتنوعها، حيث يمكن تحويل أي كائن مادي إلى جزء من إنترنت الأشياء ببساطة عن طريق توصيل جهاز استشعار به.

من المتوقع أن تصل كمية البيانات التي يتم إنشاؤها بواسطة أجهزة إنترنت الأشياء إلى 79.4 زيتابايت من البيانات بحلول عام 2025[5]، كما أنه من المتوقع أيضاً أن يتضاعف عدد أجهزة إنترنت الأشياء في جميع أنحاء العالم ثلاث مرات تقريبًا من 8.74 مليار في عام 2020 إلى أكثر من 25.4 مليار جهاز إنترنت الأشياء في عام 2030[4] (انظر الشكل 2 .**Error! Reference source not found**).



الشكل 2. عدد أجهزة انترنت الأشياء المتصلة من عام 2019 و حتى 2030 حسب موقع www.statista.com

تعد البيانات التي يتم إنشاؤها بواسطة أجهزة إنترنت الأشياء أكثر ثراءً من أنواع البيانات الأخرى، نظرًا لأنه يمكن توصيل المستشعرات بأي جهاز مادي، وبالتالي فإن بيانات إنترنت الأشياء متنوعة ودقيقة وتتدفق بسرعة فتتراكم وتنتج كماً هائلاً من البيانات لتحليلها .

تُستخدم أجهزة إنترنت الأشياء في جميع أنواع قطاعات الصناعة والأسواق الاستهلاكية، مثل قطاعات الصناعة الرئيسية كالكهرباء والغاز والبخار والتكييف وإمدادات المياه وإدارة النفايات وتجارة التجزئة والجملة والنقل والتخزين والحكومة.

على سبيل المثال، يمكن للمباني الذكية Smart Buildings جمع البيانات المتعلقة بما يلي:

- الظروف البيئية، مثل جودة الهواء ومستوى التلوث ودرجة الحرارة والرطوبة والسطوع، حتى تتمكن من معرفة ما يجب تغييره من أجل سلامة الإنسان و راحته.
- أنماط استخدام الطاقة، حتى يتم فهم كيف ومتى يستخدم مبنى ما الطاقة ويمكن اتخاذ خطوات لتحسين كفاءة الطاقة.
- استخدام المياه .
- معدات المبنى الخاص بك واستخدام المعلومات للصيانة التنبؤية .

يتم تجميع بيانات انترنت الأشياء وتحليلها، وتستخدم العديد من منصات إنترنت الأشياء التعلم الآلي لتحليل البيانات. على سبيل المثال، يمكن أن تساعد بيانات المستشعر التي تقيس مستوى اهتزاز المعدات وغيرها في اكتشاف الحالات الشاذة والتنبؤ بالمشكلات قبل ظهور مشكلات خطيرة، ينتج عن امتلاك القدرة على استخدام جميع البيانات رؤى أكثر قابلية للتنفيذ وعائد استثمار أكبر نتيجة لذلك .

تولد أجهزة انترنت الأشياء (ولا سيما حساسات انترنت الأشياء التي تقرأ قيمة ما) تدفقات بيانات ضخمة عالية السرعة نحتاج في معظمها لمعالجتها وتحليلها في الزمن الحقيقي (real time)، حيث يتم العمل على مراقبة القيم واكتشاف الشاذة منها أو استخدامها للقيام بالعمليات الحسابية والتجميعية اللازمة وغيرها من الاستخدامات التي تساعد في قراءة الواقع واكتساب الرؤى بشكل أسرع.

وبما أن قيمة انترنت الأشياء في البيئات الذكية تأتي من أهمية المعرفة المتأتية من تحليل البيانات التي يقدمها، لذا نحن بحاجة لتحليل البيانات بطرق فعالة تدعم عمليات المعالجة في الزمن الحقيقي لتحقيق الاستفادة القصوى والحصول على أفضل النتائج بوقت قليل، مما أدى إلى حاجة ملحة لأطر عمل تجمع بين تقنيات البيانات الضخمة وخدمات انترنت الأشياء.

1.1 مشكلة البحث

مع تزايد أجهزة انترنت الأشياء (من حساسات وغيرها) في البيئات الذكية تزايدت كمية البيانات المتدفقة باختلاف أنواعها وأصبح من غير الممكن التعامل معها بالطرق التقليدية. أما التحدي الأساسي لبيانات إنترنت الأشياء هو طبيعتها المتأتية من كونها بيانات تحتاج للمعالجة والتحليل في الزمن الحقيقي. بحلول عام 2025، ستكون 30٪ من جميع البيانات هي بيانات زمن حقيقي، حيث تمثل إنترنت الأشياء ما يقرب من 95٪ منها، وستكون 20٪ من جميع البيانات حرجة، وستكون 10٪ من جميع البيانات شديدة الأهمية[14]. يجب أن تحدث التحليلات في الزمن الحقيقي حتى تستفيد الشركات من هذه الأنواع من البيانات.

لذلك ظهرت الحاجة إلى ضرورة استخدام تقنيات جديدة لمعالجة هذه البيانات والحاجة إلى عرض النتائج بطرق فعالة أكثر. **وبالتالي تطرح هذه الدراسة التساؤل التالي : ماهي كفاءة أطر تحليل البيانات الحالية على تحليل بيانات انترنت الأشياء الضخمة في الزمن الحقيقي؟**

1.2 أهداف البحث

تهدف هذه الدراسة إلى دراسة أنظمة تحليل البيانات وتحديد مدى قدرتها وكفائتها في تحليل كم البيانات الكبير المتولد في الزمن الحقيقي والمتدفق من انترنت الأشياء والذي يزداد بشكل مضطرد مع تزايد الأجهزة المتصلة بالشبكة وتحسين مراقبة

النتائج واستخلاص المعلومات الناتجة عن انترنت الأشياء. ويتم ذلك من خلال تحقيق الأهداف الفرعية التالية

1- دراسة أنظمة تحليل البيانات الحالية
2- مقارنة كفاءتها في تحليل بيانات انترنت الأشياء الضخمة
3- تحديد مدى قدرتها وقابليتها على تحليل بيانات انترنت الأشياء في الزمن الحقيقي

2 المفاهيم البحثية المستخدمة

2.1 انترنت الأشياء Internet of Things (IoT) والبيئات الذكية Smart Environment

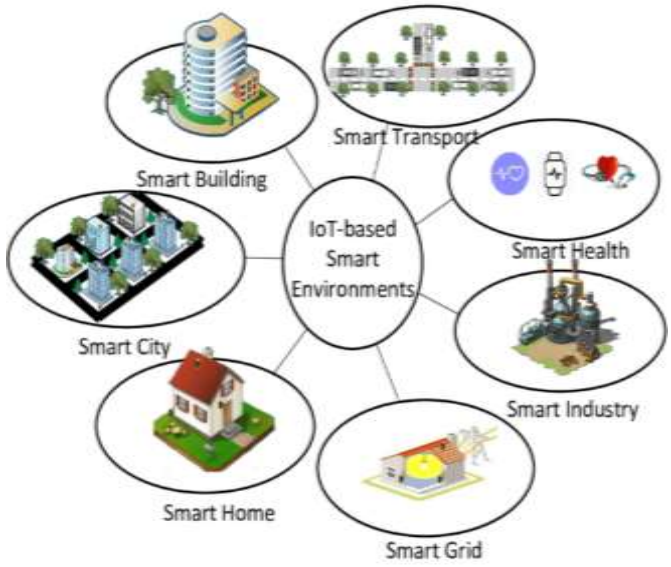
انترنت الأشياء Internet of Things (IoT)، هو نظام من أجهزة الحوسبة المترابطة والآلات الميكانيكية والرقمية والأشياء والحيوانات أو الأشخاص التي يتم تزويدها بمعرفات فريدة (IP)، والقدرة على نقل البيانات عبر الشبكة دون الحاجة لتفاعل إنسان مع إنسان أو إنسان مع كمبيوتر [8] .

يمكن أن يكون الشيء شخصًا مرتبطاً بحساس لمراقبة القلب أو سيارة تحتوي على مستشعرات مدمجة لتنبيه السائق عندما يكون ضغط الإطارات منخفضًا، وبالتالي الشيء هو أي جهاز أو طرفية أو نحو ذلك يمكن تعريفه على الإنترنت من خلال عنوان IP يساعد على التفاعل مع البيئة الخارجية.

انبثق مفهوم البيئة الذكية استنادًا إلى الفكرة الأولية للحوسبة في كل مكان، وهي تشجع على فكرة "عالم يمكن تصورهُ بأنه مليء بالمستشعرات والأجهزة المرتبطة ببعضها البعض وتندمج معنا لتصبح جزءاً من حياتنا اليومية من خلال الارتباط مع شبكة دائمة ومستمرة"، أيضاً تم تعريف البيئة الذكية بأنها "عالم صغير تعمل فيه أنواع مختلفة من الأجهزة الذكية باستمرار لجعل حياة البشر أكثر راحة" [14].

ويوضح (الشكل 3) بعضاً من تطبيقاتها حيث انتشرت البيئات الذكية على نطاق واسع بسبب المميزات التي تملكها ومنها:

- أجهزة التحكم عن بعد للتحكم في الأجهزة
- الاتصال بين الأجهزة باستخدام برمجيات وسيطة
- الحصول على المعلومات من خلال الحساسات ونقلها
- التنبؤية والقدرة على اتخاذ القرار



الشكل 3 تطبيقات البيئة الذكية

2.2 البيانات الضخمة Big Data

البيانات الضخمة تعني مجموعة كبيرة من البيانات، تستعصي لضخامتها أو تعقيدها على التخزين على جهاز واحد وإنما تتطلب مجموعة من الأجهزة (تسمى عناقيد)، ويصعب معالجتها والوصول إليها بسرعة بإحدى الأدوات أو التطبيقات المعتادة لإدارة البيانات [15].

تتدفق البيانات بمعدل مرتفع (بمعنى معدل تولدها وتغيرها) ويجب معالجتها بأقل وقت استجابة.

تتطلب البيانات الضخمة تقنيات خاصة ومتطورة وأشكال جديدة من المعالجة لتحليلها واستخلاص النتائج حيث تأتي أهميتها من القيمة المضافة بفهمها حيث تعزز عملية صنع القرار والفهم العميق في شتى المجالات.

2.3 أطر معالجة البيانات الضخمة Hadoop و Spark [17] [16]

يوجد اليوم العديد من أطر معالجة البيانات الضخمة كـ Apache Hadoop و Apache Spark وApache Hive وApache Cassandra وغيرها، من أكثرها شيوعًا واستخداماً اليوم هما اطارا Apache Hadoop وApache Spark.

استخدم كل منهما في دراسات مختلفة تبحث في معالجة البيانات الضخمة بشكل عام ومعالجة البيانات الضخمة الناتجة عن انترنت الأشياء بشكل خاص، ولكونهما الإطارين الرائدين لدينا دائمًا سؤال حول الإطار الذي يجب استخدامه Hadoop أم Spark، لذا سنقوم بتعريف كل إطار ومكوناته ثم إجراء مقارنة لمعرفة الإطار الأنسب لمعالجة البيانات الناتجة عن انترنت الأشياء .

2.3.1 هادوب Hadoop

Apache Hadoop عبارة عن إطار عمل أو منصة تتعامل مع مجموعات البيانات الكبيرة بطريقة موزعة، يستخدم طريقة MapReduce التي تعمل على تقسيم البيانات إلى كتل وتعيينها للعقد ضمن العنقود ،ثم تعالج الطريقة MapReduce البيانات بالتوازي على كل عقدة لإنتاج مخرجات فريدة.

كل آلة في العنقود تخزن البيانات وتعالجها، يقوم Hadoop بتخزين البيانات على الأقراص باستخدام نظام الملفات Hadoop Distributed File System

(HDFS)، كما يقدم خيارات سلسة للتوسع حيث يمكن البدء بجهاز واحد ثم التوسع إلى الآلاف، مع إضافة أي نوع من الأجهزة المنتجة من قبل شركات مختلفة.

نظام Hadoop متسامح للغاية مع الأخطاء، تم تصميم Hadoop للبحث عن حالات الفشل في طبقة التطبيق من خلال نسخ البيانات عبر العنقود، عندما يحدث فشل في جهاز ما يمكن للإطار بناء الأجزاء المفقودة من موقع آخر.

يتكون مشروع Apache Hadoop من أربع وحدات رئيسية:

- Hadoop Distributed File System– HDFS . وهو نظام الملفات الذي يدير تخزين مجموعات كبيرة من البيانات عبر العنقود، يمكن لـ HDFS التعامل مع البيانات المهيكلة وغير المهيكلة، كما يمكن أن تتراوح أجهزة التخزين من محركات أقراص صلبة فردية إلى محركات أقراص المؤسسات.
- MapReduce. مكون المعالجة لنظام Hadoop، يقوم بتوزيع أجزاء البيانات على العقد في العنقود ليقوم بمعالجتها على التوازي ومن ثم الدمج للحصول على النتيجة المرجوة.
- Yet Another Resource Negotiator– YARN. مسؤول عن إدارة موارد الحوسبة وجدولة الوظائف.
- Hadoop Common. مجموعة المكتبات العامة والأدوات المساعدة، اسم آخر لهذه الوحدة هو Hadoop core، لأنه يوفر الدعم لجميع مكونات Hadoop الأخرى.

2.3.2 سبارك Spark

Apache Spark أداة مفتوحة المصدر، يمكن تشغيل إطار العمل هذا في وضع مستقل (standalone) أو على سحابة. تم تصميمه للحصول على أداء سريع حيث يستخدم ذاكرة الوصول العشوائي للتخزين المؤقت للبيانات ومعالجتها.

يقوم Spark بأداء أنواع مختلفة من أعمال البيانات الضخمة، يتضمن ذلك معالجة الدُفعات (batch processing) المشابهة لـ MapReduce ، بالإضافة إلى معالجة التدفق في الزمن الحقيقي (real-time stream processing)، والتعلم الآلي، والرسم البياني، والاستعلامات التفاعلية، ومن خلال واجهات برمجة التطبيقات عالية المستوى سهلة الاستخدام يمكن أن يتكامل Spark مع العديد من المكتبات المختلفة، بما في ذلك PyTorch و TensorFlow.

تم إنشاء محرك Spark لتحسين كفاءة MapReduce والحفاظ على فوائده، وعلى الرغم من أن Spark لا يحتوي على نظام الملفات الخاص به، إلا أنه يمكنه الوصول إلى البيانات الموجودة على العديد من حلول التخزين المختلفة، وتسمى بنية البيانات التي يستخدمها Spark مجموعة البيانات الموزعة المرنة resilient distributed Datasets (RDD) .

هناك خمسة مكونات رئيسية لـ Apache Spark:

- Apache Spark Core. أساس المشروع بأكمله، يعد Spark Core مسؤولاً عن الوظائف الضرورية مثل الجدولة، وإرسال المهام، وعمليات الإدخال والإخراج، واسترداد الأخطاء، وما إلى ذلك .
- Spark Streaming. يتيح هذا المكون معالجة تدفقات البيانات في الزمن الحقيقي، حيث يمكن أن تنشأ البيانات من العديد من المصادر المختلفة .
- Spark SQL. يستخدم Spark هذا المكون لجمع معلومات حول البيانات المنظمة (المهيكلة) وكيفية معالجة هذه البيانات.
- مكتبة التعلم الآلي (MLlib). تتكون هذه المكتبة من العديد من خوارزميات التعلم الآلي، هدف MLlib هو قابلية التوسع وجعل التعلم الآلي أكثر سهولة.
- GraphX. مجموعة من واجهات برمجة التطبيقات المستخدمة لتسهيل مهام تحليلات الرسم البياني.

2.3.3 الاختلافات الرئيسية بين Hadoop و Spark

توضح الأقسام التالية أوجه التشابه والاختلاف الرئيسية بين الإطارين، سنلقي نظرة على Hadoop مقابل Spark من زوايا متعددة مثل الأداء والكلفة وسهولة الاستخدام لإيجاد الإطار الأكثر ملائمة للتعامل مع بيانات انترنت الأشياء.

- الأداء

يعمل Hadoop من خلال الوصول إلى البيانات المخزنة محليًا على القرص الصلب (HDFS) وهذا لا يتطابق مع معالجة Spark في الذاكرة، وفقًا لادعاءات Apache فإن Spark أسرع 100 مرة عند استخدام ذاكرة الوصول العشوائي للحوسبة من Hadoop مع MapReduce، ويحتاج spark إلى عدد أقل من العقد بمعدل 10 أضعاف لمعالجة 100 تيرابايت من البيانات على HDFS.

السبب الرئيسي لهذا التفوق لـ Spark هو أنه لا يقرأ ويكتب البيانات الوسيطة على الأقراص ولكنه يستخدم ذاكرة الوصول العشوائي بينما يقوم Hadoop بتخزين البيانات على الأقراص الصلبة ثم معالجة البيانات على دفعات باستخدام MapReduce، وهذا ما يجعل Spark أفضل للتعامل مع بيانات انترنت الأشياء التي تتصف أصلاً بالسرعة في الإنتاج وتحتاج لسرعة في المعالجة.

- الكلفة

يمكن استخدام النظامين بشكل مجاني تمامًا، ومع ذلك يجب أن تؤخذ تكاليف البنية التحتية والصيانة والتطوير في الاعتبار للحصول على التكلفة الإجمالية التقريبية.

العامل الأكثر أهمية في فئة التكلفة هو الأجهزة الأساسية التي تحتاجها لتشغيل هذه الأدوات، نظرًا لأن Hadoop يعتمد على أي نوع من أنواع التخزين على القرص لمعالجة البيانات، فإن تكلفة تشغيله منخفضة نسبيًا.

من ناحية أخرى، يعتمد Spark على العمليات الحسابية في الذاكرة لمعالجة البيانات في الزمن الحقيقي، لذلك فإن استخدام العقد التي تحتوي على الكثير من ذاكرة الوصول العشوائي يزيد من التكلفة.

تشير النقاط أعلاه إلى أن البنية التحتية لنظام Hadoop أكثر فعالية من حيث التكلفة و لكن نحتاج إلى تذكير أن Spark يعالج البيانات بشكل أسرع، وبالتالي يتطلب الأمر عددًا أقل من الأجهزة لإكمال نفس المهمة.

- معالجة البيانات

على الرغم من أن كل من Hadoop و Spark يعالجان البيانات في بيئة موزعة، فإن Hadoop أكثر ملاءمة لمعالجة الدُفعات. في المقابل ،يتألق Spark من خلال المعالجة في الزمن الحقيقي.

هدف Hadoop هو تخزين البيانات على الأقراص ثم تحليلها بالتوازي على دفعات عبر بيئة موزعة، لا يتطلب MapReduce قدرًا كبيرًا من ذاكرة الوصول العشوائي (RAM) للتعامل مع كميات البيانات.

يعمل Apache Spark باستخدام العمليات الحسابية في الذاكرة (in-memory computations) وواجهات برمجة التطبيقات عالية المستوى، لذا يتعامل Spark بشكل فعال مع التدفقات المباشرة للبيانات، وهذا ما يميزه عن غيره في التعامل مع بيانات انترنت الأشياء .

- قابلية التوسع

يصبح الخط الفاصل بين Hadoop و Spark ضبابيًا في هذا القسم.

يستخدم Hadoop نظام الملفات (HDFS) للتعامل مع البيانات الضخمة، عندما ينمو حجم البيانات بسرعة يمكن لـ Hadoop التوسع بسرعة لاستيعاب الطلب.

نظرًا لأن Spark لا يحتوي على نظام ملفات خاص به، فإنه يتعين عليه الاعتماد على نظام الملفات (HDFS) عندما تكون البيانات كبيرة جدًا بحيث لا يمكن التعامل معها.

يمكن للعناقيد توسيع وتعزيز قوة الحوسبة بسهولة عن طريق إضافة المزيد من الخوادم إلى الشبكة. نتيجة لذلك، يمكن أن يصل عدد العقد في كلا الإطارين إلى الآلاف، لا يوجد حد ثابت لعدد الخوادم التي يمكنك إضافتها إلى كل مجموعة وكمية البيانات التي يمكنك معالجتها.

- سهولة الاستخدام ودعم لغة البرمجة

قد يكون Spark هو إطار العمل الأحدث مع عدم وجود العديد من الخبراء المتاحين مثل Hadoop، ولكن من المعروف أنه أكثر سهولة في الاستخدام. في المقابل، يوفر Spark دعمًا للغات متعددة بجانب اللغة الأصلية (scala): Java و Python و R و Spark SQL، حيث يتيح ذلك للمطورين استخدام لغة البرمجة التي يفضلونها.

يعتمد إطار Hadoop على Java، اللغتان الرئيسيتان لكتابة كود MapReduce هما Java أو Python. لا يحتوي Hadoop على وضع تفاعلي لمساعدة المستخدمين.

بالإضافة إلى دعم واجهات برمجة التطبيقات بلغات متعددة، يفوز Spark في قسم سهولة الاستخدام من خلال وضعه التفاعلي. يمكن استخدام Spark shell لتحليل البيانات بشكل تفاعلي مع Scala أو Python، كما يوفر ملاحظات فورية على الاستعلامات، مما يجعل استخدام Spark أسهل من Hadoop MapReduce.

الشيء الآخر الذي يعطي Spark اليد العليا هو أنه يمكن للمبرمجين إعادة استخدام الكود الموجود عند الاقتضاء، وبالتالي تقليل وقت تطوير التطبيقات، كما يمكن دمج

البيانات التاريخية والبيانات المتدفقة لجعل هذه العملية أكثر فعالية، وهذا ما نواجهه بوضوح في تطبيقات إنترنت الأشياء.

- التعلم الآلي

التعلم الآلي هو عملية تكرارية تعمل بشكل أفضل باستخدام الحوسبة في الذاكرة. لهذا السبب، أثبت Spark أنه حل أسرع في هذا المجال.

السبب في ذلك هو أن Hadoop MapReduce يقسم الوظائف إلى مهام متوازية قد تكون كبيرة جدًا بالنسبة لخوارزميات التعلم الآلي و يخلق مشكلات في أداء الإدخال / الإخراج في تطبيقات Hadoop.

يأتي Spark مع مكتبة تعلم الآلة الافتراضية MLlib، تقوم هذه المكتبة بالحسابات التكرارية في الذاكرة، حيث يتضمن أدوات لأداء الانحدار (regression) ، والتصنيف(classification)، وبناء خطوط الأنابيب(pipeline constructing) ، والتقييم(evaluating)، وغير ذلك الكثير.

أثبت Spark مع مكتبة MLlib أنه أسرع بأشواط، وأنه الخيار الأفضل للتعلم الآلي، مما يعطيه الأفضلية في التعامل مع بيانات انترنت الأشياء التي غالباً ما نستخدمها للقيام بعمليات التصنيف والتنبؤ.

2.3.4 حالات استخدام Hadoop مقابل Spark

بالنظر إلى Hadoop مقابل Spark في الأقسام المذكورة أعلاه، يمكننا استخراج بعض حالات الاستخدام لكل إطار عمل.

تشمل حالات استخدام Hadoop:

- بناء البنية التحتية لتحليل البيانات بميزانية محدودة.
- إكمال الوظائف التي لا تتطلب نتائج فورية، والوقت ليس عاملاً مقيدًا.

- معالجة الدفعات مع المهام التي تستغل عمليات القراءة والكتابة على القرص.
- تحليل البيانات التاريخية والأرشيفية.

يمكننا فصل حالات الاستخدام التالية حيث يتفوق Spark بالأداء:

- تحليل بيانات التدفق في الوقت الحقيقي.
- عندما يكون الوقت جوهريًا، تقدم Spark نتائج سريعة مع عمليات حسابية في الذاكرة.
- التعامل مع سلاسل العمليات المتوازية باستخدام الخوارزميات التكرارية.
- معالجة موازية للرسم البياني لنمذجة البيانات.
- جميع تطبيقات التعلم الآلي.

3 الدراسات المرجعية

تم من خلال أدبيات الدراسة توضيح أهمية وكيفية تحليل البيانات الضخمة في مختلف المجالات وشتى أنواع البيانات. تختلف المنصات المستخدمة في التحليل ويبقى أكثرها شيوعاً منصتي هادوب وسبارك. كما تختلف مصادر البيانات الضخمة فمنها ما هو ناتج عن الشركات ومعاملات العملاء والمناقلات البنكية ووسائل التواصل الاجتماعي وغيرها بشكل عام، ومنها ما ينتج عن أجهزة انترنت الأشياء بشكل خاص.

يتم في هذا لاقسم إلقاء الضوء على مجموعة من الدراسات التي صنفت إلى دراسات استخدمت بيانات من مصادر عامة و دراسات تخصصت في مجال انترنت الأشياء مع بيان المنصات المستخدمة في التحليل .

3.1 بيانات عامة

تدرك شركات البيع بالتجزئة الحاجة إلى التحليل والتنبؤ بمبيعاتها وسلوك العملاء مقابل منتجاتها وفئات منتجاتها، حيث يتم مساعدة شركات البيع بالتجزئة على إنشاء صفقات وعروض ترويجية مخصصة لعملائها من خلال أطر عمل البيانات

الضخمة التي تسمح لهم بالتعامل مع أحجام مبيعات ضخمة بطرق أكثر كفاءة، استخدم إطار Apache Spark لتحليل بيانات مبيعات الجمعة السوداء (Black Friday) [2] و تم تدريب نماذج للتعلم الآلي باستخدام مكتبة التعلم الآلي المدمجة MLlib للتنبؤ بالأسعار والمبيعات في المستقبل ،حيث تم أولاً تنفيذ نموذج الانحدار الخطي Linear regression [18] ونموذج الغابة العشوائية Random Forest [19] دون استخدام إطار Spark وكانت الدقة 68٪ و 74٪ على التوالي. بعد ذلك، تم تدريب هذه النماذج على إطار عمل البيانات الضخمة للتعلم الآلي في Spark حيث حقق نتائج أفضل بدقة 72٪ لنموذج الانحدار الخطي و 81٪ لنموذج الغابة العشوائية، تم العمل بأسلوب معالجة الدفعات (batch processing) دون الاستفادة من قدرة Spark على المعالجة في الزمن الحقيقي.

وفي مجال آخر، لوحظ أن تحليل المشاعر هو الاتجاه الأكثر شيوعًا في عالم اليوم حيث تم إنجاز الكثير من العمل في هذا القطاع، وتعتبر وسائل التواصل الاجتماعي هي مصدر حيوي للمعلومات في هذه الحالة. يستقبل موقع Twitter، أحد أكبر مواقع التواصل الاجتماعي، ملايين التغريدات كل يوم، تحاول الصناعات المختلفة استخدام هذه البيانات النصية الضخمة لاستخراج آراء الناس تجاه منتجاتهم، حيث استخدم إطار Hadoop لتحليل عدد كبير من التغريدات التي تعبر عن رأي المستخدم لتصنيفها وتخصيص القطبية لكل تغريدة ما إذا كان المستخدم يعبر عن رأي إيجابي أو سلبي[6]، حيث بلغ متوسط الدقة في تصنيف الرأي من إيجابي أو سلبي أو حيادي قيمة مقدارها 72.27 مع محاولة الحصول على زمن استجابة مقبول بتقليل عمليات الوصول إلى القرص الصلب .

3.2 بيانات انترنت الأشياء

أدى تطور الالكترونيات والشبكات ووسائل الاتصال إلى دخول انترنت الأشياء في كافة المجالات الحياتية والخدمية والصناعية والصحية وغيرها .

حيث تم اقتراح إطار عمل يجمع بين تقنيتي انترنت الأشياء وتحليل البيانات الضخمة [7]، يعتمد على المعالجة المتوازية للبيانات الموزعة، يتألف الإطار المقترح من عدة مستويات ويركز بشكل أساسي على المشكلات المتعلقة برؤية المدينة الذكية وقدرتها على اتخاذ القرار. تكون الطبقة الأولى مسؤولة عن الاتصال وتوليد البيانات، أما الطبقة الثانية مهمتها جمع و تخزين البيانات في بيئة موزعة، في الطبقة الثالثة تتم معالجة البيانات المخزنة باستخدام تقنيات البيانات الضخمة مثل MapReduce. حيث إن MapReduce هو نموذج معالجة عالي للمعالجة الموزعة والمتوازية والمعتمد من قبل نظام Hadoop. طبقة التحليل هي الأخيرة والتي توفر وسيلة للتفاعل بين الأشخاص والأجهزة مباشرةً لاتخاذ القرارات والتنبؤ وإنشاء التقارير والتوصيات. رسمت هذه الدراسة خارطة طريق للباحثين في مجال معالجة البيانات الضخمة الناتجة عن انترنت الأشياء ولكن لم يتم التطبيق على بيانات حقيقية وبالتالي لا نستطيع التحقق من دقة النتائج.

تعتبر قضية توفير الطاقة أمراً هاماً ،حيث تم اقتراح نظام إدارة الطاقة Energy Management System (EMS) للمنازل الذكية[1] . في هذا النظام، يتم ربط كل جهاز منزلي بوحدة اكتساب البيانات، حيث كل جهاز هو كائن إنترنت أشياء بعنوان IP فريد مما يؤدي إلى شبكة لاسلكية كبيرة من الأجهزة، تجمع وحدة نظام الحصول على البيانات بيانات استهلاك الطاقة من كل جهاز في كل منزل ذكي وتنقل البيانات إلى خادم مركزي لمزيد من المعالجة والتحليل. تتراكم هذه المعلومات في خادم عام باعتبارها بيانات كبيرة، استخدمت برمجيات الأعمال الجاهزة Business Intelligence (BI) وتحليلات البيانات الضخمة لإدارة استهلاك الطاقة بشكل أفضل وتلبية طلب المستهلكين، ونظراً لأن تكييف الهواء يساهم في 60٪ من استهلاك الكهرباء في دول الخليج العربي، فقد تم أخذ وحدات HVAC (التدفئة والتهوية وتكييف الهواء) كدراسة حالة للتحقق من صحة النظام المقترح. تم بناء نموذج أولي واختباره في المختبر لتقليد أنظمة التدفئة والتهوية وتكييف الهواء في المناطق السكنية الصغيرة، حيث يمكن

لمالكي المنازل مراقبة استهلاكهم اليومي والشهري والسنوي على شكل مخططات بيانية ،كما يمكن للمسؤولين عن قطاع الكهرباء في المدينة استعراض استهلاك المنازل في مناطق محددة على شكل مخططات جغرافية . وقد ظهر أن كمية نقل البيانات واستهلاك الموارد يتأثران طرداً بازدياد عدد الزبائن كما يزداد زمن الوصول. في هذه الدراسة كان الاهتمام الأكبر بالبنية وبروتوكولات النقل وطرق العرض.

و كدمج بين إنترنت الأشياء وتحليلات البيانات الضخمة في مجال السياحة الذكية والتراث الثقافي المستدام، تم تقديم نظام TreSight [11] في مدينة ترينتو– إيطاليا ، وهو نظام توصيات للسائحين، تم استخدام البيانات من OpenData Trentino فيما يتعلق بالنقاط المثيرة للاهتمام، والمناخ، والمطاعم النموذجية الموصى بها وما إلى ذلك، وتم توسيعها ببيانات إضافية تم جمعها من خلال حساسات منتشرة مع سوار يمكن ارتداؤه من أجل توفير تفاصيل أوفى وأدق متعلقة بالطقس ومستويات الازدحام ومتابعة الأنشطة السياحية، تم استخدام Hadoop بالإضافة لأدوات مساعدة من أجل معالجة جميع البيانات الثابتة والبيانات الديناميكية لتوفير قدرات متقدمة من حيث التحليل واستخراج المعرفة، وهو نظام فعّال قيد الاستخدام في مدينة ترينتو.

أما في مجال البيئة ومع انتشار الملوثات و تدني جودة الهواء برزت ضرورة التحرك للحد من التلوث وانبعاث الغازات السامة، فكان لابد من استخدام حساسات انترنت الأشياء لقياس درجة انتشار كل ملوث في الهواء. حيث تمت دراسة إمكانية الدمج بين مفهومي البيانات الكبيرة وإنترنت الأشياء في سياق التنبؤ بتلوث الهواء الذي يحدث عند زيادة نسبة الغازات الضارة في الجو مثل NO_2 ، SO_2 وغيرها، ليتم رفع الإنذار في حال الوصول لعتبة محددة (المستوى الحرج)، وتم استخدام Apache Spark مع مكتبته المدمجة Spark MLlib من أجل بناء نموذج التنبؤ [10]، وقد تم العمل على بيانات مخزنة مسبقاً دون الخوض في أهمية المعالجة للبيانات في الزمن

الحقيقي (real time)، وبمعدل خطأ root–mean–square error (RMSE) يساوي 0.13 ويعتبر جيداً طالما أنه أصغر من 0.3 .

ومن ناحية إنسانية، استخدم إنترنت الأشياء إلى جانب تحليل البيانات الضخمة لرصد المرضى في المناطق النائية، حيث يتم جمع المؤشرات الحيوية للمرضى باستخدام حساسات انترنت أشياء وإرسالها مباشرة، يتم تحليل المعلومات التي تم جمعها على الفور باستخدام نهج التعلم الآلي في Apache MAHOUT لمعرفة مدى خطورة الحالة بزمن استجابة مقبول [9]، يتم نقل نتائج التحليل إلى الأطباء القريبين في الصحة الأولية وكذلك الطبيب المسؤول عن المريض وأفراد الأسرة، ليتعيّن نقل المريض إلى المستشفى في الحالات الخطرة أو يستجيب الطبيب على الفور لاقتراح فوري بشأن الإسعافات الأولية و العلاج في الحالات الأقل خطورة، ولا تزال هكذا أمور متعلقة بحياة الإنسان قيد الدراسة دائماً لتقليل زمن الاستجابة والحصول على دقة أفضل.

وأما في الصناعة، تسير إنترنت الأشياء والبيانات الضخمة جنبًا إلى جنب في لعب دور مهم في القطاعات الصناعية ووجب تحليل أداء الجمع بين التقنيتين لتحقيق الاستدامة في الإنتاج الصناعي، حيث أدى تحديث الصناعات باستخدام الآلات المزودة بأجهزة الاستشعار المضمنة إلى زيادة سريعة في توليد البيانات وضرورة معالجتها، اقترح استخدام نظام Hadoop للتعامل مع الكميات الهائلة للبيانات بالإضافة إلى أدوات التعلم الآلي للتنبؤ بالاعطال و غيرها[13]، وقد نتج عن ذلك توفير في استخدام الطاقة وبزمن استجابة جيد وزمن تأخير قليل، كما ساهم في زيادة الإنتاج العام مع تقليل كمية النفايات الناتجة.

4 الإطار العملي

يتم في هذ القسم إجراء مقارنة بين أداء كل من Hadoop و Spark لتحليل البيانات الضخمة مع بيان خصوصية Spark في التعامل مع البيانات المتدفقة والتي يتم الحصول عليها من انترنت الأشياء .

يتم تنفيذ مقارنة الأداء على بيانات جاهزة (csv file) من شبكة الانترنت، وقد أتيحت هذه البيانات لأغراض البحث العلمي ، وتم تجميعها من قبل منظمة حكومية أميركية (U.S. EPA (Environmental Protection Agency ، وذلك باستخدام مستشعرات انترنت أشياء منتشرة في أنحاء الولايات المتحدة الأميركية تقيس مستوى الغازات في الجو لا سيما السامة منها (NO3 , SO2 , CO , O3) في كل يوم منذ عام 2000 وحتى عام 2016 .

تحوي مجموعة البيانات هذه على أكثر من 1.7 مليون سجل مع 29 عمود تحوي التاريخ ورقم الموقع ومعلومات مفصلة عن عنوان الحساس (ولاية ، مقاطعة ، منطقة) بالإضافة لأعمدة تحوي القيمة المسجلة لكل غاز من الغازات (NO3 , SO2 , CO , O3) ومؤشر الجودة لها.

قبل إجراء التجربة، تم القيام بمعالجة مسبقة للبيانات للتعرف عليها وتنظيفها، تم استخدام مكتبة Pandas وهي أداة مفتوحة المصدر بُنِيت باستخدام لغة بايثون سريعة وقوية ومرنة وسهلة الاستخدام تساعد في تحليل ومعالجة البيانات تقدم العديد من التوابع التي تساعد في قراءة ملفات البيانات واستكشافها والتعديل عليها، حيث تمت قراءة الملف واستعراض محتوياته والتعرف على أنماط المتغيرات، والإبقاء على الأعمدة اللازمة كالتاريخ وقيم مؤشر الجودة للغازات، كما تم تحويل أنماطها من قيم نصية (String) إلى قيم عددية (Integer)، كما وُجِد عدد كبير من السجلات التي تحوي على قيمة (Null) في عمودي SO2 و CO الأمر الذي استدعى معالجتها باستبدالها بالقيمة

الأكثر تكراراً ضمن العمود، كما وتم إنشاء عمود جديد يعبر عن جودة الهواء بشكل عام (AQI) تعتمد قيمه على قيم مؤشر الجودة للغازات كلها.

أجريت التجربة باستخدام جهاز حاسوب محمول يعمل بنظام تشغيل windows 10. والجدول (جدول 1 . مواصفات الجهاز) يبين مواصفات الجهاز

جدول 1 . مواصفات الجهاز

م	الجهاز	المواصفات
1	المعالج	Intel Core i5-6200U @2.4 GHz
2	ذاكرة وصول عشوائي	8.00 GB
3	قرص صلب	1 TB , 5200 r.p.m

تم استخدام إطار العمل سبارك ذو النسخة spark-3.1.1 مع موجه الأوامر jupyter ، وإطار العمل هادوب ذو النسخة hadoop-3.3.1 مع موجه أوامر نظام تشغيل windows 10 .

استخدمت لغة بايثون في كلا الإطارين لقراءة مجموعة البيانات وإيجاد عدد السجلات في المجموعة ومن ثم إيجاد المعلومات الإحصائية من القيمة الكبرى NO2_max، والقيمة الصغرى NO2_min، وقيمة الانحراف المعياري NO2_sd، للمتغير المعبر عن قيم غاز NO2 مع تسجيل الوقت الذي استغرقه كل إطار على حدا، و قد تمت التجربة على إطار العمل هادوب باستخدام طريقة المعالجة المعتمدة فيه وهي طريقة map-reduce .

كذلك قمنا بتجربة أداء كلا الإطارين في العمليات التكرارية ولا سيما التعلم الآلي، استخدمت المكتبة المدمجة sparkML مع إطار spark، بينما احتجنا للاستعانة بمكتبات خارجية لإتمام نفس العمل على إطار Hadoop، حيث تم بناء نموذجي تصنيف ،الأول باستخدام خوارزمية Random Forest Classifier، والثاني باستخدام خوارزمية Decision Tree Classifier، تم تدريب النماذج باستخدام الأعمدة التي أبقينا عليها وهي أعمدة مؤشر الجودة للغازات كلها ومؤشر جودة الهواء العام، ونختبرها فيما بعد باستخدام أعمدة مؤشر الجودة للغازات كلها ليعطينا مؤشر جودة الهواء العام.

والجدير بالذكر أننا قمنا بمحاكاة عملية استقبال بيانات إنترنت الأشياء في نظام Spark باستخدام مكون Spark Streaming الذي يسمح باستقبال البيانات ومعالجتها بالزمن الحقيقي، الأمر الذي تعذر القيام به في نظام Hadoop نتيجة افتقاره لمكونات تدعم عمليات التدفق الحقيقي ، ولتحقيق موضوع تدفق البيانات كما لو أنها تأتي من حساسات حقيقية قمنا باستخدام Netcat وهي أداة تقرأ وتكتب البيانات عبر اتصالات الشبكة، باستخدام بروتوكول TCP أو UDP، حيث استطعنا من خلاله بفتح منفذ (Port) بمعرف (9999) نقوم بكتابة البيانات عليه، حيث يقوم بكتابة القيمة تلو الأخرى بفارق زمني يمكننا تحديده بأنفسنا (مثلاً 1 ثانية)، بينما يقوم Spark بالتنصت على المنفذ واستقبال البيانات لنقوم بعرضها على شكل مخطط بياني تفاعلي ضمن متصفح الويب، ولرسم المخطط البياني التفاعلي تم استخدام Plotly وهي مكتبة تفاعلية مفتوحة المصدر تساعد في رسم أنواع مختلفة من المخططات البيانية وتحتوي على إمكانيات أداة التمرير (hover tool capabilities) التي تسمح لنا باكتشاف أي قيم متطرفة أو شذوذ في عدد كبير من نقاط البيانات، كما تم استخدام Dash وهو إطار عمل يعتمد لغة بايثون (python framework) تم إنشاؤه بواسطة Plotly لبناء تطبيقات ويب تفاعلية، تم تطويره بالاعتماد على flask (web framework)

ومكتبات Plotly.js و React.js، ويُعد Dash مفتوح المصدر ويقوم بعرض نتائجه ضمن تبويب في متصفح الويب، وبذلك استطعنا عرض المخطط البياني ضمن تبويب في المتصفح.

ونبين فيما يلي أهم النتائج التي حصلنا عليها لتحليل البيانات والحصول على مستويات NO2 في مجموعة البيانات، وكذلك التعلم الآلي باستخدام كل من الإطارين المذكورين Hadoop و Spark ،وأيضاً عرض البيانات المتدفقة في Spark .

5 النتائج والمناقشة

5.1 التجربة الأولى

يتم فيها قراءة مجموعة البيانات وإيجاد عدد السجلات في المجموعة، ثم إيجاد المعلومات الإحصائية من القيمة الكبرى، والقيمة الصغرى، وقيمة الانحراف المعياري للمتغير المعبر عن قيم غاز NO2 .

5.1.1 زمن التنفيذ

يعبر زمن التنفيذ عن الزمن المستغرق للحصول على المعلومات الإحصائية المذكورة لتحليل مستويات NO2 عند استخدام كل من الاطارين Hadoop و Spark اعتباراً من بداية قراءة البيانات وحتى طباعة النتائج والحصول على التقرير.

يبين (جدول 2 . زمن التنفيذ) زمن التنفيذ لكلا الاطارين.

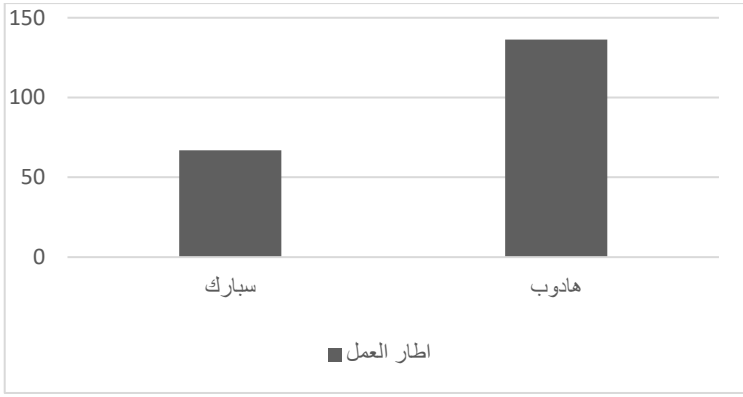
جدول 2 . زمن التنفيذ

الإطار	*زمن التنفيذ (ثا)*
Hadoop (map-reduce)	136.27
Spark	67

وقد استغرق Hadoop ضعف الوقت الذي استغرقه نظام Spark للقيام بنفس العمل، حيث نجد أن Hadoop استغرق دقيقتين و بضع ثوانٍ، بينما سبارك استغرق دقيقة و بضع ثوانٍ .

لدينا (

مخطط *1*) يوضح الوقت المستغرق من كلا الإطارين و يُظهِر الفرق بينهما



مخطط 1.الزمن المستغرق لتنفيذ المهمة

5.1.2 استخدام الموارد

من المعلوم أن أسلوب البرمجة map-reduce يعمل على مرحلتين mapping و من ثم reducing، حيث يتم استخدام القرص الصلب لتخزين البيانات الوسيطة (Intermediate output) الناتجة عن الـ mapper والتي ستكون دخل يقرأه الـ reducer لإتمام العملية .[20]

أما أسلوب المعالجة in-memory والمتّبع من قبل Spark فإنه يعمل على الاحتفاظ بالبيانات ضمن الذاكرة دون استخدام القرص الصلب، وهذا هو السبب الرئيسي للسرعة في أداء المهام باستخدام Spark .

يوفر Apache Spark واجهة ويب (Spark UI) لمراقبة حالة تطبيق Spark قيد التشغيل (الوظائف ، والمراحل ، والمهام ، .....) واستهلاك الموارد، و تتوفر على http: // [driver]: 4040 افتراضيًا ، وبمراقبة أداء التجربة باستخدام Spark UI تبيّن لنا أن المهمة تمت باستخدام الذاكرة فقط دون الاحتياج للقرص الصلب أثناء التنفيذ، وهذا ما يوضحه (جدول 3 جدول 3.استخدام القرص الصلب في Spark) الذي تم اقتصاصه من واجهة Spark UI .

جدول 3.استخدام القرص الصلب في Spark

Address	Status	RDD Blocks	Storage Memory	Disk Used	Cores
DESKTOP-H9KELC7:12665	Active	0	88.4 KiB / 366.3 MiB	0.0 B	1

5.2 التجربة الثانية (العمليات التكرارية)

تعبر عن قدرة الإطار على القيام بنفس الأعمال لعدد كبير من المرات كعمليات التعلم الآلي، والتي من المعلوم أنها تعمل بشكل أفضل في حال استخدام الحوسبة في الذاكرة وهذا ما أظهرته النتائج في (جدول 4) وذلك عند استخدام خوارزمية Random Forest Classifier .

جدول 4. زمن تنفيذ خوارزمية Random Forest Classifier

الإطار	*زمن التنفيذ (ثا)*
Hadoop	240

110	*Spark*

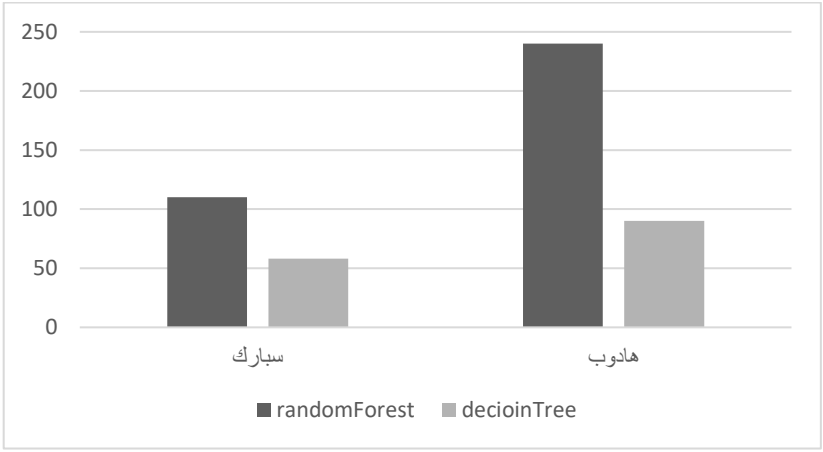
حيث نجد أن أداء Spark كان الأفضل ،حيث استغرق Hadoop ضعف الوقت تقريباً، و بتنفيذ خوارزمية Decision Tree Classifier والتي تستهلك وقتاً أقل من سابقتها كانت النتيجة كما في (جدول 5) والتي تظهر تفوّق Spark أيضاً.

جدول 5 . زمن تنفيذ خوارزمية Decision Tree Classifier

الإطار	*زمن التنفيذ (ثا)*
Hadoop	90
Spark	58

لدينا (

مخطط2) يوضح أداء كلا الإطارين في التعلم الآلي و يُظهِر الفرق بينهما



مخطط2 . زمن تنفيذ التعلم الآلي

5.3 التجربة الثالثة: تدفق البيانات

وهنا تظهر خصوصية Spark في التعامل مع بيانات انترنت الأشياء التي تتميز بتدفقها الدائم والتي استطعنا محاكاتها وتحليل البيانات مباشرة وعرضها بشكل بياني .

قمنا برسم مخطط بياني (Live Chart) يتحدث باستمرار لرسم القيمة الجديدة من غاز NO2 مع لحظة الوصول، وهذا ما يوضحه (مخطط 3) حيث يعبر المحور (X) عن لحظة الوصول بينما يعبر (Y) عن القيمة الواصلة، ويتميّز بأنه تفاعلي (Interactive) حيث يعطينا القيمة المسجلة عند المرور في نقطة محددة باستخدام مؤشر الفأرة .

مخطط 3. قيم غاز NO2 المتدفقة

نلاحظ أن spark تفوّق على hadoop بسرعته، حيث قام بإنجاز المهام مستغرقاً تقريباً نصف الوقت الذي احتاجه hadoop، كما أن spark يعمل مستخدماً الذاكرة دون الرجوع إلى القرص الصلب، وتميّز spark بمكتبة التعلم الآلي المدمجة MLlib ومكون Spark Streaming الذي سمح لنا باستقبال البيانات المتدفقة ومعالجتها بالزمن الحقيقي.

6 الخاتمة

تمت المقارنة نظرياً بين Apache Hadoop و Apache Spark من زوايا متعددة ودراسة مدى ملائمتها لتحليل بيانات انترنت الأشياء مع إلقاء الضوء على أهم ما يميزها عن غيرها من البيانات، وتم القيام بمقارنة عملية لإظهار سرعة وأداء كلا الإطارين في تنفيذ نفس المهمة.

يلعب كلا الإطارين دورًا مهمًا في تطبيقات البيانات الضخمة، بينما يبدو أن Spark هو الأنسب لبيانات انترنت الأشياء وذلك :

1. لسرعته وسهولة استخدامه .
2. يلائم وجود بيانات متدفقة بسرعة كحالنا في معالجة بيانات انترنت الأشياء والتي تحتاج إلى التعامل معها بالزمن الحقيقي .
3. الأفضل في حالات استخدام التعلم الآلي للحصول على نتائج مفيدة من البيانات الموجودة، وذلك لامتلاكه مكتبة MLlib المدمجة حيث يقوم بالعمليات الحسابية التكرارية مستخدماً الذاكرة .
4. قد يتطلب Spark ميزانية أكبر للصيانة ولكنه يحتاج إلى أجهزة أقل لأداء نفس الوظائف وبسرعة أكبر .

نتيجة لما سبق عرضه يمكن التأكيد على أهمية النتائج التي توصلنا إليها وإمكانية اعتماد Spark لتحليل البيانات الضخمة المتدفقة من أجهزة انترنت الأشياء بحيث يمكن بناء معماريات تحلل البيانات الواصلة بشكل مباشر .

7 الأعمال المستقبلية

لابد من متابعة العمل على إطار Spark والعرض البياني للمخططات بشكل يُظهِر القيم الإحصائية الضرورية بالزمن الحقيقي مما يسمح بمعالجة محتملة للحالات الحرجة .

8 المراجع

1. AL-ALI, A.R., I.A. ZUALKERNAN, M. RASHID, R. GUPTA, and M. ALIKARAR 2017- A smart home energy management system using IoT and big data analytics approach. **IEEE Transactions on Consumer Electronics**, Vol. 63, N. 4, 426–434.

2. AWAN, M., M. SHAFRY, M. RAHIM, H. NOBANEE, A. YASIN, I. KHALAF, U. ISHFAQ, and M. JAVED 2021- A Big Data Approach to Black Friday Sales. **Intelligent Automation and Soft Computing**, Vol. 27, 785–797.

3. HOLST, A. 2021- • Total data volume worldwide 2010-2025 | Statista. https://www.statista.com/statistics/871513/worldwide-data-created/.

4. HOLST, A. • IoT connected devices worldwide 2019-2030 | Statista. https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/.

5. IOTACOMM 2020- How Does IoT Affect Big Data? » Iota Communications, Inc. https://www.iotacommunications.com/blog/iot-big-data/.

6. MANE, S.B., Y. SAWANT, S. KAZI, and V. SHINDE Real Time Sentiment Analysis of Twitter Data Using Hadoop. .

7. MOHBEY, K. 2019- An Efficient Framework for Smart City Using Big Data Technologies and Internet of Things. In 319–328p.

8. S. GILLIS, A. What is IoT (Internet of Things) and How Does it Work? https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT.

9. SMYS, S. and J. RAJ 2019- INTERNET OF THINGS AND BIG DATA ANALYTICS FOR HEALTH CARE WITH CLOUD COMPUTING. **Journal of Information Technology and Digital World**, Vol. 01, 9–18.

10. SOSSI ALAOUI, S., B. AKSASSE, and Y. FARHAOUI 2019- Air pollution prediction through internet of things technology and big data analytics. **International Journal of Computational Intelligence Studies**, Vol. 8, 177.

11. SUN, Y., H. SONG, A.J. JARA, and R. BIE 2016- Internet of Things and Big Data Analytics for Smart and Connected Communities. **IEEE Access**, Vol. 4, 766–773.

12. 2021- How Much Data Is Created Every Day? [27 Powerful Stats]. SeedScientific. https://seedscientific.com/blog/how-much-data-is-created-every-day/.

13. The world's data explained: how much we're producing and where it's all stored. https://theconversation.com/the-worlds-data-explained-how-much-were-producing-and-where-its-all-stored-159964.

14. What is Smart Environments | IGI Global. https://www.igi-global.com/dictionary/smart-environments/27179.

15. Big Data: What it is and why it matters | SAS. https://www.sas.com/en_us/insights/big-data/what-is-big-data.html.

16. Hadoop vs. Spark: What's the Difference? | IBM. https://www.ibm.com/cloud/blog/hadoop-vs-spark.

17. Hadoop vs Spark: Detailed Comparison of Big Data Frameworks. https://phoenixnap.com/kb/hadoop-vs-spark.

18. ML | Linear Regression - GeeksforGeeks. https://www.geeksforgeeks.org/ml-linear-regression/.

19. Random Forest | Introduction to Random Forest Algorithm. https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/.

20. Hadoop - Mapper In MapReduce - GeeksforGeeks. https://www.geeksforgeeks.org/hadoop-mapper-in-mapreduce/.