Performance Evaluation of RAM Chunking Algorithm in Cloud Backup Service

Dr. Ahmad Mahmoud Ahmad*

(Received 15 / 6 / 2023. Accepted 30 / 8 / 2023)

□ ABSTRACT □

With the current explosion of digital data, the regulatory back-up data methods are the pending issue to be resolved. Deduplication is one of the main solutions to restrain the increase of duplicated copies of data and achieve cost-savings in data centers. Data deduplication is a technique for effectively reducing the storage requirement of practical backup by eliminating redundant data to ensure that only a single instance is stored in the storage medium. Chunking based deduplication is one of the most effective deduplication strategies, which replaces duplicated data with references to data already stored. In this thesis, we evaluate the performance of three different Content-based chunking algorithms (TTTD-S, AE, and RAM) by implementing them in C# in Visual Studio 2015 environment and then performing a series of systematic experiments. The results show superior performance of RAM in runtime and deduplication rate, while AE came second with a slight difference from RAM.

Keywords: Cloud Computing, Backup, Deduplication, Content-based chunking.

Copyright **:**Tishreen University journal-Syria, The authors retain the copyright under a CC BY-NC-SA 04

* **Associate Professor- Faculty of Information Engineering- Tishreen University- Lattakia- Syria.**

تقييم أداء خوارزمية التقطيع RAM في النسخ الاحتياطي في السحابة

د. احمد محمود احمد*

(تاريخ الإيداع 15 / 6 / 2023. قُبِل للنشر في 30/ 8 / 2023)

□ ملخّص □

مع الازدياد الهائل في كمية المحتوى الرقمي تبرز الحاجة إلى تطوير طرق الخزن الاحتياطي التقليدية. إزالة التكرار Deduplication هو أحد الحلول الأساسية المقترحة لمنع تراكم البيانات المكررة وتحقيق التوفير في تكاليف التخزين في مراكز البيانات Datacenters. إزالة تكرار البيانات هي تقنية فعالة لتقليل المتطلبات التخزينية الملائمة للخزن الاحتياطي؛ وذلك عن طريق استبعاد البيانات المكررة والاحتفاظ بنسخة واحدة فقط من البيانات على مستوى وسط التخزين.

إزالة التكرار بالتقطيع هي أكثر استراتيجيات إزالة التكرار فاعلية والتي تقوم على الإشارة للبيانات المكررة بمرجع للنسخة الموجودة سابقاً في وسط التخزين بدلاً من تخزينها مرة أخرى. تمّ في هذا البحث مقارنة ثلاث خوارزميات للتقطيع المعتمد على المحتوى Content-based chunking هي (TTTD-S, AE, RAM) وذلك بتحقيقها أولاً بلغة #C ضمن بيئة Visual Studio 2015 ومن ثم استخدام مجموعة من المعايير المنهجية للمقارنة فيما بينها. بناءً عليه، أبدت خوارزمية RAM تفوقاً على الخوارزميتين الباقيتين من حيث زمن التنفيذ ومعدل إزالة تكرار، في حين حلت خوارزمية AE ثانياً بفارق بسيط عنها.

الكلمات المفتاحية: حوسبة سحابية، خزن احتياطي، إزالة التكرار، التقطيع المعتمد على المحتوى.

حقوق النشر : مجلة جامعة تشرين- سورية، يحتفظ المؤلفون بحقوق النشر بموجب الترخيص CC BY-NC-SA 04

* **أستاذ مساعد- كلية الهندسة المعلوماتية- جامعة تشرين- اللاذقية- سورية.**

مقدمة:

أدى التحول حديثاً إلى تقنيات الاتصال الرقمية إلى زيادة كمية البيانات المخزنة بشكل رقمي بسرعة كبيرة، ففي عام 2007 ولأول مرة تجاوز الحجم الإجمالي للبيانات الرقمية Digital Data المراد تخزينها سعة التخزين الموجودة عالمياً. بل وأبعد من ذلك، إذ تتجاوز كمية البيانات الرقمية المولدة بشكل آلي كمية البيانات الناتجة عن المستخدمين البشريين.[1]

إضافة إلى المشكلة التخزينية الناتجة عن هذه الزيادة المتسارعة تبرز لدينا مشكلة جديدة عند تخزين البيانات الرقمية وهي أنها أكثر عرضة للخطأ مقارنة مع البيانات المخزنة تقليدياً (ورقياً مثلاً)، فعندما يتم حفظ هذه البيانات في نظام حاسوبي قد يؤدي خطأ تخزيني واحد أو انقطاع التيار الكهربائي إلى جعل كمية كبيرة من هذه البيانات عرضة للخطر أو الضياع، لذلك تم استخدام عدد من التقنيات لتجاوز هذه المشاكل وحماية البيانات لتعزيز توافرية Availability وموثوقية Reliability البيانات الرقمية المخزنة. من أهم هذه التقنيات الخزن أو النسخ الاحتياطي Backup، فعلى الرغم من ارتفاع الثقة بأداء العتاد Hardware والبرمجيات Software في السنوات الأخيرة إلا أن احتمال تعرضها للعطب لا يزال قائماً مما سيؤدي إلى تلف البيانات أو ضياعها.

من ناحية أخرى، جلبت الحوسبة السحابية سبلاً جديدة لاستخدام منخفض التكلفة للتطبيقات وتطويرها وامتلاك موارد حوسبية إضافية. التخزين السحابي Cloud storage هي خدمة تخزين مقدمة عبر الإنترنت تعتمد على مبدأ الدفع حسب الاستخدام pay-as-you-go حيث يدير مزود الخدمة Cloud Storage Vendors السعات التخزينية والأمن والخصوصية للبيانات المخزنة بما يضمن وصول المستخدم لها في أي وقت ومن جميع أنحاء العالم. [2]

مؤخراً، أصبحت خدمة الخزن الاحتياطي على السحابة حلاً فعالاً من حيث التكلفة وتم تبنيه من قبل عدة منظمات كاستراتيجية حماية بيانات بديلة من الاستراتيجيات التقليدية.

أهمية البحث وأهدافه:

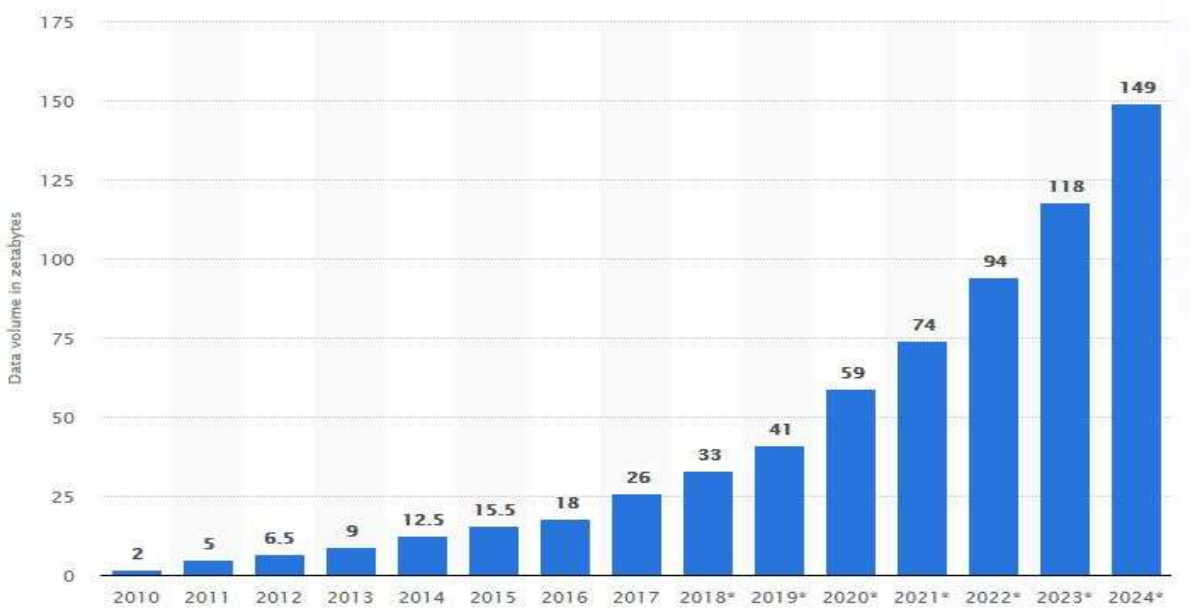
في العالم الرقمي الحديث، تأتي البيانات من مصادر مختلفة وبأشكال مختلفة. وقد أشارت آخر الإحصائيات[3] إلى أن حجم البيانات الرقمية بلغ 41ZB في عام 2019 ومن المتوقع وصولها إلى 149 ZB في 2024 كما هو موضح في الشكل رقم (1). هذا التضاعف فرض تحديات تخزينية وإدارية كبيرة لاسيما أنّ حوالي ثلاثة أرباع هذه البيانات فائضة (مكررة). [4]

كما أن توجه المستخدمين والمنظمات والشركات لاستخدام السحابة في تخزين نسخ احتياطية من بياناتهم عزز هذا التضاعف، خاصة أن الدراسات تشير إلى أن حوالي 90% من البيانات المخزنة احتياطياً هي بيانات مكررة أيضاً.

حاولت الكثير من الأبحاث التطرق لهذه المشكلة من خلال محاولة تقليل كمية البيانات التي يتم نقلها خلال عملية الخزن الاحتياطي وبالتالي تقليل أعباء إدارة هذه البيانات واستغلال المساحة التخزينية المتاحة بأفضل شكل ممكن، وأحد الحلول المقترحة هو تطبيق: إزالة التكرار من البيانات Deduplication.

تقوم عملية إزالة التكرار على تجزيء الملف إلى مجموعة من القطع Chunks وحساب بصمتها التشفيرية Hash value ومقارنتها مع جدول البصمات التشفيرية Hash table المخزن سابقاً في النظام. في حال وجود تطابق تعتبر القطعة مكررة؛ فتُحذف ويستعاض عنها بمؤشر، أما في حال عدم الحصول على تطابق فتعتبر فريدة ويتم تخزينها.[5]

يهدف هذا البحث إلى إيجاد أفضل خوارزمية تقطيع من حيث المعايير المدروسة لضمان تطبيق إزالة التكرار Deduplication بالشكل الأمثل مما سيؤدي إلى تحسين استغلال الموارد المتاحة (مساحة تخزينية – عرض حزمة).



الشكل (1): تضاعف أحجام البيانات عبر السنين

طرائق البحث ومواده:

قمنا في هذا البحث بمقارنة أداء ثلاث خوارزميات تقطيع معتمد على المحتوى هي: (TTTD-S, AE, RAM) تبعاً لمجموعة من المعايير هي: زمن التنفيذ – عدد القطع الناتجة – متوسط حجم القطع – نسبة إزالة التكرار.

نظراً لعدم وجود كود مفتوح المصدر Open Source، فقد قمنا بكتابة الكود الخاص بهذه الخوارزميات بلغة C# ضمن بيئة Visual Studio 2015، واستخدام Sql Server 2014 للتعامل مع قاعدة البيانات التي احتوت نتاج عملية التقطيع. أمّا مخططات النتائج فقد تم رسمها باستخدام برنامج MATLAB 2015، حيث تمّ استخدام حاسب شخصي Laptop بمواصفات العتاد التالية:

- CPU: Intel Core i5 2.3GHz
- RAM: 8GB
- Hard Drive: 240GB SSD

1- الحوسبة السحابية:[6]

الحوسبة السحابية حسب المعهد الوطني للمعايير والتكنولوجيا NIST هي نموذج لتمكين نفاذ شبكي ملائم عند الطلب لمستودع مشترك من الموارد الحوسبية القابلة للإعداد (الشبكات، المخدمات، التخزين، التطبيقات والخدمات) والتي يمكن توفيرها وتحريرها بأقل عبء إدارة ممكن أو تدخل من قبل مزود الخدمة. الحوسبة السحابية هي بيئة توزيع بنية تحتية جديدة تقوم على ضمان توفير خدمات عند الطلب كالحساب Computation، البرمجيات Software، والنفاذ للبيانات بشكل مرن من خلال جدولة عرض الحزمة، والمساحة التخزينية والموارد الحوسبية، بدون الحاجة لمعرفة المستخدم النهائي بالموقع الفيزيائي أو إعدادات النظام الذي يوفر الخدمة.

في جوهرها، لا تمثل الحوسبة السحابية أكثر من كونها حوسبة تعتمد على شبكة الإنترنت، يتم فيها تقديم التطبيقات السحابية للمستخدم عن طريق متصفح الويب Web browser بدون الحاجة لوجود برامج مثبتة مسبقاً على جهازه وبدون شراء تطبيق سحابي خاص بالخدمة.

نموذج الحوسبة السحابية نموذج مرن، ممّا يعني أن تخصيص الموارد يمكن أن يكون أكبر أو أصغر تبعاً للطلب، كما أنّ المرونة التي تتميز بها الحوسبة السحابية تُمكن من قابلية التوسع Scalability بمعنى أنّ الخدمات المقدمة يمكن أن تتدرج لخدمات أكبر أو أصغر حسب الحاجة.

2- التخزين السحابي Cloud Storage:[6]

التخزين السحابي Cloud Storage هو مساحة تخزينية هائلة متوفرة للعامة عبر الإنترنت. تدعى هذه الخدمة اصطلاحاً وفقاً لخدمات السحابة: خزن البيانات كـ خدمة (DaaS) Data storage as a Service وعليه فإنّ القدرات التخزينية التي تقدم كخدمة عبر شبكة الإنترنت بشكل مرن وقابل للتوسع توصف على أنها خزن سحابي.

أهم ما في الخزن السحابي هو النسخ أو الخزن الاحتياطي Backup على السحابة. إذ يحفظ الخزن الاحتياطي على السحابة بيانات من جانب الزبون على جانب مزود خدمة الخزن السحابي عبر الشبكة ليمكّن الزبون من استعادة بياناته لاحقاً.

خدمة الخزن الاحتياطي على السحابة وعلى الرغم من فوائدها، إلا أنّها الخزن تواجه عدة تحديات كبيرة. أحد أهم هذه التحديات هو عرض حزمة الاتصال بالإنترنت Internet Bandwidth، حيث أن عرض حزمة الاتصال بالإنترنت هي أقل بشكل ملحوظ من عرض حزمة الاتصال في الشبكة المحلية (LAN). وعليه فإن عملية الخزن الاحتياطي والاستعادة أكثر بطأً وتكلفة من الخزن الاحتياطي التقليدي المحلي on-site.

تحدٍ آخر، هو النافذة الكبيرة للخزن الاحتياطي Backup Window والتي تمثل الزمن اللازم لإرسال مجموعة محددة من البيانات إلى وجهة النسخ الاحتياطي، بسبب انخفاض عرض الحزمة بين المستخدم ومزود الخدمة والذي يحد من سرعة انتقال البيانات. على سبيل المثال، سيستغرق الخزن الاحتياطي لـ 1 تيرا من البيانات على نظام Amazon S3 أكثر من 14 يوماً في حال كانت سرعة الاتصال kbps800 .

أيضاً يواجه الخزن الاحتياطي على السحابة تحديات ناتجة عن مساحات التخزين الهائلة والتكلفة الكبيرة لإدارة البيانات الناتجة عن الزيادة السريعة للبيانات المخزنة في جانب مزود الخدمة.

3- فوائد الخزن السحابي Cloud Storage Advantages: [7]

- ❖ عند استخدام التخزين السحابي، سيدفع المستخدم مقابل المساحة التخزينية التي سيستخدمها فقط.
- ❖ البيانات المخزنة على السحابة يمكن الوصول إليها بسرعة وفعالية بأي وقت ومن أي مكان.
- ❖ حماية البيانات بشكل أفضل في حال حدوث الكوارث.
- ❖ مسؤولية تجاوز حالات الفشل والمشاكل التخزينية الناشئة عن عطب العتاد تقع على عاتق مزود الخدمة وليس على المستخدم مما يؤدي إلى تجنب انقطاع الخدمة.
- ❖ يوفر الخزن السحابي سعات تخزينية افتراضية غير محدودة.
- ❖ يساعد مزود الخدمة المستخدمين على تحقيق أفضل أداء من خلال موازنة أعباء الأعمال workloads.
- ❖ يعطي الخزن السحابي رؤية موحدة لاستخدام السعات التخزينية.

4- فعالية الخزن الاحتياطي على السحابة:[6]

عند تقييم الأداء يبرز تحدٍ جديد هو زمن الاستعادة Recovery Time Objective (RTO) والذي يحدد زمن التوقف عن العمل الأعظمي الذي ينتظره المستخدم قبل استقبال بياناته المستعادة بعد حدوث الكارثة. وهو ما يمثل تحدياً أكبر لمزودي خدمة الخزن الاحتياطي على السحابة نظراً لعرض حزمة الاتصال بالـ WAN المنخفض نسبياً.

وعليه، فإنّه من المهم والضروري أن يتم اعتماد أساليب شبكية ذات فعالية للاتصال ببيئة الخزن الاحتياطي على السحابة لتحسين أداء كل من عملية الخزن والاستعادة لتكون بذلك هذه الخدمة عملية وفعالة من حيث التكلفة.

حاولت الكثير من الأبحاث التطرق لهذه المشكلة من خلال محاولة تقليل كمية البيانات التي يتم نقلها خلال عملية الخزن الاحتياطي، أحد الحلول المقترحة هو تطبيق: إزالة التكرار Deduplication.

5- مفهوم إزالة التكرار:[5]

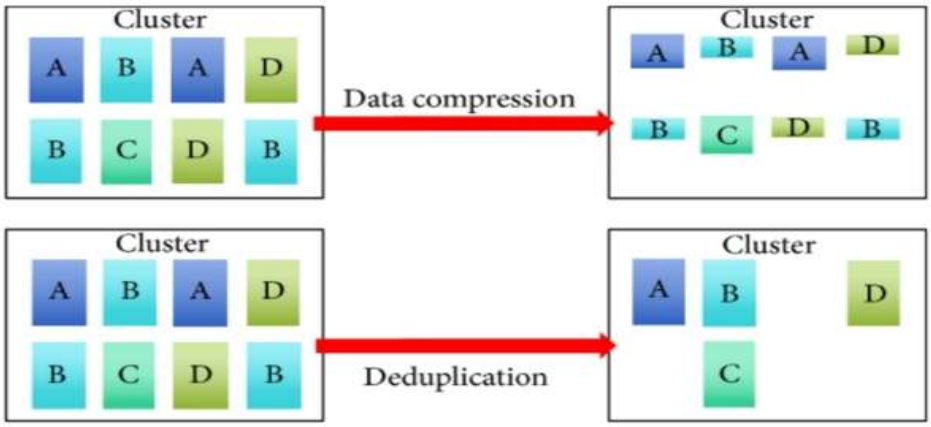
تعتمد تقنية إزالة التكرار في البيانات على التعرف إلى البيانات المكررة ومن ثم إزالة هذا التكرار لتقليل الحاجة إلى نقل هذه البيانات أو تخزينها. إزالة التكرار تقنية فعالة في أَمثلة استخدام المساحة التخزينية، ويمكن أن تخفض بشكل كبير من كميات البيانات وبالتالي تخفض من استهلاك الطاقة وعرض الحزمة التخزينية في مراكز البيانات السحابية.

خلال عملية إزالة التكرار، يتم تحديد البيانات المكررة وتخزين نسخة واحدة منها مع مؤشرات إلى النسخة الفريدة الباقية؛ وبهذا يتم حذف البيانات الفائضة. من أكثر تقنيات إلغاء التكرار شيوعاً تجزيء البيانات إلى قطع Chunks من البيانات غير المتداخلة. حيث يتم حساب البصمة التشفيرية Hash Value لكل قطعة منها باستخدام تابع تلبيد Hash Function –مثل الـ SHA-1– وتُخزّن البصمة لكل قطعة ضمن جدول Hash Table يعتبر فهرساً للقطع. كل قطعة مخزّنة في نظام التخزين لها بصمة فريدة ضمن فهرس القطع، فلتحديد فيما إذا كانت القطعة مخزنة سابقاً في النظام أو لا، يتم البحث عن البصمة التشفيرية للقطعة الواردة في فهرس القطع أولاً. إذا وُجد تطابق ما، يخزن النظام مرجعاً إلى القطعة الموجودة بدلاً من تخزينها مرة أخرى، وإلا فإن القطعة الجديدة الواردة تعتبر فريدة؛ تُخزّن في النظام وتُضاف بصمتها التشفيرية إلى فهرس القطع.

5-1- الفرق بين Data compression و Data Duplication:[8]

تقنية ضغط البيانات Data compression هي تقنية منتشرة على نطاق واسع لتحسين استخدام المساحات التخزينية ونقل البيانات عبر الشبكة. تقوم بالمبدأ على تقليل عدد البتات bits الممثلة للبيانات مما سيقلل حجمها عن الحجم الأصلي. هذا الضغط سينتج عنه توفير في المساحة التخزينية وتقليل كمية البيانات المنقولة عبر الشبكة وبالتالي تخفيض زمن الإرسال.

كما هو موضح في الشكل رقم (2) لا يهتم الضغط بكون البيانات مكررة أو لا حيث يعمل على تخفيض تمثيل البيانات كاملةً من حيث الحجم، على عكس إزالة التكرار التي تعمل على توفير المساحة التخزينية من خلال الاحتفاظ بنسخة واحدة من البيانات المكررة فقط.



الشكل (2): الفرق بين الضغط وإزالة التكرار

6- فوائد إزالة التكرار:[6]

- انخفاض متطلبات مساحات التخزين.
- انخفاض كلفة التخزين.
- تحسّن الأداء.
- زيادة فعالية استخدام الشبكة.
- إجراء عمليات النسخ الاحتياطي بفعالية أكبر نظراً لانخفاض كمية البيانات المراد نسخها.

7- خوارزميات التقطيع المعتمد على المحتوى Content-based Chunking:

تكون القطع الناتجة عن هذا النوع من الخوارزميات متغيرة الطول، ويعتمد طول القطع الناتجة على مدى مطابقة محتوى البيانات لشرط محدد مسبقاً يتم عند تحققه تحديد حد القطع Chunk Breakpoint.

لذلك فإنّ استخدام خوارزميات التقطيع المعتمد على المحتوى مفضّل في أنظمة النسخ الاحتياطي لملائمتها لعمليات الإضافة والحذف، سنستعرض تالياً بعض هذه الخوارزميات:

7-1- Two Thresholds Two Divisors-S (TDDD-S):[9]

تعرّف هذه الخوارزمية 5 متحولات هم:

- العتبة العليا Maximum threshold.
- العتبة الدنيا Minimum threshold.
- القاسم الرئيسي Main divisor
- القاسم الثانوي Second divisor
- متحول التبديل switchP.

وتوضح الخطوات التالية مبدأ عملها:

* تتحرك الخوارزمية بايتاً واحداً في كل مرة على طول الملف.

* إذا كان حجم القطعة من آخر حد قطع وحتى الموقع الحالي أكبر من العتبة الدنيا Minimum Threshold تبدأ عندها الخوارزمية البحث عن موقع حد قطع جديد قبل الوصول إلى العتبة العليا Maximum Threshold. نلاحظ هنا وجود 4 حالات:

1- إذا وجد حد قطع باستخدام القاسم الرئيسي Main Divisor يتم اعتماده مباشرة.

2- في حال الفشل في تحقيق الشرط السابق يعاد الاختبار باستخدام القاسم الثانوي Second Divisior. في حال الحصول على تطابق يتم الاحتفاظ بحد القطع الناتج عنه كحد قطع احتياطي، ومن ثم الإزاحة بايتاً واحداً نحو الأمام والبحث عن قطعة جديدة تحقق تطابق مع القاسم الرئيسي Main Divisor قبل وصول حجم القطعة المختبرة إلى العتبة العليا Maximum Threshold.

3- في حال وصول حجم القطعة إلى العتبة العليا Maximum Threshold ولم ينتج لدينا أي تطابق مع القاسم الرئيسي Main Divisor نعتمد حد القطع الاحتياطي الناتج عن القاسم الثانوي Second Divisior.

4- في حال وصول حجم القطعة إلى العتبة العليا Maximum Threshold ولم ينتج لدينا أي تطابق مع القاسم الرئيسي Main Divisor وليس لدينا حد قطع احتياطي نعتمد حد القطع عند العتبة العليا Maximum Threshold.

أما بالنسبة لمتحول التبديل switchP فمهمته اختبار حجم القطعة المختبرة حالياً، فعندما يتجاوز حجمها قيمة switchP يأخذ القاسمان الرئيسي والثانوي قيماً جديدة كالتالي:

القاسم الرئيسي الجديد = القاسم الثانوي القديم.

New Main Divisor = Old Second Divisor.

القاسم الثانوي الجديد = نصف قيمة القاسم الثانوي القديم.

New Second Divisor = ½ Second Divisor.

وعند إيجاد حد قطع تبعاً لهذه القيم الجديدة، تتم العودة إلى القيم الأصلية.

2-7- Asymmetric Extremum Chunking Algorithm (AE):[10]

في خوارزمية AE يعطى للبايت سمتين أساسيتين هما رقم الموقع والقيمة.

رقم الموقع: لكل بايت في الدخل Data Stream رقم يدل على موقعه في هذا الدخل فرقم الموقع للبايت ذي الترتيب n في الدخل هو n حيث أن (طول الدخل $\geq n \geq 1$)

القيمة: تعامل كل مجموعة S من البايتات/المحارف المتعاقبة في الدخل كقيمة واحدة Value، حيث ترتبط هذه القيمة برقم الموقع لأول بايت من هذه المجموعة S. وبذلك يكون لجميع بايتات الدخل قيمة مرتبطة بها ما عدا آخر S-1 بايت.

ملاحظة:

القيمة الحديّة Extreme Value في خوارزمية AE قد تكون كبرى أو صغرى ولكننا سنعتمد القيمة الحديّة العظمى Extreme Maximum Value.

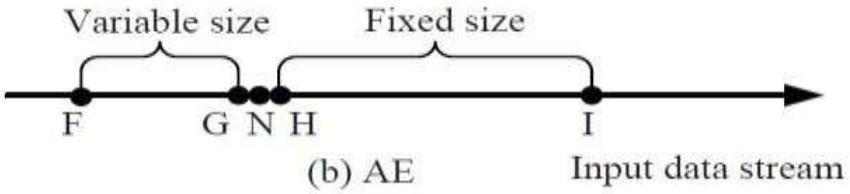
مبدأ عمل الخوارزمية:

تبدأ الخوارزمية عملها من أول بايت في الدخل (أو من أول بايت بعد آخر حد قطع) كما هو مبين في الشكل رقم (3) لتبحث عن أول بايت يحقق الشرطين التاليين:

1. أن تكون قيمته أكبر من قيم جميع البايتات السابقة له.
2. أن تكون قيمته ليست أصغر من قيم جميع البايتات الموجودة في النافذة اليمنى له.

يسمى أول بايت ينطبق عليه الشرطان السابقان النقطة العظمى Maximum Point. يضمن تحقق هذين الشرطين أن النقطة العظمى Maximum Point هي قيمة محلية عظمى Local Maximum Value.

في حال حصولنا على نقطة عظمى، تعتبر AE البايت الأخير في النافذة اليمنى لها (مجموعة البايتات المتعاقبة مباشرة ذات الطول W) حداً للقطع.



الشكل (3): خوارزمية AE

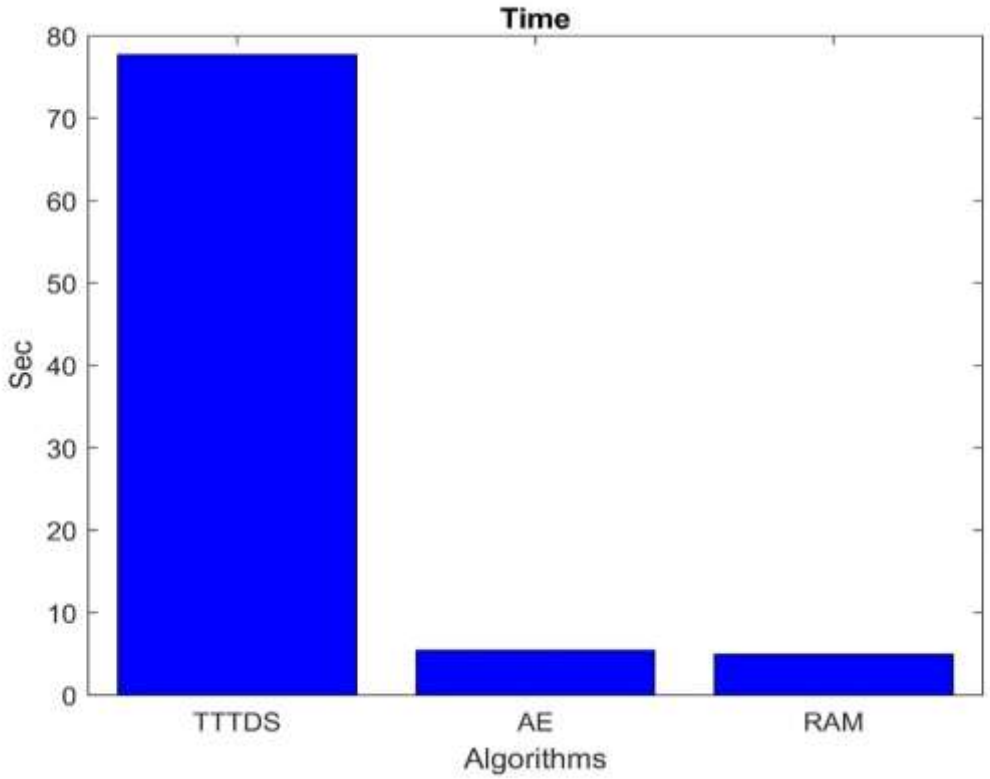
3-7- Rapid Asymmetric Maximum (RAM) [11]:

تشابه في مبدأ عملها خوارزمية AE، إلا أنها تطبّق النافذة ذات الطول الثابت في البداية؛ حيث يتم تحديد القيمة العظمى Maximum Value في هذه النافذة. ليتم بعدها مقارنة كل بايت بعد النافذة بالقيمة العظمى، حيث يحدد حد القطع عند وجود بايت أكبر من القيمة العظمى الموجودة في النافذة ذات الطول الثابت.

النتائج والمناقشة:

قام الباحثون في [12] بمقارنة أداء أربعة من خوارزميات التقطيع المعتمد على المحتوى وهي: خوارزمية النافذة المنزلقة BSW، خوارزمية TTTD ، خوارزمية TTTD-S ، وخوارزمية AE. وقد خلصوا إلى تفوق خوارزمية AE من حيث زمن التنفيذ ومعدل إزالة التكرار. سنقوم في هذه المقالة بمقارنة هذه النتائج مع خوارزمية RAM بناءً على: زمن التنفيذ – نسبة إزالة التكرار – متوسط حجم القطع الناتجة. سيتم التنفيذ باستخدام الـ Dataset ذات الحجم الكلي 1.1 GB المذكورة في الملحق رقم 1 وهي نفس الـ Dataset في [12].

1- زمن التنفيذ:



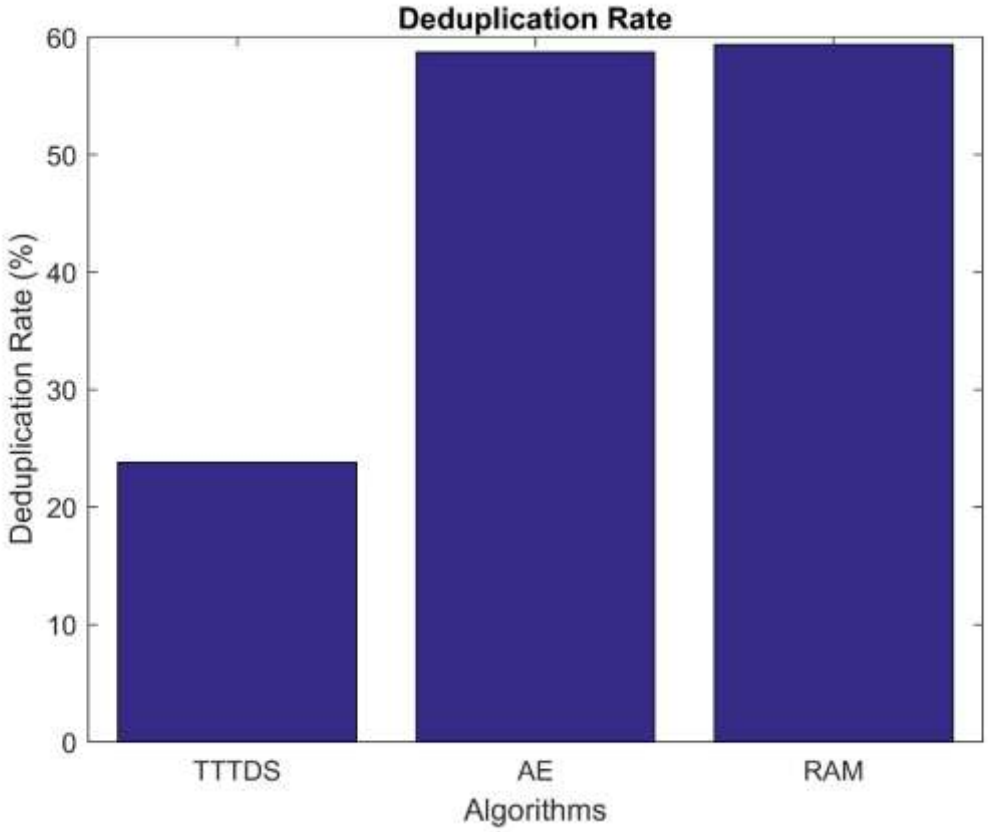
الشكل (4): مقارنة زمن التنفيذ الذي استغرقته الخوارزميات الثلاث

يوضح الشكل رقم (4) السابق الزمن الذي استغرقه تنفيذ الخوارزميات الثلاثة، حيث حققت خوارزمية RAM أقل زمن تنفيذ (3.07 ثانية) تلتها AE (5.41 ثانية) وأخيراً خوارزمية TTTD-S (77.7 ثانية)

يعود سبب تفوق خوارزميتي RAM و AE في زمن التنفيذ لكونهما تعتمدان آلية عمل مختلفة لا تتطلب حساب بصمة تشفيرية (والتي تعتبر مكلفة زمنياً) في تعيين حدود القطع كما هو الحال في خوارزمية TTTD-S القائمة على مبدأ النافذة المنزلقة.

بينما تتفوق خوارزمية RAM على خوارزمية AE في زمن التنفيذ لأنها تقلل من عدد عمليات المقارنة اللازمة كونها تحدد القيمة العظمى ضمن النافذة الثابتة في البداية ليتم بعدها مقارنة كل بايت بعد النافذة بهذه القيمة العظمى، على عكس خوارزمية AE التي يتم مقارنة القيمة العظمى فيها بما قبلها وما بعدها.

2- نسبة إزالة التكرار:



الشكل (5): نسبة إزالة التكرار في كل من الخوارزميات الثلاث

تحسب النسبة المئوية لإزالة التكرار كما يلي:

$$\text{نسبة إزالة التكرار} = \frac{\text{عدد القطع المكررة}}{\text{عدد القطع الكلي}} * 100$$

يبين الجدول (1) عدد القطع الكلي الناتج عن كل خوارزمية بالإضافة إلى القطع المكررة منها وغير المكررة:

الجدول (1): عدد القطع الكلي الناتج عن كل خوارزمية

الخوارزمية	عدد القطع الكلي	عدد القطع المكررة	عدد القطع غير المكررة
TTTD-S	1183527	282941	900586
AE	869973	510710	359263
RAM	496439	294655	201784

يوضح الجدول السابق عدد القطع الكلي الناتج عن كل خوارزمية ممثلاً بالعمود الأول، يليه عدد القطع غير المكررة ممثلاً بالعمود الثاني، في حين يمثل العمود الأخير عدد القطع المكررة. نلاحظ أن خوارزمية TTTD-S أعطت أكبر عدد قطع كلي / 1183527 / قطعة منها / 282941 / قطعة مكررة، تليها خوارزمية AE بـ / 869973 / قطعة منها / 510710 / قطعة مكررة، لتحل ثالثاً خوارزمية RAM بـ / 496439 / قطعة منها / 294655 / قطعة مكررة.
يعود ذلك لأنّ خوارزمية TTTD-S وبإضافتها للمتحول switchP خفّضت من عدد القطع التي يتم فيها القطع عند الحد الأعظمي Maximum Threshold لعدم تحقيقها الشرط مع Main Divisor وعدم وجود حد قطع احتياطي باستخدام Second Divisor، مما قلّل من أحجام القطع الناتجة عن TTTD-S وأدى بالتالي إلى زيادة عددها. تمكنت خوارزميتا AE وRAM من كشف تكرارات أكبر من TTTD-S وذلك لكونهما لا تفرضان قيداً على الحد الأعظمي لحجم القطعة Maximum Threshold كما هو الحال في TTTD-S، فهما تعينان حد القطع عند تحقق شروطهما فقط؛ مما يفسر قلة عدد القطع بالإضافة إلى ارتفاع نسبة التطابقات.
بناءً على الجدول السابق فقد حققت خوارزمية RAM أعلى نسبة إزالة تكرار هي 59.35% تلتها خوارزمية AE بفارق بسيط بنسبة 58.7% وأخيراً خوارزمية TTTD-S بنسبة 23.8%.

3- متوسط حجم القطع:

يبين الجدول (2) متوسط حجم القطع الناتجة عن الخوارزميات المدروسة، حيث سجلت خوارزمية TTTD-S أقل متوسط لحجم القطع لأنّ عدد القطع الناتجة عنها كان الأكبر، بينما سجّلت خوارزمية RAM أكبر متوسط للحجم كون عدد القطع الناتجة عنها كان الأقل بين الخوارزميات الثلاث.

الجدول (2): متوسط حجم القطع الناتج عن كل خوارزمية

الخوارزمية	متوسط حجم القطع
TTTD-S	950.7
AE	1293.4
RAM	2796.6

الاستنتاجات والتوصيات:

الاستنتاجات:

من التجارب السابقة يتبين لدينا:

- ✓ تفوّق خوارزمية RAM على باقي الخوارزميات من حيث إزالتها للتكرار بنسبة 59.35% تليها خوارزمية AE بنسبة 58.7%.
- ✓ تفوّق خوارزمية RAM أيضاً من ناحية زمن التنفيذ، حيث سجّلت زمناً كلياً قدره 3.07 ثانية تليها أيضاً خوارزمية AE بزمن قدره 5.41 ثانية.
- ✓ سجّلت خوارزمية RAM أقل عدد قطع كلي من بين الخوارزميات الأربعة 496439 قطعة بينما سجّلت خوارزمية TTTD-S أكبر عدد قطع وهو 1183527 قطعة.
- ✓ نتج عن خوارزمية RAM أكبر متوسط لحجم القطع (2796.6B) في حين أن أصغر متوسط للحجم سُجل لخوارزمية TTTD-S وهو (950.7B).

الأعمال المستقبلية:

- ✓ دراسة إمكانية تطوير خوارزمية RAM بفرض قيود على حجوم القطع الناتجة عنها.
- ✓ دراسة إمكانية تطبيق خوارزميات Machine Learning لتحديد قيمة متحول التبديل switchP بشكل متغير في خوارزمية TTTD-S.

الخاتمة:

قدمنا في هذا البحث مقارنة بين ثلاث خوارزميات تقطيع معتمد على المحتوى، أظهرت فيه خوارزمية RAM تفوقاً من ناحية تقليل عدد العمليات الحسابية اللازمة للتقطيع وبالتالي تقليل زمن التنفيذ مع محافظتها على نسبة إزالة تكرار مرتفعة مما يجعل أداءها مناسباً للاستخدام في الأجهزة المحمولة والـ IoT وتطبيقاتها التي تحتاج سرعة في الأداء وتوفيراً في المساحة التخزينية.

References:

[1] J. F. Gantz and S. Minton, "The Diverse and Exploding Digital Universe An Updated Forecast of Worldwide," 2011.

[2] U. H. Rao and U. Nayak, "Data Backups and Cloud Computing," in *The InfoSec Handbook: An Introduction to Information Security*, Berkeley, CA: Apress, 2014, pp. 263–288.

[3] "Data created worldwide 2010-2024 | Statista." https://www.statista.com/statistics/871513/worldwide-data-created/ (accessed Sep. 06, 2020).

[4] J. Malhotra, "A Survey and Comparative Study of Data Deduplication Techniques," vol. 00, no. c, pp. 0–4, 2015.

[5] M. P. Taware and S. B. Rathod, "A Survey Paper on Deduplication Approaches," vol. 2, no. 6, pp. 239–242, 2015.

[6] P. Neelaveni and M. Vijayalakshrni, "A Survey on Deduplication in Cloud Storage," vol. 13, pp. 320–330, 2014.

[7] S. L. Obrutsky, "Cloud Storage : Advantages , Disadvantages and Enterprise Solutions for Business," 2009.

[8] R. S. Chang, C. S. Liao, K. Z. Fan, and C. M. Wu, "Dynamic deduplication

decision in a hadoop distributed file system," *Int. J. Distrib. Sens. Networks*, vol. 2014, 2014, doi: 10.1155/2014/630380.
[9] T. R. Nisha, S. Abirami, and E. Manohar, "Experimental Study on Chunking Algorithms of Data Deduplication System on Large Scale Data," doi: 10.1007/978-81-322-2674-1.
[10] Y. Zhang *et al.*, "AE: An Asymmetric Extremum content defined chunking algorithm for fast and bandwidth-efficient data deduplication," *Proc. - IEEE INFOCOM*, vol. 26, no. April 2016, pp. 1337–1345, 2015, doi: 10.1109/INFOCOM.2015.7218510.
[11] M. E. Student, "Scheduling For Workflows With Security- Sensitive Intermediate Data By Selective Tasks," vol. IV, no. V, pp. 1–8, 2018.

[12] م. رهام كمال صقر. تقييم أداء خوارزميات التقطيع المعتمد على المحتوى في النسخ الاحتياطي في السحابة (رسالة أعدت لنيل درجة الماجستير). جامعة تشرين – كلية الهندسة المعلوماتية – قسم النظم والشبكات الحاسوبية 2020.