ارائه مدل بهینه پایگاه‌داده NoSQL برای شبکه اینترنت اشیاء

محمد اسماعیلی [۱*]، جمشید باقر زاده [۲]
۱.دانشجوی کارشناسی ارشد دانشگاه ارومیه mohammadesmaili595@yahoo.com
۲.استادیار دانشگاه مهندسی کامپیوتر دانشگاه ارومیه j.bagherzadeh@urmia.ac.ir

چکیده : پایگاه‌های داده غیرساختارمند طی سالیان اخیر تحولی نوین در جهان فنآوری اطلاعات به وجود آورده و با عبور از مرزهای نامتعارف، معضلات لاینحل کلان داده در زمینه دسترسی بالا به اطلاعات، امکان تعدد و توزیع پذیری مراکز تولید محتوا، تعامل چندگانه کاربران، مولدان محتوا، سرورهای توزیع شده، امکان تبادل وسیع تمامی فرمت های متصور داده در میان پایگاه داده، تجمیع داده‌های ناسازگار و فاقد ساختار در درون سیستم‌های واحد و دهها امکان جدید را فراهم نمایند. به موازات نیز با ورود گوشی‌های هوشمند و تجهیز ریزترین عناصر به قطعات هوشمند و حسگرهای دارای ارتباطات شبکه‌ای، به صنعت IT، تلاقی این دو رهیافت بشر منجر به شکل گیری کلان داده در جهان اینترنت اشیاء شده است. در این پژوهش ما به دنبال آن هستیم که مدل‌های مختلف پایگاه داده را تحلیل و ارزیابی نموده و به مدل‌های مطلوب برای شبکه اینترنت اشیا برسیم. این مدل باید بتواند برای داده‌های سری زمانی بهینه باشد. در نتیجه بررسی مدل‌ها، دومدل کلید مقدار و ستون عریض قابلیت‌های مدنظر IoT ما را داشتند که دو محصول برتر و شاخص ریاک و کاساندرا را انتخاب نموده و با پیکره بندی ساختار کلاستری برای این دو سیستم، برنامه خود را در محیط جاوا طراحی و برای عملیات‌های پایه در پایگاه داده زیر بار تست‌های سنگین قرار دادیم

کلید واژه‌ها: پایگاه‌های داده غیرساختارمند، ریاک، کاساندرا، اینترنت اشیاء، حسگر

۱- مقدمه

در ساده‌ترین و عامیانه‌ترین تعبیر و تفسیر، هر شی از جهان فیزیکی و یا جهان مجازی که قابلیت شناسائی شدن و یکپارچه شدن با شبکه‌های ارتباطی را داشته باشد می تواند عضوی از شبکه اینترنت اشیا بوده و در فرآیند مبادله داده سهیم گردد. این اشیا می توانند تبلت‌ها، عناصر هوشمند نظیر لوازم خانگی دارای سیستم عامل و یا حسگرها باشند. شبکه‌های رایانه‌ای علاوه بر توسعه جغرافیائی و فیزیکی، درکنار بسط شبکه‌های بی‌سیم در قلمرو شبکه حسگر و اینترنت اشیا (IoT) ابعاد جدید یافته‌اند و عینیت یافتن واقعیت ملموس کلان داده [۱]که منجر به پدیدار شدن مشکلات و معضلات در فناوری اطلاعات گردیده است. این مساله سبب شده تا پایگاه‌های داده غیرساختارمند (NoSQL) [۲]را در سطح اول دانش فناوری اطلاعات برای مهار شکاف‌های فناوری و نیازمندی قرار گیرد.در این میان پایگاه‌های داده غیرساختارمند توانسته‌اند همزمان داده‌های نیمه ساختارمندو غیرساختارمند را در کنار ساختارهای داده سنتی پشتیبانی کرده و با قابلیت‌هایی نظیر سیستم فایل توزیع شده، خوشه بودن سرویس‌ها، دسترسی بالا و انعطاف در طراحی بسترهای جدیدی برای توسعه فراهم نماید. قابلیت‌های متنوع و گسترده پایگاه‌های داده NoSQL سبب گردیده تا هیچگاه نتوان به قطع یقین و در مواردی حتی به صورت تقریبی بر بهینه بودن مدلی از این نوع پایگاه‌های داده در یک محیط رای داد. از این رو به دلیل تنوع بالای مدل‌های مختلف ارائه شده، و همچنین نیازمندی‌ها، مدل ارتباطی، مدل طراحی جریان داده و ماهیت داده‌های سیال در شبکه، سیستم های NoSQL پاسخ‌های متفاوتی در اکوسیستم های متنوع ارائه می‌دهند.

از این رو ما در این پژوهش و پایان نامه به دنبال ارائه مدل بهینه و مطلوب برای این شبکه هستیم. در گام اول هدف ما این است که

1 Big Data
2 Not Only SQL

ابتدا مدل‌های پایگاه داده غیرساختارمندمطلوب برای این سیستم را بشناسیم، سپس از میان مدل‌های مزبور مدلی که توانایی عملیات بر روی داده سری زمانی را داشته باشد را مدنظر قرار دهیم و در نهایت با بررسی تجربیات دانشمندان حوزه فنآوری اطلاعات، به بررسی سایر ابعاد توانمندی و قابلیت های غیرساختارمندها رفته و ویژگی-های کیفی این پایگاه‌های داده را به دست آوریم. هدف بعدی ما رسیدن به آستانه مطلوب در حجم بالای عملیات‌های پایگاه‌داده نظیر درج و بروز رسانی داده‌های سری زمانی در مقیاس وسیع هستیم.در این مسیر با اتکا به پژوهش‌های معتبر، اولا تلاش خواهیم کرد تا به پیش فرض هایی برای انتخاب مدل پایگاه داده دست یابیم و در ادامه براساس ساختار داده‌ای مدنظرمان، به مدل بهینه و کارآمد پایگاه داده دست یابیم. رسیدن به سرعت بهینه در عملیات پایه پایگاه‌داده به ویژه درج، خواندن و بروزرسانی با تکیه بر قابلیت های ویژه پایگاه‌های داده غیرساختارمندنظیر ساختار خوشه‌ای، توزیع پذیری در عین حفظ قابلیت انعطاف در برابر خرابی‌ها و رفع خرابی با تکیه بر توانمندی منحصر به فرد پایگاه‌های داده غیر ساختارمنداهداف کلی ما هستند.

۲-چالش میان پایگاه داده ساختارمند و غیر ساختارمند

زاهد [1] و بکر و رابر [2] و همکاران در سال ۲۰۱۲ و ۲۰۱۳ بر روی سیستم‌های میان‌افزار پایگاه داده‌ای پژوهش‌های مفصلی انجام دادند و ضمن اثبات کارایی برتر پایگاه‌های داده غیرساختارمند، استنتاج کردند که برخلاف پایگاه‌های داده ساختاری، در پایگاه‌های داده غیرساختارمند، ویژگی‌هایی همانند معماری و مدل داده، عدم سازگاری فضاهای کاری و تحول در فضاهای عملیاتی، تنوع بالا و توانمندی وسیع انواع سیستم‌های پایگاه‌های داده غیرساختارمند، دارند که تفاوت بنیادی با مدل‌های پیشین داشته و از این رو دست طراحان برای طراحی مدل‌های نوین مبتنی بر کلان داده را باز گذاشته و امکان تحولات بنیادین در طراحی سامانه‌های فنآوری اطلاعات در مقیاس وسیع را فراهم می‌نماید.

ونکارتمان[6] و همکاران با اشاره به پژوهش دانشمندان صنعت فناوری اطلاعات در رابطه با چالش‌های حاکم بر کلان داده نظیر سرعت، تنوع، حجم و پیچیدگی بالای داده‌ها که سبب بن بست در فناوری مبتنی برRDBMS شده است، به تحقیق پیرامون کارآیی پایگاه‌های داده ساختارمند و غیر ساختارمند پرداختند و شاخص-های برتری هرکدام را مورد بررسی قرار دادند.

بربنر [3] و کلین [4] سال ۲۰۱۴ بر روی سیستم‌های پایگاه داده غیرساختارمند در مدل کلان‌داده مطالعه و ارزیابی انجام دادند و در زمینه خوشه بندی و طراحی سیستم‌های نرم افزاری پایگاه‌داده به نتایجی دست یافتند که حاکی از برتری پایگاه‌های داده غیرساختارمندبر مدل‌های سنتی و ساختارمنداست.

نایاک [16] و همکاران همچنین نتیجه گرفتند که انعطاف ریاک و کاساندرا در برابر زبان های برنامه نویسی و امکان برنامه نویسی مستقیم در این سیستم‌ها را یکی دیگر از دلایل برتری مدل‌های غیرساختارمند بر ساختارمند می‌باشد.

چوئی [7] ، بازار و لوسیف [8] ، منیرالزمان [12] و پراساد و همکاران [10] بر روی ویژگی‌های کیفی و ماهیتی پایگاه‌های داده NoSQL پژوهش نمودند و به نتایج ملموسی در مورد برتری برخی مدل‌های پایگاه داده دست یافتند. در این پژوهش‌ها، انواع سیستم‌های مدیریت پایگاه‌داده مورد مطالعه و سنجش قرار گرفته و در زمینه دسترس پذیری بالا ۳، تحمل افرازپذیری (Partition Tolerance)، کنترل همروندی در مقیاس وسیع و تحمل خرابی ۴، کارآیی آنها مورد سنجش قرار گرفت.

پژوهش دوم نیز بر روی پایگاه‌های داده سندگرا صورت گرفت که در نتیجه آن اثبات شد که سندگراهایی نظیر مونگو دی.بی۵، , کوچ دی.بی، ریاک، دینامو دی.بی۶، ویژگی های کارآیی بالا، مقیاس پذیری متغیر و غالبا بالا، انعطاف بالا، پیچیدگی کم و قابلیت عملیاتی متغیر و غالبا پایین دارند.

سومین پژوهش تیم تحقیقاتی مزبور نیز بر روی پایگاه‌های داده غیرساختارمند گرافی بود که در این میان Neo4j ، اورینت دی.بی[۷]، ,گراف[۸] ، ,تیتان[۹]مورد پژوهش قرار گرفتند که در نتیجه آن، مدل گرافی، کارآیی متغیر و پایین تر از سه مدل دیگر داشته و همچنین مقیاس پذیری ضعیف تر از سه مدل قبلی ، انعطاف بالا،

[3] High Availability
[4] Fault Tolerance
[5] MongoDB
[6] DynamoDB
[7] Orient DB
[8] Giraph
[9] Titan

پیچیدگی بالا و قابلیت عملیاتی بر مبنای تئوری گراف را دارا هستند.

پراساد و همکاران [10] همچنین به بررسی نقاط قوت وتوانمندی پایگاه‌های داده غیر ساختارمند برجسته از هر چهار مدل پرداخته و به نتایج دیگری دست یافتند. نتایج حاصله از پژوهش‌های پراساد و همکاران که در سال ۲۰۱۴ صورت گرفته، طی سالیان اخیر به عنوان مرجعی برای معیار و شاخص پژوهش‌های عملیاتی در اغلب دانشگاههای جهان قرار گرفته و به عنوان یکی از بالاترین ارجاعات در صنعت فنآوری اطلاعات، در زمینه پایگاه‌های داده مورد استفاده قرار گرفت.

در این پژوهش، سرعت، تنوع، حجم و پیچیدگی ساختار داده را از مولفه‌های اصلی در تحلیل کلان داده اعلان و نتیجه گرفتند که ریاک و کاساندرا در دسترس پذیری بالا، شاخص‌های بهتری دارند و ریاک در این میان در رده اول و کاساندرا در رده دوم می باشد. Google Big Table و Couch DB نیز در رده های سوم و چهارم قراردارند.

در زمینه تحمل افرازپذیری[10] نیز ریاک و کاساندرا عملکرد خوب و قابل قبولی دارند. در این بررسی مونگو در رده اول تحمل افرازپذیری بوده و کاساندرا بعد از مونگو در رده دوم قرار دارد. ریاک بعد از Google Big Table و Couch DB در رده پنجم این قابلیت قرار گرفته است. همچنین پایگاه داده نوظهور Hbase که شاخص های بالائی در توانمندی و سرعت دارد در رده ششم بعد از ریاک قرار گرفته است.

در زمینه تشابه پایگاه‌های داده غیرساختارمند به ساختارمند نیز مونگو، ردیس، Hbase نیز سنخیت‌هایی با مدل‌های ساختارمند داشته و ویژگی سازگاری و قفل کنترل هم روندی[11] را دارا می‌باشند که در این خصیصه‌ها به دلیل استقلال کامل ریاک و کاساندرا از سیستم‌های ساختارمند می‌باشد که سبب گردیده تا این دو محصول حضوری در ارزیابی نداشته باشند.

در این پژوهش ریاک توانمندی بسیار بالائی در تحمل خرابی به وسعت شبکه داشته و در این میان رقیبی ندارد. ریاک با تکیه بر سیستم گره‌های مجازی در ساختار حلقه‌ای ذخیره سازی داده‌ها بر روی خوشه، امکان بی بدیلی برای راه اندازی خرابی در مقیاس

پایگاه‌های داده NoSQL دارای بهترین نتیجه	ویژگی
Riak, Cassandra, Google Big Table, Couch DB	دسترسی بالا
MongoDB, Cassandra, Google Big table, CouchDB, Riak, Hbase	تحمل افراز پذیری
Google Big table	مقیاس پذیری بالا
MongoDB, Google Big Table, Redis, Hbase	سازگاری
MongoDB	Auto-Sharding
MongoDB, Redis, Cassandra	فرکانس بالای درج اطلاعات
Riak	رفع خرابی در مقیاس کل شبکه
Riak, Dynamo, CouchDB, Cassandra, Google Big Table	کنترل همزمانی (MVCC)
MongoDB, Redis, Google Big Table	کنترل همزمانی (Locks)

(جدول ۱ مقایسه پایگاه‌های داده غیرساختارمند دارای قابلیت‌های پایه نظیر تئوری CAP)

وسیع دارد. این ویژگی ریاک در میان تمامی پایگاه‌های داده غیرساختارمند منحصر بفرد اعلام شده است.

همچنین در زمینه کنترل همروندی در مقیاس وسیع نیز ریاک در رده اول قرارگرفته و کاساندرا بعد از Couch DB و Dinamo در رده چهارم قرار گرفته است.

۲- مدل پیاده‌سازی طرح پیشنهادی

برای بررسی رفتارهای ریاک و کاساندرا در برخورد با حجم وسیع و گستره کلان داده در محیطی شبیه محیط واقعی، اقدام به طراحی جدول پایگاه داده مرتبط و منطبق با مجموعه داده‌های دریافتی محتمل از یک شی اینترنتی با چندین حسگر هستیم. همچنین این پایگاه داده باید قابلیت سری زمانی داشته باشد. از این رو پایگاه داده‌مان را با جدولی که شامل داده‌هایی نظیر شناسه منحصر بفرد شی حاوی حسگر، برچسب زمانی، مختصات جغرافیایی، متغیرهای دما، ارتفاع، فشار، موقعیت مکانی، توصیف محیط و متغیرهایی حاوی آخرین وضعیت شی نظیر ظرفیت ذخیره انرژی و توان

[10] Partition tolerance
[11] Concurrenvy Control Locks

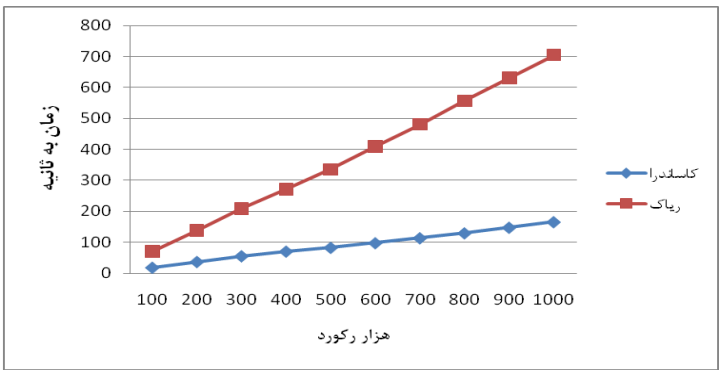
دریافتی آنتن در وضعیت بی‌سیم می‌باشد طراحی خواهیم کرد.در صورتی که فیلدهای مزبور در سیستم پایگاه داده ریاک طراحی گردد، لازم است تا شناسه به همراه مختصات مکانی در ترکیب با برچسب زمانی کلید ما را تشکیل دهند و در صورتی که در محیط کاساندرا این اقدام صورت پذیرد، برچسب زمانی، مختصات جغرافیایی و شناسه شی به عنوان حداقل کلیدهای ما خواهند بود. لازم است تا برای سنجش و مقایسه صحیح میان قابلیت های این دو پایگاه داده، زبان برنامه نویسی یکسانی نیز انتخاب گردد.

۳-نتایج پیاده سازی

حال پس از پیاده سازی طرح در ریاک و کاساندرا، به استخراج داده ها از فایل‌های ثبت سیستم و فایل متعلق به سنجش‌گر نانوثانیه پرداخته و نتایج را وارد جدول می کنیم. نتایج مزبور به عنوان داده-های اولیه برای تولید جداول تاخیر زمانی و گذردهی سیستم (Throughput) مورد استفاده قرار خواهند گرفت.

۳-۱ بررسی عملکرد مقایسه ای ریاک و کاساندرا در عملیات درج اطلاعات

طی تست ها و سنجش های انجام یافته در زمینه ورود ده هزار، بیست هزار، هشتاد هزار، صد هزار، دویست هزار، پانصد هزار و یک میلیون رکود، متوسط زمان پاسخ ریاک و کاساندرا به این عملیات به صورت نمودار در شکل ۱ می باشد.



(شکل ۱ نمودار مقایسه ای زمان پاسخ کاساندرا و ریاک در عملیات درج اطلاعات در حالت سه سروره)

مشاهده می‌شود که با افزایش حجم ورود اطلاعات از محدوده صدهزار رکورد به بالا، رشد تاخیر زمانی ریاک نسبت به کاساندرا شتاب گرفته و در این میان کاساندرا علیرغم وجود حجم بالای اطلاعات ورودی توانائی راه اندازی درخواست ها را با شیب بسیار ملایمی کنترل می نماید. در بررسی عملیاتی در سیستم خوشه ای، با بیشتر شدن حجم ورود اطلاعات به چند میلیون رفتار ریاک بدتر نیز می شود.

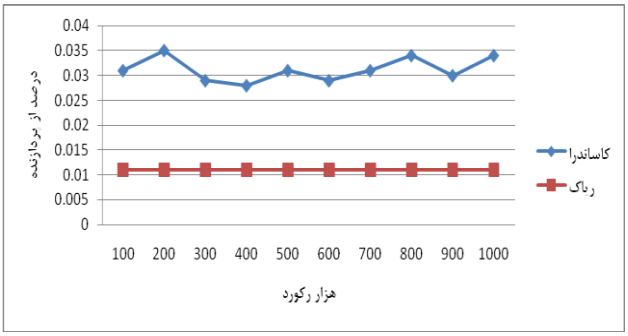
۳-۲ رفتار ریاک و کاساندرا در استفاده از حافظه

بررسی مقایسه‌ای از استفاده ریاک و کاساندرا از حافظه اصلی نشانگر آن است که ریاک با صرف نظر از زمان ورود انفجاری داده در ابتدای اجرای الگوریتم، و در بازه زمانی که دیسک به گلوگاه تبدیل میشود استفاده چندانی از حافظه اصلی ندارد ولی کاساندرا، افزونگی چندین برابری از داده‌های ورودی را در حافظه اصلی قرار داده و میان سیستم های کلاستر شده مبادله می نماید. کاساندرا علیرغم کاهش بار پردازشی و اتمام عملیات نیز همچنان نرخ استفاده خود از حافظه اصلی را تا اتمام نهایی فرآیند حفظ می نماید. این مساله در شکل ۲ آورده شده است.

(شکل ۲ نمودار مقایسه ای استفاده کاساندرا و ریاک از حافظه اصلی در عملیات درج اطلاعات در حالت سه سروره)

۳-۳ رفتار ریاک و کاساندرا در استفاده از پردازنده

بررسی رفتار ریاک و کاساندرا در استفاده از پردازنده حاکی از آن است که ریاک بار پردازشی خاص و محاسبات دامنه داری به سیستم تحمیل نمی نماید اما کاساندرا، استفاده بیشتری از پردازنده داشته و همزمان نیز مصرف حافظه بیشتری دارد. شکل ۳ نمودار مقایسه میزان استفاده کاساندرا و ریاک از پردازنده در عملیات درج اطلاعات در حالت سه سروره می‌باشد.



(شکل ۳ نمودار مقایسه ای استفاده کاساندرا و ریاک از پردازنده در عملیات درج اطلاعات در حالت سه سروره)

کاساندرا صف‌های متعدد و ساختمان داده‌اش را تا پایان آخرین مرحله از الگوریتم در حافظه اصلی نگه می‌دارد و هیچکدام از منابع تصرف شده‌اش را آزاد نمی کند و تلاش می‌نماید تا هیچ صفی میان ریزپردازنده و دیسک و میان حافظه اصلی و دیسک ایجاد نگردد زیرا صف در پشت دیسک یعنی سرباره و تبدیل شدن دیسک به گلوگاه و از این رو تمامی عملیات خود را میان پردازنده و حافظه انجام و در نهایت با نهایی شدن اسلوب درج اطلاعات در دیسک، به صورت صف دارای نرخی هماهنگ با نرخ نوشتن دیسک روانه دیسک می-نماید و در این میان حافظه اصلی را به روش متمرکزی مدیریت می‌نماید. در مقابل ریاک حافظه را به صورت نامتمرکز و توزیع شده مدیریت نموده و پس از آزاد شدن هر کدام از بخش‌های حافظه اصلی بلافاصله آن را به سیستم عودت می‌دهد که این مساله رجوع مکرر ریاک برای تصالحب مکرر حافظه را سبب می‌گردد. در این میان ریاک اجازه تشکیل صف در هر مرحله را داده و با استفاده کمتر از پردازنده و حافظه اصلی سبب تشکیل صف های متعدد به صورت توزیع یافته و مبادله مکرر آن می‌گردد که این مساله سبب بالا رفتن زمان پاسخ در ریاک نیز می‌گردد.

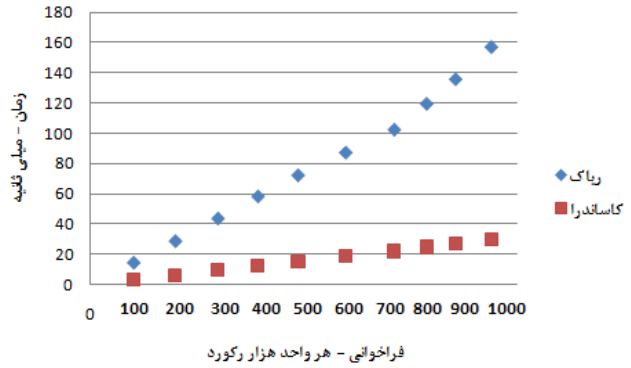
در نتیجه کاساندرا مصرف حافظه بالائی دارد و زمان استفاده بیستری نسبت به ریاک از پردازنده دارد. اما سرعت هفت برابری در درج اطلاعات سبب گردیده تا تفاوت ها معنا دار گردد و براساس پژوهش های سابق با افزایش گستره ساختار خوشه ای نرم استفاده از حافظه اصلی نیز با شیب ملایمی کاهش می‌یابد.

در این میان ریاک علیرغم مصرف معقول‌تر حافظه زمان پاسخ بالائی داشته و تنها مزیت ریاک اطمینان پذیری بالای آن می‌باشد که در صورت چشم پوشی کاربران از سرعت و تکیه بر ویژگی غلبه بر خرابی در گستره وسیع، شاید این سیستم مدیریت پایگاه داده بتواند پاسخگوی برخی مدل‌های نیازمندی باشد. در هر صورت کاساندرا، همانند آنچه در پژوهش‌های سابق بیان شده، سرعت قابل قبولی در درج اطلاعات و خواندن اطلاعات دارد و در این میان توانسته از رقبا پیشی گرفته و در کنار آن شاخص های اطمینان پذیری را نیز داشته باشد.

۳-۴ رفتار ریاک و کاساندرا در عملیات خواندن اطلاعات از پایگاه داده

با نگاهی به نمودارهای استفاده ریاک و کاساندرا از حافظه اصلی و پردازنده و مقایسه رفتارهای ریاک و کاساندرا در می یابیم که ریاک استفاده محدودی از حافظه اصلی داشته و عدم نوسان در بهره مندی از پردازنده نیز احتمالا حاکی از عدم وجود الگوریتم‌های متعدد چند لایه و سیاست های متنوع در برخورد با چالش های انفجاری و متعدد می باشد. بررسی نمودار رفتار ریاک و کاساندرا در قبال حجم بالای عملیات فراخوانی از پایگاه داده (SELECT) حاکی از آن است که ریاک ضعیف تر از کاساندرا بوده و این اختلاف در تست‌های با حجم پایین نظیر ده هزار رکورد کمتر بوده و با افزایش حجم فراخوانی و عملیات خواندن، از محدوده یکصد هزار، این تفاوت معنادار شده و با افزایش بیشتر حجم فراخوانی کاساندرا تاخیر خاصی از خود بروز نداده ولی در مقابل تاخیر عملیات ریاک روند صعودی به خود می‌گیرد.

تجربه سنجش‌های قبل حاکی از آن است که کاساندرا استفاده بهتری از حافظه اصلی و پردازنده دارد و برای برخورد با تمامی حالات متصور، الگوریتم‌ها و سناریوهای مطلوبی دارد که حجم سنگین بار ورودی را به بهینه‌ترین شیوه ممکن مهار می‌نماید.



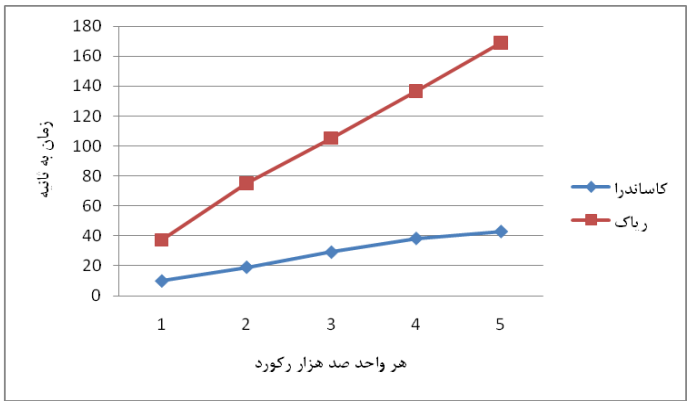
(شکل ۴ نمودار مقایسه ای زمان پاسخ کاساندرا و ریاک در عملیات درج اطلاعات در حالت سه سروره)

۳-۵ رفتار ریاک و کاساندرا در عملیات بروزرسانی اطلاعات از پایگاه داده

در بررسی قابلیت‌های ریاک و کاساندرا برای عملیات بروزرسانی، براساس اطلاعات شبیه سازی شده و وارد شده به سیستم، عملیات بروزرسانی سیستم با دامنه جانشانی و درج اطلاعات به میزان ۵۰٪ و عملیات خواندن و عدم تطبیق شرط برای ۵۰٪ دیگر انجام گردید که در این میان نمودار شکل ۵ نتایج این آزمایش برای صد هزار الی پانصد هزار رکورد می‌باشد.

همانند استدلالات و نتایج به دست آمده از مراحل قبل، کاساندرا استفاده وسیعی از حافظه اصلی و استفاده بیشتری از ریزپردازنده نموده و این مساله سبب می‌گردد تا کاساندرا در عملیات درج اطلاعات صفهای خود را میان پردازنده و حافظه اصلی تشکیل داده و در نهایت امر برای درج نهایی و به صورت مدیریت متمرکز اقدام به انتقال آن به دیسک نماید. در این میان ریاک به دلیل نامتمرکز بودن مدیریت حافظه و عدم استفاده بلندمدت از فضای گارانتی شده، هرچند در مصرف حافظه صرفه جویی می‌نماید ولی این مساله سبب مبادله مکرر صف‌ها میان سیستم های توزیع شده گردیده این مساله زمان پاسخ ریاک را به صورت صعودی افزایش می‌دهد.

در مقابل کاساندرا توانائی این را دارد که حجم سنگین کلان داده را با تکیه بر ظرفیت گارانتی شده بلند مدت مهار نماید.



(شکل ۵ نمودار مقایسه ای زمان پاسخ کاساندرا و ریاک در عملیات بروز رسانی اطلاعات در حالت سه سروره)

۵- تحلیل و جمع بندی

بررسی عملکرد این دو پایگاه داده نیز حاکی از آن است که علیرغم وجود تفاوت‌های تکنیکی در پیاده‌سازی و ساختار پایگاه داده، این تفاوت‌ها تنها ناشی از مکتب ذهنی و مدل فکری طراحان این دو سیستم بوده که با اهدافی در بلندمدت صورت گرفته است. اما پارامتر تعیین کننده این دو پایگاه داده که در خروجی واقعی، گذردهی عملیاتی و تاخیر زمانی عملیات‌های آن تاثیر بسزایی دارد، مدل جریان داده در درون سیستم، از پردازش در پردازنده تا درج در حافظه جانبی می‌باشد.

به صورت تجربی و علمی ثابت گردیده که ریاک از حافظه RAM به صورت توزیع شده استفاده نموده و مدیر کلاستر مدیریت غیرمتمرکز آن را بر عهده می‌گیرد. ریاک به هیچ عنوان از حافظه مجازی به دلیل سرباره زیاد آن در عملیات پایگاه داده استفاده نمی-نماید. علاوه بر آن دیسک جانبی همواره به عنوان گلوگاه در سیستم عمل نموده و با توجه به پردازش کلان داده توسط ریاک، هرگونه تکیه پایگاه داده به حافظه مجازی و Swappingسبب افت شدید کارآیی سیستم، افزایش شدید تاخیر اجرا و کاهش گذردهی سیستم می‌گردد.

این گونه استنباط می‌گردد که ریاک برای عملیات درج اطلاعات میان پردازنده و دیسک اقدام به ایجاد صف می نماید. همچنین ریاک به صورت توزیع شده منابع حافظه اصلی را تملک نموده و برای اجتناب از تصرف بیجای منابع سیستم به محض خروج رکوردها از حافظه اصلی اقدام به رها سازی آن نموده و سویچ های متعددی به حافظه اصلی می نماید. این مساله سبب رجوع مکرر ریاک به حافظه اصلی و مبادله مکرر داده میان حافظه اصلی و پردازنده و دیسک می‌گردد.

در مقابل کاساندرا از حافظه RAM به صورت متمرکز و با روش Master/Slave مدیریت شده توسط مدیر کلاستر بهره می برد و در زمان اجرای برنامه نیز هیچگاه از حافظه مجازی استفاده نمی نماید. علت ان نیز همانند موارد ذکر شده ریاک می باشد. کاساندرا نیز برای اجتناب از سرباره سنگین ناشی از Swappingصفحات حافظه میان حافظه اصلی و مجازی هیچگاه از حافظه مجازی استفاده نمی‌نماید. همچنین دیسک برای کاساندرا نیز گلوگاه بوده و کاساندرا در طی عملیات خود، همواره تلاش دارد تا تنها زمان درج در مکان حافظه جنبی به دیسک مراجعه نماید و علاوه بر آن

تمهدیات لازم خود برای عملیات خواندن اطلاعات دارد که یکی از آنها بکارگیری روش های هوش مصنوعی در بافر نمودن می‌باشد.

مقایسه این دو سیستم حاکی از آن است که برخورد انحصاری و متمرکز کاساندرا با حافظه اصلی و انتقال صف درج اطلاعات به میان پردازنده و حافظه اصلی، علیرغم مصرف بالای حافظه سرعت بسیار بالایی برای کاساندرا به ارمغان می آورد و در مقابل مدیریت توزیع شده و منابع توزیع شده حافظه اصلی و کشیده شدن صف درج اطلاعات به میان پردازنده با دیسک و یا حافظه اصلی با دیسک سبب افزایس تاخیر زمانی اجرا و کاسته شدن از توانمندی ریاک در برابر کاساندرا می‌گردد.

۶- نتیجه گیری

جهت نتیجه گیری در زمینه بهینه بودن سیستم پایگاه داده از لحاظ زمان عملیات‌های پایه در سیستم، به بررسی نتایج دریافتی از ریاک و کاساندرا در حالت تک سروره و حالت خوشه ای دو و سه سروره می پردازیم. بررسی اجمالی در نتایج خروجی تست‌های انجام شده حاکی از وجود فاصله معنادار میان ریاک و کاساندرا است. کاساندرا بسیار سریع تر از ریاک در عملیات درج اطلاعات می باشد. کاساندرا در درج اطلاعات شاخص‌هایی نظیر سه برابر و حتی در برخی موارد حد هفت برابر سریع تر از ریاک، در حالت کلاستر شده از خود به نمایش گذاشته و ثابت نموده که خوشه بندی سبب بهبود شاخص های کاساندرا در مقابل ریاک می‌گردد. در هر صورت خوشه بندی سبب رشد کارآیی ریاک و کاساندرا می‌گردد. بررسی نمودار رفتار کاساندرا و ریاک حاکی از آن است که با ورود حجم کلان داده به سیستم، تاخیر زمانی ریاک افزایش می‌یابد و در این میان کاساندرا ، ترافیک سنگین ورودی را با شیبب ملایمی کنترل نموده و از افزایش تاخیر زمانی جلوگیری می‌نماید.

همچنین در عملیات بروز رسانی و خواندن اطلاعات نیز، با افزایش حجم اطلاعات، کاساندرا شاخص‌ها خود را حفظ و با استفاده بیشتر از پردازنده‌های سیستم‌های کلاستر شده به میزان حدود سه برابر ریاک، و بهره بردن بیشتر از حافظه اصلی توانست مانع از افزایش تاخیر زمانی گردد. همچنین بررسی رفتارهای کاساندرا با CPU و RAMحاکی از وجود الگوریتم های چندلایه و سیاست های متنوع در توزیع پذیری می‌باشد که بسیار موفق‌تر از ریاک نشان می‌دهد.

کاساندرا از حافظه و پردازنده در حالت خوشه ای استفاده بهتری دارد هرچند که متوسط میزان استفاده کاساندرا از حافظه اصلی در مواردی چندین برابر ریاک می باشد.

لازم به ذکر است که سیاست مدیریت حافظه در کاساندرا به صورت ارباب/رعیتی [12]در برابر مدل توزیع یافته چندارباب‌ی[13] در ریاک می-باشد که این مساله سبب کاهش مصرف حافظه از سوی ریاک و افزایش مصرف حافظه از سوی کاساندرا می‌گردد.

با کلاستر شدن، کاساندرا در عین کاستن از مصرف منابع سیستم نظیر حافظه و پردازنده کارآیی خود را افزایش داده و توانائی بالائی در مهار ورود انفجاری داده پیدا می‌نماید.

ریاک نیز با خوشه ای شدن، شاخص‌هایش را ارتقا می‌دهد ولی همواره مدت زمان اجرا و تاخیز زمانی اش بین سه الی هفت برابر کاساندرا می‌باشد.

از این رو به نظر می‌رسد که کاساندرا انتخاب خوبی برای پایگاه‌های داده غیرساختارمند شبکه اینترنت اشیاء می‌باشد که توانائی پشتیبانی به میزان سالهای متمادی را به ما می‌دهد. همچنین قابلیت های سهل برنامه نویسی با پایتون و جاوا امکان ارتباط راحت تر گوشی های هوشمند و شبکه سنسور را برای مان فراهم می‌نماید.

۷- نوآوری

کارهای انجام شده پیشین در این حوزه بر داده‌های تلفیقی و ترکیبی از داده‌های با حجم کم در کنار داده‌های با حجم بالا نظیر داده‌های حسگر و داده‌های چندرسانه‌ای بود که محدوده حجم اطلاعات از چند بایت الی چندین گیگابایت را شامل می گردید که ترکیبی از انواع نامتنجانس داده و فرمت‌های گوناگون را شاما=ل می گردید. همچنین تاکنون پژوهش مستقلی در باب داده‌های سری زمانی میان مدل‌های ستونی و کلید مقدار صورت نگرفته بود. سایر پژوهش‌ها نیز بر روی حجم کلان داده غالبا با ورودی چند رسانه‌ای به ویژه مبادلات وب سایت‌ها می‌باشد که ماهیت و مدل داده‌ای آنها متفاوت از داده‌های حسگرها می‌باشد.. تاکنون بر روی داده‌های جمع‌آوری شده شبکه حسگرها و اینترنت اشیاء با ماهیت سری زمانی، تحقیقات مستقل و منحصر بفردی صورت نگرفته است. از

[12] Master/Slave
[13] MultiMaster

این رو در این پژوهش ما به طور خاص به دنبال بررسی داده‌های سری زمانی ارسال شده از شبکه حسگرها و اینترنت اشیاء هستیم.

سپاسگزاری

در پایان از مرکز آپای دانشگاه ارومیه که بدون هیچ چشمداشتی یک سال تمام امکانات سخت‌افزاری و سرورهای خود را در اختیار بنده گذاشت تا این پژوهش را به نتیجه برسانم سپاسگذارم، به ویژه از محبت و دلسوزی ریاست این مرکز دکتر جمشید باقرزاده، معاونت آن دکتر اصغریان اصل و برادری‌های دکتر سامان تاجبخش.

مراجع :

[1] J. Zahid, A. Sattar, and M. Faridi. Unsolved Tricky Issues on COTS Selection and Evaluation. Global Journal of Computer Science and Technology 12.10-D 2012.

[2] Becker, C., Kraxner, M., Plangg, M., & Rauber, A. Improving decision support for software component selection through systematic cross-referencing and analysis of multiple decision criteria. In Proc. 46th Hawaii Intl. Conf. on System Sciences (HICSS), , pp. 1193-1202. 2013

[3] I. Gorton, A. Liu, and P. Brebner, Rigorous evaluation of COTS middleware technology, Computer, 36(3), pp,no.50-55. 2003.

[4] I. Gorton and J. Klein, Distribution, Data, Deployment:Software Architecture Convergence in Big Data Systems, IEEE Software, 11(3). PP , no. 99-117. 2014.

[5]John Klein, Ian Gorton, Neil Ernst,Patrick Donohoe, Kim Pham, Chrisjan Matser: Performance Evaluation of NoSQL Databases:A Case Study . In Proc. 15th Performance Analysis Intl.Conf of Big Data system. PP,no.5-10. 2012

[6]Sitalakshmi Venkatraman, Kiran Fahd, Samuel Kaspi, Ramanathan Venkatraman: SQL Versus NoSQL Movement with Big Data Analytics,I.J.Information Technolog and computer science, 10(12), PP,no.59-66 . 2016

[7] Choi, Y., Jeon, W., & Yo, S. (2014), Improving Database System Performance by Applying NoSQL, Journal Of Information Processing Systems, 10(3), 355-364. 2014

[8] Bazar, C., & Losif, C. (2014), 'The Transition from RDBMS to NoSQL. A Comparative Analysis of Three Popular Non-Relational Solutions: Cassandra, MongoDB and Couchbase', Database Systems Journal, 5(2), PP, no. 49-59 . 2014.

[9]Tilmann Rabl, Mohammad Sadoghi, HansArno Jacobsen : Solving Big Data Challenges for Enterprise Application Performance Management,Journal of the VLDB Endowment (PVLDB), 5(12),PP,no.1724-1735 . 2012

[10]Abhishek Prasad, Bhavesh N. Gohil : A Comparative Study of NoSQL Databases, International Journal of Advanced research in computer science, 28(3), PP, no. 1-6 . 2014

[11] Abdullah Talha Kabakus, Resul Kara: A performance evaluation of in-memory databases. Journal of King Saud University –Computer and Information Sciences,28(3),PP.no 1-6 . 2016

[12]Moniruzzaman, A. B., & Hossain, S. A. (2013), NoSQL database: New era of databases for big data analytics - Classification, characteristics and comparison. International Journal of Database Theory and Application, 6(4), PP, no. 1-14. 2013

[13]Dr. Norbert Jesse : Internet of Things and Big Data – The Disruption of the Value Chain and the Rise of New Software Ecosystems . IFAC-PapersOnLine 49(29), PP, no. 275–282. 2016

[14]NoSQL Benchmarking. http://www.cubrid.org/blog/dev-platform/nosql-benchmarking/. visited at march 2017

[15]Key-Value:Cassandra. http://www.ibm.com/developerworks/cn/opensource/os-cn-cassandra/?ca=drs-tp4608 . visited at march 2017.

[16]Cassandra Website. http://cassandra.apache.org/. visited at march 2017.

[17]Basho Riak Website. http://www.basho.com. . visited at march 2017.

[18]Abramova, V., Bernardino, J., Furtado. Which NoSQL database? A performance overview. Open J. Databases), 7(11), PP, no. 1,17–24. 2014