فصلنامه علمی- ترویجی پدافند غیرعامل
سال نهم، شماره ۲، تابستان ۱۳۹۷، (پیاپی ۳۴): صص ۶۸-۶۱

چارچوب هدوپ، کاربردها و چالش‌های پیش‌روی آن

محمد درویشی[۱]، مهدی نقوی[۲*]

تاریخ دریافت: ۱۳۹۶/۰۲/۱۱
تاریخ پذیرش: ۱۳۹۶/۰۹/۰۱

چکیده

در سال‌های اخیر، شاهد افزایش چشم‌گیر تولید داده بوده‌ایم. به گفته IBM تقریباً، ۹۰ ٪ کل داده‌های ذخیره‌شده موجود در دنیا، در دو سال اخیر تولید شده‌اند و برای اولین بار در تاریخ، در سال ۲۰۰۷ میلادی بود که حجم داده‌های تولیدی فراتر از فضای موجود برای ذخیره‌سازی آن‌ها رفت. همچنین، طیف گسترده‌ای از برنامه‌های کاربردی مانند موتورهای جستجو، تحقیقات پزشکی، پیش‌بینی آب‌وهوا و برنامه‌های علمی برای پردازش و تجزیه‌وتحلیل مقادیر داده‌ای بزرگ نیازمند محیط‌های توزیع‌شده هستند. داده‌های عظیم همانند سایر فن‌آوری‌ها، فرصت‌ها و چالش‌های متعددی را پیشروی استفاده‌کنندگان قرار داده است، استفاده از فرصت‌ها و مزایای آن در کسب‌وکار و مدیریت صحیح چالش‌ها به یکی از موضوعات داغ در عرصه فن‌آوری اطلاعات تبدیل شده است. بنابراین، وجود راه‌حلی برای پردازش اطلاعات عظیم با هزینه‌ای مقرون به‌صرفه بسیار اهمیت دارد، لذا یکی از بهترین راه‌حل‌ها برای رفع مشکل پردازش اطلاعات عظیم استفاده از چارچوب آپاچی هدوپ است. تعریف گارتنر از هدوپ این است که «هدوپ یک چارچوب مدیریت داده است که حجم زیادی از داده‌های دارای ساختار و بدون ساختار را که تقریباً در تمامی لایه‌های سازمانی اثر می‌گذارد، در کنار هم می‌آورد که موجب قرارگیری آن در بطن مراکز داده می‌شود». هدوپ بخشی از پروژه آپاچی است که بنیاد نرم‌افزار آپاچی آن را حمایت می‌کند. در واقع، هدوپ یک چارچوب برنامه‌نویسی رایگان و مبتنی بر جاوا است که ما را در پردازش مجموعه‌های عظیمی از داده‌ها در یک محیط پردازش توزیعی پشتیبانی می‌کند. لذا در این مقاله قصد داریم مقایسه‌ای بین پایگاه داده‌های ساخت‌یافته و نیافته داشته باشیم و سپس به بررسی معماری آپاچی هدوپ و کاربردهای وسیع آن در داده‌های عظیم امروزی و همچنین چالش‌های پیشروی این فناوری نوپا مثل پردازش دسته‌ای، گلوگاه و پردازش‌های بلادرنگ بپردازیم.

کلیدواژه‌ها: پردازش دسته‌ای، پردازش بلادرنگ، آپاچی هدوپ، سیستم‌های توزیعی، مقیاس‌پذیری، داده‌های عظیم

۱- دانشجوی کارشناسی ارشد، دانشگاه جامع امام حسین(ع)
۲- هیئت علمی دانشگاه امام حسین(ع)، Naghavi@ihu.ac.ir- نویسنده مسئول

۱- مقدمه

هدوپ[1] بخشی از پروژه آپاچی است که بنیاد نرم‌افزار آپاچی آن را حمایت می‌کند. هدوپ این امکان را در اختیار ما قرار می‌دهد تا برنامه‌هایی را روی سیستم‌هایی مجهز به هزاران گره و حاوی هزاران ترابایت داده به اجرا درآوریم. هدوپ این امکان را فراهم می‌آورد تا سرعت انتقال داده‌ها در میان گره‌ها افزایش یابد و سیستم بتواند درصورت ازکارافتادگی یک گره همچنان بی‌وقفه به کار خود ادامه بدهد. این رویکرد عملاً خطر فاجعه‌بار خرابی سیستم راحتی وقتی تعداد قابل‌ملاحظه‌ای از گره‌ها غیرفعال می‌شوند، کاهش می‌دهد [۱]. ایده هدوپ الهام‌گرفته از MapReduce گوگل است. مپ ردیوس یک چارچوب نرم‌افزاری است که در آن برنامه‌ها به بخش‌های کوچک‌تر تقسیم می‌شوند. هرکدام از این بخش‌ها (که همچنین بخش یا بلوک نامیده می‌شوند) می‌تواند روی هر نودی در یک مجموعه خوشه‌ای از نودها اجرا شود. «Doug Cutting » خالق هدوپ این نام را از نام فیل اسباب‌بازی کودکش گرفته است. اکوسیستم کنونی هدوپ آپاچی از Hadoop kernel، مپ ردیوس، سیستم فایل توزیعی هدوپ یا HDFS[2] و چند پروژه مرتبط دیگر مثل آپاچی Hive، Zookeeper و ... تشکیل شده است. چارچوب هدوپ مورد استفاده بازیگران بزرگ مثل گوگل، یاهو و ای‌بی‌ام قرار می‌گیرد. این شرکت‌ها از این چارچوب برای برنامه‌های مرتبط با موتورهای جست‌وجو و تبلیغات بهره می‌گیرند. استفاده از فرصت‌ها و مزایای آن در کسب‌وکار و مدیریت صحیح چالش‌ها به یکی از موضوعات داغ در عرصه فن‌آوری اطلاعات تبدیل شده است. یکی از مهم‌ترین چالش‌ها در این زمینه کمبود نیاز انسانی‌مستعد و ماهر است. نیاز به کارشناسان حرفه‌ای و آشنا به داده در بخش‌های مختلف نظیر مراکز دولتی، مراکز خصوصی و سازمان‌های غیرانتفاعی روندی رو به رشد را دارد و این نوید را می‌دهد که در آینده‌ای نه‌چندان دور شاهد تحولات مهمی در بازار عرضه و تقاضا باشیم. چراکه از یک طرف تأمین کارشناسان حرفه‌ای که قادر به کار مؤثر با داده باشند بسیار محدود است و از طرف دیگر، دستمزد مهندسین داده، دانشمندان داده، آمارگران و تحلیل‌گران داده روندی کاملاً تصاعدی به خود را گرفته است. در ۱۰ سال اخیر Hadoop از منشأ خود که به موتور جستجو مرتبط بود به یکی از مشهورترین بسترها برای حل چالش‌های همه‌منظوره بزرگ‌داده تبدیل شد. کلان‌داده دارای ویژگی‌های مشترکی هستند که می‌توان به حجم بالای داده، نرخ تولید بالا و تنوع محتوا اشاره کرد. شرکت تحقیقاتی IDC پیش‌بینی می‌کند که تجارت بزرگ‌داده تا سال ۲۰۱۶ میلادی بیش از ۲۳ میلیون دلار سرمایه در هدایت خود خواهد داشت. از زمان شروع به کار اولین شرکت مبتنی بر هدوپ به‌نام Cloudera در سال ۲۰۰۸ میلادی، بسیاری از شرکت‌های نوبنیاد مبتنی بر هادوپ صدها میلیون دلار برای سرمایه‌گذاری دریافت کرده‌اند. به‌طور خلاصه، سازمان‌ها دریافتند که هدوپ رویکرد مورد تأییدی برای تحلیل بزرگ‌داده پیشنهاد می‌دهد [۳-۲].

۲- کارهای مرتبط

در ابتدا بهتر است که بدانیم هدوپ چیست؟ هدوپ پایگاه داده نیست و همچنین هدوپ یک نرم‌افزار نیست، بلکه هدوپ یک چارچوب یا مجموعه‌ای از نرم‌افزارها و کتابخانه‌هایی است که سازوکار پردازش حجم عظیمی از داده‌های توزیع‌شده را فراهم می‌کند. هدوپ توسط Doug Cutting سازنده Apache Lucene که به‌صورت گسترده برای عملیات جستجوی متن‌ها استفاده می‌شود، تولید شد. در حقیقت، به‌وجودآمدن هدوپ از کار بر روی Nutch شروع شد. Apache Nutch یک فریم ورک متن‌باز برای ایجاد موتور جستجو است که به‌صورت گسترده، عملیات جستجوی متن‌ها را به روشی که Crawler نام گرفت انجام می‌دهد. درخصوص نام‌گذاری نام هدوپ باید گفت که مخفف عبارت خاصی نیست، این نامی است که پسر Dougبر روی عروسک فیل خود که زردرنگ بود گذاشته بود. برای شروع، Doug و همکارش Mike ایده ساختن یک موتور جستجوگر وب را در سر داشتند اما این تنها چالش آن‌ها نبود، قیمت سخت‌افزار یک موتور جستجوگر که ۱ میلیون صفحه وب را جستجو و نمایه می‌کند در حدود ۵۰۰۰۰۰ دلار بود. با این وجود، آن‌ها باور داشتند که این پروژه یک هدف ارزشمند است. Nutch در سال ۲۰۰۲ شروع به کار کرد و در همان زمان عملیات و روش‌های جستجوی صفحات وب به‌سرعت رشد کرد. طی زمانی معماران پروژه دریافتند که این پروژه قابلیت و توانایی کارکردن با میلیون‌ها صفحه وب را ندارد، در همان برهه در سال ۲۰۰۳ میلادی، مقاله‌ای از شرکت گوگل منتشر شد که توانست راه‌گشای مشکل آن‌ها باشد و معمار GFS را توصیف می‌کرد. GFSتوانست مشکل ذخیره‌سازی داده‌های عظیم را حل کند. علاوه‌بر آن، مدیریت ذخیره‌سازی نودها دیگر چالشی بود که در معماری‌های قبلی بود و با استفاده از این روش آن مشکل نیز برطرف شد. در سال ۲۰۰۴ میلادی، گروه Nutch توانست نسخه متن‌باز خود را با نام HDFS منتشر کنند. در سال ۲۰۰۴ میلادی، گوگل با مقاله‌ای MR را به جهان معرفی کرد، خیلی زود در سال ۲۰۰۵ میلادی، برنامه‌نویسان Nutch شروع به‌کار با MR کردند و تا اواسط همان سال، Nutch نسخه جدید خود را که با HDFS و MR کار می‌کرد به جهان معرفی کرد. بعد از چندی معماران Nutch دریافتند که عملکرد آن فراتر از فقط یک موتور جستجوگر است و در فوریه

1- Hadoop
2- Hadoop Distributed File System

۲۰۰۶ میلادی، آن‌ها از پروژه Nutch که خود زیرپروژه Lucene به‌حساب می‌آمد به سمت پروژه‌ای آمدند که آن را Hadoop (هدوپ) نامیدند. تقریباً همان سال، Doug به یاهو پیوست تا با استفاده از یک گروه مستقل هدوپ را آزمایش و پیاده‌سازی کند. در سال ۲۰۰۸ میلادی، شرکت یاهو، موتور جستجویی را معرفی کرد که توسط ۱۰۰۰۰ خوشه هدوپ عملیات جستجو را انجام می‌داد. در همان سال و در ماه ژانویه هدوپ در بالاترین سطح پروژه‌های Apache قرار گرفت. در آن زمان دیگر Yahoo تنها استفاده‌کننده این محصول نبود و شرکت‌هایی نظیر Last.fm، Facebook و New York Times نیز شروع به فعالیت در این حوزه کرده بودند. در همان سال، New York Times تعداد زیادی از روزنامه‌های خود را که در قسمت آرشیو وجود داشت اسکن کرد که حجم آن نزدیک به ۴ ترابایت داده می‌شد و سپس با استفاده از پردازش ابری EC2 آمازون و با استفاده از ۱۰۰ ماشین در کمتر از ۲۴ ساعت پردازش خود را به پایان برساند. در آوریل سال ۲۰۰۸ میلادی، هدوپ رکورد جهان را شکست و سریع‌ترین سیستمی شد که توانست ۱ ترابایت داده را ظرف ۲۰۲ ثانیه و با استفاده از ۹۱۰ نود کلاستر پردازش کند. این رکورد در سال قبل با ۲۹۷ ثانیه ثبت شده بود. در نوامبر همان سال، گوگل طی گزارشی اعلام کرد که این رکورد را به ۶۸ ثانیه ارتقاء داده است. در آوریل ۲۰۰۹ میلادی، یاهو اعلام کرد با استفاده از هدوپ توانسته است ۱ ترابایت داده را ظرف ۶۲ ثانیه پردازش کند. و بالاخره در سال ۲۰۱۴ میلادی، یک گروه از شرکت Data Bricks اعلام کرد که توانسته با استفاده از ۲۰۷ گره خوشه Spark حدود ۱۰۰ ترابایت داده را ظرف ۱۴۰۶ ثانیه که تقریباً ۴,۲۷ ترابایت در دقیقه می‌شود پردازش کند. امروزه هدوپ به‌صورت وسیعی و در زمینه‌های بسیاری از فعالیت‌های دانشگاهی تا تجارت، از علوم تا نجوم مورد استفاده قرار می‌گیرد [۴]. هدوپ مکانی امن برای ذخیره و تحلیل داده‌های کلان به‌شمار می‌رود، مقیاس‌پذیر، توسعه‌پذیر و متن‌باز است. هدوپ هدف اصلی کمپانی‌های بزرگ تولید و ذخیره داده‌هاست ازجمله Facebook، IBM، EMC، Oracle و Microsoft است. از کمپانی‌های متخصص در زمینه سرویس‌های هدوپ می‌توان به MapR، Cloudera و HortonWorks اشاره کرد.

به‌طور خلاصه، Hadoop را می‌توان به یک سیستم‌عامل تشبیه کرد که طراحی‌شده تا بتواند حجم زیادی از داده‌ها را بر روی ماشین‌های مختلف پردازش و مدیریت کند [۲].

۳- چرا به هدوپ نیازمندیم؟

ابتدا در این بخش از علل گرایش سوئیچ شرکت‌های بزرگ مانند گوگل، یاهو، فیس‌بوک به سمت Hadoop و Map Reduce را بیان می‌کنیم. بهتر است قبل از هر چیز نگاهی اجمالی به قبل از مدل مپ ردیوس و هدوپ بیندازیم، این به ما کمک می‌کند که نگاهی دقیق‌تر نسبت به این موضوع داشته باشیم که چرا ما نیازمندیم به‌سوی چیزی بهتر از قبل حرکت کنیم و یا به عبارت ساده‌تر چرا به هدوپ نیازمندیم. از اولین مدل‌های کامپیوتری که شروع به استفاده از مدل‌های توزیع‌شده کردند، سوپرکامپیوترها بودند. در مدل سوپرکامپیوترها که قبلاً و هم‌اکنون در مسائل خاص مورد استفاده قرار می‌گرفتند از مجموعه‌ای از کامپیوترها استفاده می‌شد که مدل طراحی آن را Clustring مدل می‌نامیدیم، البته نوع و مدل سخت‌افزاری، طراحی و تجهیزات شبکه آن با تصورات ما نسبت به کامپیوتر بسیار متفاوت است.

MPI یک فریم ورک توزیع‌شده است که قبلاً بسیار مورد استفاده بود. اما MPI چگونه کار می‌کند؟ MPI به زبان ساده از ۲ عملکرد تقسیم می‌شود:

MPI_SEND

MPI_RECIVE

برای مثال ماشین M اطلاعاتی را به ماشین K می‌فرستد و ماشین K پیام دریافت اطلاعات را به ماشین M می‌فرستد، تکرار این عملیات در مقیاس بسیار وسیع باعث ایجاد بن‌بست می‌شود. راه‌حل‌های متفاوتی برای حل این مسئله وجود دارد، ولی در کل این وظیفه برنامه‌نویسان است که راهی برای آزادکردن منابع درخواستی ارائه دهند زیرا در محیط واقعی هزاران ماشین هستند که در هر دقیقه به ده‌ها بن‌بست برمی‌خورند و مدیریت این‌چنین سیستمی بسیار وقت‌گیر و طاقت‌فرسا هست [۳ و ۵].

جدول (۱): مقایسه پایگاه‌های داده رابطه‌ای و هدوپ

ویژگی	RDBMS	Hadoop
ترابایت و پتابایت	گیگا بایت	حجم داده‌ای
خطی یا مقیاس‌پذیری افقی (Scale Out)	غیرخطی یا مقیاس‌پذیری عمودی (Scale Up)	مقیاس‌پذیری
دسته‌ای	تعاملی و دسته‌ای	دستیابی
شِما قابل‌تغییر و پویا	شِما غیرقابل تغییر	ساختار
کم	بالا	یکپارچگی
بالا	خیلی کم	گسترش‌پذیری
نوشتن یک‌بار و خواندن در هر زمان	خواندن و نوشتن در هر زمان	بروز رسانی
ساخت‌یافته و غیرساخت یافته	ساخت‌یافته	نوع داده‌ای
تأخیر زیاد	تأخیر کم	زمان پاسخ
بازیابی داده و پردازش‌های کلان	پردازش‌های بانکی	موارد استفاده

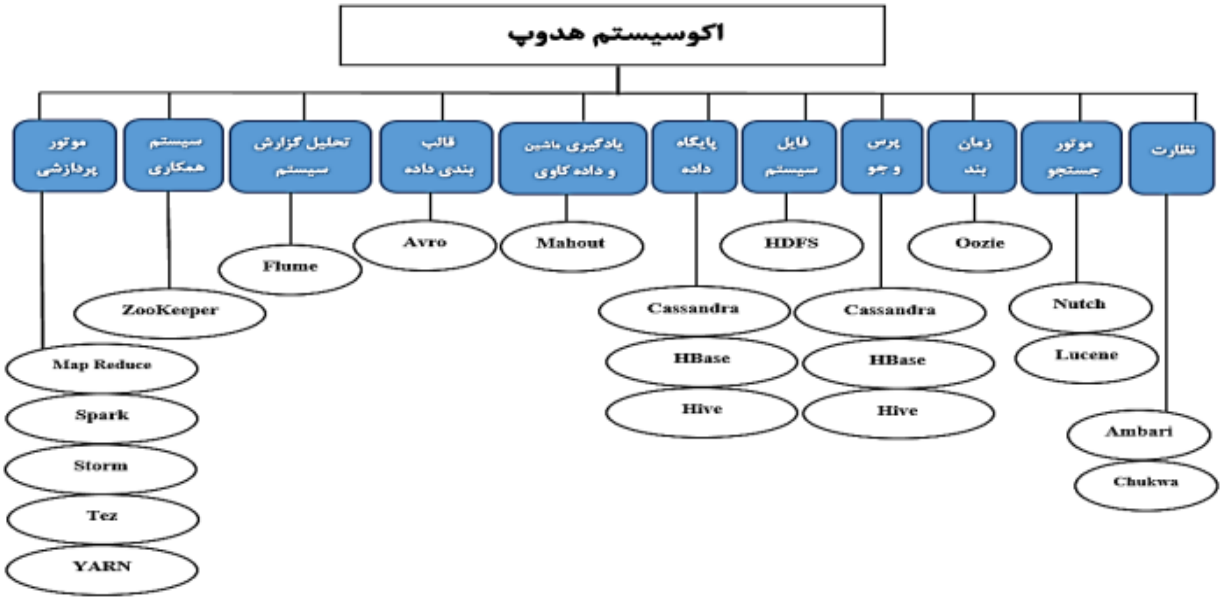
از آن گذشته در دهه اخیر به‌دلیل ظهور شبکه‌ها و رسانه‌های اجتماعی اطلاعات شروع به رشد کردند و فایل‌ها از حداکثر سایز چندین گیگابایت به چندین ترابایت تبدیل شدند و دیگر با ساختارهای قبلی قابل ذخیره و تحلیل نبودند، از این‌رو، برای حل این مشکل کارشناسان به یک جمع‌بندی کلی رسیدند که در ابتدا باید نحوه ذخیره و بازیابی اطلاعات تغییر کند. آن‌ها تعداد ۲۴ سیستم کامپیوتری را بر روی یک رک سوار کردند که هرکدام به‌صورت مستقل یک کامپیوتر بود ولی نحوه ذخیره اطلاعات به نحوی بود که اگر قرار بود فایلی روی آن ذخیره شود، براساس الگوریتمی آن فایل تقسیم و بر روی کامپیوترها به‌صورت موازی و با سرعت بسیار بالایی ذخیره می‌شد و در مرحله دوم اگر ما قصد پردازش اطلاعات آن را داشتیم باید تمام فایل‌های تکه‌شده را جمع‌آوری می‌کردیم و سپس فایل واحد را پردازش می‌کردیم که در مقیاس بالا باعث گلوگاه در شبکه می‌شد. برای بهبود و حل این مشکل آن‌ها تصمیم گرفتند به‌جای این‌که فایل‌های تکه‌تکه شده را در ابتدا واکشی کرده و پردازش کنند، از قدرت پردازش هر کامپیوتر در رک استفاده کرده و پردازش را در همان‌جا انجام دهند. در این روش قدرت پردازش چند برابر شده و به‌جای انتقال داده‌ها اطلاعات پردازش‌شده انتقال پیدا می‌کنند [۱]. البته در روش کنونی نیز مشکلاتی وجود دارد، مثلاً اگر برای یکی از نودها مشکلی به‌وجود آید چه اتفاقی برای پردازش و فایل‌ها به وجود خواهد آمد یا اگر مشکلی برای سوئیچ روی رک به‌وجود آمد چه راه‌حلی وجود خواهد داشت. شرکت‌های مطرح مانند Google، Yahoo، Facebook و ... که در زمینه داده‌های بزرگ پیشگام هستند از راه‌حلی‌هایی با نام DFS و GFS استفاده می‌کنند که به‌صورت Open source در اختیار کاربران قرار دارد. با استفاده از این روش‌های جدید نحوه ذخیره و بازیابی اطلاعات در داده‌های کلان کاملاً عوض شد. با توجه به حجم داده امروزی نیاز به پردازش این حجم داده در زمان مطلوب و همچنین نیاز به بهینگی در پردازش دسته‌ای و ذخیره‌سازی داده‌های عظیم با هزینه‌ای مقرون‌به‌صرفه بیش از پیش مورد توجه است، با مطرح‌شدن هدوپ تا حدودی این نیازمندی‌ها رفع شدند، چارچوب هدوپ در واقع برای ذخیره‌سازی و فراخوانی اطلاعات عظیم (در حد گیگابایت، ترابایت و یا حتی پتابایت) مورد استفاده قرار می‌گیرد [۶]. این اطلاعات می‌تواند شامل فایل و یا پردازش باشد. برای مثال چندی قبل شرکت یاهو که بزرگ‌ترین سیستم هدوپ را در اختیار دارد، موفق شد رقم ۲،۰۰۰،۰۰۰،۰۰۰،۰۰۰،۰۰۰ ام عدد پی و چند رقم بعد و قبل آن را محاسبه کند، جالب است بدانید که این عملیات که بر روی ۱۰۰۰ سرور صورت گرفته به مدت ۲۳ روز به طول انجامید، درحالی‌که اگر این عملیات را بر روی یک سیستم اجرا کنیم، حدود ۵۰۳ سال به طول خواهد انجامید [۱۰]. به‌طورکلی، هدوپ توانست آنالیز و تحلیل بر روی داده‌هایی را ممکن سازد که تا قبل آن انجام تحلیل بر روی آن‌ها به‌دلیل پیچیدگی و ساخت نیافتگی آن‌ها غیرممکن بود. در پایان این بخش در جدول (۱) مقایسه‌ای بین پایگاه‌های داده رابطه‌ای که عمدتاً قبل از مطرح‌شدن هدوپ استفاده می‌شدند و پایگاه‌های داده بعد از هدوپ بیان شده است.

۴- کاربردهای هدوپ

همان‌طورکه قبلاً هم اشاره شد، هدوپ یک پایگاه داده نیست و همچنین هدوپ یک نرم‌افزار نیست، بلکه هدوپ یک فریم و رک یا مجموعه‌ای از نرم‌افزارها و کتابخانه‌هایی است که سازوکار پردازش حجم عظیمی از داده‌های توزیع‌شده را فراهم می‌کند، در شکل (۱) مجموعه تمام اجزایی که هرکدام به‌نوعی وظایفی را بر عهده دارند و در کنار هم به هدوپ معنا می‌بخشند نشان داده شده است. بنابراین، اگر تمامی اجزا بتوانند به‌خوبی نقش خود را ایفاء کنند می‌گوییم هدوپ کارایی بالایی از خود نشان داده است. امروزه هدوپ به‌طور گسترده در شرکت‌های بزرگی مثل گوگل و یاهو و فیس‌بوک مورد استفاده قرار می‌گیرد، برخی شرکت‌ها مثل یاهو از هدوپ به‌عنوان موتور جستجو خود استفاده می‌کند و برخی هم مثل گوگل و فیس‌بوک از آن به‌عنوان پردازشگر لاگ‌ها و تجزیه‌وتحلیل ویدئوهای خود بهره می‌برند، در شکل (۲) درصد کاربردهای مختلف هدوپ در صنایع مختلف نشان داده شده است که نقش هدوپ در فن‌آوری و تبلیغات نسبت به سایر صنایع بیشتر به چشم می‌خورد، علاوه‌بر موارد ذکرشده در شکل، هدوپ کاربردهای وسیع دیگری نیز دارد:

۴-۱- پیش‌بینی وضع هوا

در چند سال اخیر، بحث هواشناسی و بحران آب یکی از مباحث داغ و مهم جوامع بین‌المللی بوده است و کشورهای مختلف بنا بر ظرفیت‌های خود به‌دنبال راه‌حل‌های مختلف جهت پیش‌بینی و جلوگیری از تخریب روزافزون آن هستند. ایالات متحده نیز در این بین با استفاده از سازمان‌های گوناگون خود ازجمله Nasa به‌دنبال یافتن راه‌حل‌های موجود است. دانشمندان هواشناسی برای پیش‌بینی آب‌وهوا نیاز دارند تا داده‌های جمع‌آوری‌شده که مقدار آن بسیار زیاد است را به‌صورت کاملاً نامرتب جمع‌آوری کنند، سپس، بعد از مراحل مرتب‌سازی و تمیزکردن داده‌ها آن‌ها را ذخیره و استفاده کنند. در این بین، زمان بسیار زیادی هدر خواهد رفت. علاوه‌بر آن، پردازش این حجم عظیم داده نیازمند هزینه‌ای بسیار بالاست. در این موقعیت، دولت شروع به سرمایه‌گذاری برای حل این مشکل می‌کند. ناسا تصمیم گرفت به‌وسیله کاوش اطلاعات، نسل آینده تحلیل و پیش‌بینی هواشناسی را تولید و راه‌اندازی کند. ناسا برای شبیه‌سازی و تحلیل آب‌وهوا از Apache Hadoop به‌دلیل بهره‌وری بالااستفاده می‌کند.



شکل (۱): اجزاء مختلف چارچوب هدوپ

شکل (۲): کاربردهای مختلف هدوپ در صنایع مختلف [۱۱]

زیرا هدوپ با توزیع داده‌ها مشکلاتی از قبیل دسترس‌پذیری داده‌ها و قابلیت اعتماد را حل کرده و علاوه‌بر آن با قابلیت پردازش موازی داده‌ها سرعت پردازش را به‌طور چشمگیری افزایش داده است. Glenn Tamkin از تولیدکنندگان و طراحان نرم‌افزار در NASA با کمک یکی از همکارانش با استفاده از ۳۴ نود کلاسترشده هدوپ توانست برنامه‌ای تولید کند تا با استفاده از آن بتوان وضعیت آب‌وهوا را پیش‌بینی و شبیه‌سازی کنند. به این صورت که اطلاعات را روی هدوپ ذخیره می‌کنند و به‌وسیله پردازش توزیع‌شده سرعت پردازش را چندبرابر می‌کنند. عملیاتی ازقبیل جمع‌کردن، شمارش، میانگین، انحراف معیار و دیگر عملیات محاسباتی و آماری با روش Map Reduce بسیار سریع‌تر و قابل اعتمادتر بودند. اجرای عملیات با استفاده از هدوپ توانست بسیاری از مشکلات دانشمندان در این زمینه را برطرف کند [۷].

۴-۲- مدیریت کشوری و امور نظامی

با استفاده از هدوپ می‌توان مدیریت کشوری را با کنترل و ارزیابی بهتری اداره کرد. از کاربردهایی که هدوپ می‌تواند در این زمینه داشته باشد می‌توان به پیش‌بینی نتایج انتخابات و جمع‌آوری و شمارش آرا، تحلیل و جمع‌آوری رفتار سربازان در مرزهای کشوری، جمع‌آوری داده‌های رفتاری شهروندان و کنترل عبور و مرور شهری اشاره کرد [۸].

۴-۳- پزشکی و داروسازی

با استفاده از هدوپ می‌توان به پزشکان در تشخیص دقیق‌تر بیماری‌هایی مثل سرطان کمک کرد و پیشنهادهایی برای تجویز

دقیق‌تر دارو بر طبق تغذیه و ژنتیک ارائه کرد و یا می‌توان به تحلیل عواقب جانبی داروها پرداخت [۶].

اگر بخواهیم از دید مهندسی کامپیوتر به هدوپ نگاه کنیم، امروزه هدوپ به‌عنوان ابزار هدف بسیاری از شرکت‌های بزرگ مثل یاهو و فیس‌بوک و آمازون کاربرد دارد. به‌طور مثال، آمازون با استفاده از تحلیل رفتار مشتریان خود از سود و زیان خود در دوره‌های زمانی مشخص مطلع خواهد شد. آقای ERIC PETERSON تحلیل‌گر آمازون می‌گوید "همواره هزینه پیداکردن مشتری جدید خیلی بیشتر از حفظ مشتریان فعلی است". بنابراین، می‌توان با دراختیارداشتن اطلاعات بیشتری از مشتریان نرخ ریزش مشتری را تا حدود قابل توجهی کاهش داد، یا مثلاً کاربرد هدوپ در سیستم توصیف‌گر Netflix است که با دراختیارداشتن حجم زیادی از داده‌های مشتریان، یک سری شباهت بین مشتریان مختلف پیدا می‌کند و براساس رفتار برخی مشتریان، رفتار برخی دیگر را پیش‌بینی می‌کنند و به آن‌ها پیشنهادهایی در زمینه جستجوی کاربر داده می‌شود. در انتها به‌طورکلی می‌توان گفت در هرجایی که داده‌های تولیدی به حجمی برسند که جمع‌آوری و پردازش آن‌ها در توانایی دستگاه‌ها و ابزارهای موجود نباشد می‌توان از بستر هدوپ استفاده کرد.

۵- چالش‌های هدوپ

همان‌طورکه در فناوری‌های نوظهور چالش‌هایی بر سر راه آن‌ها وجود دارد هدوپ هم از این قضیه مبرّا نیست و چالش‌های خاص خود را دارد که در ادامه به تشریح چند مورد از آن‌ها خواهیم پرداخت:

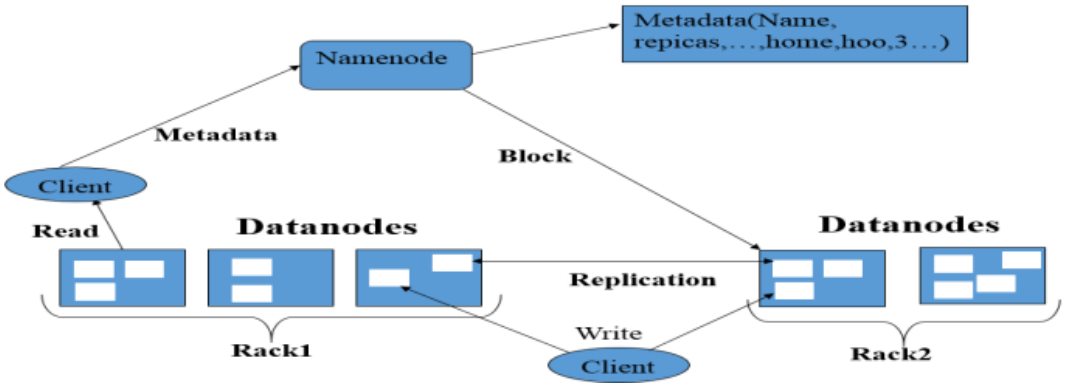
۵-۱- هدوپ و چالش پردازش پردازش‌های بلادرنگ

در حوزه پردازش داده داده‌های عظیم، دو نوع پردازش داریم : ۱- پردازش بلادرنگ یا همان پردازش جریانی: در این نوع پردازش اطلاعات دریافتی به‌دلیل درخواست‌های بلادرنگ به مرور زمان ارزش خود را از دست می‌دهند. بنابراین، باید بلافاصله بعد از دریافت مثلاً داده‌های دریافتی از ناسا، مورد پردازش قرار گیرند. ۲- پردازش انبوه: در این نوع پردازش اطلاعات دریافتی به‌صورت کامل ذخیره‌سازی شده و بعداً به‌صورت آفلاین یا آنلاین مورد پردازش قرار می‌گیرند مثل داده‌های نمرات دانشجویان دانشگاه یا اطلاعات یارانه‌ای افراد [۹]. در حوزه فن‌آوری‌های اصلی کلان‌داده برای پردازش داده‌های عظیم فن‌آوری‌های مانند روش توزیع (مپ)، تجمیع (ردیوس) و جدیداً اسپارک برای پردازش انبوه داده‌ها طراحی‌شده‌اند. در سال‌های اخیر، فن‌آوری‌های پردازش بلادرنگ و داده‌های جریانی مانند داده‌های دریافتی از حسگرها و تصاویر ترافیک و ماهواره، داده‌های شبکه‌های اجتماعی و مانند آن‌که یکسره در حال تولید هستند و جریان آن‌ها به‌صورت پیوسته درحال تزریق به برنامه‌های پردازشی هستند، پیشرفت زیادی کرده‌اند و فقط در اکوسیستم آپاچی (مجموعه پروژه‌های بنیاد آپاچی) امروزه بیش از ده پروژه مختلف متن‌باز مختلف در این حوزه وجود دارد. بعضی از آن‌ها، تفاوت بسیار کمی با یکدیگر دارند که این امر، انتخاب درست ابزار و کتابخانه‌های مورد نیاز برای پردازش جریان را امری زمان‌بر و تخصصی نموده است و استفاده از هدوپ برای پردازش آن‌ها کارایی لازم را ندارد [۱]. البته با هدف سهولت تصمیم‌گیری مهندسین داده، وبلاگ Database Line در اقدامی تحسین‌برانگیز، این فناوری‌ها را در یک جدول با هم مقایسه کرده است که می‌توانید آن را در [۱۲] مشاهده کنید. هدوپ به‌کاررفته در مپ ردیوس تا زمانی که تمامی داده‌های ورودی مپ نشوند نمی‌تواند به مرحله ردیوس برود. دلیلش هم واضح است چون ردیوس برای انجام مراحل خود نیاز به تمام اطلاعات ورودی که مپ شدند دارد. بنابراین، برای بلادرنگ مناسب نیست. راه‌حلی که امروزه در حال گسترش است استفاده از آپاچی اسپارک است که امکان پردازش کلان‌داده به‌صورت بلادرنگ را فراهم کرده است. پس اگر نیاز به انجام تجزیه‌وتحلیل در جریان داده‌ای، مانند حسگرهای روی کف کارخانه و یا برنامه‌های کاربردی که نیاز به عملیات‌های متعدد دارند داشته باشیم، باید از اسپارک استفاده کرد. به‌عنوان‌مثال، یک شبکه اجتماعی ممکن است مایل به تشخیص موضوعات پرطرفدار گفتگو در دقیقه باشد، مثلاً یک سایت اینترنتی ممکن است مایل به بررسی فایل‌های لاگ برای تشخیص شکست‌ها در چند ثانیه باشد که همه این‌ها نیاز به پردازش بلادرنگ دارد، راه‌حل آن استفاده از آپاچی اسپارک است [۱].

۵-۲- هدوپ و چالش پردازش بلوک‌های داده کوچک

طرز کار معماری هدوپ در شکل (۳) قابل مشاهده است که نحوه عملکرد آن به این صورت است که برای ذخیره‌سازی داده‌های عظیم، ابتدا داده‌ها را تکه‌تکه کرده و تکه‌ها را در سرورهای مختلف ذخیره می‌کند. به‌طور پیش‌فرض، اندازه هر تکه می‌تواند ۶۴ مگابایت باشد حال مسئله این‌جا است که هرگاه اندازه فایل تکه‌تکه شده آن‌قدری بزرگ نباشد که بتوان آن را در سرورهای مختلف ذخیره کرد و یا حتی گاهی اوقات اندازه فایل آن‌قدر کوچک است که فایل ذخیره‌شده فقط به یک‌تکه تقسیم شود و در درون یک سرور ذخیره‌سازی شود در این حالت اگر فایل کوچک ذخیره‌شده توسط کاربران زیادی مورد دستیابی قرار گیرد این کار باعث می‌شود همه کاربران فقط به یک سرور (به‌جای چندین سرور) مراجعه کنند و به‌نوعی در سیستم گلوگاه ایجاد می‌شود و کارایی سیستم پایین می‌آید. علاوه‌بر این، اگر تعداد فایل‌های کوچک زیاد باشد درگیری سرور مستر زیادتر شده و امکان بن‌بست در سیستم وجود دارد [۱۰]. یکی از راه‌حل‌هایی که می‌توان بر ای این چالش مطرح کرد ایجاد برچسب سقف مراجعه

برای هریک از بلوک‌های داده است، مثلاً اگر تعداد مراجعات همزمان به یک بلوک یا تکه داده بیش از صد بود اجازه دستیابی به آن بلوک داده را محدود کند تا از بروز بن‌بست در آن کلاستر جلوگیری شود.



شکل (۳): معماری آپاچی هدوپ [۲]

۵-۳- هدوپ و چالش وابستگی به گره نام

مطابق شکل (۳)، معماری هدوپ از سه بخش اصلی تشکیل شده است: گره نام، گره داده و Client که هرکدام وظایفی خاص را برعهده دارند. این معماری مبتنی بر سرویس دهنده- سرویس گیرنده است، گره نام، بخش اصلی این معماری می‌باشد و ازجمله وظایف مدیریت فضای نام، تعیین نوع دسترسی کاربران به فایل گره نام و تعیین تعداد کپی‌های بلوک‌های یک فایل می‌باشد. سیستم فایل هدوپ برای مقاوم‌بودن در برابر خطا و آسیب‌های احتمالی، بلوک‌های داده‌ای را در دو گره داده اضافی کپی‌سازی می‌کند تا در مواقع بروز خطا بتواند از آن‌ها برای بازیابی داده استفاده کند. اما در کنار مزیت‌هایی که بیان شد این معماری معایب‌هایی نیز دارد که یکی از این مشکلات وابستگی به گره نام است، چون گره نام به‌عنوان هسته مرکزی معماری هدوپ انجام وظیفه می‌کند و هرگونه بروز مشکل در این بخش باعث از دسترس خارج‌شدن خوشه هدوپ می‌شود و تنها در صورت ترمیم گره نام است که خوشه هدوپ می‌تواند مجدداً به فعالیت خود ادامه بدهد. مشکل دیگر این است که دسترسی‌ها به بلوک‌های داده‌ای ذخیره‌شده به‌صورت خواندنی است چون‌که فایل‌ها در هدوپ یک‌بار نوشته و بارها خوانده می‌شوند، بنابراین، در جاهایی که عملیات نوشتن در آن‌ها زیاد است چارچوب هدوپ نمی‌تواند کارایی لازم را داشته باشد [۸].

۵-۴- هدوپ و چالش همه‌منظوره نبودن آن

هدوپ هنوز یک فناوری نوظهور است. به‌عبارتی در لبه فناوری قرار دارد، از این جهت که برای افرادی که به‌تازگی با آن آشنا شدند دردسرساز می‌شود زیرا یک کاربر هدوپ اگر با زبان جاوا آشنایی نداشته باشد با مشکل مواجهه می‌شود. پس قبل از هدوپ باید به‌صورت حرفه‌ای زبان جاوا را فرا بگیرد تا در مواجهه با خطاهای متداول هدوپ عاقلانه‌ترین تصمیم را بگیرد. علاوه‌بر زبان جاوا، یک کاربر هدوپ باید آشنایی نسبی از شبکه و خطاهای آن داشته باشد [۶].

۵-۵- هدوپ و چالش عملکرد ضعیف در مراجعات بعدی

در هدوپ متأسفانه هیچ سیستم‌کشی مدنظر گرفته نشده است که این امر ممکن است به‌دلایل امنیتی باشد، با این‌حال هرگاه سرویس گیرنده‌ای تقاضای یک فایل را داشته باشد آدرس فایل مورد نظر را به گره نام می‌دهد و گره نام فایل مورد نظر را در میان کلاسترهای متفاوت جمع‌آوری کرده و آن را تحویل سرویس گیرنده خواهد کرد، حال اگر ظرف چند دقیقه بعد سرویس دهنده دیگری تقاضای همان فایل را بکند باید به همان اندازه که سرویس گیرنده اولی منتظر ماند، صبر کند تا فایل مورد نظر را واکشی نماید. پس هنوز هدوپ هیچ مکانیزمی برای دسترسی سریع‌تر سرویس گیرنده به فایل اخیراً استفاده شده ندارد [۱۳].

۶- نتیجه‌گیری

بستر هدوپ یک راه‌حل برای مدیریت پردازش داده‌های عظیم در حجم بسیار بالا می‌باشند. با وجود چالش‌های مطرح‌شده هنوز هم یکی از ابزارهای کلیدی غول‌های تجاری دنیا محسوب می‌شوند و علی‌رغم تصور برخی افراد که معتقد هستند با آمدن اسپارک، هدوپ دیگر جایی در پردازش‌های کلان داده ندارد، می‌تواند سال‌های زیادی مورد استفاده قرار گیرد. نگارش این مقاله در راستای آشنایی با اجزا بستر هدوپ و کاربرهای مختلف آن در حوزه‌های مختلف و چالش‌های عمده آن انجام شده است. در این کار پژوهشی سعی کردیم معماری هدوپ و اجزای آن‌که هرکدام وظایفی را برعهده دارند، مطرح نمایم. در دو بخش مطالب را بررسی کردیم، در بخش اول پیشینه تاریخی

هدوپ و همچنین ضرورت استفاده از سیستم فایل توزیع‌شده با توجه به افزایش حجم داده‌ها و همچنین ویژگی‌های برجسته هدوپ که می‌تواند تأثیر به‌سزایی در کارایی آن داشته باشد را بیان کردیم، در بخش دوم کاربردهای مختلف هدوپ در شرکت‌های مشهور دنیا مثل گوگل و فیس‌بوک و ناسا و همچنین کاربردهای آن در مهندسی داده مثل تحلیل ریسک و تحلیل تهدیدات سایبری و تحلیل رفتارهای مشتری برای سودآوری بیشتر فروشگاه‌های اینترنتی مثل آمازون بیان شد و بخش آخر هم چالش‌های عمده چارچوب هدوپ که تا حدودی راه‌حل‌هایی برای آن‌ها مطرح شد. برخلاف تصور برخی افراد هدوپ و اسپارک نتیجه دو ابزار با تعاریف متفاوت هستند، مقایسه این دو ابزار با هم نه‌تنها از منظر تخصصی بلکه ازنظر اعتبار استدلال منطقی، حکم قیاس مع‌الفارق را دارد زیرا که دو طرف مقایسه می‌بایست شباهت کاملی با هم داشته باشند درحالی‌که این‌گونه نیست.

سخن آخر این‌که در هر جایی‌که داده‌های تولیدی به حجمی برسند که جمع‌آوری و پردازش آن‌ها در توانایی سیستم‌ها و ابزارهای موجود نباشد می‌توان از بستر هدوپ استفاده کرد.

۷- مراجع

1. Gurusamy, Vairaprakash, S. Kannan, and K. Nandhini. "A Study on Distributed Computing Framework: Hadoop, Spark and Storm." (2018).
2. S. Blazhievsky and W. Nice, “Introduction to Hadoop and MapReduce,” SNIA Education All Right Reserved Storage Networking Industry Association, 2013.
3. K. Grolinger, M. Hayes, A. Higashino, A. L'Heureux, and Allison, “Challenges for Map Reduce and Hadoop in Big Data,” Department of Electrical and Computer Engineering Western University, London, IEEE 2014.
4. Ch. Wong Lee, S. Hong Cho, J. Wook Kim, and D. HoonHwang, “Development of electric trading system using big data,” International Journal of Multimedia and Ubiquitous Engineering, vol .9, 2014.
5. H. wardhan, Bh. Devendra, and P. Gadekar, “A Review Paper on Big Data and Hadoop,” International Journal of Scientific and Research Publications, vol. 4, Issue 10, October 2014.
6. A. Madaan, et al., “Hadoop: Solution to Unstructured Data Handling,” Big Data Analytics, Springer, Singapore, 2018.
7. J. Schnase, D. Duffy, S. Strong, D. Nadeau, and H. Thompson, “Applying Apache Hadoop to NASA’s Big Climate Data,” National Aeronautics and Space Administration, 2014.
8. R. R. Parmar, et al., “Large-Scale Encryption in the Hadoop Environment: Challenges and Solutions,” IEEE Access 5, pp. 7156-7163, 2017.
9. S. Sakr, A. Liu, D. M. Batista, and M. Alomari, “A survey of large scale data management approaches in cloud environments,” Communications Surveys & Tutorials, IEEE, vol. 13, pp. 311-336, Jaiswal, Er Shalika, and Amandeep Singh Walia, “Big Data and Hadoop challenges and issues,” International Journal 8.4, 2017.
10. J. Anuradha, “A brief introduction on Big Data 5Vs characteristics and Hadoop technology,” Procedia computer science 48, pp. 319-324, 2015.
11. http://www.bigdatacompanies.com/top-5-hadoop-distributions-for-big-data/
12. http://cakesolutions.net/teamblogs/comparison-of-apache-stream-processing-frameworks-part-1
13. J. E. Shalika and E. A. Singh Walia, “Big Data and Hadoop challenges and issues,” International Journal, vol. 8, no. 4, 2017.

Hadoop Framework and Uses and its Challenges

M. Darvishi, M. Naghavi*

Abstract

In recent years, an ever-increasing trend in mass data production is observed over the recent years. According to IBM, interestingly, around 90% of the existing data in the world is produced only in the last two years. It was in 2007, when the size of data exceeded the available storage resource for the first time. Also a wide range of applications such as search engines, medical research, weather forecasting and scientific programs needed distributed data for the processing and analysis ofbig amounts of data , Big Data, as in other technologies, has numerous opportunities and challenges in front users. The use of opportunities and benefits in the business and proper management challenges is converted into one of the hot topics in the field of IT, So there is a very important mechanism for processing mass at a cost effective, Therefore, one of the best ways to solve the problem of massive information processing is the use of the Apache Hadoop. Gartner's definition of the Hadoop is "Hadoop is a data management system that brings together large volumes of structured and unstructured data that affects almost all organizational layers. this causes the positioning in the heart of data centers". Hadoop is part of the Apache Software Foundation supported byApache projects , in fact Hadoop is a free Java-based programming framework that allows us to process massive sets of data in a distributed processing environment supports. Therefore, in this article, we have a comparison of structured and unstructured Database and then, we investigate the Apache Hadoop architecture and its wide range of applications in today's Big Data as well as challenges facing this emerging technology, such as batch processing, real-time processes and bottlenecks.

Key Words: *Batch Process, Real-Time Process, Apache Hadoop, Distributed Systems, Big Data, Scalability*

* Imam Hossein University (Naghavi@ihu.ac.ir) - Writer-in-Charge