فناوری های اطلاعات بزرگ و مدیریت:

چه مدل سازی مفهومی را می توان انجام داد

چکیده

دوران داده‌های بزرگ منجر به توسعه و کاربرد فناوری‌ها و روش‌هایی شد که به طور مؤثر با استفاده از حجم وسیع داده‌ها به پشتیبانی تصمیم‌گیری و فعالیت‌های کشف دانش کمک می‌کنند. در این مقاله، پنج V داده بزرگ، حجم، سرعت، تنوع، صحت و ارزش، و همچنین فناوری‌های جدید شامل پایگاه داده NoSQL که مطابق با نیازهای ابتکاری داده‌های بزرگ ارائه شده، بررسی می‌شوند. سپس نقش مدل‌سازی مفهومی برای داده‌های بزرگ بررسی شده و پیشنهاداتی درباره تلاش‌های مدل‌سازی مفهومی مؤثر با توجه به داده‌های بزرگ ارائه می‌شود.

1. مقدمه

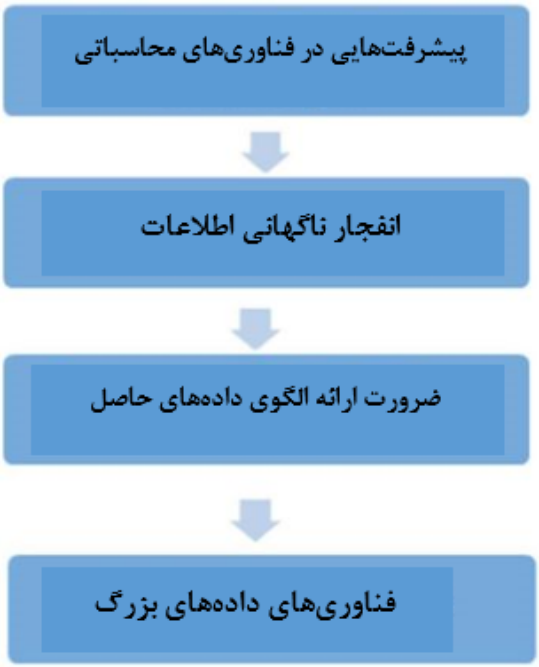
داده‌های بزرگ به طور گسترده به عنوان مقادیر بسیار زیاد داده‌ها شناخته می‌شوند، ساختاریافته و غیرساختاریافته، که در حال حاضر سازمان‌ها قادر به دست‌یابی و تلاش برای تحلیل معنی‌دار بودن آنها هستند به طوری که تحلیل تصمیم‌گیری بر پایه داده ها و بینش عملی بدست می‌آید. انجام این کار مستلزم توسعه تکنیک‌ها و روش‌های تحلیل، ایجاد روش‌های جدید برای ساخت داده‌ها و برنامه های جالب در علم و مدیریت است (به عنوان مثال، [1، 8، 19]). با وجود به چالش کشیدن ارزش داده‌های بزرگ، چشم انداز داده‌ها همچنان رشد می‌کند [28].

هدف این مقاله بررسی پیشرفت داده‌های بزرگ در تلاش برای شناسایی چالش‌های موجود است؛ و نقشی که مدل‌سازی مفهومی می تواند در پیشبرد کار در این حوزه مهم بازی کند را تعیین می کند. بخش بعدی، توصیف

داده‌های بزرگ و ویژگی های ذاتی شناخته شده است. سپس، پیش از تحلیل نقش به خصوصی که مدل‌سازی مفهومی در درک و پیشرفت تحقیق و کاربرد داده‌های بزرگ بازی می کند، فناوری داده‌های جدید و در حال ظهور ارائه می‌شوند.

2. داده‌های بزرگ

حجم داده‌ها در دهه گذشته به طور نمایی افزایش یافته است، تا جایی که مدیریت دارایی داده‌ها با استفاده از روش‌های سنتی امکان‌پذیر نیست (ریبریو و همکاران، 2015). همان طور که در شکل 1 نشان داده شده است، روند پیشرفت داده‌ها با پیشرفت فناوری‌های محاسباتی امکان‌پذیر شده است که منجر به انفجار ناگهانی داده‌های منابع مختلف مانند وب، رسانه‌های اجتماعی و سنسورها شده است. سیل داده‌ها سبب ظهور الگویی مبتنی بر داده‌ها شد تا از فناوری‌های جدید محاسبات در دسترس استفاده شود. فناوری‌های داده‌های بزرگ، منجر به الگوی مبتنی بر داده‌ها می‌شود و آن را به طور فزاینده‌ای پیچیده‌تر و مفیدتر می‌کند.



شکل 1. شاخص‌های منتهی به داده‌های بزرگ

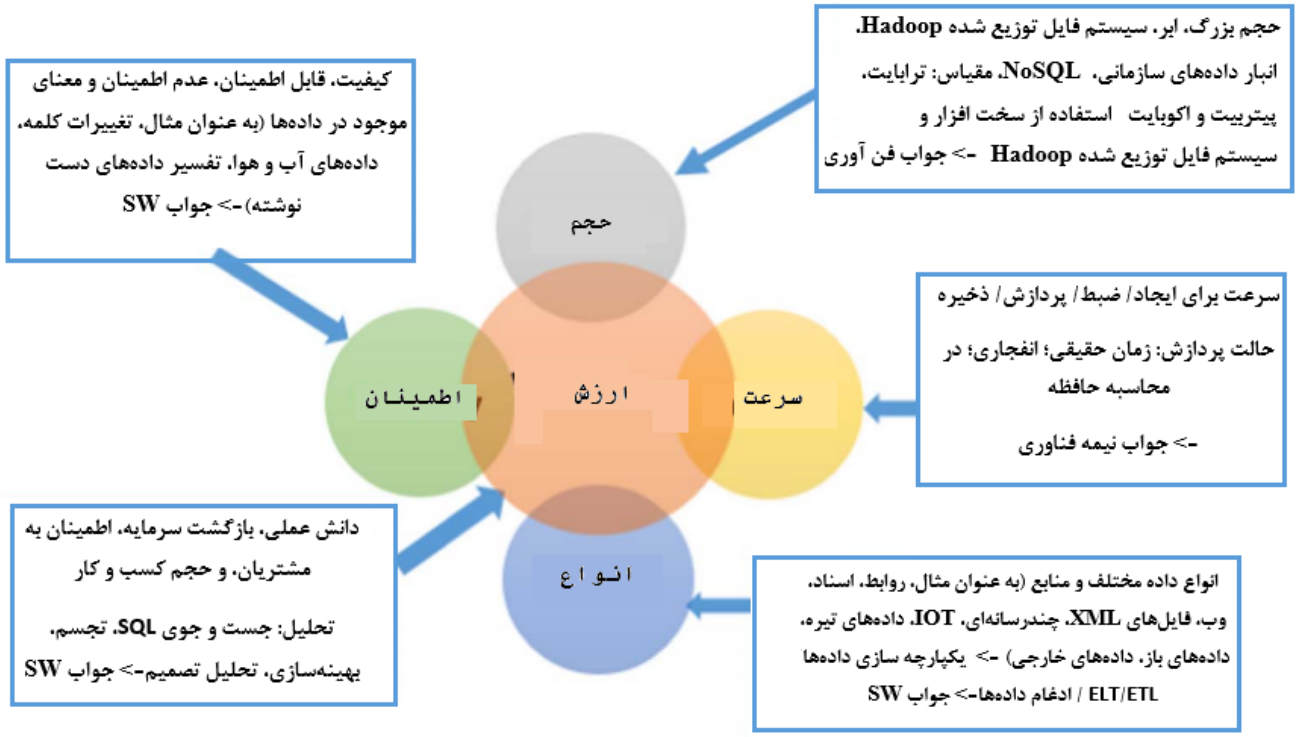
داده‌های بزرگ به حجم بالا، سرعت و انواع دارایی‌های اطلاعاتی اشاره می کنند که به پردازش‌های جدید و نوآورانه برای تصمیم گیری پیشرفته، بینش کسب و کار و بهینه سازی روند کار احتیاج دارد [23]. به عنوان مفهوم نسبتاً جدید، مفهوم اصلی داده‌های بزرگ شامل تکنیک‌ها و فناوری‌های مورد نیاز برای مدیریت حجم بسیار زیاد داده‌ها است. علاوه بر فناوری، برای تحلیل و طراحی با مهارت لازم جهت مدیریت این منبع، متخصصان ماهر مورد نیاز هستند [2 و 21].

مایر- اسچونبرگر[1] و کوکیر[2] (2013) استدلال کردند که داده‌های بزرگ باعث تغییر رفتار افراد، کار و تفکر می‌شوند، هر چند که لازم است بسیاری از موانع برطرف شوند. داده‌ها باید بدست آمده، پردازش شوند و به طور مؤثر مورد استفاده قرار گیرند، موضوعات مربوط به چگونگی نمایش داده‌ها و مدل‌سازی داده‌ها افزایش می‌یابد. هر چند که درک چالش‌های مرتبط با نمایش و مدل‌سازی داده‌های بزرگ، ابتدا به درک ویژگی‌های داده‌های بزرگ نیاز دارد.

2.1. Vهای داده های بزرگ

داده‌های بزرگ به طور سنتی با استفاده از سه V حجم، تنوع و سرعت مشخص می‌شود که برگرفته از پیشرفت در سنجش، ارزیابی و فناوری های محاسبات اجتماعی است (Gartner.com). علاوه بر این Vها، درستی (دقت) و به ویژه ارزش، مهم هستند. هر یک ار ارزش‌ها دارای چالش های منحصر به فردی هستند. حجم بیش از حد بزرگ به انواع تحلیل ساختاریافته و غیرساختاریافته احتیاج دارد و سرعت بسیار بالا حتی ممکن است منجر به عدم تشخیص سؤالات معقول شود [14]. درستی منجر به عدم اطمینان می شود، و حجم با سرعت رقابت می‌کند [34]. با این وجود، این حجم برای استخراج وقت گیرتر است، و برای اطمینان دشوار است. شکل 2 خلاصه ای از چالش‌های پنج V را در عملکردهای داده‌های بزرگ و تلاش‌های تحقیقاتی نشان می‌دهد.

[1] . Mayer-Schonberger
[2] . Cukier



شکل 2. 5 V داده بزرگ

حجم: حجم زیاد داده‌ها موجب دسترسی به داده‌ها در جریان‌های داده‌ای متنوع می‌شود و اغلب به موقعیت مکانی وابسته است که شامل انواع مختلفی داده می‌شود که با سرعت بسیار بالا از بانک‌های بزرگ سنسورهای فیزیکی، دیجیتال و انسانی ایجاد می‌شود [16]. منابع داده شامل فناوری‌های قابل پوشش، خدمات مبتنی بر ابر (به عنوان مثال، خدمات وب آمازون)، انبار داده‌های سازمانی (EDW) و پایگاه‌های داده NoSQL است. مقیاس در حال حاضر ترابایت، پیتربیت و اکوبایت است. چالش حجم با استفاده از سخت افزار و سیستم فایل توزیع شده Hadoop (HDFS) از نظر تکنیکی مورد توجه قرار گرفته است.

سرعت: سرعت، سرعت ایجاد، ضبط، استخراج، پردازش و ذخیره داده‌ها است. برای مقابله با چالش سرعت، راه‌حل نیمه تکنولوژی با بخش راه‌حل نرم‌افزاری مورد نیاز است که دارای پردازش در زمان واقعی، جریان و محاسبات حافظه، است.

انواع: انواع داده‌ها و منابع مختلف روابط (از پایگاه داده‌های ارتباطی)، اسناد، داده‌های وب، فایل‌های XML، داده‌های سنسور، فایل‌های چندرسانه‌ای و غیره را ارائه می‌دهند. چالش‌های گوناگون عمدتاً با راه‌حل‌های پیچیده حل می‌شوند، زیرا ادغام داده‌های ناهمگن به تلاش گسترده برای بررسی انواع مختلف نیاز دارد.

اطمینان: درستی به عدم اطمینان داده‌ها اشاره دارد. درستی مسائل مربوط به کیفیت، قابلیت اطمینان، عدم اطمینان، ناتمام بودن و معنای موجود در داده‌ها را افزایش می‌دهد (به عنوان مثال، تغییرات کلمه، داده‌های آب و هوا، تفسیر داده‌های دست نوشته). با این وجود، در نهایت، درستی باید به صورت خودکار پردازش شود. چالش درست باید باید با راه‌حل‌های صحیح حل شود.

ارزش: اطمینان به ارزش داده‌های بزرگ ممکن است دشوار باشد. اقدامات منطقی بر شناسایی دانش «عملی»؛ محاسبه، در صورت امکان، بازده سرمایه گذاری (ROI)؛ شناسایی روابط مشتریان (تحقیقات زیادی درباره تحلیل معادلات متن و احساسات ارائه شده است)؛ و سایر اقدامات متمرکز هستند. تحلیل مورد نیاز داده‌های بزرگ برای شناسایی ارزش از راه‌های مختلفی انجام می‌شود، از جمله: جست‌وجوی سنتی SQL، روش‌های یادگیری ماشین، داده‌کاوی، آمار، بهینه‌سازی، و تحلیل پشتیبانی از تصمیم. نتایج ممکن است در اشکال مختلف، از جمله تولید سند، استاندارد و ارائه گزارش و تجسم ارائه می‌شود. چالش ارزش برای رسیدن به اهداف سخت است، زیرا راه‌حل‌های پیچیده آن باید در درون حوزه کسب و کار یا مسأله آدرس‌دهی شوند. علاوه بر این، دانشمندان می‌توانند داده‌های موجود و واجد شرایط داده‌های بزرگ را با درک صحیح دامنه در نظر گیذند و با استفاده از ابزارهای تحلیلی به راحتی پیدا کنند.

بسط هفت V شامل اعتبار و نوسان است [18]. قابلیت اجرا، Vهای اضافی شامل تجسم، تنوع، آسیب‌پذیری، قابلیت مشاهده، عدم تردید و غیره، همگی باید مدیریت شوند. با این وجود، «درک» و برخورد با معانی داده‌ها همچنان یک چالش بزرگ است [38، 39 و 44].

2.2. زیرساخت مدیریت داده‌های بزرگ

برای تطابق با ابتکارات داده‌های بزرگ، شرکت‌ها زیرساخت‌هایی را برای مدیریت آن ایجاد کرده‌اند، داده‌های بزرگ را به عنوان دارایی شرکت ساخته‌اند. زیرساخت نیاز به ابزار درست دارد و به طور کلی از اجزای زیر تشکیل شده است (کالاکوتا، 2012):

• پشته داده‌ها: شامل اطلاعات ساختار یافته و همچنین بدون ساختار است.

• اکوسیستم داده‌های بزرگ: شامل جست و جو و تجسم، تنظیم داده‌ها و دسترسی به داده‌ها است. خدمات حرفه‌ای برای داده‌های بزرگ عبارتند از شخت افزار، مانند کامپیوتر، ذخیره سازی و شبکه‌ها. خدمات مربوط به پایگاه داده خاص شامل سیستم‌های مبتنی بر SQL، سیستم های NoSQL و Hadoop و اکوسیستم‌های آن است.

• مدیریت اطلاعات سازمانی: بر مسائل مربوط به مدیریت داده‌ها، ادغام داده‌ها، کیفیت داده‌ها، تجسم داده‌ها و مدیریت داده‌های اصلی تمرکز دارد.

• سیستم‌های علمی داده‌ها و ابزار: ابزارهای مختلفی برای استخراج الگو تجسم ارائه شده است. این‌ها عبارتند از: الگوریتم یادگیری ماشین؛ تحلیل پیش‌بینی [12]؛ تکنیک‌های تجویزی (به عنوان مثال، شبیه‌سازی با متغیرهای جایگزین و زیرمجموعه داده‌ها)؛ تکنیک‌های توصیفی (به عنوان مثال، آمار و تکنیک‌های گزارش‌های تاریخچه)؛ و گزارش (به عنوان مثال، ارائه کارت امتیازی یا تابلوهای علامت).

2.3. الگوهای مبتنی بر داده‌ها

الگوهای هدایت‌کننده دارای پیامدهای تحلیل و تصمیم‌گیری هستند. تفاوت‌های مهم میان مدیریت داده‌های سنتی و الگوی جدیدتر به صورت زیر خلاصه می‌شود.

به جای استفاده از کل مجموعه داده‌ها می توان از نمونه‌گیری استفاده کرد. لین[1] و همکاران [19] برای کاهش هزینه جمع‌آوری داده‌ها و تحلیل، آن را به عنوان نمونه برداری با اندازه «N» ارائه دادند و داده‌ها را به اندازه‌ای که ممکن است جمع‌آوری کردند [15].

همان طور که در فیلم مونیبال[2] نشان داده شد، تحلیل داده‌ها به منظور درک صورت می‌گیرد، به عنوان مثال، ورزش بیس‌بال، به شیوه‌ای که قبلاً در نظر گرفته نشده استفاده می‌شود (http://blog.minitab.com/blog/the-statistics-game/money ball-shows-the-power-of-statistics).

سیستم کامپیوتری، دپ بلو[3] [17]، یک قهرمان شطرنج را بر اساس تکنیک های ارائه شده در هوش مصنوعی شکست داد. بررسی آمازون[4] (Amazon.com) که به ساکنین وب کمک می‌کند، ارزش آنها را بیش از توصیه کارشناسان نشان می‌دهد.

شناسایی دلیل همیشه ممکن نیست. سیستم‌های توصیه کننده مانند موارد استفاده شده در آمازون، مواردی را که با هم خرید می‌شوند شناسایی می‌کند. با این وجود، دلیل انجام آن را نشان نمی‌دهد.

این روند بسیاری از جنبه‌های غیرمعمول زندگی اجتماعی را به داده‌های کامپیوتری قابل تحلیل تبدیل می‌کند . ایده اصلی رفتار یا افکار انسان، جامعه و ماشین آلات برای ارائه برنامه‌های دیجیتالی جدید است.

داده‌های زیر به سطح جزئیاتی اشاره می‌کنند که داده‌ها در آن ذخیره می‌شوند (به عنوان مثال، سطح پایین اطلاعات به اطلاعات خلاصه بالاتر). با استفاده از تکنولوژی داده‌های بزرگ، داده‌های ریزتر پتانسیل بیشتری برای استخراج ارزش کسب و کار دارند و امکان استفاده مجدد در برنامه‌های متعدد را افزایش می‌دهند.

اجازه دهید داده ها برای خود صحبت کنند. این نشان دهنده ارزش آیتم فیزیکی مارک‌ها، ایده‌ها و حقوق فکری است.

---

[1] . Lin
[2] . Moneyball
[3] . Deepblue
[4] . Amazon

2.4. رشد و چالش‌ها

پیشرفت در ذخیره‌سازی، شبکه، پردازنده مرکزی، و پهنای باند قابل توجه است. برای مثال هزینه ذخیره‌سازی ($ در هر ترابایت) از 14000000 دلار (1980) به حدود 50 دلار کاهش یافته است. تعداد گره‌های یک شرکت ممکن است از 1 (1969) به 1 ملیارد میزبان افزایش یافته باشد. هزینه هر GFLOPS هزینه مجموعه‌ای از سخت‌افزار است که از لحاظ نظری با 1 میلیارد نقطه شناور در ثانیه کار می‌کند. هزینه‌های پردازنده مرکزی از 1100000000 دلار (1961) به 0/08 دلار کاهش یافته است. پهنای باند ($ در هر مگابایت در ثانیه) که در سال 1998 تقریباً 120 دلار بوده است، در حال حاضر حدود 5 دلار هزینه دارد.

انفجار ناگهانی داده‌ها از ضبط خودکار، سنسورها و سایر منابع که به الگو هدایت داده منتهی می‌شود، برای تصمیم‌گیری در زمان واقعی و سایر فعالیت‌های متمرکز بر داده‌ها در نظر گرفته می‌شود. میزان بازار جهانی داده‌های بزرگ (سخت افزار، نرم‌افزار و خدمات) بیش از 50 میلیارد دلار تخمین زده شده است (ویکیبون، 2015). با توجه به پیشرفت‌های تکنولوژی‌های محاسباتی با قابلیت پردازش سریع‌تر، ارزان‌تر و قدرتمندتر به علت قانون مور[1] و دیگر راه‌های جایگزین برای پیشرفت قابلیت محاسبات، داده‌های بزرگ بدست می‌آیند.

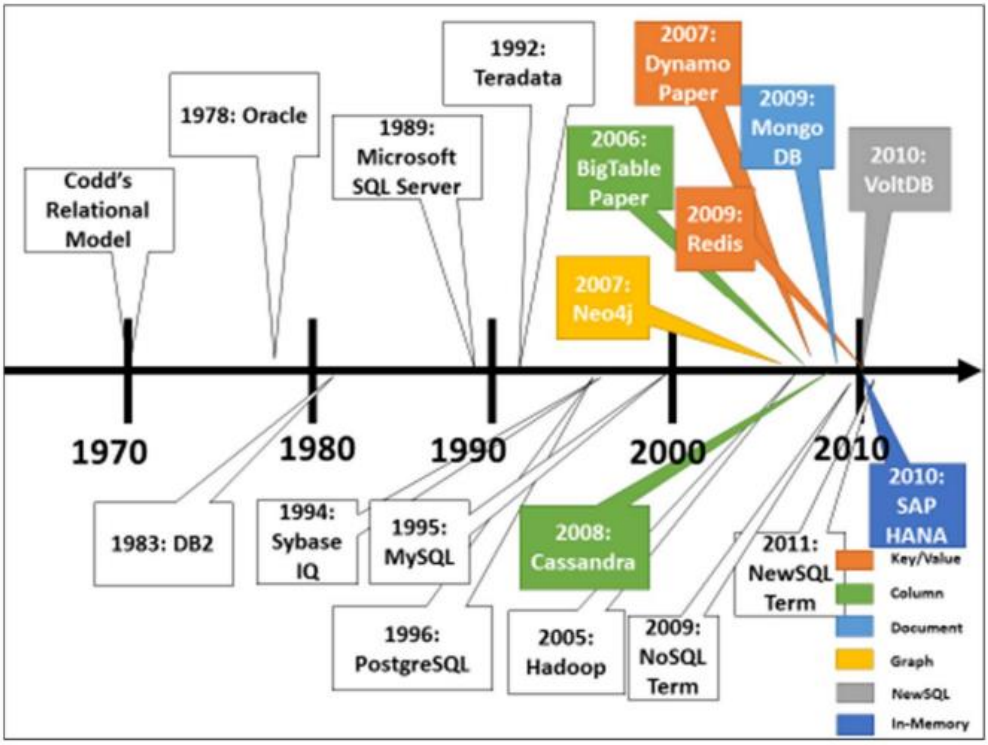
چالش‌های زیادی در رابطه با داده‌های بزرگ وجود دارد [45]، از جمله مشکلات، درک مقیاس آن برای انسان است. حجم بیش از حد بزرگ است و سرعت خیلی سریع است [14]. تنوع و صحت اطلاعات در ارائه راه‌حل‌های نرم‌افزاری چالش برانگیز است. ارزش بسیار جالب است و از طریق فرصت‌های بزرگ داده‌ها، راه‌حل‌های نوآورانه‌ای برای مسائل ارائه می‌دهد. علاوه بر این، تأثیر آن بر جامعه کسب و کار قابل توجه است.

3. فناوری پایگاه داده جدید برای داده‌های بزرگ

فناوری‌های پایگاه داده به طور خاص برای داده‌های بزرگ طراحی شده و مورد استفاده قرار می‌گیرند. شکل 3 خط زمانی را نشان می‌دهد که طی آن نقاط عطف مربوط به ارائه تکنیک‌های داده بزرگ رخ داده اند. در دهه 1970،

[1] . Moore

روش مدیریت پایگاه داده ارتباطی (کد[1]، 1979) ارائه شد. در دهه 1980، به روش ساختاری قابل اعتماد و کارا برای مدیریت داده‌ها تبدیل شد. در مدل ارتباطی، داده‌ها به صورت تابعی در جدول، روابط را فراخوانی می‌کنند. تلاش‌هایی برای درک نحوه تفسیر مدل‌های مفهومی برای مدل‌های ارتباطی انجام شد (به عنوان مثال، [43 و 47]). همان طور که حجم داده‌ها در پایگاه داده ذخیره می‌شوند، مفاهیم مرتبط با پایگاه‌های «بسیار بزرگ» ظاهر می‌شوند، همان طور که توسط کنفرانس‌ها و مجلات پایگاه‌های بسیار بزرگ شناخته می‌شوند. این پایگاه‌های بسیار بزرگ (در محدوده ترابایت) به مدل ذخیره‌سازی متفاوت احتیاج دارند، به دلیل: (1) بار زیاد در مدل ارتباطی (اگرچه اکنون تنها در حد نظریه است)؛ و (2) خواسته‌های بازیابی و کاربرد متفاوت. به عنوان مثال، بازیابی ساده رکورد شخصی مشتری، جان دویی[2]، کافی نبود. در مقابل، تنها نتیجه مطلوب برای همه کسانی که فهرست ذخیره را مشاهده می‌کنند، چیزی شبیه به «مسیریاب انسانی» است.



شکل 3. سیر تکاملی مدیریت داده‌ها

[1] . Codd
[2]. John Doe

در اویل دهه 1990، یک ترابایت از داده‌ها به عنوان مقدار زیاد اطلاعات در پاسخ به تمایل و نیاز به دریافت اطلاعات بیشتر مورد توجه قرار گرفت. در نتیجه انواع مختلف ابزار و سیستم‌های مدیریت ارتباطی داده‌ها در بازار تسلط یافتند (به عنوان مثال، اوراکل[1]، IBM DB 2، سرور SQL مایکروسافت). در دهه اول این قرن، سیستم‌هایی گسترش یافتند که کمتر به ساختار پایگاه داده ارتباطی، انحراف کلیدها، ستون‌ها، استاد، نمودارها، مدیریت حافظه و سایر روش‌ها وابسته بودند. به عنوان مثال، Hadoop چارچوب منبع باز برای مدیریت داده‌های بدون ساختار با استفاده از روش پردازش موازی با MapReduce است که ابتدا توسط گوگل ارائه شد. گسترش بیشتر شامل مجموعه‌ای از پایگاه داده‌های NoSQL و NewSQL است. اکثر انواع جدید پایگاه داده با استفاده از معماری مقیاس‌پذیر از مقیاس‌پذیری و عملکرد پشتیبانی می‌کنند.

3.1. چه چیزی در سیستم‌های مدیریت پایگاه داده ارتباطی سنتی اشتباه است؟

سیستم‌های مدیریت پایگاه داده ارتباطی (RDMS) سنتی به سادگی نمی‌توانند داده‌های بزرگ را مدیریت کنند. داده‌های بیش از حد بزرگ، برای ذخیره و دستکاری بسیار سریع و متنوع هستند. پایگاه‌های داده ارتباطی پیش از نوشتن برای پایگاه داده به طرحی نیازمندند که برای کنترل حجم داده‌های زمان حقیقی با ساختار متنوع بسیاری قوی باشد. ویژگی‌های ACID (ثبات، انسجام، انزوا و دوام) برای برخی کاربردها بسیار سخت است. خوشه‌های SPOF ACID (تنها نقطه شکست) گران هستند و عدم تقارن و عدم انتطباق (مجموع در مقابل داده‌های با ثبات) دارند. این‌ها به الزامات معماری‌های جدید و مدیریت معاملات جدید مانند BASE (اساساً موجود، قابل انعطاف، ثبات احتمالی) منتهی می‌شوند که ویژگی‌های ACID را در سیستم‌های مدیریت داده توزیع شده آزاد می‌کنند. BASE در دستگاه‌های NoSQL متداول است.

[1]. Oracle

Hadoop .3.2

Map/Reduce یک چارچوب برنامه نویسی با نرمال سازی خودکار است. قسمت نقشه مربوط به داده‌های ورودی است. این قسمت کلیدها و مقادیر کاهشی را با تولید مرتب و تقسیم شده برای جنبه کاهشی منتشر می‌کند. تابع Reduce بر داده‌های گروهی با کلید کاهش داده اعمال می‌شود. تابع Reduce داده‌ها را کاهش می‌دهد، بدین معنا که داده‌ها را می‌توان با افزودن مقادیر انتخاب شده جمع کرد. Map و Reduce را می توان برای محاسبات پیچیده به هم متصل کرد. نتیجه، مقیاس‌پذیری است که برای معماری مقیاس‌پذیر مناسب است که از سخت افزار محصول کم هزینه با ویژگی‌های تحمل خطا استفاده می‌کند. Hadoop مقادیر بزرگ داده‌های ساختاریافته، ساختار نیافته و نیمه ساختاریافته را پردازش و ذخیره می‌کند.

Hadoop (hadoop. apache. org) یک نسخه منبع باز الگوریتم Map/Reduce است که برای تحلیل داده‌های بدون ساختار بزرگ ساخته شده و به استاندارد واقعی در تحلیل داده‌های بزرگ تبدیل شده است. در پایگاه داده سنتی از جست‌وجویی که در زبان جست‌وجو ساختار یافته نوشته شده استفاده می‌شود، داده‌ها به عنوان ذخیره شده در پایگاه داده رابطه‌ای قابل دسترس می‌شوند و نتیجه بدست می‌آید. با این وجود، این نوع نمایش داده محدود است، بنابراین ممکن است خروجی مطلوب نباشد. با استفاده از Hadoop، داده‌های غیرساختاری را می‌توان به روش‌های مختلف برای داده‌کاوی الگوهای مفید ترکیب کرد. Hadoop 1.0 سیستم تک کاربره برای برنامه‌های کاربردی دسته‌ای است؛ Hadoop 2.0 پلت‌فرم داده چندمنظوره است که از برنامه‌های دسته‌ای، تعاملی، جریان و برنامه‌های گرافیکی پشتیبانی می‌کند. پیشرفت اکوسیستم Hadoop ذخیره‌سازی داده‌ها، پردازش داده‌ها و دسترسی به داده‌ها برای مدیریت داده‌ها به صورت زیر تعریف می‌شوند.

ذخیره‌سازی داده‌ها- HDFS (سیستم فایل توزیع شده Hadoop) و Hbase (ذخیره‌سازی ستونی پایگاه داده).

پردازش داده‌ها- MapReduce (پردازش خودکار داده‌های موازی).

دسترسی به داده‌ها- Hive (SQL- لینک)، Pig (جریان داده)، Mahout (یادگیری ماشین)، Avro (سریال داده‌ها و موافقت‌نامه پردازش از راه دور)، Sqoop (رابط مدیریت پایگاه داده ارتباطی).

مدیریت- Oozie (گردش کار)، Chukwa (نظارت)، Flume (نظارت)، و ZooKeeper (مدیریت).

Hadoop به طور ذاتی مقیاس‌پذیر است و برای پردازش مقدار زیاد داده‌ها با توازن بار خودکار خوب است. با این وجود، زمانی که چند تکرار مورد نیاز است و هنوز اجرای عملیات پیچیده مانند اتصالات بر اساس زمینه های چندقطبی احتیاج، Hadoop بیش از حد به Hbase وابسته است. این محدودیت‌ها چارچوب پردازش موازی پردازنده حافظه جدید را ارائه می‌دهند که Spark نامیده می‌شود.

Apache spark .3.2.1

Apache spark پلت‌فرم محاسباتی حول محور حافظه است که مخصوص پردازش‌های تحلیلی در مقیاس بزرگ طراحی شده است (http://spark .apache.org/). یک موتور سریع و عمومی با مدل برنامه نویسی ساده است که بیانگر پشتیبانی از طیف گسترده‌ای از برنامه‌های کاربردی از جمله ETL (استخراج، تبدیل، بار)، یادگیری ماشین، پردازش جریان و محاسبات گراف است. Apache spark، 100 برابر سریع‌تر از Hadoop MapReduce در حافظه و 10 برابر سریع‌تر از دیسک است. Spark دارای موتور اجرایی پیشرفته DAG (نمودار بدون دور جهت‌دار) است که از جریان داده‌ها و داده‌ها در محاسبات حافظه پشتیبانی می‌کند. 80 اپراتور سطح بالا را قادر می‌سازد که برنامه‌های موازی را با استفاده از تعامل پوسته‌های Scala، پایتون[1] و R ایجاد کند. این پلت‌فرم SQL، جریان و تحلیل پیچیده را ترکیب می‌کند. Spark یک ستون از کتابخانه‌ها را در برنامه واحد ترکیب می‌کند که شامل SQL و چارچوب های داده، Mllib برای یادگیری ماشین، GraphX و جریان Spark است.

Spark می‌تواند به منابع داده‌های متنوعی مانند HDFS (سیستم فایل توزیع شده Hadoop)، Cassandra (پایگاه داده مبتنی بر ستون)، Hbase (ذخیره‌سازی ستونی پایگاه داده)، Hive و Tachyon دسترسی پیدا کند. Spark از مجموعه داده‌های توزیع انعطاف‌پذیر (RDDs) استفاده می‌کند که انتزاعی از توزیع حافظه توزیع شده با

[1] . Python

خطایی است که در آن پرهیز تکرار Spark می‌تواند تعاملی از 1 تا 2 ترابایت داده را در کمتر از یک ثانیه جست‌وجو کند.

در حالی که Hadoop برای برنامه‌های کاربردی مناسب است، Spark برای اجرای برنامه‌های زمان حقیقی و یا تکراری مانند پردازش گرافیکی مناسب‌تر است و برای برنامه از Hadoop آسان‌تر است.

3.3. پایگاه‌های داده NOSQL و NewSQL

پایگاه داده NoSQL «نه تنها SQL» برای نشان دادن آن است که SQL نیز می‌تواند مورد استفاده قرار گیرد، حتی اگر نماینده سیستم‌های مدیریت داده ارائه شده برای مدیریت داده‌های غیرارتباطی باشد. این پایگاه در قالب ذخیره‌های کلیدی ارزش، ستون، سند و پایگاه‌های داده گراف است. پایگاه‌های داده NoSQL [32 و 36] محدودیت‌های سازگاری کمتری نسبت به پایگاه‌های اطلاعاتی مرتبط دارند. آنها مدل‌های داده‌ای خاص را برای برنامه‌های خاص ارائه می‌دهند. سیستم‌های NoSQL از قاعده نهایی بر پایه قضیه CAP (تطابق، در دسترس بودن، تفکیک) استفاده می‌کنند، بدین معنی که هر سیستم از دسترسی داده‌ها یا ثابت داده‌ها در حالتی پشتیبانی می‌کند که داده‌ها در شبکه تفکیک می‌شوند.

NewSQL کلاسی از پایگاه داده‌های جدید است که نقاط قوت هر دو پایگاه داده‌های ارتباطی و NoSQL را دارد. بنابراین، از SQL و ویژگی‌های ACID پشتیبانی می‌کند و بر پایه معماری مقیاس با پشتیبانی از مقیاس‌پذیری و تحمل خطا ساخته می‌شود. بنابراین، پایگاه دادهNewSQL عملکرد مقیاس‌پذیر قابل مقایسه با سیستم‌های NoSQL را برای کارهای OLTP ارائه می‌دهد. پشتیانی آن از تنوع به دلیل نیاز به طرح (به عنوان مثال، Google spanner، VoltDB،MemSQL ، MemSQL، NuoDB، Clustrix) محدود است. جدول 1 استانداردهای SQL، NoSQL و NewSQL را مقایسه می‌کند.

جدول 1. مقایسه پایگاه‌های داده SQL، NoSQL و NewSQL

	Old SQL	NoSQL	NewSQL
ارتباطی	Yes	No	Yes
SQL	Yes	No	Yes
ذخیره‌سازی ستونی	No	Yes	Yes
نهایت سازگاری	Limited	Yes (Horizontally)	Yes (Horizontally)
BASE (اساساً موجود، قابل انعطاف، ثبات احتمالی)		Yes	
مقادیر بزرگ داده‌های دستی	No	Yes	No
طرح کمتر	No	Yes	Yes
	No	Yes	No

3.3.1. محاسبات در حافظه

محاسبات در حافظه، کلیه داده‌ها را به جای دیسک در RAM دارد. بنابراین هیچ بافر دیسک وجود ندارد. پایگاه‌های داده حافظه (IMDBs) پایگاه‌های ارتباطی سازگار با ACID هستند که SQL را ارائه می‌دهند. طول عمر با تصویرهای لحظه‌ای، ثبت تراکنش و سایر ویژگی‌ها پشتیبانی می‌شود. برای پایگاه داده در حافظه، RDBMS داده را در حافظه با پشتیبانی SQL ذخیره می‌کند. شبکه داده در حافظه (IMDG) چندین ویژگی دارد. داده‌ها را در RAM سرورهای توزیع شده در یک خوشه ذخیره می‌کند. قابلیت بزرگ‌تری در مقایسه با پایگاه داده حافظه وجود دارد اما SQL با پیچیدگی کمتر را پشتیبانی می‌کند. جست‌وجوی SQL توزیع‌شده و قابلیت نمایه‌سازی را محدود می‌کند. ساختار داده کلیدی/ ارزش قابلیت انعطاف را برای توسعه‌دهندگان نرم‌افزار فراهم می‌کند.

3.3.2. محاسبه ابر ترکیبی

محاسبه ابر، همان طور که توسط مؤسسه ملی استاندارد و فناوری تعریف شده است، عبارت است از: «مدلی برای دسترسی به شبکه گسترده و راحت، بر پایه تقاضا برای دسترسی به ائتلاف مشترک منابع محاسباتی قابل تنظیم (مانند شبکه، سرور، ذخیره‌سازی، برنامه‌های کاربردی و خدمات) است که می‌تواند به سرعت با حداقل تلاش مدیریتی و یا تعامل ارائه دهنده خدمات ارائه شود. مدل ابر شامل 5 ویژگی اساسی، 3 مدل خدمات و 4 مدل استقرار

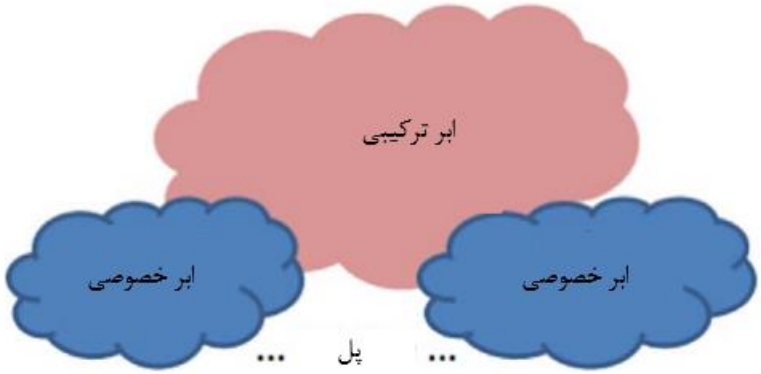
است [29، صفحه 3]». ویژگی های اصلی عبارتند از خدمات بر تقاضا؛ دسترسی به شبکه گسترده جمع‌آوری منابع؛ قابلیت ارتجاعی سریع؛ و خدمات ارزیابی. مدل‌های خدمات عبارتند از نرم‌افزار خدمات (SaaS)، پلت‌فرم خدمات (PaaS) و زیرساخت خدمات (IaaS). چهار مدل استقرار عبارتند از خصوصی، اجتماعی، عمومی و ترکیبی.

با توجه به زیرساخت خدمات، برنامه‌ها، داده‌ها، زمان اجرا، میان افزار[1]، و سیستم عامل توسط مشتری مدیریت می‌شوند، فروشنده مسئول شبکه، ذخیره‌سازی، سرورها و مجازی‌سازی است. با پلت‌فرم خدمات، مشتری تنها برنامه‌ها و داده‌ها را مدیریت می‌کند، در حالی که در نرم‌افزار خدمات، فروشنده مسئولیت کامل مدیریت کلیه عملکردهای فوق را بر عهده دارد.

روش ابر ترکیبی به طور گسترده برای مدیریت داده‌های بزرگ مورد استفاده قرار می‌گیرد. محاسبه ابر ترکیبی اطلاعاتی خصوصی، حساس و بحرانی را در سرور پیش‌فرض ذخیره می‌کند و داده‌های مشترک را در ابر عمومی ذخیره می‌کند که برای کارهای پویا یا بسیار متغیر مفید است. این روش در شکل 4 نشان داده شده است. مزیت در نظر گرفته شده در ابر ترکیبی، امکان بارگذاری کار برای تغییر میان ابرهای خصوصی و عمومی به عنوان تغییرات مورد نیاز برای محاسبات سازمانی را فراهم می‌کند، در نتیجه کسب و کار با انعطاف بیشتر و گزینه‌های بیشتر برای مدیریت داده‌ها ارائه می‌شود.

پلت‌فرم یکپارچه خدمات (iPaas) مجموعه‌ای از خدمات است. این فرایندها، خدمات، برنامه‌ها و داده‌های ابر را در میان سازمان‌های متعدد و یا در سراسر سازمان بهم متصل می‌کنند و در قبال پشتیبانی از توسعه، اجرا و مدیریت جریان‌های ادغام را پشتیبانی می‌کنند (gartner.com). این پلت‌فرم به عنوان پلت‌فرم یکپارچه ترکیبی است که واحدهای تجاری مختلف می‌توانند به شرکت و سایر خدمات متصل شوند؛ مشتریان و تأمین‌کنندگان مرتبط شوند؛ و ادغام یکپارچه‌سازی اینترنت اشیا، رسانه‌های اجتماعی و دستگاه‌های تلفن همراه اتفاق بیافتد.

[1] . middleware



شکل 4. ابر ترکیبی ابر خصوصی و عمومی را ترکیب می‌کند.

محاسبات ابر، استراتژی است که تعمیر و حفظ ناسازگاری، مقیاس‌پذیری سریع و هزینه افزایشی را بر پایه استفاده برای سازمان‌ها فراهم می‌کند. با این وجود، استفاده از محاسبات ابری مستلزم مدیریت خوب داده‌ها، فراداده‌هایی با کیفیت بالا و پردازش یکپارچه اطلاعات است.

3.3.3. روند ETL در مقابل ELT

در ETL سنتی (استخراج، تبدیل و بارگذاری)، تبدیل در مرحله پیش- پردازش قبل از بارگذاری داده‌ها در سرور انجام می‌شود. با داده‌های بزرگ، که در آن مقدار زیادی از اطلاعات به سرعت تولید می‌شوند، ELT (استخراج، بار و تبدیل)، مکان تبدیل در پایگاه داده هدف پس از اولین بارگذاری است. مجازی‌سازی داده‌ها در یک مکان متوقف شده و داده‌ها تبدیل نمی‌شوند. در مقابل، داده‌هایی مورد نیاز در زمان اجرا تبدیل می‌شوند. مجازی‌سازی داده‌ها بدون نیاز به انتقال داده‌ها از مکانی به مکان دیگر، لایه انتزاعی نرم‌افزاری در بالای چندین منبع داده را ارائه می‌دهد. این فناوری اجازه می‌دهد تا بازار داده‌های مجازی و بکپارچه‌سازی زمان حقیقی پلت‌فرم‌های ادغامی برای تحلیل ایجاد شود. نتایج بر ذخیره داده‌ها تأثیر می‌گذارد که در آن، مکان‌های ذخیره‌سازی ذخیره سازمانی می‌توانند داده‌ها را در پایگاه‌های اطلاعاتی ارتباطی سنتی، خوشه‌های داده‌های بزرگ و ذخیره‌های زمان حقیقی ترکیب کنند. داده‌ها برای انواع برنامه‌های کاربردی از جمله OLAP سنتی (پردازش تحلیل آنلاین) و داشبوردها، و همچنین یادگیری ماشین و پارامترهای گزارش تکی استفاده می‌شوند.

3.4. خلاصه‌ای از فناوری‌های داده بزرگ

فناوری‌های داده‌ها بسیار پیچیده هستند و به تکامل نیاز دارند تا مسائل مرتبط با وب، تلفن همراه، رسانه‌های اجتماعی، محاسبات ابری و تحلیل داده‌های بزرگ را حل کنند. اکوسیستم Hadoop می‌تواند تکامل یابد اما هنوز هم برای پردازش دسته‌بندی متوالی بزرگ است. Spark برای پردازش موازی و زمان حقیقی، غالب و مؤثر است. سیستم‌های NoSQL به زبان جست و جوی استاندارد و منظم برای مدل‌سازی احتیاج دارند. محاسبات در حافظه به طور گسترده در NoSQL و NewSQL مورد استفاده قرار می‌گیرند و برای تحلیل مفید هستند. Hadoop و ذخیره داده‌های سازمانی هم‌زیستی دارند و با انبار داده‌های سازمانی در پلت‌فرم داده‌ها یکپارچه می‌شوند. استفاده از محاسبات ابری افزایش یافته اما ادغام ابر ترکیبی و مسائل امنیتی همچنان به عنوان موضوعات مهم مطرح می‌شوند. داده‌های بزرگ نیرو محرکه بسیاری از برنامه‌های کاربردی از جمله اینترنت اشیا و سایر برنامه‌های کاربردی مانند سلامت هوشمند و شهرهای هوشمند هستند.

4. پایگاه داده NoSQL

NoSQL به پایگاه داده‌های منبع باز، توزیع شده و مستقل اشاره دارد. مقیاس افقی مقیاس‌پذیری تحمل خطا را امکان‌پذیر می‌سازد. این پایگاه‌داده کم طرح است، اجازه می‌دهد نوع داده جدید به صورت پویا به پایگاه داده افزوده شود و عملکرد نوشتن افزایش می‌یابد. اکثراً سیستم‌های NoSQL، BASE (اساساً موجود، قابل انعطاف، ثبات احتمالی) را در نظر می‌گیرند که بر خلاف ACID (ثبات، انسجام، انزوا و دوام) برای مدیریت تبدیلات برای افزایش قابلیت دسترسی است و نسبت به انسجام شدید کارا است.

NoSQL به طور خاص برای پاسخ‌گویی به نیازهای داده‌های بزرگ، کاربران بزرگ و ابررایانه طراحی شده است. این نوع از پایگاه‌داده از داده‌های غیرساختاری یا غیرمرتبط (ساختار تو در تو، دسته‌های ستونی، سند، JSON (نشانه‌گذاری شی جاوا اسکریپت)، BSON (سریال سازی باینری JSON) و گراف) پشتیبانی می‌کند. NoSQL چند ویژگی مهم دیگر را نشان می‌دهد. کم طرح است (روش خواندنی، روش ضمنی). می‌تواند در مقیاس وسیع با

هزینه کم و ارزیابی سریع (مقیاس ارتجاعی) مدیریت هزینه عملیاتی کم هزینه را برای تعداد زیادی از کاربران ارائه دهد. این قابلیت مقیاس‌پذیری از عملکرد بالا و تحمل خطا پشتیبانی می‌کند و برای داده‌های زمان حقیقی و غیریکنواخت طراحی شده است.

از جمله فعالیت‌های مهم برای برخورد با مسائل مربوط به داده‌های بزرگ عبارتند از مقیاس‌پذیری، انعطاف‌پذیری طرح، سهولت توسعه، هزینه و در دسترس بودن گزینه‌های استقرار (کارتنر[1]، 2014). انتقال پایگاه داده‌های ارتباطی با پایگاه داده‌های NoSQL نیز به دلیل نیاز به انعطاف‌پذیری هر دو در مدل مقیاس‌گذاری و مدل داده است. از لحاظ مقیاس‌گذاری، در پایگاه داده‌های ارتباطی، افزایش با افزودن سرور بزرگ‌تر در زمانی انجام می‌شود که ظرفیت اضافی مورد نیاز است. در NoSQL، مقیاس‌سازی بدان معناست که به جای دستیابی به سرور بزرگ‌تر، می توان سرورهای کالای بیشتری را افزود.

4.1. مدل‌سازی پایگاه داده NoSQL

از دیدگاه مدل‌سازی داده‌ها، NoSQL بدون عمل اتصال شبیه‌سازی شده است و افزونگی را با تعبیه و لینک‌گذاری ادغام می‌کند. این ویژگی‌ها توسعه نرم‌افزار را با ساده‌سازی نقشه‌برداری بین ساختاری حافظه و ساختار پایگاه داده تسهیل می‌کنند. پایگاه داده ارتباطی دارای «روش بر نوشتن» است در حالی که NoSQL دارای «روش بر خواندن» است. NoSQL دارای دسته ستون پویا با نام ستون تعریف شده در زمان ورود داده‌ها و طرح ضمنی است که با استفاده از برنامه‌نویسی تعریف شده است. شکل 5 نمونه‌ای از دسته ستون پویا را در Hbase ارائه می‌دهد که در آن «کلید سطری» نشان‌دهنده شناسه داده‌های کارمند است.

[1] . Gartner

Employee

RowKey	empID
CF: emp_data	name gender city

Emp Data

empID	name	gender	city
100	Obama	M	DC
200	Smith	F	NY

```
Hbase: CREATE 'employee', 'emp_data'

Hbase: put 'employee', '100', 'emp_data:name', 'Obama'
Hbase: put 'employee', '100', 'emp_data:gender', 'M'
Hbase: put 'employee', '100', 'emp_data:city', 'DC'

Hbase: put 'employee', '200', 'emp_data:name', 'Smith'
Hbase: put 'employee', '100', ', emp_data:gender ', 'F'
Hbase: put 'employee', '100', 'emp_data:city', 'NY'
```

شکل 5. دسته ستون پویا در Hbase

Employee

RowKey	empID
CF: emp_data	name gender city

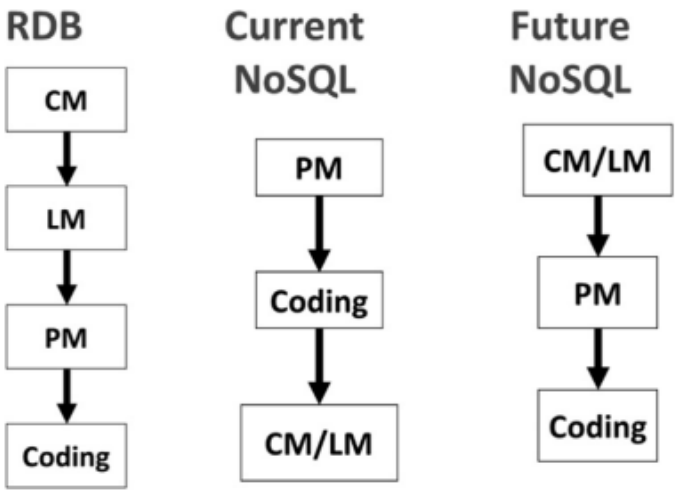
Emp Data

empID	name	gender	city	Priority
100	Obama	M	DC	
200	Smith	F	NY	Next President of US

می‌توانیم ستون جدید را به کارمند انتخاب شده بیافزاییم، بدون در نظر گرفتن سایر سطرها یا بدوم ایجاد مقدار Null برای سایر کارمندان.

```
Hbase: put 'emplyee', '200', 'emp_data:priority', 'Next President of US'
```

شکل 6. یکپارچه‌سازی ستون پویا



شکل 7. مقایسه پایگاهای داده‌ای ارتباطی اخیر NoSQL برای NoSQL (CM= مدل مفهومی، LM= مدل منطقی، PM= مدل فیزیکی).

از دیدگاه مدل‌سازی داده‌ها، چندین تفاوت وجود دارد. با توجه به عدم انطباق ظاهری، پایگاه داده‌های ارتباطی دارای داده کوچک نرمال‌سازی شده است. NoSQL دارای داده‌های غیرارتباطی برای بررسی انواع پنج V و ساختار انحرافی است. تجمیع، واحد عملیات و سازگار است. NoSQL از عملیات پیوستن یا کلیدهای خارجی پشتیبانی نمی‌کند، همان طور که در پایگاه داده‌های ارتباطی انجام می‌شود. اتصالات ترجیحاً در برنامه‌ها انجام می‌شوند. در نتیجه بیشترین جست‌وجو در جدول منفردی انجام می‌شود که باید شامل کلیه داده‌های مربوطه باشد.

پایگاه داده‌های NoSQL کنترل افزونگی را با استفاده از تعبیه و لینک‌گذاری مدیریت می‌کند. تعبیر شامل روابط انحرافی 1:N و M:N و انقباض موجودیت‌های کلاس فرعی در موجودیت‌های ابرکلاس است. این برای بازیابی و کنترل انسجام مناسب است. پیوند (ارجاع) شامل افزودن کلید به شی است و هنگامی مناسب است که اشیا اشاره شده ایستا باشند و روابط تغییر نکنند، مانند ناشر و کتاب.

اکثر پایگاه داده‌های NoSQL دارای انطباق کامل ACID را برای تضمین یکپارچگی تراکنش و انطباق داده‌ها نیستند. بسیاری از پایگاه داده‌های NoSQL، طراحی را تضمین نمی‌کنند زیرا بسیاری از برنامه‌های کاربردی برای ناسازگاری‌های احتمالی به بروزرسانی احتیاج دارند. انطباق احتمالی، استفاده از پایگاه داده‌های NoSQL را برای

برنامه‌های کاربردی انتقالی بحرانی محدود می‌کند. با این وجود، تغییرات زیادی وجود دارد، حتی در دسته مشابه سیستم‌های NoSQL.

4.2. انواع پایگاه داده NoSQL

4 نوع عمده پایگاه داده‌های NoSQL وجود دارد که هر یک همراه با سطوح مختلف مقیاس‌پذیری، انعطاف‌پذیری، پیچیدگی و عملکرد می‌باشند (academy.datastax.com).

4.2.1. ذخیره کلید- ارزش

پایگاه داده‌های ذخیره کلید- ارزش، اطلاعات ذخیره شده را به شیوه‌ای ساده‌تر با کلیه داده‌های متشکل از یک کلید شاخص و یک مقدار (مانند Oracle NoSQL و Redis) ذخیره می‌کند. جدول ساده هش[1] برای دسترسی به مقدار، تنها با یک کلید قابل دسترس است. ارزش می‌تواند هر نوع داده («ترکیبی») در هر اندازه‌ای باشد و به عنوان یک قطره محسوب می‌شود. برخی از مدل‌ها شاخص‌هایی را روی ارزش‌ها ارائه می‌دهند. هدف طراحی بارگذاری عظیم است که دسترسی آن به کلید بسیار سریع باشد، بسیار سریع‌تر از SQL.

کلید و مقدار می‌توانند شی ترکیبی پیچیده‌ای باشند. این برای داده‌هایی است که تنها از طریق یک کلید قابل دسترس هستند و برای ارتباط نیاز به پرونده‌های چندگانه ندارند. برای خواندن اجزای پست‌ها، باید تمام اشیا را تجزیه کنیم. مزایای آن دسترسی به صورت دستی از طریق کلید، مقیاس‌پذیری، توزیع آسان در میان خوشه‌ها و ارائه یک مدل ساده به عنوان جدول هش است. از جمله معایبش این است که هیچ‌گونه فیلترینگ پیچیده وجود ندارد، پیوستن باید در برنامه‌ها انجام شود و هیچ مکانی برای حمایت از ثبات چندضبطی وجود ندارد.

---

[1]. Hash

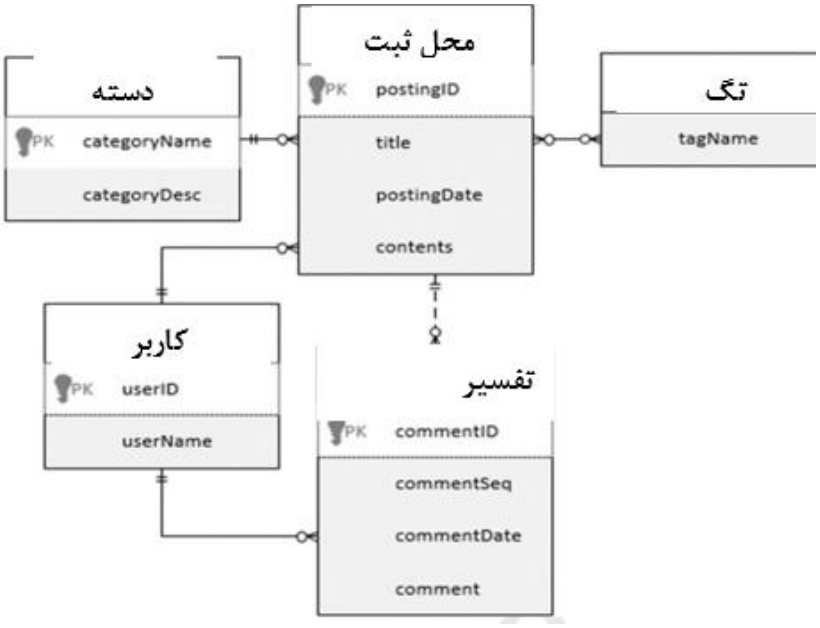
4.2.2. ذخیره ستون

ذخیره ستون (به عنوان ذخیره ستون- گسترده نیز شناخته می‌شود) توسط Google's BigTable ارائه شد. به جای ذخیره داده‌ها در ردیف‌ها، جدول داده‌ها به عنوان بخش‌هایی از ستون‌های داده ذخیره می‌شوند. این تعمیم پایگاه داده ذخیره کلید- مقدار است که در آن ستون‌ها می‌توانند ساختار پیچیده‌ای داشته باشند، نه یک مقدار قطره‌ای. پایگاه داده ذخیره ستونی از ساختار مدل‌سازی پیچیده (جداول تو در تو، گروه‌های تکراری، مجموعه، لیست و غیره) پشتیبانی می‌کند. دسته ستون (فوق دسته در کاساندرا[1] نامیده می‌شود) گروهی از ستون‌های مرتبط است که حفظ سازگاری را تضمین می‌کند. ذخیره ستون دارای عملکرد بالا و معماری بسیار مقیاس‌پذیر است.

ساختار کلید ذخیره ستون متشکل از {کلید سطری، نام دسته ستون، نام توصیف کننده ستون، زمان‌بندی} است. زمان بندی از نسخه پشتیبانی می‌کند؛ به طور پیش فرض، جدیدترین مقدار بازیابی می‌شود. بنابراین، تحلیل سریع زمان از طریق مقدار ستون امکان پذیر است. کلید ردیف می‌تواند کلید ترکیبی باشد اما به عنوان کلید تنها ذخیره می‌شود.

جنبه‌های مدل‌سازی پایگاه داده‌های ذخیره ستون: ذخیره‌های ستون از ساختارهای مدل‌سازی غنی (ستون‌ها، ستون‌های ترکیبی، ساختارهای انحراف، جداول تو در تو، ترکیبات، شاخص‌های ثانویه، نقشه، مجموعه و لیست) پشتیبانی می‌کنند. داده‌ها ذخیره می‌سوند تا تمام داده‌ها با هم قابل دسترس باشند. انحراف متداول اسا، اما منجر به مشکلات افزونگی و انسجام می‌شود.

[1] . Cassandra



شکل 8. مدل موجودیت- رابطه برای پایگاه‌های داده ذخیره‌سازی ستون NoSQL

ابتدا باید اصلاح جست‌وجو در نظر گرفته شود و سپس ساختار فیزیکی عملکرد جست‌وجو طراحی شود. توسعه‌دهندگان جدول جمع‌آوری شده از هر درخواست را ایجاد می‌کنند. روش‌های متعددی برای مدل‌سازی جست‌وجوی مورد نیاز وجود دارد. همان‌طور که در ادامه نشان داده می‌شود، نمودارهای ارتباطی اساسی می‌توانند به طور گسترده به طراحی پایگاه‌های داده ذخیره‌سازی ستون NoSQL کمک کنند. مثال شکل 8 را در نظر بگیرید. سؤال این است «چه گزینه‌هایی از طرح NoSQL را در این نمودار رابطه موجودیت داریم؟».

برای رابطه 1:N میان دسته و محل ثبت، دو جست‌وجو الزامی قابل قبول وجود دارد:

جست‌وجو 1: برای هر دسته، کلیه محل‌های ثبت را نشان می‌دهد.

جست‌وجو 2: نمایش محل‌های ثبت با دسته‌هایشان.

شکل 9 رابطه میان دسته و محل ثبت را نشان می‌دهد. جست‌وجو اول «برای هر دسته، کلیه محل‌های ثبت را نشان می‌دهد» با تعبیه داده‌های ارسالی در رده با استفاده از دسته ستون و به صورت تو در تو انجام می‌شود.

الزامات جست‌وجو

- برای هر دسته کلیه محل‌های ثبت را نشان می‌دهد.

PostingPerCategory

RowKey	categoryName
CF: postingData	Nested Posting postingID title postingDate contents

- تعبیه: محل تعبیه داده‌ها در جدول دسته
- استفاده از دسته ستون
- استفاده از ساختار تو در تو

شکل 9. رابطه 1:N با تعبیه برای جست‌وجو 1

پاسخ ارتباطی در شکل 10 ارائه شده است.

الزامات جست‌وجو

- برای هر دسته کلیه محل‌های ثبت را نشان می‌دهد.

- تعبیه داده‌های دسته درجدول محل ثبت با انحراف
- استفاده از ستون ها

PostingsPerCategory

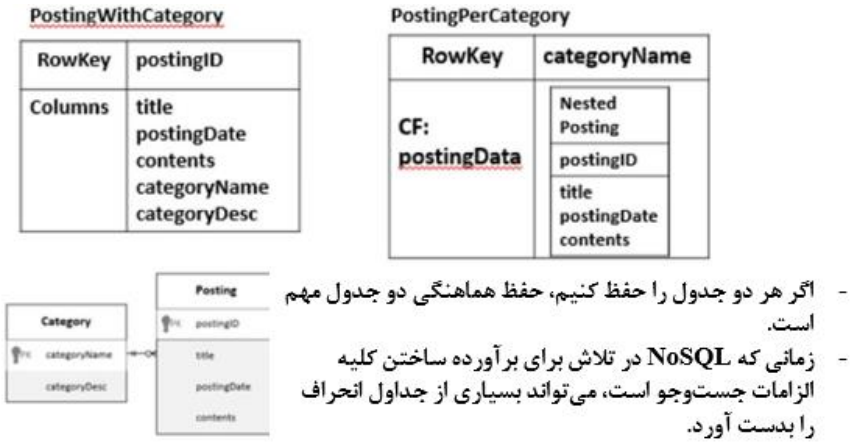
RowKey	categoryName
Columns	{postingID}

شکل 10. رابطه 1:N با ارتباط برای جست‌وجو 1

جست‌وجو دوم، « نمایش محل‌های ثبت با دسته‌هایشان» در شکل 11 ارائه شده است. داده‌های گروهی در جدول ثبت با انحراف درج شده اند.

شکل11. رابطه 1:N با تعبیه برای جست‌وجو 2

اگر هر دو جدول را حفظ کنیم، حفظ هماهنگی دو جدول مهم است. همان طور که در شکل 12 نشان داده شده است، زمانی که NoSQL در تلاش برای برآورده ساختن کلیه الزامات جست‌وجو است، می‌تواند بسیاری از جداول انحراف را بدست آورد.



شکل 12. نمایش کاهشی برای رابطه 1:N برای جست‌وجو 1 و 2.

برای رابطه M:N میان محل ثبت و تگ[1]، دو الزام جست‌وجو وجود دارد: هر تگ، کلیه محل‌های ثبت مرتبط را نشان می‌دهد؛ و تگ‌های هر ثبت نشان داده می‌شوند. روش‌های مختلفی برای مدل‌سازی وجود دارد که الزامات جست‌وجو را ساده می‌کند. اگر ویژگی رابطه M:N وجود داشته باشد، مدل NoSQL را بیشتر می‌توان تغییر داد.

[1] . Tag

چندین مسأله مهم به وجود می‌آید. برای هر مسیر دسترسی، باید جدول یکپارچه‌ای ایجاد کرد که در آن کلید ردیف با ویژگی مورد استفاده در جست‌وجو آغاز شود. برای رابطه 1:N، معمولاً از ساختار انحراف یافته استفاده می‌شود. در روابط M:N، روش‌های متعددی برای یکپارچگی وجود دارد: نگرش- متمرکز، رابطه - متمرکز، روش دسترسی، دسترسی دوجانبه، زمان معکوس و غیره. اگر مجموع وابستگی همراه با یکپارچگی پیوسته باشد، دسته ستون را می‌توان در یکپارچگی مورد استفاده قرار داد. اگر زیرگروه یک گروه تکراری باشد، می‌توان از ساختار تو در تو استفاده کرد. مهم‌ترین حقیقت این است که نمودارهای رابطه – متمرکز برای طراحی طرح NoSQL مفید هستند.

ذخیره‌های ستونی از ساختارهای مدل‌سازی متنوعی برخوردارند و دارای انعطاف‌پذیری طرح (طرح پویا) هستند. با این وجود، طراحی به مسیرهای دسترسی بستگی دارد. در جداول چندگانه داده‌های اضافی وجود دارند که به مدیریت دقیق جامع در سطح برنامه‌نویسی احتیاج دارند. زمانی که تلاش می‌کنیم تا کلیه الزامات جست‌وجو را فراهم کنیم، تعداد جداول ذخیره ستونی می‌تواند بزرگ‌تر از تعداد آن در پایگاه داده رابطه‌ای باشد. حفظ یکپارچگی داده در سطح برنامه‌نویسی مهم است. پردازش جست‌وجوی غیرپیچیده در اکثر ذخیره‌های ستونی در دسترس نیست.

4.2.3. ذخیره سند

ذخیره سند، مجموعه‌ای از ذخیره‌های کلید- ارزش است که در آن ارزش، سند است، مانند JSON، BSON و غیره. هر سند دارای یک کلید منحصر به فرد است که برای بازیابی سند تعیین شده است. هر مجموعه داده می‌تواند به عنوان ساختارهای تو در تو، نقشه‌ها، مجموعه‌ها و مقادیر اسکالر ذخیره شود. شاخص‌های ثانویه برای دسترسی به یک جزء سند وجود دارد. ذخیره‌های سند در MongoDB، CouchDB و غیره یافت می‌شوند.

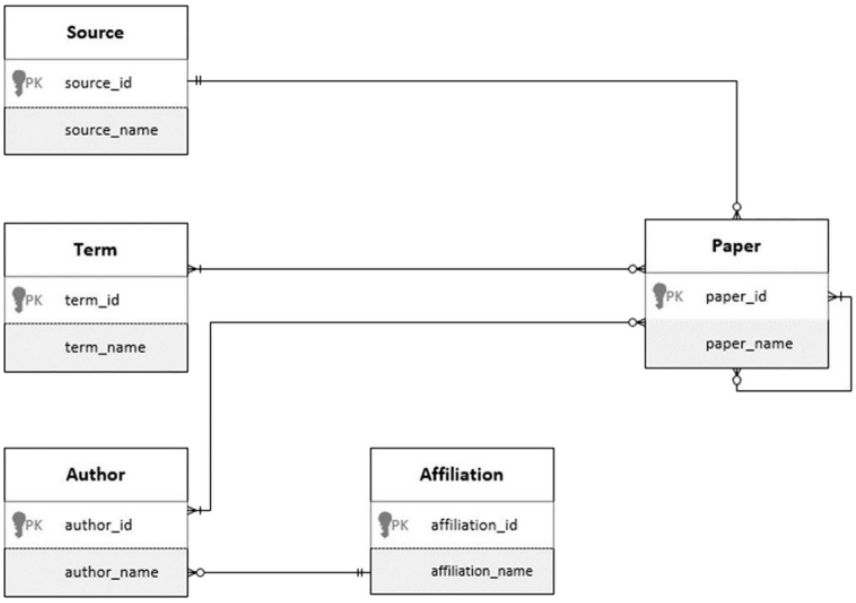
از دیدگاه مدل‌سازی، ذخیره‌های سند برای داده‌های نیمه ساختاری خوب هستند. ذخیره‌های سند از یکپارچگی و ساختارهای دوباره نرمال‌سازی شده پشتیبانی می‌کنند. کلیه روابط 1:1 و 1:N را می‌توان در یک سند تعبیه کرد. روابط ایستا مرجع، مانند ارتباط میان کتاب و ناشر، نیز امکان‌پذیر است. در ساختارهای مدل‌سازی غنی (ساختار

توجیه شده، آرایه متغیر) ذخیره‌های سند قوی‌تر از ستون‌ها هستند و برای مجموعه داده‌های کمی کوچک مناسب هستند. با این وجود، فقدان استاندارد برای ساختارهای مدل‌سازی و زبان جست‌وجو وجود دارد.

4.2.4. پایگاه داده گراف

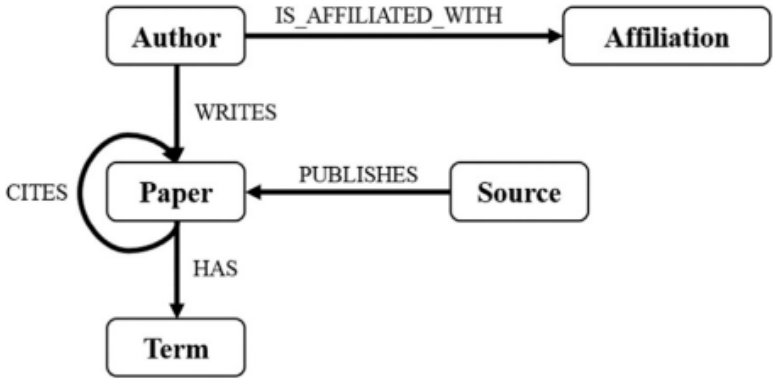
پایگاه داده گراف بر اساس نظریه گراف (مجموعه‌ای از گراف‌ها، لبه‌ها و ویژگی‌ها) بوده و برای داده‌های مربوط به روابط متقابل، مانند الگوهای ارتباطی، شبکه‌های اجتماعی و تعاملات زندگی‌نامه مفید است. این پایگاه اجازه می‌دهد تا پرسش‌های عمیق‌تر، پیچیده‌تر و پرسش‌های متقاضیان بیان شود. با این وجود، آسان نیست که مؤلفه‌های یک گراف را در میان شبکه‌های سرور به عنوان نمودارهایی که بزرگ‌تر می‌شوند، توزیع کنیم. این ممکن است به طور دقیق به عنوان غیررابطه‌ای، NoSQL غیرحقیقی، بیان شود. Neo4j، Titan و OrientDB از جمله مثال‌ها هستند.

به عنوان مثال، سیستم بازاریابی اطلاعات کتابشناختی تعاملی مبتنی بر گراف (GIBIR) به طور مؤثر در ارتباط با مقالات، نویسندگان، استنادها، وابستگی، منابع، اصطلاحات و غیره طراحی می‌شوند، ژو و همکاران ($سال$) [55]. سیستم بازاریابی اطلاعات کتابشناختی تعاملی مبتنی بر گراف در Neo4j اجرا می‌شود، به طور گسترده مقیاس‌پذیر است، و منبع بازی است که از ACID پشتیبانی می‌کند. گراف‌های در دیسک ذخیره شده در فرمت باینری ذخیره می‌شوند. مدل نگرش- متمرکز [7] برای سیستم بازاریابی اطلاعات کتابشناختی تعاملی مبتنی بر گراف در شکل 13 ارائه شده است که سودمندی نمایه مدل مفهومی را برای پایگاه داده گراف نشان می‌دهد.



شکل 13. مدل نگرش- متمرکز برای سیستم بازاریابی اطلاعات کتابشناختی تعاملی مبتنی بر گراف

نمودار گراف متناظر شکل 13 در شکل 14 ارائه شده است که نشان‌دهنده موجودیت‌ها و روابط کتابشناختی موضوعی است.



شکل 14. گراف ویژگی سیستم بازاریابی اطلاعات کتابشناختی تعاملی مبتنی بر گراف.

عملکرد: با توجه به عملکرد، سیستم مبتنی بر مدل گراف بیشتر از سیستم مبتنی بر مدل رابطه ای در بازاریابی اطلاعات کتابشناختی عمل می‌کند، زیرا سیستم گراف به جای اتصال به آن، حرکت می‌کند. پایگاه داده ارتباطی در اجرای نمایش داده‌ها بهتر عمل می‌کند و تنها با دو گره اجرا می‌شود. با این وجود، چون جست‌و جوها پیچیده می‌شوند، پایگاه داده گراف بیشتر از پایگاه داده ارتباطی اجرا می‌شود.

مدل‌سازی مفهومی: نمودار نگرش- رابطه به راحتی به مدل نمودار ویژگی تبدیل می‌شود و به یک مدل مفهومی برای پایگاه‌های داده گراف نیاز دارد. این نمودار کمک می‌کند تا بدانیم کدام موجودیت‌ها را می‌توان به صورت منطقی به سایر موجودیت‌ها مرتبط کرد. پایگاه‌های داده گراف تنها از روابط باینری پشتیبانی می‌کنند. مدل‌سازی گراف بسیار ساده‌تر از مدل رابطه‌ای است زیرا اشیای دنیای حقیقی از لحاظ ارتباطات صریح هستند.

ارزیابی: پایگاه داده گراف برای برنامه‌های کاربردی با روابط پیچیده مناسب است. هنگامی که چند گره درگیر هستند، سریع‌تر از مدل‌های ارتباطی است. با این وجود، پایگاه‌های داده گراف قابل خواندن نیستند. گراف‌های بزرگ ممکن است در گره واحد در حافظه قرار نگیرند. تفکیک گراف باعث مشکلات زیادی می‌شود. پیام‌های ارتباطی بین گره بحرانی هستند. در حال حاضر APIsهای استاندارد و یا زبان جست‌وجو برای پایگاه‌های داده گراف وجود ندارند.

4.3. مسائل و موارد استفاده از پایگاه‌های داده NoSQL

NoSQL برای OLTP (فرایند تراکنش آنلاین) خوب نیست و به ویژگی‌های ACID و به روز رسانی مکرر نیاز دارد. در برخی از برنامه‌های کاربردی که در آن عملکرد اهمیت بیشتری دارد، سازگاری احتمالی قابل قبولی دارد. مثال‌ها شامل موتورهای جست‌وجویی هستند که به سرعت نتایج جست‌وجو مختلف را نمایش می‌دهند و یا به سیستم‌های پیشنهاددهنده‌ای احتیاج دارد که می‌تواند نتایج بررسی‌های مختلف را نمایش دهند. اگرچه، تولیدات NoSQL به سرعت در حال تکامل هستند. برخی از سیستم‌ها مانند FoundationDB و OrientDB ادعا می‌کنند که از ACID پشتیبانی می‌کنند.

تأثیرات پایگاه‌های داده NoSQL را می‌توان در بسیاری از موارد به کار برد. برنامه‌های موبایل، اینترنت اشیا، و تحلیل داده‌ها در زمان حقیقی، از پایگاه‌های کلید- مقدار، ذخیره ستون و سند استفاده می‌کنند. سیستم‌های مدیریت اسناد ساختار یافته و غیرساختاری چند منبعی از ذخیره‌های سند استفاده می‌کنند. شخصی‌سازی، سیستم‌های مدیریت محتوا، یادگیری و سیستم‌های کاتولوگ از پایگاه‌های ذخیره ستون و ذخیره سند استفاده می‌کنند. سیستم‌های مدیریت سلامت و نسخه بیماران از پایگاه‌های داده کلید- مقدار مانند بازی زمان حقیقی

استفاده می‌کنند. سرویس‌های بازاریابی بزرگ در تلاش برای بدست آوردن دید کلی مشتریان ذخیره ستون مورد استفاده قرار می‌گیرند. مدیریت داده‌ها از شبکه‌های اجتماعی پایگاه‌های داده گراف استفاده می‌کنند. پایگاه‌های داده گراف برای مدیریت داده‌های اصلی و سیستم‌های مدیریت نسخه و مدیریت ارتباط پیچیده دارو مورد استفاده قرار می‌گیرند.

5. مدل‌سازی مفهومی و مدیریت داده‌های بزرگ

مدل‌سازی مفهومی از ابتدا بر سازمان‌دهی داده‌ها متمرکز شده بود [7 و 14]. این مستلزم ایجاد نمای مفهومی دامنه کاربرد سیستم اطلاعات است [51 و 53]. برای داده‌های بزرگ، اهمیت مدل‌سازی مفهومی را می‌توان از نظر فنی و مدیریتی مورد توجه قرار داد.

مدل‌سازی مفهومی برای توصیف معانی کاربردی نرم‌افزار در نظر گرفته می‌شود. طراحان مفهومی مدل‌های ساختاری، رفتار یا مدل‌های عملکردی، و همچنین تعاملات و روابط کاربر را توصیف می‌کنند. چنین مدل‌هایی مشتریان و تحلیل‌گران را قادر می‌سازند تا یکدیگر را درک کنند، در نتیجه ارتباطات مفید بین آنها تقویت می‌شود (www. conceptualmodeling.org). مدل‌سازی مفهومی برای مفاهیم انتزاعی و تحلیلی مفید است و می‌تواند به درک ، مدیریت، طراحی و آموزش تکنیک‌های پایگاه داده کمک کند. تکنیک‌های مدل‌سازی مفهومی باید مسائل مربوط به ادغام داده‌ها، انبوه‌سازی داده‌ها، ابر، ابرداده و پردازش را در نظر بگیرند. امبلی[1] و لیدل[2] در [14] نشان دادند که داده‌های بزرگ و همچنین اطلاعات منظم مورد به مدل‌سازی دقیق احتیاج دارند تا از آن به عنوان نماینده دنیای حقیقی حاصل به طور مناسب استفاده کنند. دانشمندان و طراحانی که در مدل‌سازی مفهومی آموزش دیده‌اند باید «متفکران با استعداد» باشند بدین معنی که آنها قادر به انتزاع، نمایندگی، استخراج، مدیریت، تحلیل و تجسم هستند.

[1] . Embley
[2] . Liddle

اگرچه مدل‌سازی مفهومی برای مدل‌سازی داده‌های بزرگ مفید است، اما مدل‌سازی مفهومی هنوز چالش‌های بسیاری برای اطمینان از موفقیت پروژه‌های داده‌های بزرگ به همراه دارد:

• درک/ استفاده/ اعمال چرخه زندگی- درک چرخه زندگی سنتی برای پایگاه داده‌های منظم و همچنین چرخه زندگی داده بزرگ که چالش‌های داده بزرگ را بررسی می‌کند.

• شناسایی نیازهای داده‌های بزرگ و مرتبط ساختن آن با اهداف تجاری و فناوری- حجم داده‌های بزرگ بسیار بزرگ است، بنابراین طراحان مفهومی باید قادر به شناسایی الزاماتی باشند که مربوط به اهداف کسب‌وکار و انتخاب فناوری اطلاعات مناسب برای مسأله و شرایط حاصل است.

• شناسایی موارد استفاده، ساخت معماری و انتخاب سیستم عامل- این می‌تواند در ایجاد سیستم عامل مناسب مفید باشد.

• سرپرست پروژهای داده بزرگ- مدل‌سازان مفهومی می‌توانند نقش مسئول داده‌های اصلی (CDO) را بازی کنند که مسئول مدیریت پروژه‌های داده بزرگ است.

• انجام تحلیل در سطح بالا- پیش از شروع پروژه تحلیل اطلاعات دقیق داده‌های بزرگ، طراحان مفهومی باید قادر به تحلیل سطح بالا برای درک بازگشت سرمایه در پروژه تحلیل باشند.

• ارزیابی پروژه‌های داده بزرگ- هر پروژه مدیریت داده‌ها، باید پروژه‌های داده بزرگ را امکان‌سنجی کند، و سودمندی و کیفیت آنها را ارزیابی کند.

• کمک به صحت مدیریت داده‌های بزرگ- شناخت مستمر سودمندی مشخصه‌های داده بزرگ مورد نیاز است.

• تحلیل دامنه، حکمرانی و مدیران ارشد داده‌ها- فعالیت عمومی مرتبط با پروژه‌های سنتی و همچنین داده‌های بزرگ باید مدیریت شوند.

این فعالیت‌ها به طور کل به موفقیت پروژه‌های داده بزرگ کمک می‌کنند. مدل‌سازی مفهومی به طور کلی، پنج V داده‌های بزرگ را به صورت زیر شرح می‌دهد:

• حجم: یک ویژگی فیزیکی است، اما مدل‌سازی مفهومی می‌تواند داده‌های مهم و فراداده‌ها را سازمان‌دهی، شناسایی و توصیف کند.

• سرعت: داده‌ها باید فیلتر شوند، اما مدل‌سازی مفهومی می‌تواند به استخراج داده‌های مهم کمک کند.

• تنوع: مدل‌سازی مفهومی می‌تواند تنوع، سلسله مراتب و شبکه‌های داده را مدل‌سازی کند، داده‌ها را ادغام کرده و مسائل مربوط به انبوه‌سازی داده‌های بزرگ را ارائه دهد.

• اطمینان: مدل‌سازی مفهومی می‌تواند کیفیت، تکمیل بودن و سازگاری را بررسی کند.

• ارزش: مدل‌سازی مفهومی می‌تواند پروژه‌های بزرگ را مدیریت کند، شامل تحلیل آنها برای استخراج ارزش و ارزیابی نتایج است.

بنابراین، مدل‌سازی مفهومی می‌تواند نقش مهمی در ابتکارات داده‌های بزرگ بازی کند.

5.1. چالش‌ها

چالش‌ها و مشکلات زیادی در ارتباط با مدیریت داده‌های بزرگ وجود دارد که قابل توجه‌ترین آنها به شرح زیر است.

فرهنگ داده محور: داده‌ها باید به صورت عینی و بدون تکیه بر شهود در نظر گرفته شوند. این از مدیریت موفق زنجیره- تأمین شرکت‌هایی مانند اپل[1]، گوگل[2] و وال- مارت[3] مشهود است.

اهداف کسب و کار: پروژه را به اهداف کسب‌وکار مرتبط کنید. اطمینان حاصل کنید که پروژه داده‌های بزرگ متناسب با نیازهای کاربرد کسب‌وکار است.

به‌کارگیری استعداد تحلیلی اثبات شده: استعداد تحلیلی مورد نیاز را بدست آورید. نوآوری که توسط چنین استعدادی شناسایی می‌شود، می‌تواند از تشخیص، پیش‌بینی و تحلیل مجدد استفاده کند.

درک پاسخ‌های داده بزرگ: پاسخ‌ها می توانند شامل اکوسیستم‌های Hadoop، Spark، NoSQL، ابر، محاسبات در حافظه و مجازی‌سازی داده‌ها باشد.

---

[1] . Apple
[2] . Google
[3]. Wal-mart

امنیت: همانند پروژه‌های مدیریت منظم داده‌ها، امنیت داده‌ها همیشه یک چالش بزرگ است.

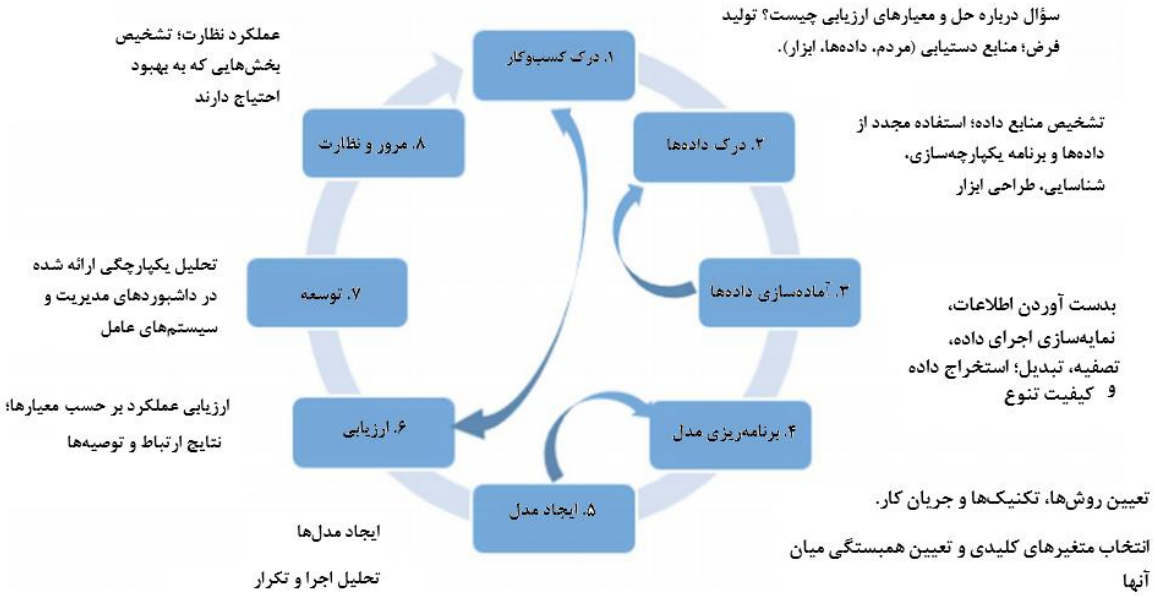
رهبری: رهبری در طول کل چرخه عمر تحلیلی داده‌ها مورد نیاز است.

5.2. چرخه حیات تحلیل داده‌ها

فرایند استاندارد صنایع- متقابل برای داده‌کاوی، CRISP- DM [6] شامل چندین جزء است. درک کسب و کار شامل درک سؤالاتی است که باید پاسخ داده شوند و درک معیارهایی است که برای ارزیابی مناسب هستند. درک داده‌ها به شناسایی منابع داده و ابزار مناسب احتیاج دارد. ارائه داده‌ها شامل ارائه داده‌ها، مرتب کردن و تبدیل داده‌ها، و تأیید کیفیت داده‌ها است. برنامه‌ریزی مدل، روش‌ها، تکنیک‌ها و جریان‌های کاری را شناسایی می کند. متغیرهای کلیدی انتخاب شده و همبستگی میان آنها مشخص می‌شود. ارزیابی شامل نتایج ارزیابی معیارها و برقراری ارتباط با نتایج است. در پایان، استقرار، یکپارچه‌سازی روش‌های تحلیل در داشبورد مدیریت و سیستم عامل است.

چرخه زندگی تحلیلی داده‌ها در شکل 15 ارائه شده است [41]. این نتیجه حاصل استراتژی کسب‌وکار برای سؤالات، کسب اطلاعات، ارائه راه‌حل، ارزیابی و نظارت است.

مدیر ارشد اطلاعات (CDO) شخصی است که به عنوان فرد مسئول اداره داده‌ها برای اهداف حکومت و نوآوری معرفی شده است. مدیر ارشد اطلاعات باید بتواند دیدگاه و استراتژی را برای ابتکارات مدیریت داده‌ها ارائه دهد و سایر وظایف را همان طور که در شکل 16 نشان داده شده انجام دهد [41]. این وظایف فعالیت‌ها را از تحلیل مفهومی تا استقرار پوشش می‌دهد و اهمیت دستیابی به دانش مدل‌سازی مفهومی را برجسته می‌سازد.



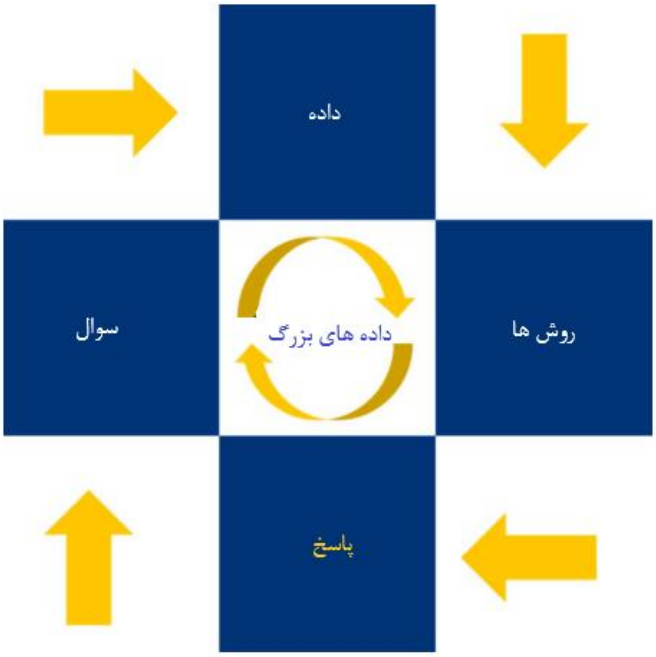
شکل 15. چرخه زندگی تحلیل داده‌ها

شکل 16. مدیر ارشد اطلاعات

6. بررسی

مسائل داده‌های بزرگ نیز از جمله مسائل علوم کامپیوتر هستند. برای حل آنها، ابتدا باید حالت کسب‌وکار را برای تحلیل داده‌های بزرگ ایجاد کرد. داده‌ها باید مورد بررسی قرار گیرند تا شناسایی و ادغام شوند و منابع مختلف

چندگانه را ترکیب کنند. داده‌های خروجی مورد نظر باید برای تعیین نحوه محاسبه نتایج مورد نظر ارزیابی شوند. پس از آن نتایج حاصل باید برای یافتن جنبه‌های جدید کشف دانش تفسیر و ارزیابی شوند. انتزاع مسائل مدیریت



شکل 17. ابتکارات مدیریت داده‌های بزرگ

فناوری‌های داده‌های بزرگ، پیچیده و در حال تکامل هستند. برای پاسخ به بسیاری از مسائل مربوط به وب، تلفن همراه، اجتماعی، ابر و تحلیل داده‌های بزرگ به آن احتیاج داریم زیرا همچنان به تکامل ادامه می‌دهند. اکوسیستم Hadoop نیز در حال تکامل است. با این وجود، Hadoop علی رغم هشدار، نرخ پذیرش نسبتاً کندی دارد. با این وجود، Spark نیز به سرعت در حال پردازش موازی زمان حقیقی است. سیستم‌های NoSQL برای مدل‌سازی به زبان‌های استاندارد و منظمی احتیاج دارند. محاسبات در حافظه به طور گسترده در NoSQL مورد استفاده قرار می‌گیرد، زیرا NoSQL برای تحلیل مفید است. با وجود همکاری فعلی Hadoop و انبار داده‌های سازمانی (EDW)، انبار داده‌های سازمانی به پلتفرم مدیریت یکپارچه داده‌ها تبدیل شده است. تحولات اجتماعی و فناوری جدید همچنان به داده‌های بزرگ مانند اینترنت اشیا (IoT) [54] می‌پردازد و به طور گسترده از محتوای شخصی و حرفه‌ای کاربر در رسانه‌های اجتماعی و سایر رسانه‌هایی استفاده می‌کند که برای تحلیل حساسیت و فرم‌های اضافی مورد استفاده قرار می‌گیرند. برای پایگاه‌های داده NoSQL، موضوعات تحقیقاتی فنی خاص عبارتند از: الگوهای

طراحی، اجرای دو مرحله‌ای، تجسم، تقسیم‌بندی، نمودار با استفاده از داده‌کاوی و مدل‌سازی مفهومی در پایگاه داده‌های گراف و زبان‌های جست‌وجو استاندارد.

از دیدگاه مدیریت، مدیریت چرخه عمر داده‌های بزرگ باید بر پنج V متمرکز باشد. سؤالات تحقیق عبارتند از: چگونه می‌توان یکی از متاداده‌های درون چرخه داده‌های بزرگ (خرید، انتقال، جمع‌آوری، توزیع، ادغام، ارائه، تحلیل، تجسم، حکمرانی، مهاجرت، نگهداری، مراقبت) را بدست آورد؟ حداقل مجموعه‌های ضروری برای پشتیبانی از مدیریت چرخه عمر اطلاعات ضروری چیست؟ چگونه می‌توان برچسب زدن، مدیریت، پیگیری و استفاده از ابرداده‌ها را در پروژه‌های حقیقی انجام داد؟ بهترین روش برای انجام چنین کاری چیست؟

پروژه‌های تحقیقاتی در حال انجام زیادی در حوزه‌های مختلف وجود دارند که مبتنی بر تأثیر استفاده گسترده از داده‌های بزرگ می‌باشند. از جمله موضوعات در میان آنها عبارتند از موضوعات هوشمند، سلامت هوشمند، پیری هوشمند و شناخت خود. برخی از سیستم‌ها بر پشتیبانی از این موضوعات ساخته شده اند، برنامه‌ریزی بسیار دشوار است، و اخذ تصمیم سخت است. بقیه سریع‌تر تصمیم می‌گیرند (به عنوان مثال، با استفاده از برنامه‌های تلفن همراه). پیشرفت‌های پیشرو در مراقبت های بهداشتی به عنوان مثال شامل کاربرد واتسون[1] (https://www.ibm.com/watson/health/) برای مرقبت از بیمار است. پیری هوشمند، با استفاده از فناوری‌های پزشکی، رایانه و ارتباطات، تأثیری پیری را بر سالمندان کاهش می‌دهد. نتایج موقعیت‌هایی برای پیری سالم هستند. با استفاده از فناوری‌های مختلف پوششی و سنسورهای مختلف برای نظارت بر فعالیت‌های بیمار و سالخورده می‌توان مقادیر زیادی از داده‌های مرسوم مختلف مانند جریان‌های ویدئویی، تصاویر، صوت و داده‌های متنی را ارائه داد. آنها عبارتند از مفاهیم الگو مبتنی بر داده که مردم را به سوی انتخاب‌های سلامتی «بفرستند»، مانند: تشخیص زودهنگام عادات نادر (به عنوان مثال، تغذیه، خواب، فعال بودن)؛ مداخله زودهنگام برای جایگزین‌های سالم پیشنهادی؛ ترجیحات ترکیبی در بیمه، درمان، خدمات اجتماعی و فعالیت‌ها. مدل‌سازی مفهومی

[1] . Watson

می‌تواند نقش مهمی را در مدل‌سازی، حاشیه‌نویسی، ادغام و استخراج داده‌های مربوطه و ابرداده برای تحلیل و گزینش بازی کنند.

پروژه‌هایی که شامل داده‌های بزرگ هستند، می توانند تأثیر زیادی بر نیازهای مدیریت پروژه داشته باشند، به ویژه با توجه به توان بالقوه آن‌ها برای مدیریت عملیات و جامعه، به عنوان مثال، استفاده از مجموعه داده‌های بزرگ‌تر نسبت به گذشته بر پایه تحلیل داده‌های جدید و یا استخراج الگوی نتایج. در جهان داده‌های بزرگ، فعالیت‌های مدل‌سازی مفهومی مفید و معتبر هستند. باید الزامات داده‌های بزرگ شناسایی، درک و نمایش داده شوند. مدل‌سازی پایگاه داده NoSQL متفاوت از مدل‌سازی پایگاه داده ارتباطی سنتی است و برای درک مفاهیم داده و معماری طراحی، به مدل‌سازان احتیاج دارد.

تلاش برای مدل‌سازی مفهومی برای درک مفاهیم مربوط به داده‌های بزرگ و ایجاد معماری طراحی ضروری است. Vهای داده‌های بزرگ، روش مفید برای در نظر گرفتن دلایل خاص برای مدل‌سازی مفهومی هستند. با توجه به کار روی تنوع، صحت و ارزش، مدل‌سازی مفهومی نقش‌های متفاوتی را بازی می‌کنند. در رابطه با انواع داده، NoSQL، الگوهای طراحی، روش‌ها و ابزار مورد نیاز هستند. درستی داده‌ها به حل مسائل مربوط به داده‌ها و کیفیت سیستم می‌پردازد. مدیریت کیفیت چالش برانگیز اما مهم است. برای بدست آوردن ارزش، باید معماری و راه‌حل‌های صحیح مدل‌سازی شوند که می‌توانند دانش عملی را برای بهبود فرایندهای کسب‌وکار ارائه دهند. همچنین مسائل قدیمی مربوط به کیفیت داده‌ها [52] و معناشناسی [39 و 49] وجود دارد.

با تمرکز بر ابزار و مهارت فنی برای بدست آوردن داده‌های بزرگ، تکنیک‌های مدل‌سازی برای نشان دادن چگونگی استفاده از داده‌های بزرگ مورد استفاده قرار می‌گیرند، قابلیت مدیریتی برای استخراج ارزش داده‌های بزرگ نیز به همان اندازه مهم است. فرصت‌های زیادی در حال ظهور هستند، بنابراین طراحان مفهومی دامنه خود را گسترش می‌دهند. برای تقویت تحلیل مدلسازان مفهومی، آنها ممکن است دانشمند داده شهری باشند (به عنوان مثال، [26])، در نقش مدیریت به عنوان مدیر ارشد اطلاعات (برای شناسایی فرصت‌های تجاری و اطلاعات شناخت) یا معمار داده‌های بزرگ (برای انتخاب سیستم عامل، معماری طراحی و فناوری) باشند.

NoSQL DBs هنوز در فاز اول تکامل هستند. آنها برای پذیرش و محبوبیت به روش‌ها و ابزارهای خوب برای طراحی، یکپارچگی و سازگاری احتیاج دارند. زبان جست‌وجوی استاندارد نیز مورد نیاز است. بدون شک، پایگاه‌های ارتباطی ارتباطات برای سیستم‌های مبتنی بر مبادله استفاده می‌شوند. NoSQL و سیستم‌های ارتباطی همسو خواهند شد. NoSQL جست‌وجوی گسترده‌ای را در رابطه با موتورهای جست‌وجو انجام می‌دهند. جست‌وجوی گسترده‌ای را در رابطه با موتورهای جست‌وجو بدست می‌آوریم، از جمله سیستم‌های مبتنی بر وب؛ در زمان حقیقی، ابر، و تلفن همراه؛ تعامل اولیه کم هزینه؛ اینترنت اشیا و .... .

7. نتیجه‌گیری

داده‌های بزرگ می‌توانند تأثیرات بالقوه‌ای بر کسب و کار و جامعه داشته باشند، که برخی از آنها در حال تحقق است. در این مقاله، تحولات اخیر در تکنولوژی داده‌های بزرگ مورد بررسی قرار گرفت و چالش‌های داده‌های بزرگ از نظر پنج «V» همراه با روش‌های بالقوه برای بررسی‌شان مورد بحث قرار گرفت، و نشان داده شد که دستیابی به ارزش داده‌های بزرگ بسیار سخت است. با مرور تکنولوژی‌های داده‌های بزرگ، مشارکت‌های مدل‌سازی به طور بالقوه شناسایی می‌شوند و از فرصت‌های راه‌حل‌های نوآورانه برای مسائل پیچیده پشتیبانی می کنند.

در پایان، آموزش فناوری‌های داده‌های بزرگ به نسل بعدی دانشمندان کامپیوتر، طراحان مفهومی و متخصصان سیستم‌های مدیریت داده‌ها (مدیریت پایگاه داده، تحلیلگران سیستم و غیره) مهم است [41]. فناوری‌های داده‌های بزرگ باید درک شوند، از جمله روش‌های نرم‌افزاری و سخت‌افزاری برای مواجه با داده‌های بزرگ، روش‌های تحلیل داده‌های بزرگ با استفاده از ابزارهای پیشرفته و خودکار است. همکاران محققان در زمینه مدل‌سازی مفهومی و مدیریت و توسعه پایگاه داده برای رفع مسائل مربوط به داده‌های بزرگ، اینترنت اشیا و بسیاری از برنامه های جالب، مورد نیاز است.

References

[1] A. Abbassi, A. Sarker, R.H.L. Chiang, Big data research in information systems: toward an inclusive research agenda, J. Assoc. Inf. Syst. 17 (2) (2016) (i-xxxii).

[2] N. Abdullah, S.A. Ismail, S. Sophiayati, S.M. Sam, Data quality in big data: a review, Int. J. Adv. Softw. Comput. Appl. 7 (3) (2015).

[3] P. Chapman, J. Clinton, R. Kerber, T. Khabaza, T. Reinartz, C. Shearer, & R. Wirth, CRISP-DM 1.0 Step-by-step data mining guide, 2000.

[4] P.P.S. Chen, The entity-relationship model—toward a unified view of data, ACM Trans. Database Syst. ((TODS)) 1 (1) (1976) 9–36.

[5] H. Chen, R.H. Chiang, V.C. Storey, Business intelligence and analytics: from big data to big impact, MIS Q. 36 (4) (2012) 1165–1188.

[6] E.F. Codd, A relational model of data for large shared data banks, Commun. ACM 13 (6) (1970) 377–387. http://dx.doi.org/10.1145/362384.362685.

[7] V. Dhar, Data science and prediction, Commun. ACM 56 (12) (2013) 64–73.

[8] D.W. Embley, S.W. Liddle, Big data—conceptual modeling to the rescueConceptual Modeling, Springer Berlin Heidelberg, 2013, pp. 1–8.

[9] Gartner, ⟨https://www.gartner.com/doc/2191415/big-data-strategy-components-business⟩.

[10] J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, Internet of Things (IoT): a vision, architectural elements, and future directions, Future Gener. Comput. Syst. 29 (7) (2013) 1645–1660.

[11] F.H. Hsu, Behind Deep Blue: Building the Computer that Defeated the World Chess Champion, Princeton University Press, 2002.

[12] Kalakota, R. (2012). Big Data Infographic and Gartner 2012 Top 10 Strategic Tech Trends.

[13] M. Khan, M. Uddin, and N. Gupta, Seven V's of Big Data: Understanding Big Data to extract Value, in: Proceedings of 2014 Zone 1 Conference of the American Society for Engineering Education (ASEE Zone 1), 2014.

[14] G. King, Ensuring the data-rich future of the social sciences, Science (2011) 719–721.

[15] R. Kitchin, T.P. Lauriault, Small data in the era of big data, Geo. J. (2014) 1–13.

[16] P. Hitzler, K. Janowicz, Linked data, big data, and the 4th paradigm, Semant. Web. 4 (3) (2013) 233–235.

[17] A. Lakshman, P. Malik, Cassandra - a decentralized structured storage system, ACM SIGOPS Oper. Syst. Rev. 44 (2) (2010) 35. http://dx.doi.org/10.1145/ 1773912.1773922.

[18] Laney, D., "3D Data Management: Controlling Data Volume, Velocity and Variety" (PDF). Gartner, 2001.

[19] M. Lin, H.C. Lucas Jr., G. Shmueli, Research Commentary – Too Big to Fail: large Samples and the p-Value Problem,, Inf. Syst. Res. 24 (4) (2013) 906–917.

[20] R. Lukyanenko, J. Parsons, Y. Wiersma, Citizen science 2.0: data management principles to harness the power of the crowdService-Oriented Perspectives in Design Science Research, Springer Berlin Heidelberg, 2011, pp. 465–473.

[21] V. Mayer-Schönberger, K. Cukier, Big data: A Revolution that Will Transform how We Live, Work, and Think, Houghton Mifflin Harcourt, 2013.

[22] A. McAfee, E. Brynjolfsson, Big Data: the management revolution, Harv. Bus. Rev. 90 (10) (2012) 60–68.

[23] P. Mell, T. Grance, The NIST definition of cloud computing, Natl. Inst. Stand. Technol. 53 (6) (2009) 50.

[24] C.W. Mineau, R. Missaoui, R. Godin, Conceptual modeling for data and knowledge management, Data Knowl. Eng. 33 (2) (2000) 137–168.

[25] J. Pokorny, NoSQL databases: a step to database scalability in web environment, Int.J. Web Inf. Syst. 9 (1) (2013) 69–82.

[26] A. Ribeiro, A. Silva, A.R. da Silva, Data modeling and data analytics: a survey from a big data perspective, J. Software Eng. Appl. 8 (12) (2015) 617.

[27] P.A. Rubin, Big data in small bites, Decis. Line (2015) 10–12.

[28] M. Russo, Redis, from the Ground Up. Retrieved November 26, 2015, from ⟨http://blog.mjrusso.com/2010/10/17/redis-from-the-ground-up.html⟩, (2010, October 17).
[29] P.J. Sadalage, M. Fowler, NoSQL distilled: a brief guide to the emerging world of polyglot persistence, Pearson Education, 2012.
[30] SAP - In-Memory Computing, New Reality of Real Time With Launch of SAP® High-Performance Analytic Appliance. Retrieved November 26, 2015, from ⟨http://global.sap.com/corporate-en/news.epx?Pressid=14457⟩, (2010, December 1).
[31] K.D. Schewe, B. Thalheim, Conceptual modelling of web information systems, Data Knowl. Eng. 54 (2) (2005) 147–188.
[32] K.D. Schewe, B. Thalheim, Semantics in Data and Knowledge Bases, Springer Berlin Heidelberg, 2008, pp. 1–25.
[33] I.-Y. Song, Y. Zhu, Big data and data science: what should we teach?, Expert Syst. 33 (4) (2016) 364–373.
[34] M. Stonebraker, New SQL: An Alternative to NoSQL and Old SQL for New OLTP Apps | blog@CACM | Communications of the ACM. Retrieved November 26, 2015, from ⟨http://cacm.acm.org/blogs/blog-cacm/109710-new-sql-an-alternative-to-nosql-and-old-sql-for-new-oltp-apps/fulltext⟩, June 16 2011.
[35] V.C. Storey, Relational database design based on the entity-relationship model, Data Knowl. Eng. 7 (1991) 47–83.
[36] V.C. Storey, The Real Challenges of Big Data Decision Line, January 2016.
[37] V.C. Storey, J.C. Trujillo, S.W. Liddle, Conceptual modeling: themes, topics, and introduction to the special issue, Data Knowle. Eng. (2015).
[38] T.J. Teorey, D. Yang, J.P. Fry, A logical design methodology for relational databases using the extended entity-relationship model, ACM Comput. Surv. ((CSUR)) 18 (2) (1986) 197–222.
[39] B. Thalheim, Syntax, semantics and pragmatics of conceptual modelling.Natural Language Processing and Information Systems, Springer Berlin Heidelberg, 2012, pp. 1–10.
[40] A. Vasiliev,. World of the NoSQL Databases. Retrieved November 26, 2015, from ⟨http://leopard.in.ua/2013/11/08/nosql-world/⟩, (November 8) 2013.
[41] Y. Wand, R. Weber, Research commentary: information systems and conceptual modeling—a research agenda, Inf. Syst. Res. 13 (4) (2002) 363–376.
[42] R.Y. Wang, V.C. Storey, C.P. Firth, A framework for analysis of data quality research, IEEE Trans. knowl. data eng. 7 (4) (1995) 623–640.
[43] R. Weber, Research review paper conceptual modelling and ontology: possibilities and pitfalls, J Database Manag 2003a, 14(3) (1-20) (2003).
[44] Wikibon (2015), ⟨http://premium.wikibon.com/executive-summary-big-data-vendor-revenue-and-market-forecast-2011-2020/⟩, 2013.
[45] A. Zanella, N. Bui, A. Castellani, L. Vangelista, M. Zorzi, Internet of things for smart cities, Internet Things J., IEEE 1 (1) (2014) 22–32.
[46] Y. Zhu, E. Yan, I.-Y. Song, The use of a graph-based system to improve bibliographic information retrieval: system design, implementation, and evaluation, J. Assoc. Inf. Sci. Technol. 68 (2) (2016) 480–490.