یادگیری ماشین در داده های بزرگ: فرصت ها و چالش ها

چکیده

یادگیری ماشین (ML) به‌طور مداوم قدرت خود را در طیف گسترده‌ای از برنامه‌های کاربردی نشان می‌دهد. این مسئله در سال های اخیر تا حدودی با توجه به ظهور داده‌های بزرگ بیشتر مورد توجه قرار گرفته است. الگوریتم ML هرگز بهترین عملکرد خود را نداشت تا اینکه توسط داده‌های بزرگ به چالش کشیده شد. داده‌های بزرگ، الگوریتم ML را قادر به کشف الگوهای دقیقتر و پیش‌بینی به موقع تر و دقیق تر از قبل کردند. از سوی دیگر، چالش‌های بزرگی در ML مانند مقیاس‌پذیری مدل و محاسبات توزیع شده مطرح کرد. در این مقاله، یک چارچوب از ML در داده‌های بزرگ (MLBiD) برای هدایت بحث به فرصت‌ها و چالش‌های آن معرفی خواهد شد. چارچوبML محور، شامل مراحل پیش پردازش، یادگیری و ارزشیابی است. علاوه براین، چارچوب شامل چهار جزء دیگر، مانند داده‌های بزرگ، کاربران، دامنه و سیستم است. مراحل ML و اجزای MLBiD برای شناسایی فرصت‌های مرتبط و چالش‌ها و روشن کردن مسیر کاری آینده در بسیاری از موارد ناشناخته و یا در پژوهش حاضر ارائه شده است.

کلیدواژه‌ها: یادگیری ماشین، داده‌های بزرگ، پیش‌پردازش داده‌ها، ارزیابی، موازی‌سازی

1. معرفی

تکنیک‌های یادگیری ماشین (ML) تاثیرات اجتماعی بزرگی در طیف گسترده‌ای از برنامه‌های کاربردی مانند بینایی کامپیوتر، پردازش سخنرانی، درک زبان طبیعی، علوم اعصاب، بهداشت و اینترنت اشیا داشته است. ظهور عصر داده ای بزرگ موجب توجه به ML گردید. الگوریتم ML هرگز بهترین نتایج را به همراه نداشت و توسط داده‌های بزرگ برای

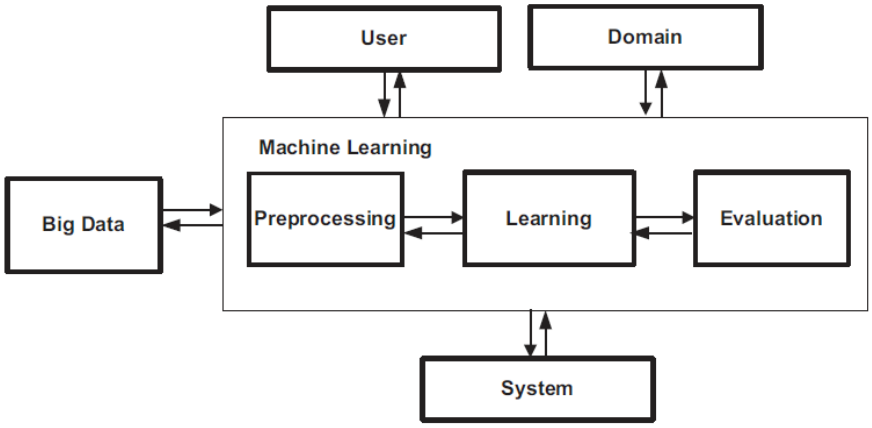
به دست آوردن بینش جدیدی در برنامه‌های کاربردی مختلف کسب و کار و رفتار انسان به چالش کشیده شد. از یک طرف، داده‌های بزرگ اطلاعات بی‌سابقه‌ای غنی برای الگوریتم ML برای استخراج الگوهای اساسی و ساخت مدل‌های پیش‌بینی فراهم می‌کند. از سوی دیگر، الگوریتم‌های سنتی ML با چالش‌های مهمی مانند مقیاس‌پذیری مقادیر واقعی و پنهان داده های بزرگ رو به رو هستند. با گسترش وسیع داده‌های بزرگ، ML در جهت تبدیل داده‌های بزرگ به هوش عملی رشد و پیشرفت کرد.

ML به این پرسش که چگونه یک سیستم کامپیوتری بسازیم که به طور خودکار از طریق تجربه بهبود یابد پاسخ می‌دهد[1]. مشکل ML به‌عنوان مشکل یادگیری از تجربه با توجه به برخی از وظایف و اندازه‌گیری عملکرد اشاره دارد. تکنیک‌های ML کاربران را قادر به کشف ساختار زیرین و پیش‌بینی از مجموعه داده‌های بزرگ می‌کند. ML در تکنیک‌های یادگیری کارآمد (الگوریتم)، داده‌های بزرگ غنی و محیط‌های محاسبات قدرتمند بسیار کارآمد است. بنابراین، ML پتانسیل زیادی دارد تا بخش مهمی از تجزیه و تحلیل داده‌های بزرگ [2] گردد.

در این مقاله در مورد تکنیک‌های ML در زمینه داده‌های بزرگ و محیط‌های محاسبات مدرن تمرکز داریم. به‌طور خاص، هدف ما بررسی فرصت‌ها و چالش‌های ML بر روی داده‌های بزرگ است. داده‌های بزرگ فرصت‌های جدیدی برای ML ارائه می‌کنند. به‌عنوان مثال، داده‌های بزرگ قادر به استفاده از یادگیری الگو در چند دانه‌ای و تنوع، از دیدگاه‌های زیادی در حالت موازی هستند. علاوه براین، داده‌های بزرگ فرصت‌ها را برای استنتاج علیت براساس زنجیره‌ای از دنباله‌ها فراهم می‌کنند. با این وجود، داده‌های بزرگ چالش‌های عمده‌ای در ML مانند ابعاد بالای داده‌ها، مدل مقیاس‌پذیری، محاسبات توزیع شده، جریان داده [3]، سازگاری و قابلیت استفاده معرفی می‌کنند. در این مقاله، یک چارچوب ML در داده‌های بزرگ (MLBiD) برای هدایت بحث به فرصت‌ها و چالش‌های آن معرفی می‌کنیم. این چارچوب ML محور، مراحل پیش‌پردازش، یادگیری و ارزشیابی را به همراه دارد. علاوه براین چارچوب از چهار جزء دیگر که توسط ML تحت تاثیر قرار می‌گیرند تشکیل شده است، داده‌های بزرگ، کاربران، دامنه و سیستم. اجزای MLBiD و مراحل ML جهت شناسایی فرصت‌ها و چالش‌ها و کارهای آینده در بسیاری از حوزه‌های ناشناخته پژوهش ارائه شده است.

2. چارچوب یادگیری ماشین در داده‌های بزرگ

چارچوب ML در داده‌های بزرگ (MLBiD) در شکل 1 نشان داده شده است. MLBiD بر جزء یادگیری ماشین (ML) استوار است، که با چهار جزء دیگر، از جمله داده‌های بزرگ، کاربر، دامنه و سیستم تعامل برقرار می‌کند. فعل و انفعالات در هر دو جهت اتفاق می‌افتد. به عنوان مثال، داده‌های بزرگ به‌عنوان ورودی به ML وارد می‌شوند و خروجی تولید می‌شود، که به نوبه خود تبدیل به بخشی از داده‌های بزرگ می‌گردد؛ کاربر ممکن است با ML برای ارائه دامنه دانش، ترجیحات شخصی و بازخورد قابلیت استفاده و با اعمال نفوذ نتایج یادگیری به‌منظور بهبود تصمیم‌سازی تعامل برقرار می‌کند؛



شکل1. چارچوب یادگیری ماشین در داده‌های بزرگ (MLBID)

دامنه می‌تواند هم به‌عنوان یک منبع دانش برای خدمت به راهنمای ML و هم به‌عنوان زمینه اعمال در مدل یادگیری استفاده شود؛ معماری سیستم بر چگونگی اجرای الگوریتم‌های یادگیری و چگونگی اجرای کارآمد آن‌ها تاثیر دارد و به‌طور همزمان پاسخگویی به نیازهای ML ممکن است به یک شرکت طراحی معماری سیستم منجر شود. سپس جزئی از MLBiD به‌طور جداگانه معرفی می‌کنیم.

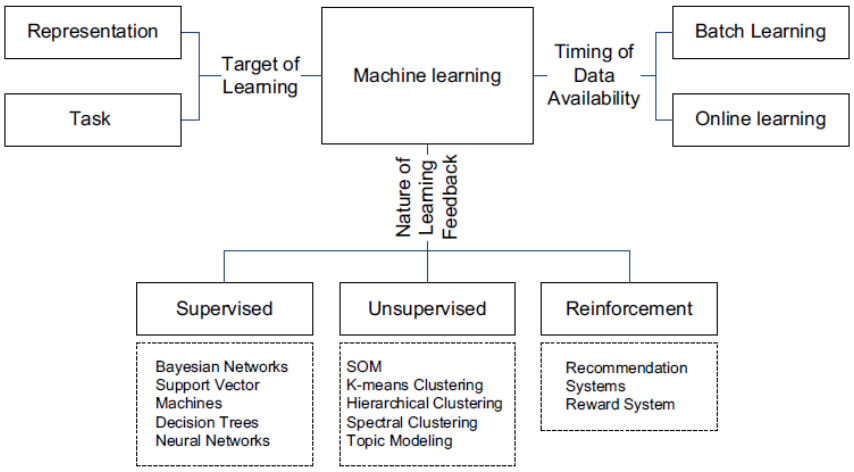
2.1 یادگیری ماشین

ML معمولا از طریق پردازش داده‌ها، یادگیری و مرحله ارزیابی (شکل 1 را ببینید) پیگیری می‌شود. پیش پردازش داده‌ها کمک می‌کند تا داده‌های خام به "شکل درست" برای مراحل یادگیری‌های بعدی آماده شود. داده‌های خام به احتمال زیاد بدون ساختار، نویزدار، ناقص و متناقض هستند. گام پیش‌پردازش، داده‌ها را به شکلی که می‌توان به عنوان ورودی برای یادگیری داده‌ها از طریق، استخراج، تبدیل، و همجوشی استفاده کرد تبدیل می‌کند. فاز آموزش، الگوریتم‌های یادگیری را انتخاب و پارامترهای مدل را برای تولید خروجی مورد نظر با استفاده از پیش پردازش داده‌های ورودی به کار می‌گیرد. برخی روش‌های یادگیری، به‌ویژه یادگیری بازنمودی، نیز می‌تواند برای پیش پردازش داده‌ها استفاده شود. ارزیابی برای تعیین عملکرد مدل به دست آمده بسیار مفید است. برای مثال، ارزیابی عملکرد یک طبقه‌بندی شامل انتخاب مجموعه داده، اندازه‌گیری عملکرد، برآورد خطا و آزمون‌های آماری است [4]. بررسی نتایج ممکن است به تنظیم پارامترهای انتخاب شده در الگوریتم‌های یادگیری و / یا انتخاب الگوریتم‌های مختلف منجر شود. ML می‌تواند در ابعاد مختلف مشخص گردد: ماهیت یادگیری بازخورد، هدف از وظایف یادگیری و زمان دردسترس بودن داده‌ها. براین اساس، یک طبقه‌بندی چندبعدی از ML، همانند شکل نشان داده شده در 2 پیشنهاد می‌دهیم.

• براساس ماهیت بازخورد دردسترس برای یک سیستم یادگیری، ML را می‌توان به سه نوع اصلی طبقه‌بندی کرد: یادگیری نظارت شده، یادگیری بدون نظارت و یادگیری تقویتی [5]. در یادگیری نظارت شده، یک سیستم یادگیری با نمونه‌هایی از جفت‌های ورودی-خروجی ارائه می‌گردد و هدف ا یادگیری یک تابع است که ورودی را به خروجی نگاشت کند. در یادگیری بدون نظارت، سیستم با بازخورد صریح یا خروجی مورد نظر ارائه نشده است و هدف کشف الگوهایی در ورودی است. همانند یادگیری بدون نظارت، یک سیستم یادگیری تقویتی با جفت ورودی و خروجی ارائه نشده است. مانند یادگیری نظارتی، یادگیری تقویتی براساس تجربه‌های قبلی عمل می‌کند. برخلاف یادگیری نظارت شده، بازخورد در یادگیری تقویتی پاداش یا مجازات مرتبط با اقدام به جای خروجی مورد نظر و یا اصلاح صریح و روشن اقدامات مطلوب است. یادگیری نیمه نظارت شده بین یادگیری نظارت شده و نظارت نشده قرار دارد، که در آن

سیستم با تعداد کمی از جفت‌های ورودی-خروجی و یک تعداد زیادی از ورودی‌های نامشخص ارائه شده است. هدف از یادگیری نیمه نظارتی شبیه به یادگیری نظارت شده است.

• براساس اینکه آیا هدف از یادگیری، وظایف خاص با استفاده از ویژگی‌های ورودی است، ML را می‌توان به یادگیری بازنمودی و یادگیری وظیفه طبقه‌بندی کرد. هدف از یادگیری بازنمودی، یادگیری نمایش‌های جدید داده است که استخراج اطلاعات مفید در هنگام ساختن و یا قبل از طبقه‌بندی را آسانتر می‌کند[6]. نمایش خوب باید دارای عوامل زمینه‌ای از تنوع باشد. که اغلب توزیع عوامل اکتشافی زمینه‌ای برای خروجی‌های مشاهده شده در مورد مدل احتمالاتی است [6].



شکل 2. یک طبقه‌بندی چندبعدی از یادگیری ماشین

یادگیری نمایشی اغلب با برآورد چگالی و کاهش ابعاد سروکار دارد. برآورد تراکم، تابع چگالی احتمال یک متغیر تصادفی را می‌یابد. کاهش ابعاد، ورودی از فضای با ابعاد بالا را به یک فضای با ابعاد کمتر نگاشت می‌کند. بنابراین ایجاد یک هدف روشن بسیار دشوار است. در مقابل، یادگیری وظیفه معمولا خروجی مورد نظر را دارا بوده و بر اساس طبقه‌بندی، رگرسیون، و خوشه‌بندی عمل می‌کند. در طبقه‌بنید، تکنیک‌های ML یک مدل را تولید می‌کنند که ورودی نهان را به یک یا چند کلاس تعریف از پیش تعریف شده اختصاص می‌دهد. رگرسیون متفاوت از طبقه‌بندی است زیرا خروجی آن به جای مقادیر مجزا، پیوسته است. خوشه‌بندی گروه‌هایی از داده‌ها را تولید می‌کند و این گروه‌ها ناشناخته

هستند، که خود را طبقه‌بندی متمایز می‌کنند. به‌طور سنتی، طبقه‌بندی و رگرسیون با عنوان یادگیری نظارت شده و خوشه‌بندی به‌عنوان یادگیری بدون نظارت نامیده می‌شوند. الگوریتم نمایش آنها در شکل 2 نشان داده شده است.

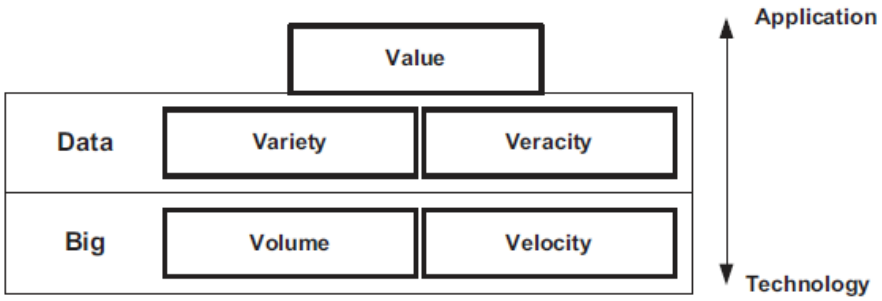
• براساس زمان‌بندی ساخت داده‌های آموزشی موجود (به‌عنوان مثال، آیا داده‌های آموزشی یک بار و یا در یک زمان در دسترس همه هستند)، ML می‌تواند به یادگیری دسته‌ای و یادگیری آنلاین تقسیم شود. آموزش دسته‌ای مدل‌هایی با یادگیری در داده‌ی آموزشی تولید می‌کند، درحالی‌که یادگیری آنلاین مدل‌ها را براساس هر ورودی جدید به روزرسانی می‌کند. الگوریتم یادگیری دسته‌ای فرض می‌کند که داده مستقل است و یکسان توزیع شده یا از توزیع احتمال یکسان تبعیت می‌کند، که معمولا با داده‌های واقعی راضی نیست. آموزش آنلاین به‌طور معمول باعث می‌شود هیچ فرض آماری در مورد داده وجود نداشته باشد [7]. اگر چه انتظار می‌رود الگوریتم یادگیری دسته‌ای، یادگیری آنلاین را تعمیم دهد زیرا انتظار می رود الگوریتم برچسب‌هایی از نمونه‌ها را که به عنوان ورودی دریافت می‌کند به دقت پیش‌بینی کند [7]. یادگیری آنلاین زمانی استفاده شده است که آموزش مجموعه داده‌ها غیرعملی باشد و / یا زمانی که داده‌ها در طول زمان ایجاد شده باشند و سیستم یادگیری نیاز به انطباق با الگوهای جدید در داده‌ها داشته باشد.

هر الگوریتم ML می‌تواند در ابعاد مختلف دسته‌بندی شود. برای مثال، درخت‌های تصمیم‌گیری معمولی متعلق به الگوریتم‌های یادگیری تحت نظارت دسته‌ای هستند.

2.2 داده‌های بزرگ

داده‌های بزرگ در پنج بعد مشخص می‌شوند: حجم (کیفیت / مقدار داده)، سرعت (سرعت تولید داده‌ها)، انواع (نوع، طبیعت و فرمت داده)، صحت (اعتماد / کیفیت داده‌های گرفته شده) و مقدار (بینش و تاثیر). در این مقاله پنج بعد دیگر سازماندهی شده است که متشکل از لایه‌های بزرگ، داده‌ها و مقادیر با شروع از پایین است (شکل 3). لایه بزرگ اساسی‌ترین و لایه داده‌ها در مرکزیت داده‌های بزرگ و لایه مقدار جنبه‌ی تاثیر برنامه‌های کاربردی داده‌های بزرگ در جهان واقعی را مشخص می‌کند. لایه‌های پایین‌تر (به‌عنوان مثال، حجم و سرعت) بستگی به شدت بیشتر در پیشرفت‌های فن‌آوری و لایه بالاتر (به‌عنوان مثال، مقادیر) بیشتر به سمت برنامه‌های کاربردی با قدرت مهار استراتژیک

داده بزرگ گرایش دارند. به منظور تحقق بخشیدن به ارزش تجزیه و تحلیل داده‌های بزرگ و پردازش کارآمد داده‌های بزرگ، پارادایم ML موجود و الگوریتم‌ها نیاز به اقتباس دارند.



شکل3. پشته داده‌های بزرگ

2.3 دیگر مولفه‌ها

2.3.1 کاربران

انواع مختلفی از ذینفعان سیستم ML مانند کارشناسان دامنه، کاربران نهایی و محققان و فعالان ML وجود دارد. به‌طور سنتی، فعالان ML کسانی هستند که برای استفاده ازML تصمیمات زیادی می‌گیرند، از جمع‌آوری داده‌ها برای ارزیابی عملکرد شروع می‌کنند. دخالت کاربر نهایی در طول این فرایند به ارائه برچسب داده‌ها، پاسخ به سوالات مربوط به دامنه و یا دادن بازخورد در مورد نتایج به دست آمده محدود شده است، که معمولا توسط فعالان، منجر به تکرار طولانی و ناهمزمان می‌گردد [8]. با این حال، کاربران نهایی تمایل به ارائه بیش از برچسب داده‌ها دارند. آنها ارزش شفافیت در طراحی یک سیستم یادگیری را بیشتر درک می‌کنند، زیرا به نوبه خود کمک می‌کند تا سیستم آنها را درک و برچسب / بازخورد بهتری ارائه کند. درگیری کاربران در ML می‌تواند به‌طور بالقوه منجر به سیستم یادگیری و تجربه کاربری موثر و بهتر گردد [8]. به‌عنوان مثال، ML فعالانه [8] اجازه می‌دهد تا کاربران به صورت تعاملی تاثیر خود بر اقدامات و انطباق ورودی‌های بعدی برای هدایت رفتارهای ML برای به دست آوردن خروجی مورد نظر را بررسی کنند.

2.3.2 دامنه

دانش دامنه موجب تسهیل ML در کشف الگوها گردد زیرا ممکن است قابل کشف از مجموعه داده‌ها نباشد. مجموعه آموزشی ممکن است به اندازه کافی بزرگ و / یا نمایشی درست برای به کارگیری تمامی الگوها نباشد. همچنین به‌دست آوردن اطلاعات کافی و پرهزینه و حتی غیرعملی است، احتمالا به علت تنوع دامنه و الزامات برنامه خاص باشد. دانش دامنه می‌تواند به بهبود کلی و استحکام الگوهای ناشی از مجموعه داده کمک کند [9]. راه‌های مختلفی برای ترکیب دانش قبلی دامنه درML [10] وجود دارد: (1) آماده‌سازی نمونه‌های آموزشی؛ (2) تولید فرضیه یا فضای فرضیه؛ (3) اصلاح هدف مورد جستجو و (4) افزایش جستجو. این الگوهای به‌دست آمده در به نوبه خود برای به روز رسانی و اصلاح دانش دامنه مورد استفاده قرار گیرند.

2.3.3 سیستم

معماری سیستم و یا پلت‌فرم، که متشکل از نرم‌افزار و سخت‌افزار است، محیطی را ایجاد می‌کند که در آن الگوریتم ML می‌تواند اجرا شود. به‌عنوان مثال، در مقایسه با همتایان ساده‌تر، از ماشین چند هسته‌ای با معماری توزیع شده انتظار می‌رود بهره‌وری ML را بهبود دهد. چارچوب و معماری سیستم جدید مانند هادوپ / اسپارک برای رسیدگی به چالش‌های داده‌های بزرگ پیشنهاد شده است. با این وجود، مهاجرت الگوریتم ML موجود برای توزیع معماری نیاز به تغییر دارد که چگونه الگوریتم ML پیاده‌سازی واجرا می‌شود. علاوه براین، نیازهای منحصربه فرد و مقادیر ML ممکن است معماری سیستم جدید را طراحی و توسعه دهند.

براساس چارچوب MLBiD، از فرصت‌های مهم و چالش‌های کلیدی را شناسایی کردیم. و آنها را برای هر سه مرحله در ML- پیش‌پردازش، یادگیری و ارزیابی، به طور جداگانه مورد بحث قرار دادیم.

3. فرصت‌ها و چالش‌های پیش‌پردازش داده‌ها

بخش عمده‌ای از تلاش واقعی در استقرار سیستم ML صرف طراحی پیش‌پردازش خطوط لوله و تحولات داده می‌شود که منجر به نمایش موثر اطلاعات با پشتیبانی ML می‌گردد [6]. پیش‌پردازش داده‌ها با هدف رسیدگی به تعدادی از مسائل مانند افزونگی داده، تناقض، نویز، عدم تجانس، تحول، برچسب گذای (برای ML (نیمه) نظارت شده)، عدم تعادل داده‌ها و نمایش/ انتخاب ویژگی است. آماده‌سازی داده‌ها و پردازش، با توجه به نیاز به کار انسانی و تعداد زیادی از گزینه‌های انتخاب معمولا پرهزینه است. علاوه براین، برخی از فرضیات معمولی داده برای داده‌های بزرگ کاربرد ندارد، در نتیجه برخی از روش‌های پیش‌پردازش غیرممکن می‌گردد. ازسوی دیگر، داده‌های بزرگ موجب کاهش فرصت تکیه به نظارت انسان با آموختن از منابع داده عظیم و متنوع می‌گردد.

3.1 افزونگی داده‌ها

تکرار زمانی رخ می‌دهد که دو یا چند نمونه داده موجودیت یکسانی را نشان دهند. تاثیر تکرار داده‌ها و یا تناقض در ML می‌تواند شدید باشد. با وجود طیف وسیعی از روش‌ها برای شناسایی موارد تکراری توسعه یافته در 20 سال گذشته [11]، روش‌های سنتی مانند مقایسه دو به دو شباهت‌ها، دیگر برای داده‌های بزرگ امکان‌پذیر نیست. علاوه براین، فرض سنتی،جفت‌هایی را که در مقایسه با جفت غیرکپی دیگر در اقلیت هستند تکرار می‌کند. برای این منظور، زمان پویا می‌تواند بسیار سریعتر از الگوریتم‌های فاصله اقلیدسی عمل کند [12].

3.2 نویز داده‌ها

مقادیر از دست رفته و نادرست، کم بودن داده‌ها و نقاط دورافتاده می‌توانند به‌عنوان نویز ML معرفی شوند. راه‌حل‌های سنتی برای مشکل نویز داده، در برخورد با داده‌های بزرگ با چالش‌هایی روبه‌رو است. به‌عنوان مثال، روش‌های دستی به علت عدم مقیاس‌پذیری آن دیگر امکان‌پذیر نیستند؛ جایگزینی به معنای از دست دادن مزایای استفاده از غنی و دانه دانه شدگی داده‌های بزرگ است. در برخی موارد، الگوهای جالبی ممکن است در این نویز داده‌ها اشتباه باشد،

بنابراین حذف ساده ممکن است یک جایگزین عاقلانه نباشد. پیش‌بینی دقیق تجزیه‌وتحلیل ترافیک داده‌های بزرگ می‌تواند برای برآورد مقادیر از دست رفته، مانند جایگزین خواندن نادرست با توجه به سنسور خراب یا کانالهای ارتباطی شکسته استفاده کند. برای رسیدن به بایاسی قابل توجهه ممکن است به پیش‌بینی با روش‌های نفوذ جمعی روی بیاوریم که، حداکثر محدودیت است در مرحله استنتاج را تحمیل کرده باشند و مجبور به پیش‌بینی با توزیع یکسان به‌عنوان برچسب مشاهده باشیم [13]. علیرغم تمام مسائل مطرح شده، ممکن است حداقل پراکندگی داده‌ها حفظ شود و حجم گسترده‌ای از داده‌های بزرگ منجر به ایجاد فرصت‌های منحصر به فرد تجزیه و تحلیل و پیش‌بینی ترافیک به دلیل فرکانس کافی برای نمونه های مختلف گردد. تلاش برای افزایش تشخیص داده‌های پرت (به عنوان مثال، ONION[14]) برای اینکه تحلیلگران موفق به کشف ناهنجاری‌ها در مجموعه داده‌های بزرگ گردند وجود دارد [14].

3.3 ناهمگونی اطلاعات

داده‌های بزرگ منجر به ارائه داده‌های چنددیدگاهی از انواع مختلف مخازن، در فرمت‌های متفاوت و از نمونه‌های مختلف جمعیت و در نتیجه بسیار ناهمگن هستند. این داده‌های ناهمگن چنددیدگاهی (به‌عنوان مثال، متن بدون ساختار، فرمت‌های صوتی و تصویری [15]) ممکن است سطوح مختلفی از اهمیت برای یک کار ذهنی را داشته باشند. بدین ترتیب، الحاق تمام ویژگی‌ها با اندازه اهمیت یکسان بعید است که نتایج یادگیری مطلوبی را به همراه داشته باشد. داده‌های بزرگ فرصت یادگیری از دیدگاه‌های متعدد را به صورت موازی و سپس پیوستن نتایج با یادگیری اهمیت ویژگی به کار را ارائه می‌دهند. انتظار می‌رود روش ارائه شده در نقاط دورافتاده داده بهتر عمل کند و مشکلات و مسائل همگرایی بهینه‌سازی [16] را نشان دهد.

3.4 مجزاسازی داده

برخی از الگوریتم‌های ML مانند درخت‌های تصمیم‌گیری و ساده بیز تنها با ویژگی‌های گسسته سروکار دارند. مجزاسازی موجب می‌شود داده‌های کمی به داده‌های کیفی جدا شده و تقسیم بدون تداخل از دامنه پیوسته صورت

گیرد. هدف از مجزاسازی ویژگی پیدا کردن اطلاعات تضمینی مختصر به‌عنوان دسته است، که برای وظیفه یادگیری برای حفظ اطلاعات به‌همان اندازه در ویژگی اصلی که ممکن است کافی هستند. بااین حال، هنگام مقابله با داده‌های بزرگ، بیشتر روش‌های گسسته موجود کارآمد نخواهند بود. برای پرداختن به چالش‌های داده‌های بزرگ، روش مجزاسازی استاندارد شده با توسعه یک نسخه توزیع شده برای به حداقل رساندن آنتروپی براساس اصل حداقل طول در پلتفرم‌های داده‌های بزرگ، افزایش کارایی و دقت [17] موازی‌سازی شده است. در مطالعه دیگری [18]، داده‌ها برای اولین بار براساس مقدار عددی مرتب شده و سپس به قطعاتی از ویژگی کلاس اصلی تقسیم می‌شوند. این قطعات، که توسط درصد ترکیب طبقات مختلف خلاصه می‌شوند، به‌عنوان نمونه فوق العاده با هدف مجزاسازی مشاهده می‌شوند.

3.5 برچسب‌گذاری داده

روش حاشیه‌نویسی داده‌های سنتی کاری فشرده است. چندین روش جایگزین برای رسیدگی به چالش اطلاعات بزرگ پیشنهاد شده است. به‌عنوان مثال، مخازن آنلاین پرجمعیت می‌توانند به‌عنوان یک منبع رایگان از داده‌های آموزشی باشند، که می‌توانند کلاسی بزرگ و تنوع درون طبقاتی را ایجاد کنند [19]. علاوه براین، مفهوم یادگیری در سطح انسان می‌تواند از طریق القای برنامه احتمالاتی به دست آید [20]. علاوه‌براین، توانایی برچسب‌گذاری اطلاعات در الگوریتم ML مانند یادگیری نیمه نظارتی، آموزش انتقال، آموزش فعال فعال گردد (به‌عنوان مثال، [21،22]). با استفاده از یادگیری فعال به‌عنوان استراتژی بهینه‌سازی برای انجام وظایف برچسب‌گذاری در پایگاه‌داده‌های منابع، می‌توان تعدادی از پرسش‌های مطرح برای به حداقل رساندن جمعیت را به برنامه‌های کاربردی منابع برای مقیاس‌گذاری محول کرد. بااین حال، طراحی الگوریتم‌های یادگیری فعال برای مجموعه منابع داده از چالش‌های عملی، مانند کلیت، مقیاس‌پذیری و سهولت استفاده است [23]. موضوع دیگر این است که چنین مجموعه داده‌ای ممکن است همه زمینه‌های خاص کاربران را پوشش ندهد، بنابراین ممکن است اغلب کارآیی پایینی نسبت به آموزش کاربر محور داشته باشد [19].

3.6 داده‌های نامتوازن

مشکل داده‌های نامتوازن توسط روش‌های نمونه‌گیری تصادفی سنتی بیان شده است. بااین حال، این روند می‌تواند بسیار وقتگیر باشد اگر شامل تکرار نسل زیر نمونه و محاسبه معیارهای خطا گردد. علاوه براین، روش‌های نمونه‌گیری سنتی نمی‌توانند نمونه‌گیری موثر داده‌ها را بر روی یک زیرمجموعه کاربر مشخص از داده‌ها پشتیبانی کنند و شامل نمونه‌گیری مبتنی بر مقدار است. داده‌های بزرگ مستلزم نمونه‌گیری موازی داده‌ها است. به‌عنوان مثال، یک چارچوب نمونه‌گیری موازی برای ایجاد مجموعه داده نمونه از مجموعه داده اصلی بر اساس بر روی چند فایل شاخص توزیع شده پیشنهاد شده است [24]. سطح موازی می‌تواند براساس اندازه‌ی مجموعه داده‌ها و فرآیندهای موجود انتخاب شود.

3.7 نمایش ویژگی و انتخاب

عملکرد ML به شدت وابسته به انتخاب نمایش اطلاعات یا ویژگی‌ها است [6]. تعمیم الگوریتم ML بستگی به مجموعه داده دارد، که به‌طور غیرمستقیم به ویژگی‌های وابسته که نشان دهنده ساختار برجسته از مجموعه داده است بستگی دارد. انتخاب ویژگی به بهبود عملکرد ML با شناسایی ویژگی‌های برجسته کمک می‌کند. که اساسا زیرمجموعه‌های مختلف از ویژگی‌ها و داده‌ها و واحدها در سطوح مختلف از دانه دانه شدگی انتخاب می‌شوند و منجر به کاهش مقدار داده بزرگ می‌گردند. با این حال، مهندسی ویژگی نیاز به دامنه دانش قبلی و ابتکار و خلاقیت انسان و اغلب کار فشرده دارد [6]. برای پرداختن به ضعف الگوریتم مهندسی ویژگی در هنگام برخورد با داده‌های بزرگ، راه‌حل‌های مختلف پیشنهاد شده است، مانند انتخاب ویژگی‌های توزیع شده [25]؛ تقریب ماتریس کم‌رتبه (به‌عنوان مثال، روش استاندارد نیستروم [26])؛ نمایش یادگیری برای ساخت الگوریتم‌های یادگیری کمتر وابسته به مهندسی ویژگی‌ها با یادگیری قبل [6]؛ طرح تطبیقی ویژگی برای انتخاب ویژگی بعدی فوق‌العاده بالا، که مکررا یک گروه از ویژگی‌ها را فعال کند و دنباله‌ای از مسائل یادگیری هسته‌ای را حل کند [27]؛ یک چارچوب یکپارچه برای انتخاب ویژگی براساس نظریه گراف طیفی است که قادر به تولید خانواده‌ای از الگوریتم‌ها برای هر دو انتخاب ویژگی تحت نظارت و نظارت نشده

است [28]؛ خوشه‌بندی فازی قبل از طبقه‌بندی، که در آن طبقه‌بندی با متوجه مرکز گروه، به دنبال خوشه‌بندی و طبقه‌بندی از طریق کاهش داده است [29]؛ و کاهش اندازه ابعاد داده‌ها (به‌عنوان مثال، رتبه‌بندی تصادفی انتخاب جنگل روبه جلو و رتبه‌بندی تصادفی انتخاب جنگل رو به عقب [30]، [31]). به تازگی، autoencoding عمیق مبتنی بر شبکه‌های عصبی ثابت کرده است که در یادگیری ویژگی‌های ویدئو، صدا و متن بسیار موثر است [32،33].

جدول 1. یک طبقه‌بندی از روش‌ها / پلتفرم‌ها برای یادگیری ماشین بر روی داده‌های بزرگ.

Parallelism	Target	Techniques	Sample Studies*
Non-parallel		Optimization	[41,42] [43]
Parallel	data	MapReduce	BN [44,45], DT [38], TM [46], GP [47,48] [49] [50] [51]
		DistributedGraph	GA [52]
		Others	SVM [37], NN [53], GP [36,39]
	model/ parameter	Multi-threading	SVM [37]
		MPI/OpenMP	NN [40], TM [46]
		GPU	NN [40,53,54]
		Others	SVM [55], NN [56], GP [36,39]

* *BN* Bayesian network learning, *DT* decision tree, *TM* topic modeling, *GP* generic platform, *SVM* support vector machine, *NN* neural network

4. فرصت‌های یادگیری و چالش‌ها

توسعه الگوریتم‌های ML مقیاس‌پذیر که قادر به استفاده از مجموعه داده‌های بزرگ هستند، موضوع تحقیق طولانی مدتی در ارتباطات ML قبل از ظهور "داده‌های بزرگ" بود. برای سازماندهی بهتر بحث در مورد فرصت‌ها و چالش، یک طبقه‌بندی از روش‌ها / پلتفرم‌ها برای ML در داده‌های بزرگ پیشنهاد شده است، که در جدول 1 نشان داده شده است.

در طبقه‌بندی، ابتدا مطالعات را براساس اینکه هر موازی‌سازی در الگوریتم‌ها / پلتفرم‌ها در نظر گرفته شده است دسته‌بندی می‌کنیم. متد و روش‌ها در دسته‌بندی غیرموازی با هدف روش‌های بهینه‌سازی بسیار سریع‌تر انجام می‌گیرد که می‌تواند با داده‌های بزرگ بدون هیچ موازی‌سازی سروکار داشته باشد. به‌طورسنتی، مقیاس‌پذیری ML به‌طور عمده

به توسعه الگوریتم‌هایی که می‌توانند بسیار موثر اجرا شوند (به‌عنوان مثال، با پیچیدگی زمانی بهتر و / یا پیچیدگی فضا بهتر) تمرکز دارد. به‌عنوان مثال، گرادیان تصادفی، یک مثال کلاسیک از یک الگوریتم ML مقیاس‌پذیر است که در اصل می‌تواند مجموعه داده‌های عظیم را بدون نیاز به حافظه پردازش کند [34]. همچنین یک معاوضه بین مقیاس‌پذیری و تحدب وجود دارد، الگوریتمی مطلوب است که تمایل به تجزیه و تحلیل تئوری داشته باشد[34]. نشان داده شده است که تحدب موجب مقیاس‌پذیری در استنباط SVM می‌گردد. به‌طور مشابه، استدلال شده است [35] که معماری عمیق مانند شبکه‌های عصبی چندلایه با چند لایه پنهان، در نمایش وظایف یادگیری معمولی مانند پیش‌بینی، ادراک بصری و درک زبان معماری مانند SVM ها کارآمدتر و در نتیجه مقیاس‌پذیرتر هستند. دسته موازی نشان‌دهنده اکثریت در روش‌های ML مقیاس‌پذیر است. برای مقابله با ظهور داده‌های بزرگ با ابعاد بزرگی از ویژگی‌ها و حجم نمونه، روش‌های موجود در این دسته از هندسه داده‌ها در ورودی و / یا فضای الگوریتم / مدل بهره‌برداری می‌کنند [30]. به‌طورخاص، روش‌های موازی‌سازی را که الگوریتمML را مقیاس‌پذیر می‌کند به دو زیرمجموعه طبقه‌بندی می‌کنیم: (1) موازی‌سازی داده‌ها: اعمال نفوذ در معماری داده‌های بزرگ موجود، پارتیشن‌بندی داده‌های ورودی به‌صورت عمودی، افقی و یا حتی خودسرانه به قطعات قابل کنترل و سپس محاسبه تمام زیرمجموعه‌ها به طور همزمان، و (2) مدل / پارامتر موازی‌سازی: ایجاد نسخه‌های موازی از الگوریتم یادگیری توسط اولین تقسیم مدل یادگیری / پارامترها و پس از آن محاسبه هر بلوک ساختاری به‌صورت همزمان. باید توجه داشته باشیم که برخی از تلاش‌ها مانند [36-39] از هر دو روش موازی‌سازی داده‌ها و موازی‌سازی مدل / پارامتر پشتیبانی می‌کنند. برخی تلاش‌های دیگر مانند [40] از یک نوع تکنیک موازی‌سازی پشتیبانی می‌کنند. در ادامه، هر نوع روش و فرصت و چالش کلیدی را در آموزش داده‌های بزرگ بحث می‌کنیم.

4.1 عدم موازی‌سازی

بهینه‌سازی در قلب بسیاری از روش‌های ML قرار دارد. روش‌های بهینه‌سازی سنتی به بهینه‌سازی ترکیبی (جستجوی حریصانه، جستجو پرتو، شاخه و حد) و بهینه‌سازی پیوسته طبقه‌بندی می‌شوند [57]. سپس به بهینه‌سازی نامحدود

(به‌عنوان مثال، گرادیان نزولی، گرادیان مزدوج، روش شبه نیوتن) و محدود (به‌عنوان مثال، برنامه‌ریزی خطی، برنامه‌نویسی درجه دوم) گروه‌بندی می‌شوند. زمانی که مجموعه داده آموزش بزرگ باشد بهینه‌سازی محدود اغلب پرهزینه است. یک راه‌حل ممکن محاسبه بهینه تقریبی است. الگوریتم بزرگ مقیاس بهینه‌سازی، از گرادیان نزولی تصادفی، فاصله مختصاتی تصادفی و بهینه‌سازی توزیع شده به خصوص برای یادگیری الگوریتم‌های تصادفی تقریبی از داده‌های بزرگ مقیاس [41] استفاده می‌کند. با این وجود، روش گرادیان نزولی تصادفی به برای موازی‌سازی دشوار است [58] و بعید است که عملکرد شگفت‌انگیزی برای مسائل در مقیاس بزرگ به همراه داشته باشد.

بهینه‌سازی پارامتر یک چالش محاسباتی برای یادگیری روش‌ها با بسیاری hyperparameters است. برای مسائل یادگیری در مقیاس بزرگ، به دست آوردن پارامترهای مدل بهینه با حرکت از طریق داده‌ها مطلوب است [42]. برای این منظور، گرادیان تصادفی مرتبه دوم و شیب تصادفی به‌طور متوسط در مجموعه آموزش کارآمد هستند [43]. علاوه براین، استفاده از توابع تحلیلی (مدل‌های نگاشت) در معماری داده‌های بزرگ جایگزین دیگری برای ایجاد بهینه‌سازی پارامتر در مقیاس گسترده است [59]. مطالعات زیادی در مورد چگونگی موازی‌سازی روش‌های بهینه‌سازی محدود در بسیاری از الگوریتم‌های یادگیری مانند SVM ها، مسائل حداقل مربعات نامنفی و رگرسیون منظم L1 (LASSO) وجود دارد [29]. با تبدیل این مسائل به مجموعه‌ای از عملیات ضرب ماتریس-بردار، موازی‌سازی می‌تواند با صراحت برای اجرای نگاشت کاهش و یا مدل برنامه‌نویسی موازی GPU استفاده شود. علاوه براین، استفاده از حافظه محدود BFGS و گرادیان مزدوج با جستجو خطی می‌تواند ساده‌سازی قابل توجهی در سرعت بخشیدن به روند آموزش ویژگی‌های بدون نظارت و الگوریتم‌های عمیق با استفاده از روش گرادیان تصادفی، به خصوص با توجه به تنظیم GPU ها و یا خوشه‌ها داشته باشد [58].

4.2 موازی‌سازی داده‌ها

مدل ML موجود می‌تواند از تکنیک‌های داده‌های بزرگ برای رسیدن به مقیاس‌پذیری استفاده کند. چنین تلاش را می‌توان به دو دسته طبقه‌بندی کرد. یکی ارائه‌ی یک لایه میان‌افزار که وظایف را مجدد پیاده‌سازی می‌کند به‌طوری‌که

بتوانند بر روی یک پلت‌فرم داده بزرگ مانند Hadoop و Spark اجرا شوند. مانند یک لایه میان‌افزار، اغلب شکلهای هندسی اولیه عمومی / عملیات‌هایی برای بسیاری از وظایف یادگیری مفید فراهم می‌کند. این روش برای کاربرانی مناسب است که می‌خواهند وظایف آموزشی / الگوریتم‌های مختلف را در همان چارچوب امتحان کنند. دسته دیگر برای تبدیل الگوریتم‌های یادگیری فردی برای اجرا بر روی پلت‌فرم داده بزرگ است. این پیاده‌سازی‌ها به‌طورمعمول فراتر از داده‌های بزرگ هستند و می‌توانند به مقیاس‌پذیری و یا نتیجه بهتر دست یابند.

4.2.1 میان‌افزار عمومی داده‌های بزرگ برای الگوریتم‌های یادگیری موجود

Spark MLlib [47] و Mahout [48] دو پروژه منبع باز هستند که از بسیاری الگوریتم یادگیری مقیاس‌پذیر حمایت می‌کنند. بسیاری از الگوریتم‌های یادگیری مشترک، از جمله طبقه‌بندی، رگرسیون، خوشه، فیلتر مشترک و کاهش ابعاد، توسط هر دو پروژه پشتیبانی می‌شوند. چون آنها یک لایه مستقل ایجاد می‌کنند که الگوریتم‌های رو به جلو از الگوریتم‌های رو به عقب جدا می‌شوند. به‌عنوان مثال، Mahout از هادوپ، Spark و H2O به‌عنوان مهندسی داده‌های بزرگ یاد کرده‌اند. علاوه براین، هر چند این الگوریتم‌ها می‌توانند برای پردازش مجموعه داده‌های بزرگ در یک محیط توزیع شده استفاده شوند، ولی استفاده از آنها بسیار شبیه به اجرا بر روی یک ماشین کوچک با مجموعه داده است. علاوه براین، این لایه مستقل قادر به بهینه‌سازی بین برنامه منطقی کاربران و برنامه‌های فیزیکی است که می‌توانند در یک محیط توزیع شده اجرا شوند. همچنین پروژه‌هایی برای یادگیری جریان داده ها در مسائل بزرگ مقیاس، مانند SAMOA وجود دارد [71]. بااین حال آنها هنوز در مراحل اولیه هستند.

4.2.2 تلاش بر روی الگوریتم‌های خاص با داده‌های موازی

اگر چه میان‌افزار بالا از بسیاری از الگوریتم‌های رایج یادگیری برای داده‌های بزرگ پشتیبانی می‌کند، هنوز هم هر دو نیاز عملی و پژوهشی را برای گسترش الگوریتم ML برای پشتیبانی از داده‌های بزرگ، به‌ویژه برای الگوریتم‌های جدید که کمتر استفاده می‌شوند به همراه ندارد. بسیاری از الگوریتم‌های ML را می‌توان با نگاشت کاهش، از جمله رگرسیون

خطی، kmeans، رگرسیون لجستیک، بیز ساده، SVM، ICA، PCA، EM، شبکه عصبی و غیره پیاده‌سازی کرد [49]. الگوریتم ضرب ساده را می‌توان برای پیاده‌سازی الگوریتم‌های ML مانند SVM ها و مسائل نامنفی حداقل مربعات در محیط‌های محاسباتی موازی با استفاده از نگاشت کاهش مورد استفاده قرار داد [60]. نگاشت کاهش برای رسیدن به داده کاوی موازی محل مشترک [61]، طبقه‌بندی نزدیکترین همسایه [62]، و شبکه یادگیری بیزی استفاده می‌شود [44،45]. بسیاری از این مسیرهای جدید به طور فزاینده‌ایاز ML / خط لوله استفاده می‌کنند که نیاز به سیستم‌هایی برای استفاده از ابزار و تکنیک‌های ترکیبی دارند [63].

4.3 مدل‌ها / موازی‌سازی پارامتر

میزبان تلاش‌ها در مورد چگونگی موازی‌سازی الگوریتم ML (به عنوان مثال، [64]) و یا تضمین عملکرد در الگوریتم‌های موازی‌سازی مختلف قرار داده شده است (به‌عنوان مثال، [65]). این تلاش‌ها به دلیل بسیاری از الگوریتم‌های ML در بهترین موازی‌سازی تضمین شده هستند [66-68]. علاوه براین، داده‌های بزرگ ML به سادگی یک نسخه کوچک از ML برای داده کوچک نیستند. آن‌ها نیاز به فرمولاسیون‌های مختلف و الگوریتم برای رسیدگی به چالش‌های فنی مرتبط با آن هستند. موازی‌سازی الگوریتم‌های یادگیری ریشه در ML توزیع شده و ML بزرگ مقیاس دارند. بنابراین، در مورد فرصت‌ها و چالش‌های ML در داده‌های بزرگ از دیدگاه‌های زیر بحث می‌کنیم: ML توزیع شده، موازی‌سازی در چندین پارادایم اصلی ML و یادگیری عمیق.

4.3.1 یادگیری ماشین توزیع شده

ML توزیع شده به‌طور طبیعی می‌تواند به حل پیچیدگی‌های الگوریتم و مشکل محدودیت حافظه در ML بزرگ مقیاس بپردازد [69]. برای نشان دادن ناتوانی الگوریتم ML برای استفاده از تمام داده‌ها برای یادگیری در مدت زمان معقول، ML توزیع شده الگوریتم‌های یادگیری با تخصیص فرایند یادگیری بر روی چندین کامپیوتر یا پردازنده [69]، و حل یک مسئله بهینه‌سازی توزیع شده را مقیاس‌گذاری می‌کنند[70]. ML توزیع شده نه تنها می‌تواند به بهره‌وری

از داده‌های موازی برسد بلکه با تکرار داده‌ها در سراسر ماشین‌ها تحمل‌پذیری خطا را افزایش می‌دهد. علاوه براین، استفاده از فرآیندهای مختلف یادگیری برای آموزش چندین طبقه‌بند از مجموعه دادههای توزیع شده امکان دستیابی به دقت و صحت بالاتر را در یک دامنه بزرگ افزایش می‌دهد [69]. یکی دیگر از مزیت‌های الگوریتم‌های توزیع شده بخش‌های یکپارچه مدیریت داده‌ها (به‌عنوان مثال، [71]) است. بااین‌حال، طراحی و اجرای الگوریتم‌های موازی کارآمد و بسیار چالش برانگیز هستند [52]. علاوه براین، الگوریتم ML موازی سنتی، محاسبات را بین گره‌ها توزیع می‌کند، تا در ماشین‌های موازی اختصاصی با ارتباطات سریع در میان گره‌ها به خوبی کار کنند اما به هنگام انتقال داده‌ها در سراسر شبکه عملکرد نه چندان خوبی دارند، که منجر به هزینه‌های ارتباطی بالا به علت زمان شبکه می‌گردد. بنابراین، دسترسی به داده‌ها را از دیسک‌های محلی بسیار ارجح است. اما بیشتر الگوریتم‌های ML برای رسیدن به محلیت خوب داده‌ها طراحی نشده‌اند. علاوه براین، تاخیر ارتباطات بین ماشین‌های مختلف ممکن است مشکلاتی در همگرایی ایجاد کند حتی اگر یک الگوریتم توزیع ندشه نرخ همگرایی خوبی را از خود [70] نشان دهد.

GraphLab یک چارچوب موازی برای ML است که از ساختار پراکنده و الگوهای محاسباتی مشترک از الگوریتم ML [52] استفاده می‌کند. این چارچوب به پژوهشگران ML کمک می‌کند تا الگوریتم‌های موازی مقیاس‌پذیر و کارآمد برای مواجه با مسائل محاسباتی خاص، وابستگی داده‌ها و برنامه‌ریزی، در یک حافظه اشتراکی طراحی و پیاده‌سازی کنند.. با این وجود، پوسته پوسته شدن موجب می‌شود که تکنیک‌های ML برای داده‌های بزرگ و توزیع شده هنوز هم به عنوان چالشی بزرگ مطرح باشند. الگوریتم براساس وزن برای یادگیری یک ساختار شبکه بیزی از مجموعه‌ای از نتایج محلی ارائه شده است، که کپلر علمی را برای ایجاد فرایند یادگیری به کار می‌برد [44].

4.3.2 موازی‌سازی الگوریتم‌های سنتی ML

در این مقاله در مورد موازی‌سازی چندین الگوریتمML نظارت شده و سنتی، از جمله SVM با هسته‌ی گاوسی و چندجمله‌ای، شبکه‌های بیزی و درخت‌های تصمیم‌گیری بحث می‌کنیم.

ماشین بردار پشتیبان (SVM ها) روش‌های طبقه‌بندی شده‌ای بنا به پایه ریاضی هستند، که دو ویژگی برجسته را نشان می‌دهند: حاشیه حداکثر و طبقه‌بندی غیرخطی با استفاده از هسته. با این‌حال، الگوریتم SVM مبتنی بر هسته از مشکل مقیاس‌پذیری رنج می برند که نیاز به محاسبات یک ماتریس هسته با زمان و فضای پیچیدگی O (N2) دارد[73]. SVM اخیرا در حوزه محاسبات با کارایی بالا از طریق قدرت / پیش‌بینی کارایی، تنظیم و برنامه‌ریزی زمان اجرا عمل می‌کند [37]. SVM های موازی یکی از بهترین روش‌های طبقه‌بندی هستند [74]. ایده اصلی این روش معرفی یک گام بهینه‌سازی موازی برای حذف سریع بسیاری از بردارهای پشتیبان است، که در آن ماتریس بلوکی مورب برای تقریب ماتریس هسته اصلی استفاده شده است به طوری‌که مشکل اصلی را می‌توان به صدها زیرمسئله که می‌توانند کاراتر حل شود تقسیم کرد. علاوه براین، برخی از استراتژی‌های موثر مانند ذخیره هسته و محاسبات کارآمد ماتریس هسته، به‌طور یکپارچه برای سرعت بخشیدن به روند آموزش [55]، و کاهش قابل توجه محاسبات و سربار حافظه مورد نیاز برای محاسبه ماتریس هسته، بدون تاثیر قابل توجه نتایج استفاده می‌شوند [73].

شبکه‌های بیزی یک نمایش قدرتمند احتمالاتی (مدل‌های گرافیکی) هستند. تحولات قابل توجهی وجود دارد که موجب، انعطاف‌پذیری و مقیاس‌پذیر یادگیری بیزی شده است. یادگیری بیزی شامل روشهای بیزی غیرپارامتری برای انطباق استنتاج پیچیدگی مدل، استنتاج بیزی منظم برای بهبود انعطاف‌پذیری از طریق تنظیم و الگوریتم‌های مقیاس‌پذیر و سیستم‌های براساس نمونه‌برداری تصادفی و محاسبات توزیع شده برای برخورد با برنامه‌های کاربردی در مقیاس بزرگ است [75].

درخت‌های تصمیم بنا به نتایج یادگیری خود برتر شناخته شده‌اند. جنگل‌های تصادفی تاثیر خود را برای تحلیل‌های پیش‌بینی بر روی داده‌های با ابعاد بالا در برنامه‌های مختلف نشان داده‌اند (به‌عنوان مثال، [76]). علاوه براین، یادگیری موازی درخت با استفاده از نگاشت کاهش در خوشه‌های رایانه برای ایجاد طبقه‌بندی و رگرسیون مقیاس‌پذیر استفاده شده است [38].

4.3.3 یادگیری عمیق

به تازگی، یادگیری عمیق عصبی مبتنی بر شبکه یکی از سریع‌ترین و هیجان‌انگیزترین مناطق در ML با داده‌های بزرگ شناخته شده است. شبکه‌های عصبی یک خانواده از مدل‌ها با الهام از شبکه‌های عصبی بیولوژیکی در سلول‌های عصبی به هم پیوسته هستند که شامل تنظیم ارتباطات سازگار با ورودی هستند. شبکه‌های عصبی عمیق را می‌توان به عنوان شبکه‌های عصبی با لایه‌های پنهان بزرگ، یا معماری لایه عمیق با هر لایه در یک تحول غیرخطی از ورودی به خروجی عنوان کرد. به‌تازگی، داده‌های بزرگ و تکنیک‌های جدیدی را برای آموزش شبکه‌های عمیق، همراه با دردسترس بودن کامپیوتر (به‌عنوان مثال، پردازنده سریع و ظهور GPU ها)، اتصال سریع به شبکه و زیرساخت‌های نرم افزاری بهتر ایجاد کرده‌ایم که فرصت‌های بزرگی برای تحقیقات یادگیری عمیق ایجاد کرده است. به‌عنوان مثال، نرم‌افزارهای مختلف یادگیری عمیق و کتابخانه‌هایی مانند Theano [53]، Caffe [54]، Torch7 [40]، Tensorflow [36]، برای توانمندسازی GPU برنامه‌های کاربردی یادگیری ایجاد شده است. در میان آنها، Theano به‌عنوان یک پردازنده ریاضی نمادین برای تمایز نمادین و یا ادغام، در توابع پیچیده غیرخطی شناخته شده است. که به طور گسترده توسط شبکه‌های عصبی و محققان ML به عنوان یک محیط مفید برای توسعه الگوریتم‌های جدید ML استفاده شده است. چارچوب Caffe شامل مجموعه بزرگی از مدل‌های مناسب شبکه عصبی از پیش آموزش دیده برای انواع کارهای طبقه‌بندی تصویر است. علاوه براین، Tensorflow گوگل یک کتابخانه منبع باز برای محاسبات عددی با استفاده از نمودار جریان داده‌ها است. با Tensorflow، محاسبات را می‌توان در یک یا چند پردازنده و یا به حد کافی GPU مستقر کرد. در چند سال گذشته، یادگیری عمیق شاهد رشد فوق العاده‌ای در گسترده وسیعی از برنامه‌های کاربردی، از جمله پردازش تصویر و بینایی کامپیوتر [77- 81]، گفتار و پردازش زبان طبیعی [82،83]، سلامت [84]بوده است.

به‌طور معمول، شبکه‌های عصبی عمیق را می‌توان در دو حالت مختلف آموزش داد (1) یادگیری نظارت شده که در آن تعداد زیادی از کارهای مرتبط به عنوان داده‌های دردسترس برچسب شده‌اند (2) یادگیری خودآموخته (که یادگیری بدون نظارت نیز نامیده می‌شود) که در آن داده‌های آموزشی می‌تواند به صورت خودکار از داده‌های بدون برچسب

بدون تلاش انسان تولید شود [85]. به‌عنوان مثال، ImageNet [79] یک مجموعه داده از تصاویر با بیش از 14 میلیون تصویر نشاندار شده با بیش از 20 هزار مفهوم است. تصاویر هر مفهوم با کیفیت کنترل شده و انسان مشروح هستند. این مجموعه داده برچسب شده غالبا در آموزش تصویر سیستم‌های تشخیص استفاده می‌شود که یادگیری عمیق را به کار می‌برند [77]. از 2015، یک قانون ارائه شده است که یک الگوریتم یادگیری عمیق تحت نظارت به طور کلی با حدود 5000 برچسب نمونه در هر رده به عملکرد قابل قبولی دست خواهد یافت که با عملکرد انسان زمانی که با یک مجموعه داده شامل حداقل 10 میلیون نمونه برچسب شده مطابقت دارد [86].

الگوریتم‌های یادگیری عمیق نیز می‌توانند از مقدار زیادی داده بدون نظارت برای یادگیری خودکار نمایش پیچیدگی استفاده کنند [3]. بهترین نتایج به‌دست آمده برای انجام وظایف یادگیری نظارت شده اغلب شامل یک گام یادگیری ویژگی بدون نظارت هستند [87]. برای مثال، در پردازش زبان طبیعی (NLP)، یادگیری بدون نظارت از دست دادن کلمه [88] ثابت کرده است که در بسیاری از کارهای NLP بسیار موثر است. اگر چه هیچ داده مشروحی برای آموزش یک مدل مورد نیاز نیست، سیستم به‌طور خودکار یک مدل شبکه عصبی که قادر به استخراج نمایش برداری از یک کلمه براساس چگونگی پیش‌بینی درست کلمات همسایه در آن زمینه است ارائه می‌کند. در پردازش تصویر، یک رمزگذار خودکار متشکل از یک رمزگذار و یک رمزگشا است که اغلب برای آموزش ویژگی‌های بدون نظارت استفاده می‌شود که در آن رمزگذار از داده‌های خام (به عنوان مثال، تصویر) به عنوان ورودی و ویژگی‌ها به‌عنوان خروجی. و رمزگشا از ویژگی استخراج شده از رمزگذار به عنوان ورودی و بازسازی داده‌های ورودی اولیه اصلی به‌عنوان خروجی استفاده می‌کند. هدف یادگیری خودکار بازنمایی تصویر برای به حداقل رساندن تفاوت بین و تصویر بازسازی شده و خام است.

شبکه‌های عصبی عمیق ماشین‌ها را ویژگی‌های طراحی شده با دست جابجا می‌کنند زیرا زمان و هزینه حافظه برای آموزش یک ماشین کرنل بنا به اندازه مجموعه داده از درجه دوم است و مجموعه داده برای این هزینه به مراتب سنگینتر از مزایای بهینه‌سازی محدب است. از سوی دیگر، دردسترس بودن داده برچسب شده بسیار متفاوت از یک دامنه به دیگری است. بنابراین، یکی از چالش‌های اصلی استفاده از یادگیری عمیق است که به خوبی از مجموعه

داده‌های کوچکتر با استفاده از مقادیر زیادی داده‌های بدون برچسب، با تکنیک‌های یادگیری نیمه‌نظارتی و نظارت نشده تعمیم یافته است.

4.4 روش‌های ترکیبی

روش ترکیبی، موازی‌سازی مدل و داده را توسط پارتیشن‌بندی داده و الگو به‌طور همزمان ترکیب می‌کند. این امر نه تنها منجر به یادگیری سریع بر روی خوشه‌های توزیع شده می‌شود، بلکه منجر به اجرای موثر برنامه‌های کاربردی ML برای داده‌ها و مدل بیش از حد بزرگ در حافظه یک ماشین می‌شود [46]. به‌عنوان مثال، DistBelief یک چارچوب نرم‌افزاری طراحی شده برای آموزش توزیع شده و یادگیری شبکه‌های عمیق با مدل‌های بسیار بزرگ (به‌عنوان مثال، چند میلیارد پارامتر) و مجموعه داده‌های بسیار بزرگ است. که موجب تعادل در خوشه‌های بزرگ از ماشین‌ها به مدیریت داده‌ها و موازی‌سازی مدل از طریق چندرشته‌ای، عبور پیام، هماهنگ‌سازی و همچنین ارتباط بین ماشین‌ها می‌شود [56]. هدفSystemML ، در ML بزرگ مقیاس فراتر از نگاشت کاهش است، چرا که در آن الگوریتم ML به‌عنوان زبان اسکریپت سطح بالا بیان می‌شود. این زبان سطح بالا در معرض چندین ساختار قرار دارد که بلوک‌های کلیدی ساختمان برای یک گروه وسیع از و الگوریتم ML بدون نظارت و نظارت شده را تشکیل می‌دهد. الگوریتم‌های بیان شده در SystemML کامپایل شده و در یک مجموعه از کارهای نگاشتک اهش که می‌توانند در یک خوشه از ماشین‌ها اجرا شوند بهینه شده است [50]. یکی از چالش‌های مهم در این زمینه چگونگی تاثیر ترکیب هر نوع موازی‌سازی برای اسکریپت دلخواه ML و حجم کار است.

4.5 فرصت‌ها و چالش‌های کلیدی

علاوه بر بحث مفصل در مورد فرصت‌ها و چالش‌های حاضر داده‌های بزرگ ML در سراسر این بخش، فرصت‌ها و چالش‌های کلیدی نیز برجسته خواهند شد.

ML در داده‌های بزرگ نیاز به یک تفکر جدید در الگوریتم‌های بسیاری برای نشان دادن چالش‌های فنی دارد [89]. داده‌های بزرگ یکی از فعال‌کننده‌های کلیدی برای یادگیری عمیق است، که موجب بهبود عملکرد در برنامه‌های مختلف می‌گردد. یادگیری عمیق معمولا می‌توانید حداقل 1000 دسته مختلف را تشخیص دهد، که حداقل 2 مرتبه بالاتر از تعداد نمونه‌های به کار برده شده در شبکه‌های عصبی سنتی است [86]. علاوه براین، داده‌های بزرگ فرصت‌هایی را برای استنتاج علیت براساس زنجیره‌ای از توالی، برای فعال کردن پشتیبانی تصمیم‌گیری موثر فراهم می‌کنند.

نیاز به ML در داده‌های بزرگ، فرصت‌های منحصربه فردی برای طراحی سیستم و ML فراهم می‌کند. ML می‌تواند چگونگی طراحی سیستم را تحت تاثیر قرار دهد. از آنجا که بسیاری از برنامه‌های ML اساسا بهینه‌سازی شده هستند، راه‌حل‌های الگوریتمی تکراری-همگرایی در تحمل‌پذیری خطا و طراحی سیستم یکپارچه ممکن است مسائلی مانند محدود خطا در هماهنگ‌سازی شبکه‌ها و برنامه‌ریزی پویا بر اساس ML را در برداشته باشد [39]. شتاب سخت‌افزار، از جمله سوپرکامپیوتر، در حال توسعه است که تنها وظایف ML [90،91] را مورد هدف قرار می‌دهد.

یادگیری در داده‌های بزرگ یک فرصت بزرگ برای تحقیق در مدیریت گردش کار و برنامه‌ریزی کار ایجاد می‌کند. دلیل این امر این است که یکی از مسائل کلیدی ML در داده‌های بزرگ چگونگی تقسیم / برنامه‌ریزی کار / داده‌ها و ادغام پیش‌بینی‌های متعدد است. تکنیک بهینه‌سازی پرس و جو پایگاه داده می‌تواند برای شناسایی طرح‌های اجرایی موثر و نتایج برنامه زمان اجرا بر روی داده‌های موتور پردازش پرس و جو استفاده شود [51،92]. یک پیش‌گویی کُند می‌تواند در زیرساخت نگاشت-کاهش برای بهبود چابکی تصمیم‌های برنامه‌ریزی شده جاسازی شود [93]. آخرین MLlib می‌تواند دنباله‌ای از الگوریتم‌های خط لوله را به کار گیرد. که از اجرای مقیاس‌پذیر و تنظیم خودکار پارامتر برای همه الگوریتم‌ها در خط لوله پشتیبانی می‌کند.

ML در داده‌های بزرگ فرصت بی‌سابقه‌ای برای یادگیری با انسان در حلقه برای چندین دلیل ارائه می‌کند. ابتدا، ML در داده‌های بزرگ نیاز به مردمی با پس زمینه در هر دو الگوریتم ML و تکنیک‌های موازی‌سازی دارد که برای اکثر کاربران بسیار چالش‌برانگیز است. بدین ترتیب، در اینجا توجه زیادی به افزایش طراحی سیستم‌های ML شده است.

دوما، صرفا تکیه بر الگوریتم‌های ML ممکن است پتانسیل کامل داده‌های بزرگ را به همراه نداشته باشد زیرا الگوریتم‌ها ممکن است بسیاری از روابط جعلی را کشف کنند. بنابراین، نقاط قوت دانش بشری / تخصص می‌تواند برای الگوریتم‌های ML مفید باشد. سوما، در ML سنتی، کاربران اغلب نقش منفعل را (به‌عنوان مصرف‌کنندگان نتایج ML) بازی می‌کنند. برای پیوستن به کاربران و کمک به آنها برای به دست آوردن بینش داده‌های بزرگ، نیاز به حرکت به سمت تعامل با ML داریم. ارتباط تاثیرگذار ML متکی بر طراحی تکنیک‌های تعاملی براساس درک درک قابلیت کاربر نهایی، رفتارها و نیازها است [8]. با یادگیری تعاملی از کاربران نهایی، سیستم‌های ML می‌توانند نیاز به نظارت توسط کارشناسان و توانمندسازی کاربران نهایی برای ملاقات سیستم ML داده‌های بزرگ برای رفع نیازهای خود را استخراج کنند.

ML در داده‌های بزرگ بر اهمیت حفظ حریم خصوصی ML تاکید دارد. داده‌های بزرگ ممکن است بسیار شخصی باشند [1]. به‌عنوان مثال، اطلاعات بهداشتی ممکن است از سازمان‌های مختلف جمع آوری شده و برای حفظ سیاست حریم خصوصی ممکن است به صراحت داده خود را برای عموم به اشتراک گذاشته نشوند. از آنجا که اتصال به ML گاهی اوقات ممکن است لازم باشد، چگونگی به اشتراک گذاشتن داده‌ها بزرگ در میان نهادهای ML توزیع شده نیز برای کاهش نگرانی‌های حریم خصوصی یک مشکل به چالش کشیده شده است. به‌عنوان مثال، حفظ حریم خصوصی ML با به کار بردن محلیت داده‌ها در معماری هادوپ به دست می‌آید و تنها تعداد محدودی از عملیات رمزنگاری در مرحله کاهش مورد نیاز است [94]. یک راه‌حل برای حفظ حریم خصوصی طبقه‌بندی SVM ارائه شده است [95].

داده‌های بزرگ تاثیر ML در دنیای واقعی را افزایش می‌دهند. برنامه‌های کاربردی در ML تاثیر وسیعی در دنیای واقعی علم (به‌عنوان مثال، طراحی فیزیکی، علوم زیستی، پیش‌بینی زلزله) کسب‌وکار (به‌عنوان مثال، سیستم‌های مالی، نظارت بر دارو پس از تصویب، رانندگی خودکار اتومبیل) و پلتفرم‌های عمومی (به‌عنوان مثال، رسانه‌های اجتماعی) قلمرو سازمانی (به‌عنوان مثال، شبکه‌های نفوذ، سیستم بهداشت و درمان) دارند.

از جمله چالش‌های موجود برای ML در داده های بزرگ، یک موضوع کلیدی بهبود بهره‌وری برای تکرارها است. چارچوب‌های موازی‌سازی موجود به ویژه برای الگوریتم‌های ML طراحی نشده‌اند. ابزارهای داده بزرگ، محاسبات را در حالت دسته‌ای انجام می‌دهند و برای انجام وظایف با پردازش تکراری و با وابستگی بالای داده در میان عملیات (به

عنوان مثال، با توجه به دیسک سنگین I / O) بهینه‌سازی نشده‌اند. کارهای فرعی تکرار شونده (به‌عنوان مثال، مراحل پردازش که مکررا تا زمانی که شرایط همگرایی مرتفع گردد اجرا می‌شوند) بر هر دو دسته از الگوریتم‌ها تسلط دارند. بهینه‌سازی تخصیص منابع خوشه در میان حجم کارهای متعدد از الگوریتم‌های تکراری اغلب نیاز به برآورد زمان اجرا دارد، که عبارتند از: (الف) پیش‌بینی تعداد تکرارها و (ب) پیش‌بینی زمان پردازش هر تکرار [96]. زیرساخت هادوپ می‌تواند از کارهای و یا انجام وظایف بسیار آهسته و رسیدگی به آنها در زمان اجرا (از طریق اجرای حدسی) اجتناب کند. Spark [92] نه تنها از نگاشت کاهش و تحمل‌پذیری خطا پشتیبانی می‌کند بلکه داده‌های کش در حافظه بین تکرارها را نیز پشتیبانی می‌کند. توجه داشته باشید که، روش‌ها برای بهبود بهره‌وری محاسبات در داده‌های بزرگ بدون قربانی کردن عملکرد ML توسعه یافته‌اند، که تنها تکه‌های کوچکی از داده‌ها را به‌جای تمام داده‌ها در حافظه سریع نگه می‌دارند و یک پیش‌بینی کننده در هر قطعه کوچک می‌سازند و سپس آنها را باهم ترکیب می‌کنند [97]. علاوه براین، نمودار مبتنی بر معماری و ابزارهای داده‌های بزرگ در حافظه برای به حداقل رساندن هزینه I / O و بهینه‌سازی پردازش تکراری توسعه یافته‌اند [98].

چالش دیگر به حداقل رساندن انتقادات / ارتباطات از / با طبقه‌بندی است. مشکل یادگیری زنجیره‌ی طبقه‌بندی بهینه در زمان اجرا به‌عنوان یک مسئله چند موردی با بازخورد محدود است [99]. که نیاز به طبقه‌بندی محلی توزیع شده برای تبادل هر گونه اطلاعات به جز نظرات محدود خود در عملکردکاوی برای فعال کردن یادگیری زنجیره طبقه‌بندی بهینه دارد [99].

چالش سوم، رسیدگی به جنبه سرعت داده‌های بزرگ در ML است. راه‌حل‌های حاضر (استاندارد) برای تجزیه و تحلیل داده‌های بزرگ برای مقابله با جریان تکامل طراحی نشده‌اند [100]. یک سیستم ML باید قادر به مقابله با هجوم تغییر دادهها در روش مستمر باشد. یادگیری ماشین در مقابل یک شات یادگیری سنتی است [101]. برای این منظور، آموزش آنلاین مورد بهره‌برداری قرار شده است تا روش هسته کارآمد و مقیاس‌پذیر برای برنامه‌های کاربردی یادگیری ماشین در مقیاس بزرگ استفاده شود. به‌عنوان مثال، دو الگوریتم مختلف آنلاین هسته ML - الگوریتم تبدیل آنلاین

فوریه گرادیان و گرادیان آنلاین نیستروم برای مقابله با سه وظیفه یادگیری آنلاین مشخص شده‌اند: طبقه‌بندی باینری، طبقه‌بندی چندکلاسی و رگرسیون [102103].

چالش چهارم رسیدگی به جنبه‌های مختلف از داده‌های بزرگ در ML است. الگوریتم‌های سنتی ML تنها می‌توانند نوع خاصی از ورودی را، مانند عدد، متن یا تصاویر دریافت کنند. در بسیاری از موارد، داده‌ای که می‌تواند برای یک هدف واحد ML مورد استفاده قرار گیرد ممکن است در انواع و فرمت‌های مختلف باشد. که موجب می‌شود ویژگی‌ها از دست بروند و گاهی اوقات به‌عنوان " ابعاد بزرگ " مشخص می‌شود [104]. به‌عنوان مثال، یک الگوریتم ML ممکن است نیاز به یادگیری داشته باشد: (1) مخلوطی از حجم زیادی از داده‌ها و جریان با سرعت بالای داده‌ها یا (2) حجم زیادی از داده‌ها با ویژگی‌های تصویر، متن، صوت و حرکت.

چالش پنجم پیچیدگی مشکل (به‌عنوان مثال، در طبقه‌بندی چند کلاس و کلاس‌های جدید) را افزایش می‌دهد. در سند و خوشه‌بندی تصویر / طبقه‌بندی، علاوه بر تعداد زیادی نقاط داده و ابعاد بالای انها، تعداد خوشه‌ها / کلاس‌ها نیز بزرگ است. بنابراین، لازم است تا به تدریج ظرفیت مدل‌های ML برای پیش‌بینی روند فزاینده‌ی کلاس‌های جدید گسترش یابد [105].

ML در داده‌های بزرگ چالش‌های متعدد دیگری نیز به همراه دارد. برای مثال، بهینه‌سازی در ML معمولی بر عملکرد متوسط تمرکز دارد، اما جلوگیری از نتایج ضعیف سخت است. الگوریتم‌های سنتی ML برای داده‌هایی که در حافظه به طور کامل لود نشده‌اند طراحی نشده‌اند. علاوه براین، داشتن مجموعه‌ای از توابع هدف با توجه به تعداد زیاد اصطلاحات و توازن بین عملکرد پیچیده است. نویز یک مسئله بزرگ برای داده‌های بزرگ به دلیل الگوهای موجود در یک زیرمجموعه کوچک از داده‌ها (به عنوان مثال، اسپم‌ها و حملات اینترنتی) است.

5. فرصت‌های ارزیابی و چالش‌ها

ML سنتی دارای مجموعه معیارهایی برای ارزیابی عملکرد، مانند دقت، میزان خطا، جامعیت، مربع خطا، احتمال، کسب اطلاعات، اختلافK-L ، هزینه، ابزار، حاشیه، خطا بهینه‌سازی، برآورد خطا، تخمین و میانگین و بدترین نتیجه

است. این معیارها بر دقت پیش‌بینی ML تمرکز دارند. علاوه براین، مقیاس‌پذیری، به صورت سنتی برای ارزیابی یک برنامه موازی، تجزیه و تحلیل داده بزرگ استفاده می‌شود. مقیاس‌پذیری به عنوان معیار داده در عملیات I / O، تحمل‌پذیری خطا، زمان واقعی استفاده از پردازش، حافظه، اندازه داده پشتیبانی، وظیفه پشتیبانی تکرارشونده، و توان استفاده می‌شود [106].

ارزیابی داده‌های بزرگ ML یک ترکیب ساده از دو نوع معیار نیست. بلکه نیاز به نشان دادن هر نوع معیار پیچیده‌ای بین آنها دارد. به‌عنوان مثال، دقت و یادآوری، دقت و زمان پاسخ، معیار سنجش عملکرد کلاسیک هستند. حمایت از کارهای تکراری در برابر تحمل‌پذیری خطا در حمایت از مقیاس‌پذیری (به عنوان مثال، نگاشت کاهش از تحمل‌پذیری خطا پشتیبانی می‌کند اما از تکرار نه) دارد. علاوه براین، الگوریتم غیرتکراری (به‌عنوان مثال، تقریب نیستروم) مقیاس بهتری از الگوریتم تکرارشونده (به‌عنوان مثال، تجزیه Eigen) دارد اما عملکرد آن کمی بد است. اگر چه در رسیدن به SVM خطی از یک مورد غیرخطی سریعتر است،اما موازی‌سازی آن از دومی سختتر است. علاوه براین، معاوضه معمولی بین محاسبات و ارتباطات به ویژه برای داده‌های بزرگML قابل اجرا است. الگوریتم باید به دقت طراحی شده باشد به‌طوری‌که زمان ذخیره شده بر روی محاسبات بتواند هزینه‌های مرتبط با ارتباطات را جبران کند. نگاهی به SVM موازی به عنوان مثال می‌اندازیم؛ هر چند هزینه های بالای محاسباتی آن (غیرخطی) است، ارتباطات داده‌ها / هزینه بارگذاری آن کمتر است. بسیاری از روش‌ها (به‌عنوان مثال، گرادیان نزولی تصادفی یا فاصله مختصاتی) ذاتا سریال هستند و در نتیجه هزینه ارتباطات جزو مسائل نگران‌کننده است. علاوه براین، پژوهش سنتی ML تنها زمان را در نظر می‌گیرد (بستگی به تعداد عملیات در حال اجرا دارد) اما زمان بارگذاری داده‌ها را (بسته به تعداد دسترسی) نادیده می‌گیرد. برای الگوریتم‌های خطی، زمان بارگزاری می‌تواند بزرگتر از زمان در حال اجرا باشد. که در مقابل برای روش های هسته درست است.

پیچیدگی الگوریتم ML موجود، اغلب قریب به اتفاق به چرایی بسیاری از کاربران و درک چالش‌ها در پارامترسازی و انتخاب بین تکنیک‌های مختلف یادگیری بستگی دارد. به‌عنوان مثال، برای اجرای یک الگوریتم ML، کاربران نیاز به مجموعه با تعدادی زیادی پارامتر دارند. از آنجا که مقادیر پارامترها بیش از حد ممکن در زمان اجرا و نتایج تأثیر دارد،

انتخاب پارامترهای مناسب در برنامه‌های کاربردی ML حیاتی است. بااین حال، سیستم ML موجود به‌طور معمول هیچ کمکی در مورد چگونگی تنظیم پارامترها ارائه نمی‌کند. علاوه براین، از آنجا که درک این الگوریتم می‌تواند برای مردم سخت باشد یک پس زمینه قوی در سیستم‌های ML یا سیستم‌های توزیع شده می‌طلبد، بنابراین پیدا کردن پارامترهای درست می‌تواند بسیار چالش برانگیز باشد [107].

توجه زیادی به قابلیت استفاده از مدل‌های ML از نظر interpretability، سهولت استفاده، ثبات، و غیره وجود دارد. در میان آنها، سهولت استفاده از معیارها، بیشتر مورد استفاده است. برخی از معیارهای سهولت استفاده عبارتند از پیچیدگی در تنظیم توابع هدف، انعطاف‌پذیری، متوسط خطا و الگوی «تنوع». بااین حال، interpretability و ثبات [99100] از ملاحظات اصلی برای طراحی بسیاری از الگوریتم‌های ML نیستند. درک و توضیح اساسی فرآیند از اطلاعات قابل مشاهده یک چالش مهم آماری برای ML است. در مقابل، مدل‌های ML-بر اساس قانون بصری هستند و قادر به بیان روابط علت و معلول نیز می‌باشند. با این اوصاف، مدل‌های مبتنی بر قوانین، با مجموعه چالش‌های پژوهش مانند قوانین حکومت، ارزیابی، اجرا، بهینه‌سازی، تعمیر و نگهداری مواجه می‌شوند.

رویکرد اعلانی به ML یک راه برای در دسترس قرار دادن ML برای غیرمتخصص است [108]. برای پرداختن به عدم پشتیبانی از داده مستقل و مشخصات اعلانی در راه‌حل‌های داده‌های بزرگ، MLbase برای مهار قدرت ML برای کاربران نهایی غیرمتخصص و محققان ML توسعه یافته است [107].

جدول 2. مسائل پژوهشی حاضر در زمینه یادگیری ماشین بر روی داده‌های بزرگ

Main Component	Aspects	Open research issues
Big Data	Volume	• Cleaning and compressing big data • Large scale distributed feature selection • Workflow management and task scheduling
	Velocity	• Real time online learning for streaming data
	Variety	• Multi-view learning for heterogeneous multimedia data • Multimedia neural semantic embedding
	Veracity	• Assessing data veracity • Learning with unreliable or contradicting data
	Value	• Explainable ML for decision support • Multi-user collaborative decision support based on big data analysis
User	Labeling Evaluation Privacy	• Crowd sourced active learning for effective large scale data annotation • Comprehensive evaluation measures for ML (e.g, usability-based measures) • Privacy preserving distributed ML
	User Interface	• Visualizing big data • Intelligent user interfaces for interactive ML • Declarative ML
Domain	Domain knowledge	• Incorporating general domain knowledge (e.g., ontology, first-order logic, business rules) in ML
System	Infrastructure	• New infrastructure that seamlessly provides decision support based on real time analysis of large amount of heterogeneous and unreliable data. • General big data middleware

سیستم فراهم می‌کند: (1) یک ساده اعلانی برای مشخص کردن وظایف ML، (2) بهینه‌ساز برای انتخاب و انطباق پویای انتخاب الگوریتم‌های یادگیری (تبدیل وظیفه ML اعلانی به یک طرح یادگیری پیچیده)، (3) مجموعه‌ای از اپراتورهای سطح بالا برای فعال کردن محققان ML برای پیاده‌سازی یک طیف وسیعی از روش ML بدون دانش سیستم‌های عمیق، و (4) زمان اجرای بهینه‌سازی شده جدید برای الگوهای دسترسی به داده‌ها در عملیات سطح بالا [107].

هر چند به لحاظ نظری طراحی یک الگوریتم که دقت بدی دارد اما استفاده از آن و قابلیت اطمینان به سیستم راحت‌تر است، کمتر جالب توجه باشد اما در عمل [109] بسیار ارزشمند است. ازاین‌رو، توسعه‌ی ML قابل استفاده در داده‌های بزرگ موجب تسهیل آموزش داده‌ها (به‌عنوان مثال، تنظیم پارامتر، بهینه‌سازی گردش کار و آماده سازی داده) و اجرای گسترده ML در عمل می‌گردد.

یکی از راه‌های متعارف برای توضیح داده، نمودار، گراف و دیگر تکنیک‌های تجسمی است، زیرا انسان‌ها به آسانی می‌توانند بر اساس الگوها تصمیم‌گیری و مقایسه کنند. با این وجود، اگر حجم داده بالا برود، این روش با محدودیت‌هایی روبه‌رو خواهد شد [110].

به منظور اینکه ML داده‌های بزرگ پذیرش اجتماعی گسترده‌ای برای اعمال اثرات و مسائلی مانند حریم خصوصی داده‌ها، امنیت، مالکیت و مسئولیت دریافت کنند باید رفتارآنها نشان داده شود [111.112].

به‌طورخلاصه، معیارهای ارزیابی پذیرفته شده، نیروی محرکه اصلی برای توسعه‌ی الگوریتم جدید ML هستند و نیاز فوری به تعیین معیارهای ارزیابی جامع‌تر و فراتر از دقت معمولی و اقدامات مقیاس‌پذیر دارند. به‌عنوان مثال، ارزیابی جامع مبتنی بر قابلیت به توسعه الگوریتم‌های جدید ML که به طور همزمان در بازده، دقت، ثبات، استحکام و سهولت استفاده تعادل ایجاد می‌کنند منجر می‌شود.

6. پژوهش‌های آینده و نتیجه‌گیری

در این مقاله یک مرور کلی در مرود فرصت‌ها و چالش‌های ML بر روی داده‌های بزرگ ارائه شده است. داده‌های بزرگ با چالش‌های متعددی برای ایجاد ML سنتی از نظر مقیاس‌پذیری، سازگاری، قابلیت استفاده، و ارائه جدید فرصت‌ها برای الهام بخشیدن به راه‌حل‌های ML و بسیاری از چالش‌های فنی مرتبط رو به رو هستند. این فرصت‌ها و چالش‌ها به‌عنوان مواردی امیدوارکننده در تحقیقات آینده به شمار می‌روند. برخی از مسائل باز مربوط به پژوهش در ML داده‌های بزرگ با توجه به اجزای چارچوب MLBiD، همانطور که در جدول 2 نشان داده شده است برجسته شده است.

بیشترین کار موجود بر روی ML برای داده‌های بزرگ به حجم و سرعت متمرکز شده است، اما کارهای بسیاری وجود دارد که به دو جنبه باقی‌مانده از داده‌های بزرگ پرداخته است: صحت و ارزش. برای رسیدگی به صحت داده‌ها، راه‌حل امیدوارکننده این مقاله توسعه الگوریتمی است که قادر به دسترسی قابل اعتماد با اعتبارسنجی به داده‌ها یا اطلاعات منابع باشد به‌طوری‌که داده‌های غیر قابل اعتماد می‌توانند در طول پردازش اولیه داده فیلتر شوند؛ یکی دیگر از جهات برای توسعه ML مدل‌های جدیدی است که می‌توانند با داده‌های غیرقابل اعتماد و یا حتی تناقض استنتاج شوند. برای پی بردن به ارزش داده‌های بزرگ در پشتیبانی تصمیم، نیازداریم تا به درک کاربران از نتایج ML و منطق پشت هر سیستم تصمیم‌گیری کمک کنیم. بنابراین، توضیح ML یک منطقه تحقیقاتی مهم در آینده خواهد بود. علاوه براین،

برای حمایت از داده‌های بزرگ انسان در حلقه ML، نیاز به رسیدگی به سوالات اساسی پژوهش مانند چگونگی به دست آوردن موثر مقدار زیادی اطلاعات از طریق جمعیت هستیم. علاوه براین، برخی مسائل تحقیقاتی باز در این زمینه عبارتند از: (1) چگونه از حریم خصوصی داده‌ها در حال اجرای ML محافظت کنیم؛ (2) چگونه ML را بیشتر اعلان کنیم به‌طوری‌که برای افراد غیرمتخصص مشخص و تعامل با آن برایشان آسان‌تر باشد؛ (3) چگونه دامنه دانش کلی را با ML ترکیب کنیم و (4) نحوه طراحی داده‌های بزرگ معماری جدید ML برای پشتیبانی یکپارچه براساس تجزیه و تحلیل زمان واقعی از مقدار زیادی از داده‌های ناهمگن که ممکن است قابل اعتماد نباشد.

به طور خلاصه، ML برای مرتفع کردن چالش‌های ناشی از داده‌های بزرگ و کشف الگوهای پنهان، دانش و بینش از داده‌های بزرگ به‌منظور ایجاد ارزش واقعی کسب و کار ضروری است. همراهی ML و داده‌های بزرگ به آینده‌ای موفق در یک مرز جدید اشاره دارد.

References

[1] M.I. Jordan, T.M. Mitchell, Machine learning: trends, perspectives, and prospects, Science 349 (2015) 255–260.

[2] C.-W. Tsai, C.-F. Lai, H.-C. Chao, A.V. Vasilakos, Big data analytics: a survey, J. Big Data 2 (2015) 1–32.

[3] M.M. Najafabadi, F. Villanustre, T.M. Khoshgoftaar, N. Seliya, R. Wald, E. Muharemagic, Deep learning applications and challenges in big data analytics, J. Big Data 2 (2015) 1–21.

[4] N. Japkowicz, M. Shah, Evaluating Learning Algorithms: a Classification Perspective, Cambridge University Press, New York, NY, USA, 2011.

[5] S. Russell, P. Norvig, Artificial Intelligence: A Modern Approach, 3rd ed., Prentice Hall, Upper Saddle River, New Jersey, USA, 2010.

[6] Y. Bengio, A. Courville, P. Vincent, Representation learning: a review and new perspectives, IEEE Trans. on Pattern Anal. Mach. Intell., Trans. 35 (2013) 1798–1828.

[7] O. Dekel, From Online to Batch Learning with Cutoff-AveragingNIPS (2008), 2008, pp. 377–384.

[8] S. Amershi, M. Cakmak, W.B. Knox, T. Kulesza, Power to the people: the role of humans in Interactive machine learning, AI Mag. 35 (2014) 105–120.

[9] V. Mirchevska, M. Luštrek, M. Gams, Combining domain knowledge and machine learning for robust fall detection, Expert Syst. 31 (2014) 163–175.

[10] T. Yu, Incorporating Prior Domain Knowledge into Inductive Machine LearningComputing Sciences, University of Technology Sydney, Sydney, Augtralia, 2007.

[11] Q. Chen, J. Zobel, K. Verspoor, Evaluation of a machine learning duplicate detection method for bioinformatics Databases, Proc. ACM Ninth Int. Workshop Data Text. Min. Biomed. Inform. (2015) 4–12.

[12] T. Rakthanmanon, B. Campana, A. Mueen, G. Batista, B. Westover, Q. Zhu, et al., Addressing Big data time series: mining Trillions of time series subsequences Under dynamic time Warping, ACM Trans. Knowl. Discov. Data 7 (2013) 10.

[13] J.J.Pfeiffer , III, J.Neville, P.N.Bennett, Overcoming relational learning biases to accurately predict preferences in large scale networks, in: Proceedings of the 24th International Conference on World Wide Web, 2015, pp. 853–863.

[14] L.Cao, M.Wei, D.Yang, E.A.Rundensteiner, Online outlier exploration over large datasets, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2015, pp. 89–98.

[15] A. Gandomi, M. Haider, Beyond the hype: Big data concepts, methods, and analytics, Int. J. Inf. Manag. 35 (2015) 137–144.

[16] X.Cai, F.Nie, H.Huang, Multi-view K-means clustering on big data, in: Proceedings of the Twenty-Third international joint conference on Artificial Intelligence, 2013, pp. 2598–2604.

Conference on in Distributed Computing Systems Workshops (ICDCSW).

[31] A.T. Azar, A.E. Hassanien, Dimensionality reduction of medical big data using neural-fuzzy classifier (04/01/2015)Soft Comput. - A Fusion Found., Methodol. Appl. 19 (2015) 1115–1127.

[32] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, Stacked denoising Autoencoders: learning useful representations in a deep network with a local denoising criterion, J. Mach. Learn. Res. 11 (2010) 3371–3408.

[33] C.-Y. Liou, W.-C. Cheng, J.-W. Liou, D.-R. Liou, Autoencoder for words, Neurocomputing 139 (2014) 84–96.

[34] R. Collobert, F. Sinz, J. Weston, L. Bottou, Trading convexity for scalability, Proc. 23rd Int. Conf. Mach. Learn. (2006) 201–208.

[35] Y. Bengio, Y. LeCun, Scaling learning algorithms towards, AI (ed), in: L. Bottou, O. Chapelle, D. DeCoste, J. Weston (Eds.), Large Scale Kernel Machines, MIT Press, Cambridge, MA, 2007.

[36] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, et al., "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems," CoRR, 2016.

[37] Y. You, H. Fu, S.L. Song, A. Randles, D. Kerbyson, A. Marquez, et al., Scaling support vector machines on modern HPC platforms, J. Parallel Distrib. Comput. 76 (2015) 16–31.

[38] B. Panda, J.S. Herbach, S. Basu, R.J. Bayardo, PLANET: massively parallel learning of tree ensembles with MapReduce, Proc. VLDB Endow. 2 (2009) 1426–1437.

[39] E. Xing, Q. Ho, W. Dai, J.-K. Kim, J. Wei, S. Lee, et al., Petuum: a new platform for distributed machine learning on Big data, IEEE Trans. Big Data (2015) 49–67.

[40] R. Collobert, K. Kavukcuoglu, and C. Farabet, Torch7: A Matlab-like Environment for Machine Learning, in: Proceedings of the Neural Information Processing Systems (NIPS) Workshop on BigLearn, 2011.

[41] T.Yang, Q.Lin, R.Jin, Big data analytics: Optimization and randomization, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2015, pp. 2327–2327.

[42] W. Xu, Towards Optimal one pass large scale learning with averaged stochastic gradient descent, 2011. Available at: arXiv:1107.2490.

[43] L. Bottou, Large-Scale Machine Learning with Stochastic Gradient Descent, in: Proceedings of COMPSTAT, 2010, pp. 177–186.

[44] J. Wang, Y. Tang, M. Nguyen, I. Altintas, A Scalable data Science workflow approach for Big data Bayesian network learning, Proc. 2014 IEEE/ACM Int. Symp. Big Data Comput. (2014) 16–25.

[45] K. Yue, H. Wu, X. Fu, J. Xu, Z. Yin, W. Liu, A data-intensive approach for discovering user similarities in social behavioral interactions based on the bayesian network, Neurocomputing 219 (2017) 364–375.

[46] A. Kumar, A. Beutel, Q. Ho, E.P. Xing, Fugue: Slow-Worker-Agnostic Distributed Learning for Big Models on Big Data, in: Proceedings of the 17th International Conference on Artificial Intelligence and Statistics (AISTATS), Reykjavik, Iceland, 2014, pp. 531–539.

[17] S. Ramírez-Gallego, S. García, H. Mouriño-Talín, D. Martínez-Rego, V. Bolón-Canedo, A. Alonso-Betanzos, et al., "Data discretization: taxonomy and big data challenge," Wiley Interdisciplinary Reviews, Data Mining and Knowledge Discovery, vol. 6, pp. 5-21, 2016.
[18] Y.Z.Y.-M.Cheung, Discretizing Numerical Attributes in Decision Tree for Big Data Analysis, in: Proceedings of the 2014 IEEE International Conference on Data Mining Workshop (ICDMW), 2014.
[19] L.-V. Nguyen-Dinh, M. Rossi, U. Blanke, G. Tröster, Combining crowd-generated media and personal data: semi-supervised learning for context recognition, Proc. 1st ACM Int. Workshop Pers. data meets Distrib. Multimed. (2013) 35–38.
[20] B.M. Lake, R. Salakhutdinov, J.B. Tenenbaum, Human-level concept learning through probabilistic program induction, Science 350 (2015) 1332–1338.
[21] G. Zhang, S.-X. Ou, Y.-H. Huang, C.-R. Wang, Semi-supervised learning methods for large scale healthcare data analysis, Int. J. Comput. Healthc. 2 (2015) 98–110.
[22] J. Suzuki, H. Isozaki, and M. Nagata, Learning condensed feature representations from large unsupervised data sets for supervised learning, in: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, Human Language Technologies, short papers, 2, 2011, pp. 636–641.
[23] B. Mozafari, P. Sarkar, M. Franklin, M. Jordan, S. Madden, Scaling up crowd-sourcing to very large datasets: a case for active learning, Proc. VLDB Endow. 8 (2014) 125–136.
[24] Y. Su, G. Agrawal, J. Woodring, K. Myers, J. Wendelberger, J. Ahrens, Effective and efficient data sampling using bitmap indices, Clust. Comput. 17 (2014) 1081–1100.
[25] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, Distributed feature selection, Appl. Soft Comput. 30 (2015) 136–150.
[26] S. Sun, Jing Zhao, J. Zhu, A review of Nyström methods for large-scale machine learning, Inf. Fusion 26 (2015) 36–48.
[27] M. Tan, I.W. Tsang, L. Wang, Towards ultrahigh dimensional feature selection for big data, J. Mach. Learn. Res. 15 (2014) 1371–1429.
[28] Z.Zhao, H.Liu, Spectral feature selection for supervised and unsupervised learning, in: Proceedings of the 24th international conference on Machine learning, 2007, pp. 1151–1157.
[29] J. Cervantes, X. Li, W. Yu, Support vector machine classification based on fuzzy clustering for large data sets, in: Proceedings of the 5th MICAI, 2015, pp. 572–582.
[30] O. Y. S. Al-Jarrah, A., M. Elsalamouny, P. D. Yoo, S. Muhaidat, and K. Kim, Machine-Learning-Based Feature Selection Techniques for Large-Scale Network Intrusion Detection, in: Proceedings of the 2014 IEEE 34th International

solution for prototype reduction in big data classification, Neurocomputing 150 (2015) 331–345 Part A.
[63] S. Landset, T.M. Khoshgoftaar, A.N. Richter, T. Hasanin, A survey of open source tools for machine learning with big data in the Hadoop ecosystem, J. Big Data 2 (2015) 1–36.
[64] R.Gemulla, E.Nijkamp, P.J.Haas, Y.Sismanis, Large-scale matrix factorization with distributed stochastic gradient descent, in: Proceedings of the 17th ACM SIGKDD international conference ion Knowledge discovery and data mining, San Diego, California, USA, 2011, pp. 69–77.
[65] D. Hsu, N. Karampatziakis, J. Langford, A.J. Smola, Parallel online learningScaling up machine learning: Parallel and distributed approaches, Cambridge University Press, 2011.
[66] P.Domingos, G.Hulten, A General Method for Scaling Up Machine Learning Algorithms and its Application to Clustering, presented at Proceedings of the Eighteenth International Conference on Machine Learning, 2001, pp. 106–113.
[67] R. Bekkerman, M. Bilenko, J. Langford (Eds.), Scaling up Machine Learning: Parallel and Distributed Approaches, Cambridge University Press, New York, 2012.
[68] C. Parker, Unexpected challenges in large scale machine learning, Proc. 1st Int. Workshop Big Data, Streams Heterog. Source Min.: Algorithms, Syst., Program. Models Appl. (2012) 1–6.
[69] D. Peteiro-Barral, B. Guijarro-Berdiñas, A survey of methods for distributed machine learning, Prog. Artif. Intell. 2 (2013) 1–11.
[70] K.L.C.Zhu, M.Savvides, Distributed class dependent feature analysis — A big data approach, in: proceedings of the 2014 IEEE International Conference on Big Data, 2014.
[71] M. Yui, I. Kojima, A database-Hadoop hybrid approach to Scalable machine learning, IEEE Int. Congr. Big Data (BigData Congr.) (2013) 1–8.
[72] F.Ö. Çatak, Classification with boosting of extreme learning machine over arbitrarily partitioned data, Soft Comput. (2015) 1–13.
[73] M. Hefeeda, F. Gao, and W. Abd-Almageed, Distributed approximate spectral clustering for large-scale datasets, in: Proceedings of the 21st international symposium on High-Performance Parallel and Distributed Computing, 2012, pp. 223–234.
[74] G. Cavallaro, M. Riedel, M. Richerzhagen, J.A. Benediktsson, A. Plaza, On Understanding Big data impacts in remotely sensed image classification using support vector machine methods, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 8 (2015) 4634–4646.
[75] J.Zhu, J.Chen, W.Hu, Big Learning with Bayesian Methods. Available: ⟨http://arxiv.org/pdf/1411.6370⟩, 2014.
[48] S. Owen, R. Anil, T. Dunning, E. Friedman, Mahout in Action, Manning Publications Co., 2011.
[49] C.T. Chu, S.K. Kim, Y.A. Lin, Y. Yu, G.R. Bradski, A.Y. Ng, et al., Map-reduce for machine learning on multicore, *NIPS* (2006) 281–288.
[50] A.K.Ghoting, R.E.Pednault, B.Reinwald, V.Sindhwani, S.Tatikonda, Y.Tian, et al., SystemML: Declarative machine learning on MapReduce, in: Proceedings of the 27th International Conference on Data Engineering (ICDE), 2011.
[51] V.R. Borkar, Y. Bu, M.J. Carey, J. Rosen, N. Polyzotis, T. Condie, et al., Declarative systems for large-scale machine learning, *IEEE Data Eng. Bull.* 35 (2012) 24–32.
[52] Y. Low, D. Bickson, J. Gonzalez, C. Guestrin, A. Kyrola, J.M. Hellerstein, Distributed GraphLab: a framework for machine learning and data mining in the cloud, Proc. VLDB Endow. 5 (2012) 716–727.
[53] Theano Development Team, Theano: A Python framework for fast computation of mathematical expression. Available: arXiv:1605.02688.
[54] Y.Jia, E.Shelhamer, J.Donahue, S.Karayev, J.Long, R.Girshick, et al., Caffe: Convolutional Architecture for Fast Feature Embedding, in: Proceedings of the 22nd ACM international conference on Multimedia, Orlando, Florida, USA, 2014.
[55] J.-x. Dong, A. Krzyzak, C.Y. Suen, Fast SVM training algorithm with decomposition on very large data sets, *IEEE* Trans. Pattern Anal. Mach. Intell. 27 (2005) 603–618.
[56] J. Dean, G. S. Corrado, R. Monga, K. Chen, M. Devin, Q. V. Le, et al., Large scale distributed deep networks, in: Proceedings of the Neural Information Processing Systems, Lake Tahoe, Nevada, United States, 2012, pp. 1232–1240.
[57] J.E. Mason, I. Traoré, I. Woungang, Machine Learning Techniques for Gait Biometric Recognition: Using the Ground Reaction Force, Springer, Switzerland, 2016.
[58] Q.V.Le, J.Ngiam, A.Coates, A.Lahiri, B.Prochnow, A.Y.Ng, On optimization methods for deep learning, in: Proceedings of the 28th International Conference on Machine Learning, Bellevue, WA, USA, 2011.
[59] Y. Ganjisaffar, T. Debeauvais, S. Javanmardi, R. Caruana, C.V. Lopes, Distributed tuning of machine learning algorithms using MapReduce Clusters, Proc. Third Workshop Large Scale Data Min.: Theory Appl. (2011) 2.
[60] C.Dijun Luo, Ding, H.Huang, Parallelization with multiplicative algorithms for big data mining, in: Proceedings of the 12th International Conference on Data Mining (ICDM), 2012, pp. 489–498.
[61] J.S.Yoo, D.Boulware, D.Kimmey, A Parallel Spatial Co-location Mining Algorithm Based on MapReduce, in: proceedings of the 2014 IEEE International Congress on Big Data, 3rd, pp. 25–31.
[62] I. Triguero, D. Peralta, J. Bacardit, S. García, F. Herrera, MRPR: A MapReduce

[93] E.Bortnikov, A.Frank, E.Hillel, S.Rao, Predicting execution bottlenecks in map-reduce clusters, in: Proceedings of the 4th USENIX conference on Hot Topics in Cloud Ccomputing, 2012, pp. 18–18.
[94] K. Xu, H. Yue, L. Guo, Y. Guo, Y. Fang, Privacy-preserving machine learning algorithms for big data systems, in: Proceedings of the 2015 IEEE 35th International Conference on Distributed Computing Systems (ICDCS), 2015, pp. 318–327.
[95] J. Vaidya, H. Yu, X. Jiang, Privacy-preserving SVM classification, Knowledge Inf. Syst. 14 (2008) 161–178.
[96] A.D. Popescu, A. Balmin, V. Ercegovac, A. Ailamaki, PREDIcT: towards predicting the runtime of large scale iterative analytics, Proc. VLDB Endow. 6 (2013) 1678–1689.
[97] L. Breiman, Pasting small votes for classification in large databases and On-Line, Machine Learn. 36 (1999) 85–103.
[98] H. Kashyap, H.A. Ahmed, N. Hoque, S. Roy, D.K. Bhattacharyya, Big Data Anal. Bioinforma.: A Mach. Learn. Perspect. (2015).
[99] J.Xu, C.Tekin, M.van der Schaar, Learning optimal classifier chains for real-time big data mining, in Proceedings 51st Annu. Allerton Conference Comm., Control and Comput. (Allerton'13), 2013.
[100] G.De Francisci Morales, SAMOA: a platform for mining big data streams, in: Proceedings of the 22nd International Conference on World Wide Web, 2013, pp. 777–778.
[101] Q.Yang, Big data, lifelong machine learning and transfer learning, in: Proceedings of the sixth ACM international conference on Web search and data mining, 2013, pp. 505–506.
[102] J. Lu, S.C. Hoi, J. Wang, P. Zhao, Z.-Y. Liu, Large scale online kernel learning, J. Mach. Learn. Res. 17 (2016) 1–43.
[103] Z. Wang, K. Crammer, S. Vucetic, Breaking the curse of kernelization: budgeted stochastic gradient descent for large-scale SVM training, *The J. Mach. Learn. Res.* 13 (2012) 3103–3131.
[104] Y. Zhai, Y.S. Ong, I.W. Tsang, The emerging big dimensionality, IEEE Comput. Intell. Mag. 9 (2014) 14–26.
[105] T.Xiao, J.Zhang, K.Yang, Y.Peng, Z.Zhang, Error-Driven Incremental Learning in Deep Convolutional Neural Network for Large-Scale Image Classification, in: Proceedings of the ACM International Conference on Multimedia, 2014, pp. 177–186.
[106] D. Singh, C.K. Reddy, A survey on platforms for big data analytics, J. Big Data 2 (2014) 1–20.
[107] T.Kraska, A.Talwalkar, J.Duchi, R.Griffith, M.J.Franklin, M.I.Jordan, MLbase: A Distributed Machine-learning System, in: Proceedings of the 6th Biennial

[76] L.Bagheri, H.Goote, A.Hasan, G.Hazard, Risk adjustment of patient expenditures: A big data analytics approach, in Proceedings of the 2013 IEEE International Conference on Big Data, 2013.
[77] A. Krizhevsky, I. Sutskever, G. Hinton, Imagen. Classif. Deep convolutional Neural Netw. (2012).
[78] Y. LeCun, K. Kavukcuoglu, and C. Farabet, Convolutional networks and applications in vision, in: Proceedings of IEEE International Symposium on Circuits and Systems, 2010, pp. 253–256.
[79] J. Deng, K. Li, M. Do, H. Su, L. Fei-Fei, Construction and analysis of a large scale image ontology, Vis. Sci. Soc. 1 (2009).
[80] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, M.S. Lew, Deep learning for visual understanding: a review, Neurocomputing 187 (2016) 27–48.
[81] X. Jiang, Y. Pang, X. Li, J. Pan, Speed up deep neural network based pedestrian detection by sharing features across multi-scale models, Neurocomputing 185 (2016) 163–170.
[82] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. Manning, A. Ng, et al., Recursive deep models for semantic compositionality over a sentiment treebank, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 2013.
[83] S. Zhou, Q. Chen, X. Wang, Active deep learning method for semi-supervised sentiment classification, Neurocomputing 120 (2013) 536–546.
[84] N. Zeng, Z. Wang, H. Zhang, W. Liu, F.E. Alsaadi, Deep belief networks for quantitative analysis of a gold immunochromatographic strip, Cogn. Comput. 8 (2016) 684–692.
[85] R.Raina, A.Battle, H.Lee, B.Packer, A.Y.Ng, Self-taught learning: transfer learning from unlabeled data, in: Proceedings of the 24th international conference on Machine learning, Corvalis, Oregon, USA, 2007.
[86] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016.
[87] D. Erhan, Y. Bengio, A. Courville, P.-A. Manzagol, P. Vincent, S. Bengio, Why does Unsupervised Pre-training help deep learning?, *The* J. Mach. Learn. Res. 11 (2010) 625–660.
[88] T.Mikolov, I.Sutskever, K.Chen, G.S.Corrado, J.Dean, Distributed Representations of Words and Phrases and their Compositionality, presented at the NIPS, Stateline, NV, 2013.
[89] X.-w. Chen, X. Lin, Big data deep learning: challenges and perspectives, Access, IEEE 2 (2014) 514–525.
[90] Y. Chen, T. Luo, S. Liu, S. Zhang, L. He, J. Wang, et al., DaDianNao: a machine-learning Supercomputer, 47th Annu. IEEE/ACM Int. Symp. Micro. (2014) 609–622.
[91] D. Mahajan, J. Park, E. Amaro, H. Sharma, A. Yazdanbakhsh, J.K. Kim, et al., TABLA: a unified template-based framework for accelerating statistical machine learning, I*EEE* Int. Symp. High. Perform. Comput. Archit. *(HPCA)* (2016) 14–26.
[92] M.Zaharia, M.Chowdhury, M.J.Franklin, S.Shenker, I.Stoica, Spark: cluster computing with working sets, presented at in: Proceedings of the 2nd USENIX conference on Hot topics in Cloud Computing, Boston, MA, 2010.
Conference on Innovative Data Systems Research, Asilomar, California, USA, 2013.
[108] V. Markl, Breaking the chains: on declarative data analysis and data independence in the big data era, Proc. VLDB Endow. 7 (2014) 1730–1733.
[109] S. Tong, Lessons learned developing a practical large scale machine learning system 2016, Google Research Blog, 2010.
[110] T.R. Armes, M, Using Big data and predictive machine learning in aerospace test environments, IEEE Autotestcon (2013).
[111] B.Thuraisingham, Big Data Security and Privacy, in: Proceedings of the 5th ACM Conference on Data and Application Security and Privacy, San Antonio, Texas, USA, 2015.
[112] B.Nelson, T.Olovsson, Security and Privacy for Big Data: A Systematic Literature Review, in: Proceedings of the 2016 IEEE International Conference on Big Data, Washington, D.C, 2016, pp. 3693–3702.