بخش عملی:

تمرین اول) در این تمرین، شما وظیفه دارید یک سیستم پردازش داده‌های جریانی (Streaming) با استفاده از Structured Streaming در PySpark پیاده‌سازی کنید. هدف این است که داده‌های ورودی که به صورت پیوسته و زنده (Real-time) از یک منبع خبری وارد سیستم می‌شوند، پردازش شده و اطلاعات مفیدی از آن‌ها استخراج گردد. این داده‌ها به‌صورت JSON هستند (news_dataset_MDA2024.json) و شامل اخبار در دسته‌بندی‌های مختلف می‌باشند. هر خبر در این دیتاست دارای اطلاعاتی از قبیل عنوان خبر، موضوع خبر، توضیحات کوتاه متن خبر، زمان ارسال خبر و... می‌باشد.

۱) در این قسمت از تمرین، هدف محاسبه و نمایش تعداد اخبار در هر دسته‌بندی (category) در بازه‌های زمانی ۲۰ ثانیه‌ای است. به‌طور خاص، شما باید تعداد اخبار ورودی را در هر بازه ۲۰ ثانیه‌ای محاسبه کنید و این نتایج را به‌صورت زنده در کنسول نمایش دهید.

۲) در بازه‌های زمانی ۳۰ ثانیه‌ای، طول عنوان اخبار(headline) را بررسی کرده و ۳ خبر با طولانی‌ترین تیتر خبری را نمایش دهید.

۳) در این بخش از تمرین، هدف این است که جریان داده‌ها را بر اساس موضوعات مشخصی فیلتر کنید، به‌طوری‌که فقط داده‌هایی که به موضوعات BUSINESS، ENTERTAINMENT و POLITICS مرتبط هستند، پردازش شوند. این کار یکی از مراحل رایج در کار با داده‌ها به صورت زنده است که به کاهش داده‌های غیرضروری و تمرکز بر اطلاعات مهم کمک می‌کند. پس از پیاده‌سازی فیلتر مربوطه، خروجی را در دو حالت قسمت‌های ۱ و ۲ نمایش دهید.

تمرین دوم) در این تمرین می‌خواهیم با الگوریتم‌های مهم در تحلیل داده‌های استریم آشنا شده و اقدام به پیاده‌سازی این الگوریتم‌ها کنیم. دیتاست در نظر گرفته شده، web_streaming_dataset.csv می‌باشد که حاوی اطلاعات ورود کاربران و بازدید از صفحات یک وبسایت است. از readStrem در PySpark برای شبیه‌سازی جریان داده از دیتاست مورد نظر استفاده کنید. ستون‌های این دیتاست عبارت‌اند از:

- UserID: شناسه عضویت هر کاربر است.
- RequestType: نوع وضعیت درخواست هر کاربر که در صورت موفقیت مقدار ۱ و در صورت عدم موفقیت مقدار ۰ خواهد داشت.
- VisitCount: تعداد دفعات مشاهده صفحات وبسایت توسط هر کاربر.

شما باید داده‌ها را به گونه‌ای پردازش کنید که بتوانید مقادیر مورد نظر را به صورت تقریبی اما با دقت بالا محاسبه کنید.

۱) پیاده‌سازی الگوریتم DGIM:

الگوریتم DGIM (Datar-Gionis-Indyk-Motwani)، یک الگوریتم کارآمد برای پردازش داده‌های استریم است که به‌ویژه برای تخمین تعداد ۱ها (یا ۰ها) در یک پنجره‌ی زمانی محدود طراحی شده‌است. این الگوریتم با استفاده از روشی به نام باکت‌های نمایی (Exponential Buckets)، فضای ذخیره‌سازی را به‌شدت کاهش می‌دهد و امکان پردازش سریع داده‌های حجیم را فراهم می‌کند. مراحل زیر را انجام دهید:

- داده‌های ستون RequestType را به صورت جریان (Stream) بخوانید.
- جریان داده را به پنجره‌های زمانی ۵۰۰ تایی تقسیم کنید.
- تعداد بیت‌های ۱ (درخواست‌های موفق کاربران) را در هر پنجره با استفاده از الگوریتم DGIM تخمین بزنید.
- نموداری رسم کنید که تعداد واقعی و تخمین زده شده درخواست‌های موفق را برای هر پنجره نشان دهد.
- حداقل ۱۰ پنجره از داده‌ها را پردازش کنید.

۲) پیاده‌سازی الگوریتم FM:

الگوریتم FM (Flajolet Martin) بر پایه استفاده از توابع هش و تحلیل موقعیت بیشترین صفرهای سمت راست در نمایش باینری خروجی از توابع هش عمل می‌کند. حال با استفاده از این الگوریتم می‌خواهیم تعداد کاربران یکتا که به وبسایت سرزده‌اند را تخمین بزنیم. مراحل زیر را انجام دهید:

- داده‌های ستون UserID را به صورت جریان (Stream) بخوانید.
- به منظور فراخوانی توابع هش از کتابخانه hashlib استفاده کنید.
- از توابع آماده sha1، sha256، md5 و sha224 استفاده کنید.
- از تابع bin بمنظور تبدیل مقادیر خروجی از هش به باینری استفاده کنید.
- الگوریتم FM را پیاده سازی و بر روی جریان داده اجرا کنید.
- تعداد کاربران یکتا واقعی و تخمینی توسط الگوریتم خود را گزارش کنید.