دانشکده مهندسی برق

تحلیل داده‌های حجیم

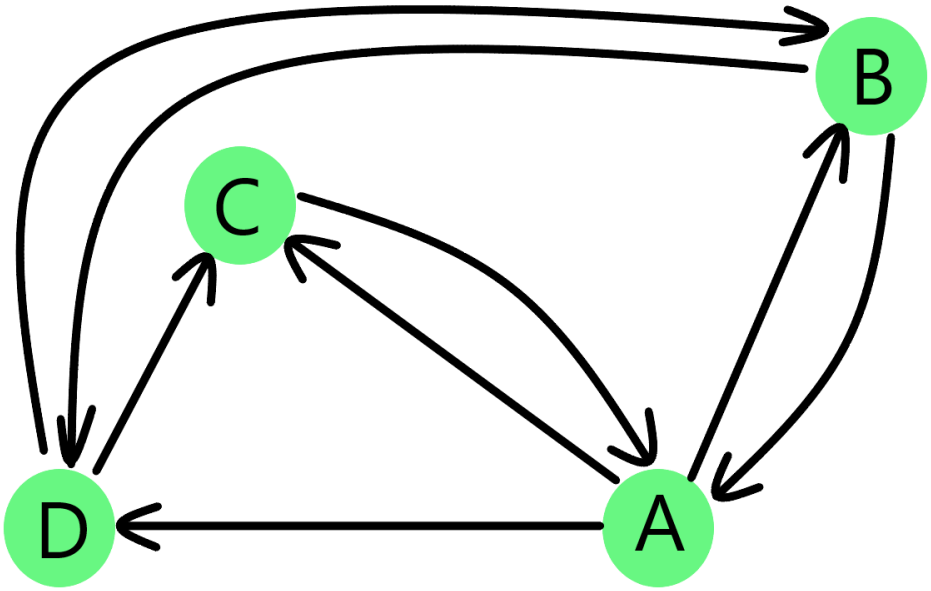
تمرین سری چهارم

استاد: دکتر ایمان غلامپور

دستیار آموزشی: بهنام رئوفی

بخش تئوری:

سوال اول) شکل زیر نمونه‌ای از گراف صفحات یک وبسایت کوچک را نشان می‌دهد که تنها شامل چهار صفحه می‌باشد. به طوری که صفحه A به هر سه صفحه B، C و D لینک دارد. صفحه B تنها به صفحات A و D لینک دارد. لینک‌های صفحات C و D نیز به همین ترتیب در شکل مشخص هستند.

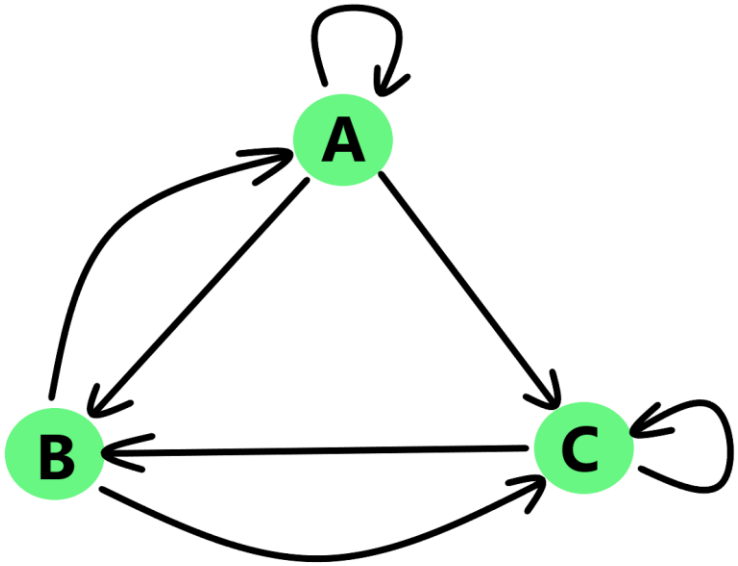


الف) در ابتدا ماتریس مجاورت و ماتریس احتمال انتقال این گراف را بنویسید.

ب) حال در نظر بگیرید که در این گراف تنها صفحه B، یک صفحه معتبر (Trusted Page) باشد. TrustRank و Spam Mass هر صفحه را محاسبه نمایید.

ج) الگوریتم Hubs and Authorities یکی از الگوریتم‌های تحلیل گراف است که برای رتبه‌بندی صفحات وب یا گراف‌ها استفاده می‌شود. مقادیر امتیاز hub و همچنین اعتبار هر node از گراف شکل بالا را محاسبه کنید.

سوال دوم) گراف زیر را که دارای سه node می‌باشد در نظر بگیرید. PageRank هر node از گراف را با در نظر گرفتن $\beta = 0.8$ محاسبه نمایید.



سوال سوم) یک Stream از تاپل‌ها به فرم زیر داریم:

Grades(university, courseId, studentId, grade)

فرض کنید دانشگاه‌ها یکتا هستند اما در هر دانشگاه صرفا courseId و studentId یکتاست. (یعنی دانشگاه‌های مختلف ممکن است courseId های مشابه برای دروس مختلف داشته باشند). حال در نظر بگیرید که می‌خواهیم سوالاتی را بر اساس یک sample شامل تنها به $\frac{1}{20}$ داده‌های اصلی پاسخ دهیم. برای هریک از سوالات زیر sample مربوطه را چگونه می‌سازید؟ (یعنی بگویید که کلید هر آیتم چیست؟)

الف) برای هر دانشگاه، تعداد متوسط دانشجویان را تخمین بزنید.

ب) درصد دانشجویانی که GPA آن‌ها برابر 3.5 یا بیشتر است را تخمین بزنید.

ج) درصد دروسی که در آن‌ها حداقل نصف دانشجویان نمره A گرفته‌اند را تخمین بزنید.

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.

تمرین سوم) هدف از این تمرین آشنایی با الگوریتم PageRank و پیاده‌سازی آن به کمک PySpark و RDD است. شما یک گراف دارید و باید مقدار PageRank هر گره را محاسبه کنید. مقادیر این گراف در قالب یک فایل متنی (Graph_Links.txt) به شما داده شده است. هر خط از فایل شامل شماره یک گره و لیستی از گره‌هایی است که از گره به آن‌ها لینک داده شده‌است.

۱) داده‌های موجود در فایل متنی را بارگذاری کرده و گراف مربوطه را رسم کنید.

۲) مشخص کنید کدام گره‌ها بصورت Dead End هستند.

۳) الگوریتم PageRank را برای گراف مربوطه پیاده‌سازی و اجرا کرده و مقادیر PageRank تمامی گره‌ها را ذخیره کنید. ($\beta =$ 0.8 و معیار همگرایی کمتر از 0.001 فرض شود.)

۴) الگوریتم HITS را پیاده‌سازی کنید. برای نرمال‌سازی، بیشترین امتیاز هرکدام از بردارهای Hubs و Authorities را برابر با یک قرار دهید. همچنین بروزرسانی‌ها را انقدر تکرار کنید تا به همگرایی برسید. حال مقادیر Hubs و Authorities تمامی گره‌ها را ذخیره کنید.

نکات تکمیلی:

۱. دیتاست‌های بخش عملی درون فولدر Datasets قرار دارند.

۲. مقادیر خروجی PageRank و HITS در تمرین عملی سوم را در یک دیتافریم با چهار ستون Node ، PageRank ، Hubs و Authorities ذخیره و در قالب یک فایل CSV به همراه پاسخ تمرین ارسال نمایید.(بخشی از نمره این تمرین به نتایج این بخش اختصاص دارد.)

۳. گزارش هر دو بخش تئوری و عملی را درون یک فایل pdf نوشته و به همراه فایل‌های قسمت عملی درون یک فایل فشرده قرار دهید. گزارش تحویلی مربوط به قسمت برنامه‌نویسی می‌بایست شامل توضیحات نحوه عملکرد برنامه، اطلاعات در مورد قسمت‌های مختلف برنامه و خروجی آنها باشد.