بسمه تعالی

مقایسه روش‌های مختلف استفاده از یادگیری نظارت شده و انتشار برچسب برای طبقه‌بندی ژن‌های مبتنی بر شبکه

سهند نوعی کردکندی – ۹۹۲۳۰۸۷

پروژه درس مبانی بیوانفورماتیک – دکتر فاطمه زارع میرک آبادی

چکیده:

اختصاص دادن ژن‌های انسانی به عملکردها، بیماری‌ها و صفات خاص، یکی از چالش‌های بزرگ در ژنتیک مدرن است. روش‌های محاسباتی، مانند یادگیری نظارت‌شده و انتشار برچسب، می‌توانند از شبکه‌های تعامل مولکولی برای پیش‌بینی ویژگی‌های ژن بهره‌مند شوند. در این مطالعه، تغییرات و بهبودهایی در فرآیند یادگیری نظارت‌شده برای طبقه‌بندی ژن مبتنی بر شبکه ارائه شده است. به‌طور خاص، برای پیاده‌سازی node2vec به جای روش‌های پیشین از PecanPy استفاده شد که منجر به افزایش چشمگیر سرعت تولید بردارهای تعبیه و کاهش حجم حافظه مورد نیاز گردید. همچنین، علاوه بر استفاده از Logistic Regression، مدل‌های SVM و Random Forest نیز در یادگیری نظارت‌شده به کار گرفته شدند و عملکرد آن‌ها در مقایسه با یکدیگر بررسی شد.

نتایج نشان می‌دهند که یادگیری نظارت‌شده با استفاده از اتصال کامل شبکه یک ژن همچنان عملکرد بالایی دارد و دقت پیش‌بینی بالایی را از طریق استخراج مؤثر ویژگی‌های محلی شبکه ارائه می‌دهد. استفاده از PecanPy نشان داد که این روش می‌تواند به‌طور قابل توجهی کارایی محاسباتی را بهبود بخشد. همچنین، مقایسه مدل‌های یادگیری نظارت‌شده نشان داد که مدل‌های SVM و Random Forest در برخی از وظایف پیش‌بینی عملکرد رقابتی داشته و می‌توانند به عنوان جایگزین‌های مؤثر در طبقه‌بندی ژن مورد استفاده قرار گیرند. این نتایج اهمیت استفاده از روش‌های نوین و بهینه‌سازی فرآیندهای یادگیری نظارت‌شده را در طبقه‌بندی ژن مبتنی بر شبکه تأیید می‌کنند.

۱ مقدمه و بیان مسئله

از مجموعه‌های geneset معتبر و در دسترس عموم که مربوط به عملکرد ژن های انسانی، مجموعه ژن های مرتبط با بیماری ها و صفات ایجاد شده در نتیجه ژن ها هستند استفاده کردیم. این کالکشن ها دارای برچسب True (مرتبط)و False (نامرتبط) با بیماری، ویژگی یا عملکرد هستند.

از networkهای معتبر و در دسترس عموم برای تشکیل گراف ارتباط بین ژن ها استفاده کردیم و ژن ها را به Entrez ID استخراج شده از NCBI نگاشت کردیم. ماتریس مجاورت(به اختصار A) این ژن ها را تشکیل میدهیم، از روی ماتریس مجاورت، ماتریس influence (به اختصار I) را تشکیل میدهیم. سپس با استفاده از روش node2vec اشاره شده در مقاله PecanPy [۱] بردار embedding هر کدام از ژن های اشاره شده در geneset collection ها را تشکیل میدهیم. (به اختصار E) گراف ها همگی بدون جهت هستند و برخی وزن دا و برخی بدون وزن هستند. از گراف حاصل و با استفاده از روش انتشار برچسب ارتباط ژن ها با بیماری ها را ایجاد میکنیم و سپس با داده های تست عملکرد آن را بررسی میکنیم. همچنین با استفاده از یادگیری نظارت شده و با هر سه ماتریش گفته به روش های مختلف مدل را تشکیل میدهیم و با داده های تست عملکرد مدل را بررسی میکنیم. همچنین از مقادیر متفاوت alpha در روش انتشار برچسب برای تشکیل گراف و روش node2vec استفاده می‌کنیم.

برای Validation مدل از روش های مختلف استفاده می‌کنیم. برای امکان پذیری استفاده از مدل برای پیش بینی های آینده از چالش ترجیحی در CAFA استفاده کردیم. همچنین از ژن های بیشتر شناخته شده برای آموزش مدل استفاده کردیم و بقیه ژن ها را برای تست گذاشتیم. همچنین از روش سنتی 5 Fold Cross Validation هم استفاده کردیم.

از امتیازهای مختلف برای تشخیص و مقایسه توانایی مدل در یاداوری مقادیر مثبت واقعی، دقت مدل، توانایی مدل در تفکیک پذیری نمونه های مثبت و منفی و چقدر مدل دقت در تشخیص و یاداوری مهم ترین عوامل ژنتیکی موثر در هر geneset دارد.

۲ مواد و روش‌ها

۱,۲ داده‌های مسئله

داده‌های مورد استفاده در این پژوهش شامل شبکه‌های مولکولی متنوعی هستند که از منابع معتبر استخراج شده‌اند. BioGRID شبکه‌ای از تعاملات پروتئین-پروتئین است که داده‌های آزمایشگاهی را جمع‌آوری می‌کند. این شبکه به دلیل وجود گره‌های هاب و توزیع نامتعادل یال‌ها چالش‌برانگیز است و برای پیش‌بینی تعاملات فیزیکی مناسب است STRING .ترکیبی از داده‌های تجربی، محاسباتی و متنی را ارائه می‌دهد و شامل نسخه‌های کامل و آزمایشی (STRING-EXP) است. این شبکه به دلیل چگالی[۱]متغیر، ابزار قدرتمندی برای تحلیل توزیع ارتباطات عملکردی محسوب می‌شود. InBioMap یک شبکه وزن‌دار است که قدرت تعاملات را با استفاده از وزن یال‌ها مشخص می‌کند و امکان تمرکز بر تعاملات قوی‌تر را فراهم می‌سازد. GIANT-TN یک شبکه ژنی بزرگ و پیچیده است که داده‌های وزن‌دار را در مقیاس ژنوم نمایش می‌دهد و برای تحلیل‌های جامع ژنتیکی ایده‌آل است. با توجه به حجم عظیم داده‌های GIANT-TN، InBioMap و STRING و کمبود منابع محاسباتی، ما فقط از شبکه‌های BioGRID و STRING-EXP در این مطالعه استفاده کردیم. در جدول-۱ به مشخصات هر کدام از این شبکه‌ها اشاره شده است.

علاوه بر این شبکه‌ها، از مجموعه‌های ژنی یا geneset collections استفاده شده است. این مجموعه‌ها در سه دسته اصلی تقسیم‌بندی می‌شوند: عملکرد زیستی، بیماری و صفات. مجموعه GOBP ، مرتبط با فرایندهای زیستی ژن‌ها، KEGGBPشامل مسیرهای زیستی ازKEGG ، برای پیش‌بینی عملکرد زیستی تعریف شده‌اند. مجموعه‌های DisGeNET و BeFree اطلاعات مرتبط با ژن‌های بیماری‌ها را ارائه می‌دهند. همچنین، مجموعه‌های GWAS و MGI مطالعات صفات پستانداران تطبیق داده‌شده به ژن‌های انسانی را پوشش می‌دهند، برای پیش‌بینی صفات استفاده می‌شوند. این مجموعه‌ها با پردازش‌های دقیق آماده شده‌اند تا وظایف پیش‌بینی مشخص و بدون تداخل تعریف شوند.

جدول-۱

Table S1. Information on the molecular networks. LT : low-throughput, HT : high-throughput, G : genetic, P : physical, DA : database annotations, CE : co-expression, NP : non-protein, R : regulation, CC : co-citation, O : orthologous.

Network	Number of Genes	Number of Edges	Edge Density	Network Construction Method	Weighted	Interaction Type
BioGRID	20,558	238,474	1.13e-3	LT	No	G, P
STRING-EXP	14,089	141,629	7.08e-4	HT	Yes	P
InBioMap	17,399	644,862	1.58e-3	HT	Yes	P, DA
GIANT-TN	25,689	38,904,929	1.92e-3	LT, HT	Yes	CE, NP, P, R
STRING	17,352	3,640,737	7.20e-3	HT	Yes	CC, CE, O, DA, P

بنابراین در نهایت ورودی برنامه شبکه(ها)ی ژنی و پروتئینی به همراه geneset collections هست و خروجی آن فایل‌های[۲]tsv. است که در آن‌ها امتیاز مربوط به هر یک از معیارهای auPRC، auROC و P@topK برای هر یک از genesetها در collection مربوطه مشخص شده و روش Validation استفاده شده و هر یک از روش های SL یا LP مشخص شده است و می‌توان از این فایل برای مقایسه نتایج به دست آمده استفاده کرد.

۲,۲ مرور روش مقاله مرجع

در این مطالعه، پنج روش پیش‌بینی از دو گروه انتشار برچسب (LP) و یادگیری تحت نظارت (SL) مورد استفاده قرار گرفته است. در گروه LP، از دو روش LP-A و LP-I استفاده شده است. در LP-A، پیش‌بینی بر اساس انتشار برچسب‌ها

[۱] Edge Density

[۲] Tab Separated Value

از گره‌های برچسب‌گذاری‌شده به گره‌های مجاور با استفاده از ماتریس مجاورت[۱] (A) انجام می‌شود. در LP-I، از ماتریس تأثیر[۲](I) که با الگوریتم Random Walk with Restart تولید می‌شود، برای گسترش اطلاعات برچسب‌ها در گراف استفاده می‌شود. این روش‌ها به دلیل استفاده از ساختار شبکه در خوشه‌های متراکم عملکرد خوبی دارند. در گروه SL، سه رویکرد به کار رفته است SL-A، SL-I و SL-E: در SL-A، ردیف‌های ماتریس مجاورت (A) به‌عنوان ویژگی‌ها برای مدل نظارت‌شده استفاده می‌شود. در SL-I، ردیف‌های ماتریس تأثیر (I) به‌عنوان ورودی به کار می‌روند. در SL-E، از بردارهای تعبیه[۳]ژن‌ها که با node2vec استخراج شده‌اند، استفاده می‌شود. الگوریتم node2vec بازنمایی‌های رتبه-پایین[۴]از گراف تولید می‌کند.

ارزیابی مدل‌ها نشان داد که روش‌های SL، به‌ویژه SL-A، عملکرد بهتری در پیش‌بینی ارتباط ژن‌ها با بیماری‌ها، صفات و عمکردهای زیستی نسبت به LP داشتند، در حالی که LP-I در داده‌های با تراکم بالا مناسب بود.

برای اعتبارسنجی مدل‌ها، سه روش اصلی به کار گرفته شده است. روش اول، Temporal Holdout، بر اساس زمان‌بندی برچسب‌گذاری داده‌ها طراحی شده و از ژن‌های دارای برچسب‌های قدیمی‌تر برای آموزش مدل و از ژن‌های جدیدتر برای تست استفاده کرده است. این روش به‌ویژه در چالش‌های CAFA[۵] برای سنجش توانایی پیش‌بینی در شرایط واقعی به کار گرفته شده است. روش دوم، Study-Bias Holdout، ژن‌هایی را که بیشترین مطالعه روی آن‌ها انجام شده است برای آموزش استفاده کرده و باقی ژن‌ها را برای تست نگه داشته است. روش سوم، 5-Fold Cross Validation، به صورت سنتی داده‌ها را به پنج بخش تقسیم کرده و به طور متناوب چهار بخش برای آموزش و یک بخش برای تست استفاده کرده است.

برای ارزیابی عملکرد مدل‌ها، از معیارهای مختلفی استفاده شده است auPRC[۶] (مساحت زیر منحنی دقت-یادآوری) برای بررسی توانایی مدل در یادآوری نمونه‌های مثبت واقعی و کاهش خطاهای مثبت کاذب به کار رفته است. P@TopK[۷] برای بررسی دقت مدل در شناسایی مهم‌ترین عوامل ژنتیکی هر مجموعه ژنی استفاده شده است. همچنین، auROC[۸](مساحت زیر منحنی ROC) برای ارزیابی توانایی مدل در تفکیک نمونه‌های مثبت و منفی به کار گرفته شده است. این ارزیابی‌ها نشان داده‌اند که مدل‌های نظارت‌شده در مقایسه با روش انتشار برچسب، به‌ویژه در پیش‌بینی عملکردهای زیستی، عملکرد بهتری داشته‌اند.

۳,۲ روش پیشنهادی

در روش به کار برده شده، رویکرد کلی روش مقاله تغییر نکرده است اما در روش‌های دیگری به کار برده شده است که در نتیجه آن در بعضی موارد مطابق انتظار منجر به بهبود نتایج و سرعت شده است و در برخی موارد تغییری ایجاد نشده یا حتی نتایج بدتری به دست آمده است.

رابطه-۱ نحوه به دست آوردن ماتریس تاثیر F را نشان می‌دهد که یکی از عوامل موثر در مقادیر ماتریس و نتیجه مدل، میزان وابستگی فرمول به وزن ماتریس در مرحله قبل است که با α تعیین می‌شود. α می‌تواند بین صفر تا ۱ باشد که ۱ نمایانگر بیشترین وابستگی به مقادیر قبلی است. در رویکرد مقاله این مقدار ۰,۸۵ بود اما در روش پیشنهادی مقدار α برای تشکیل ماتریس تاثیر در Label Propagation در سه مرتبه برابر با ۰,۲۵، ۰,۵ و ۰,۸۵ قرار گرفت تا تاثیر آن در نتایج به دست آمده مشخص شود. در نتیجه این تغییر، تفاوت قابل ملاحظه‌ای در نتایج به دست نیامد و نتایج ارزیابی‌های auPRC، auROC، P@topK با هم تقریبا مشابه و قابل چشم پوشی بود.

$$F = \alpha[I - (1-\alpha)W_D]^{-1}$$

رابطه-۱

برای به دست آوردن ماتریس تعبیه از روش node2vec استفاده می‌شود که برگرفته از روش word2vec است. در این روش گره‌های مرتبط با استفاده از پیاده‌روی تصادفی مرتبه دوم(متعصبانه)[۹]استخراج می‌شوند و توالی گره‌های مرتبط همانند آنچه به عنوان توالی کلمات در word2vec به مدل داده می‌شد، به شبکه عصبی داده می‌شود تا بردار embedding هر کلمه با ابعاد دلخواه به دست آید. رابطه-۲، تابع هدف[۱۰]روش node2vec برای به دست آوردن بردار با ابعاد-پایین[۱۱]از روی ماتریس مجاورت را نشان می‌دهد.

$$E = \max_{f} \sum_{u \in V} \log(\Pr(N_S(u)|e(u)))$$

رابطه-۲

مشکل روش اصلی پیاده‌سازی الگوریتم node2vec که در مقاله هم استفاده شده، سرعت بسیار پایین و حجم حافظه مورد نیاز بسیار بالا است. به طور مثال برای ایجاد بردار تعبیه ژن‌های شبکه STRING با این روش حدود ۵ ساعت زمان و حدود ۱۰۰ گیگابایت حافظه مورد نیاز است که در رایانه‌های معمولی قابل انجام نیست اما روش پیشنهاد شده در مقاله[۱] که موسوم به PecanPy است، توانسته فقط در ۱ دقیقه و با ۱ گیگابایت حافظه مورد استفاده بردارهای تعبیه ژن‌ها را به دست آورد.

---

۱ Adjacency Matrix
۲ Influence Matirx
۳ Embedded Vectors
۴ Low Rank
۵ Critical Assessment of protein Function Annotation algorithms
۶ Area Under Precision-Recall Curve
۷ Precision at Top K
۸ Area Under Receiver Operating Characteristic
۹ 2nd Order(Biased) Random Walk
۱۰ Objective Function
۱۱ Low-Dimensional

بهینه‌سازی‌های کلیدی در این روش شامل سه بخش اصلی است: استفاده از ساختمان داده‌های فشرده‌تر، بهره‌گیری از پردازش موازی و استفاده از حالت‌های OTF[1] برای مدیریت انتقالات تصادفی در شبکه.

در حالت اول از SparseOTF استفاده می‌کند که در آن از ماتریس‌های پراکنده برای ذخیره اطلاعات شبکه‌های بزرگ و پراکنده استفاده می‌کند. این ماتریس‌ها تنها مقادیر غیرصفر را ذخیره کرده و با دسترسی سریع به ردیف‌ها، محاسبات کارآمدتری ارائه می‌دهند. حالت بعدی DenseOTF است که برای شبکه‌های متراکم استفاده می‌شود که امکان دسترسی سریع‌تر به تمامی ارتباطات گره‌ها را فراهم می‌کند. در این حالت ماتریس به طور مستقیم در حافظه بارگذاری شده و پردازش می‌شود. در هر دو حالت، PecanPy امکان استفاده از فایل‌های ذخیره‌شده با فرمت‌های بهینه‌سازی‌شده مانند فایل‌های numpy را فراهم می‌کند که سرعت بارگذاری داده‌ها را به شدت افزایش می‌دهد.

همچنین در روش اصلی node2vec ، محاسبه احتمالات انتقال برای هر گره به صورت ترتیبی انجام می‌شود. اما در PecanPy ، این فرآیند از طریق کتابخانه Numba به طور کامل موازی شده است، به طوری که هر هسته پردازنده مسئول محاسبه احتمالات انتقال[2] برای یک بخش از گره‌ها است. تولید مسیرهای تصادفی نیز به صورت موازی انجام می‌شود. هر هسته پردازنده به طور مستقل تعدادی مسیر تصادفی را برای مجموعه‌ای از گره‌ها تولید می‌کند. این رویکرد باعث کاهش چشمگیر زمان اجرای الگوریتم در شبکه‌های بزرگ می‌شود.

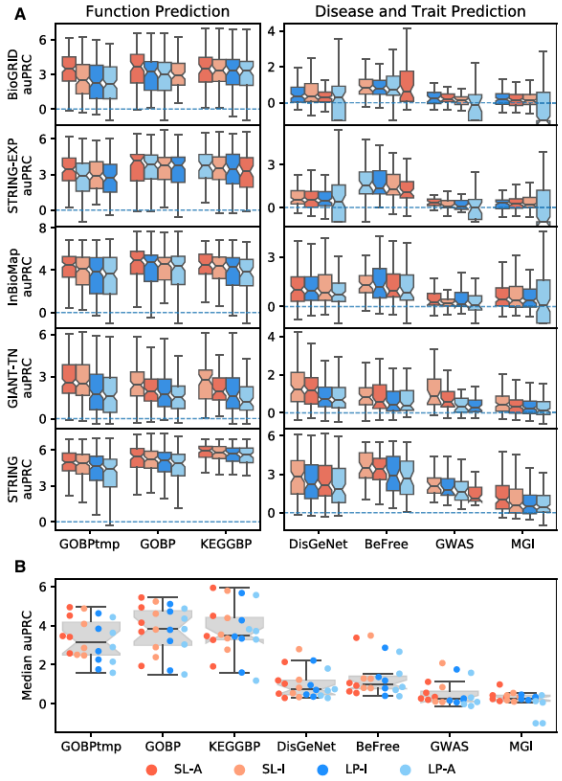
در نهایت در PecanPy ، حالت‌های OTF طراحی شده‌اند تا نیاز به ذخیره‌سازی پیش‌محاسبات (مانند احتمالات انتقال مرتبه دوم) را کاهش دهند. در SparseOTF احتمالات انتقال در هنگام تولید هر مسیر تصادفی به صورت بلادرنگ محاسبه می‌شوند. این روش باعث کاهش نیاز به حافظه می‌شود، زیرا نیازی به ذخیره تمامی احتمالات انتقال برای هر گره نیست. این حالت برای شبکه‌های پراکنده و بزرگ مناسب است.DenseOTF هم مشابه SparseOTF است اما از ماتریس‌های کامل استفاده می‌کند. این حالت برای شبکه‌های بسیار متراکم که تعداد ارتباطات بین گره‌ها بالاست، مناسب است. در این روش نیز محاسبات در هنگام تولید مسیرها انجام می‌شود، اما دسترسی مستقیم به ماتریس‌های کامل سرعت محاسبات را افزایش می‌دهد.

در نهایت در روش پیشنهادی بردارهای تعبیه به طول ۵۱۲ هم طول بردارهای تعبیه تولید شده از روش اصلی با استفاده از PecanPy از شبکه‌های BioGRID و STRING-EXP به دست آمد. با توجه به حجم عظیم داده‌ها همانطور که گفته شد فقط از شبکه‌های STRING-EXP و BioGRID در روش پیشنهادی استفاده شد. بنابراین در نهایت نتایج به دست آمده از آموزش مدل با SL در مقاله با روش node2vec اصلی که فقط مربوط به این دو شبکه بودند، با نتایج به دست آمده با همی نوع آموزش و با روش تعبیه PecanPy به دست آمده است، مقایسه شد و در قسمت نتایج درباره آن بحث خواهم کرد. هر دو این موارد با روش Logistic Regression به دست آمده بود.

همچنین به عنوان تغییر نهایی، برای به دست آوردن ژن‌های مرتبط با بیماری در SL مقاله از روش Logistic Regression استفاده کرده بود. اما در روش پیشنهادی علاوه بر این روش سعی شد از SVM[3] و Random Forest هم استفاده شود و نتایج آن‌ها با هم مقایسه شود تا بهترین روش انتخاب شود.

۳ نتایج

مقاله مرجع نتایج به دست آمده از مقایسه دو روش LP و SL را منتشر کرده است. به وضوح مشخص است که SL در اکثر مواقع عملکرد بهتری نسبت به LP داشته است که در شکل-۱ که از مقاله اصلی آورده شده است با توجه به نتایج auPRC مشخص شده است.



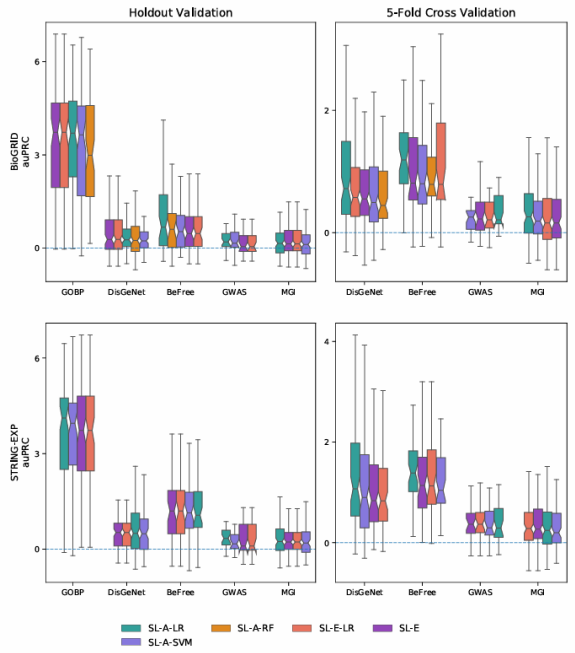
شکل-۱

با پذیرش این مورد در روش پیشنهادی، با توجه به شکل‌های ۲ تا ۳[4]، ما نتایج به دست آمده از روش‌های SVM(SL-A-SVM در شکل)، Logistic

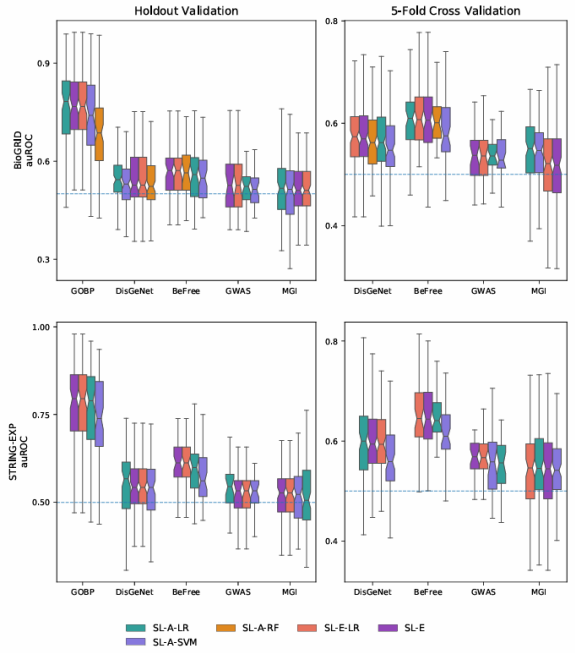
[1] On The Fly
[2] Transition Probabilities
[3] Support Vector Machine

[4] به دلیل محدودیت تعداد صفحات گزارش و کوچک بودن شکل‌ها شاید به اندازه کافی خوانا نباشند، برای مشاهده بهتر می‌توان به کد مقاله مورد نظر در قسمت my_figures_tables مراجعه کرد.

Regression(SL-A-LR در شکل) و Random Forest(SL-A-RF در شکل) را با معیارهای auPRC، auROC و P@topK را به دست آورده‌ایم. همچنین در کنار آن نتایج به دست آمده از SL در مقاله اصلی با استفاده از بردارهای تعبیه شده(SL-E در شکل) با نتایج به دست آمده با روش PecanPy(SL-E-LR در شکل) در این سه شکل نشان داده‌ایم و می‌توانیم مقایسه کنیم.



شکل-۲



شکل-۳

با توجه به نتایج به دست آمده از نمودارهای جعبه‌ای هر سه شکل، مشخص است بدون در نظر گرفتن روش، مدل در GOBP geneset collection، عملکرد بهتری داشته است که نشان‌دهنده توانایی بهتر مدل در پیش‌بینی عملکردهای زیستی نسبت به پیش‌بینی صفات و بیماری‌های مرتبط دارد. نکته بعدی عملکرد بهتر در پیش‌بینی بیماری‌ها در صورت استفاده از geneset collection BeFree به جای DisGeNet است. در بین روش‌های استفاده شده با ماتریس مجاورت، روش LR برتری مطلق نسبت به SVM و RF دارد. در صورت استفاده از شبکه BioGRID، بین دو روش SVM و RF، روش RF با وجود پیچیدگی محاسباتی بیشتر و زمانبر بودن نتایج بهتری را با معیار auPRC می‌دهد که به معنی تشخیص بهتر وجود ارتباط(عملکرد زیستی، بیماری یا صفت) در مدل RF است.

در رابطه با نحوه ایجاد بردار تعبیه با روش PecanPy و روش اصلی، هر چند تفاوت عملکردی بسیار کمی در بعضی موارد بین دو روش وجود دارد اما در اکثر موارد دیگر عملکرد مشابهی دارند که این نشان‌دهنده کارامدی روش PecanPy است که با زمان و حجم حافظه مورد نیاز بسیار کمتر عملکرد مشابهی دارد.

همچنین در پیش‌بینی عملکرد زیستی ژن‌ها، روش‌های مبتنی بر ماتریس مجاورت(A) عملکرد مشابهی با روش‌های مبتی بر بردار تعبیه(E) دارند.

۴ جمع‌بندی

در این تحقیق تلاش شده است تا عملکرد مدل با روش‌های مشابه به چالش کشیده شود. همانطور که ملاحظه شد، برای یادگیری ماشین نظارت شده، روش Logistic Regression که مقاله اصلی هم در نظر گرفته بهتر از روش‌های Random Forest و SVM عمل می‌کند. همچنین برای ایجاد بردارهای تعبیه روش پیشنهاد شده در مقاله[۱] که یک سال بعد از انتشار این مقاله منتشر شده توانایی ایجاد بردارهای تعبیه برای شبکه‌های با حجم بسیار بیشتر را با زمان بسیار کمتر دارد.

۵ منابع

[1] Liu, Renming & Krishnan, Arjun. (2020). PecanPy: a fast, efficient, and parallelized Python implementation of node2vec. 10.1101/2020.07.23.218487.