



تحلیل هوشمند تصاویر زیست پزشکی

نیم سال اول ۰۴-۰۳

مدرس: محمدحسین رهبان

تمرین چهارم

- مهلت ارسال پاسخ تا ساعت ۲۳:۵۹ روز مشخص شده است.
- در طول ترم امکان ارسال با تاخیر تمرین ها بدون کسر نمره تا سقف ۱۲ روز وجود دارد. محل بارگزاری جواب تمرین ها بعد از ۴ روز بسته خواهد شد و پس از گذشت این مدت، پاسخ های ارسال شده پذیرفته نخواهند شد.
- توجه داشته باشید که نوت بوک های شما باید قابلیت باز اجرای ۱۰۰ درصد داشته باشند و در صورت نیاز به نصب یک کتابخانه یا دسترسی به یک فایل، مراحل نصب و دانلود (از یک محل عمومی) در نوت بوک وجود داشته باشد.
- هم فکری در انجام تمرین مانعی ندارد، فقط توجه داشته باشید که پاسخ تمرین حتما باید توسط خود شخص نوشته شده باشد. همچنین در صورت هم فکری در هر تمرین، در ابتدای جواب تمرین نام افرادی که با آن ها هم فکری کرده اید را حتما ذکر کنید.
- برای پاسخ به سوالات نظری در صورتی که از برگه خود عکس تهیه می کنید، حتما توجه داشته باشید که تصویر کاملاً واضح و خوانا باشد. در صورتی که خوانایی کافی را نداشته باشد، تصحیح نخواهد شد.
- محل بارگذاری سوالات نظری و عملی در هر تمرین مجزا خواهد بود. به منظور بارگذاری بایستی تمارین تئوری در یک فایل زیپ با نام `IABI_Theo_hw4_[First-Name]_[Last-Name]_[Student-Id].zip` و تمارین عملی نیز در یک فایل مجزای زیپ با نام `IABI_Prac_hw4_[First-Name]_[Last-Name]_[Student-Id].zip` بارگذاری شوند.
- در صورت وجود هرگونه ابهام یا مشکل، در کوثرای درس آن مشکل را بیان کنید و از پیغام دادن مستقیم به دستیاران آموزشی خودداری کنید.

بخش نظری (۵۰ نمره)

سوال اول: تابع Aggregation در MIL (۱۰ نمره)

(الف) توضیح دهید که چرا انتخاب توابع aggregation با خاصیت permutation invariance برای مسائل MIL ضروری است.

(ب) در مسئله یادگیری (MIL) Multiple Instance Learning، سه تابع aggregation برای ترکیب ویژگی های پچ ها در نظر گرفته شده اند:

$$f_{\max}(x_1, x_2, \dots, x_n) = \max(x_1, x_2, \dots, x_n) \quad (\text{آ})$$

$$f_{\text{prod}}(x_1, x_2, \dots, x_n) = \prod_{i=1}^n x_i \quad (\text{ب})$$

$$f_{\text{concat}}(x_1, x_2, \dots, x_n) = [x_1, x_2, \dots, x_n] \quad (\text{ج})$$

برای هر یک از این توابع توضیح دهید که آیا مناسب برای مسائل MIL هستند یا نه و چرا.

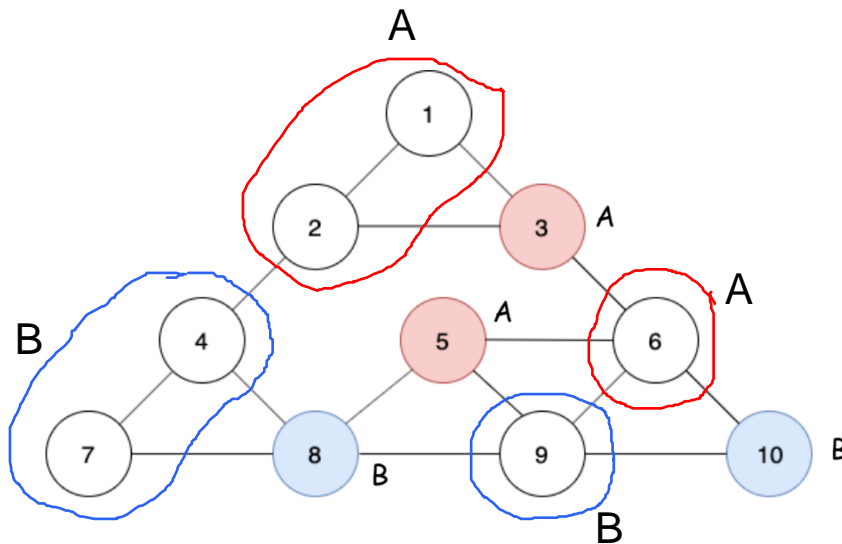
سوال دوم: طبقه‌بندی رابطه‌ای در گراف (۶ نمره)

می‌خواهیم از طبقه‌بندی رابطه‌ای برای پیش‌بینی برچسب‌های گره‌ها استفاده کنیم. گراف G را در نظر بگیرید که در شکل ۱ نشان داده شده است. می‌خواهیم گره‌ها را به دو کلاس A و B طبقه‌بندی کنیم. برچسب‌های گره‌های ۳، ۵، ۸ و ۱۰ داده شده‌اند. استفاده از یک طبقه‌بند رابطه‌ای احتمالاتی برای پیش‌بینی Y_i برای گره i به صورت زیر تعریف می‌شود:

$$P(Y_i = c) = \frac{1}{|N_i|} \sum_{(i,j) \in E} W(i,j) P(Y_j = c)$$

که در آن $|N_i|$ تعداد همسایه‌های گره i است.

- فرض کنید همه‌ی یال‌ها وزن ۱ دارند.
- برای گره‌های دارای برچسب، با مقدار واقعی Y مقارنه‌ی اولیه کنید و برای گره‌های بدون برچسب، از مقداردهی اولیه بی‌طرفانه یعنی $P(Y_i = A) = 0.5$ استفاده کنید.
- گره‌ها را به ترتیب شماره گره به‌روزرسانی کنید.



شکل ۱: گراف G

- (آ) پس از مرحله (تکرار) دوم، مقدار $P(Y_i = B)$ را برای $i = 2, 4, 6$ بدست آورید.
- (ب) اگر از ۰.۵ به عنوان آستانه احتمال استفاده کنیم، یعنی گره i به کلاس A تعلق دارد اگر $P(Y_i = A) > 0.5$ ، کدام گره در پایان به کلاس A تعلق خواهد داشت؟ و کدام به کلاس B تعلق خواهد داشت؟

سوال سوم: تابع Aggregator در GNN (۱۲ نمره)

فرض کنید یک گراف G داریم. ماتریس A ماتریس مجاورت آن است که نشان می‌دهد کدام گره‌ها به هم متصل‌اند و همچنین ماتریس قطری D که به ابعاد $|V| \times |V|$ است، ماتریس درجه گره‌های گراف است. بطوریکه عناصر قطر آن، درجه هر گره را نشان می‌دهد. فرض کنید $h_i^{(l)}$ نمایش‌دهنده Embedding گره i در لایه l باشد. همچنین از یک تجمیع‌کننده میانگین‌گیر (Mean Aggregator) برای آپدیت گره‌ها استفاده می‌کنیم به طوری که:

$$h_i^{(l+1)} = \frac{1}{|N_i|} \sum_{j \in N_i} h_j^{(l)}$$

اگر مقدار Embedding همه گره‌ها در لایه L را به شکل زیر Vectorize کنیم:

$$H^{(L)} = \begin{bmatrix} h_1^{(L)} \\ h_2^{(L)} \\ \vdots \\ h_{|V|}^{(L)} \end{bmatrix}$$

(آ) یک رابطه ماتریسی پیدا کنید که با داشتن ماتریس‌های A، D و بردار $H^{(L)}$ ، بردار $H^{(L+1)}$ که هر گره با استفاده از تابع تجميع‌کننده میانگین‌گیر آپدیت شده را محاسبه کند.

(ب) حال فرض کنید در گراف G می‌خواهیم بصورت تصادفی با احتمال یکنواخت (Uniform) حرکت کنیم. یعنی اگر گره i به گره‌های z و k متصل باشد، به احتمال مساوی از گره i به گره z یا k می‌توانیم برویم. ماتریس انتقال (Transition) را برای این حرکت تصادفی با داشتن ماتریس‌های D و A بدست بیاورید. با توجه به پاسخ بخش «آ» و این بخش چه نتیجه‌ای می‌توان گرفت؟
(راهنمایی: ماتریس Transition $|V| \times |V|$ است که درایه (i, j) در آن احتمال رفتن از گره i به گره j را نشان می‌دهد).

(ج) حال فرض کنید یک Skip Connection به تابع Aggregator اضافه کنیم که مقدار Embedding قبلی خودش نیز در مقدار Embedding لایه بعدش تاثیرگذار باشد، به طوریکه:

$$h_i^{(l+1)} = \frac{1}{3}h_i^{(l)} + \frac{2}{3} \frac{1}{|N_i|} \sum_{j \in N_i} h_j^{(l)}$$

همانند بخش «آ» رابطه ماتریسی مربوط به این تابع تجميع‌کننده را بنویسید.

(د) همانطور که تا اینجا با تابع تجميع‌کننده Mean آشنا شده‌اید، توابع تجميع‌کننده دیگری برای تجميع Embedding همسایه‌های یک گره در گراف وجود دارند. دو تابع دیگر عبارت است از:

$$h_v^{l+1} = \text{AGGREGATE}_{\min}(\{h_u^{(l)}, \forall u \in \mathcal{N}(v)\}) = \min_{u \in \mathcal{N}(v)} h_u^{(l)}$$

$$h_v^{l+1} = \text{AGGREGATE}_{\text{sum}}(\{h_u^{(l)}, \forall u \in \mathcal{N}(v)\}) = \sum_{u \in \mathcal{N}(v)} h_u^{(l)}$$

و

حال یک گراف $G(V, E)$ بیابید که دارای دو شرط زیر باشد:

$$|V| \geq 3$$

$$\forall v \in V, \deg(v) \geq 2$$

و همچنین مقدار اولیه‌ای به عنوان Embedding (همان h_i^0) داشته باشد که پس از اعمال تابع تجميع‌کننده Min در لایه اول و اعمال تابع تجميع‌کننده Sum در لایه دوم، مقدار Embedding گره‌ها در لایه دوم با مقادیر اولیه یکی شوند (یعنی $h_v^0 = h_v^2$ ، $\forall v \in V$).

سوال چهارم: چالش‌های GNN ها (۸ نمره)

میدانیم یک لایه از شبکه GNN استاندارد از تابع زیر برای Aggregation و اعمال Activation Function استفاده می‌کند تا مقادیر گره‌ها را بروزرسانی کند.

$$\mathbf{h}_v^{(l)} = \sigma \left(\sum_{u \in \mathcal{N}(v)} \mathbf{w}^{(l)} \frac{\mathbf{h}_u^{(l-1)}}{|\mathcal{N}(v)|} \right)$$

- (آ) یکی از چالش‌های اصلی GNN مشکل Over-smoothing است. این مشکل را توضیح دهید.
 دلیل بوجود آمدن این مشکل چیست؟ بدلیل چه خاصیتی شبکه‌های گرافی این چالش را دارند؟ استدلال خود را برای دو گره که در فاصله‌ی دوری از هم در گراف قرار دارند، با مثالی ترسیمی بیان کنید.
- (ب) روشی ارائه کنید که اگر نیاز به تعداد لایه‌های زیادی از GNN داریم، بتواند از بروز این مشکل جلوگیری کند. تابع آپدیت متناظر با روشی که بیان کردید را بنویسید.

سوال پنجم: Image Registration (۵ نمره)

با توجه به مطالبی که در کلاس آموختید، برای اینکه بتوانیم یک moving image را با fixed image انطباق دهیم و سپس مدلی داشته باشیم که بتوانیم فرآیند **image registration** را انجام دهیم:

(آ) یکی از روش‌ها برای این فرآیند استفاده از **VoxelMorph** است. ابتدا توضیح دهید ساختار **VoxelMorph** به چه شکل است و با توجه به اینکه این روش Unsupervised است، چه مزیتی نسبت به روش Supervised که می‌توان برای آن استفاده کرد دارد؟

(ب) همان‌طور که می‌دانید، عبارت loss در **VoxelMorph** به این صورت تعریف می‌شود:

$$\mathcal{L} = \|m \circ \phi - f\| + \lambda \text{Reg}(\phi)$$

که در آن بخش اول مربوط به تطبیق تصاویر (images match) و بخش دوم تضمین‌کننده یک میدان هموار (smooth field) است.

مهم‌ترین مشکلی که در این loss وجود دارد چیست؟ و با چه روشی می‌توان این مشکل را حل کرد؟ (راهنمایی: برای به دست آوردن اطلاعات بیشتر، به مقاله **VoxelMorph** مراجعه کنید.)

سوال ششم: SyntMorph (۹ نمره)

همان‌طور که از قبل میدانید یکی از روش‌های registration در تصاویر پزشکی **VoxelMorph** است. با توجه به سناریوی زیر به سوال پاسخ دهید:
 تعداد کمی داده‌ی T1-weighted داریم اما تصاویری که قرار است با آنها کار کنیم ممکن است از کانترست‌های مختلفی باشد (مثلاً T2)

(الف) به نظر شما استفاده از **VoxelMorph** برای چنین سناریویی مناسب است؟ چرا؟

(ب) یکی دیگر از روش‌های register کردن تصاویر پزشکی SyntMorph است. برای سناریو ذکر شده توضیح دهید استفاده از این روش مناسب است یا خیر؟ علت خود را نیز ذکر کنید. (برای آشنایی با روش Synt-Morph می‌توانید به این مقاله مراجعه کنید.)

